

软·件·工·程·师·典·藏·版

开发效率提升 10 倍

# JavaScript

## 网页特效

# 范例宝典

■ 张鑫 薛莉颖 高茹 编著



 人民邮电出版社  
POSTS & TELECOM PRESS

更多电子书资料请搜索「书行天下」：<http://www.sxpdf.com>



软·件·工·程·师·典·藏·版

开发效率提升 10 倍

# JavaScript

## 网页特效

# 范例宝典

■ 张鑫 薛莉颖 高茹 编著



 人民邮电出版社  
POSTS & TELECOM PRESS



# 目 录

---

[封面](#)

[扉页](#)

[前言](#)

[第1章 窗口/框架与导航条设计](#)

[1.1 弹出窗口控制](#)

[实例001 打开新窗口显示广告信息](#)

[实例002 定时打开窗口](#)

[实例003 通过按钮创建窗口](#)

[实例004 自动关闭的广告窗口](#)

[实例005 控制弹出窗口居中显示](#)

[实例006 弹出的窗口之Cookie控制](#)

[实例007 为弹出的窗口加入关闭按钮](#)

[实例008 关闭弹出窗口时刷新父窗口](#)

[实例009 关闭IE主窗口时，不弹出询问对话框](#)

[1.2 弹出网页对话框](#)

[实例010 弹出网页模式对话框](#)

[实例011 弹出全屏显示的网页模式对话框](#)

[实例012 网页拾色器](#)

[实例013 日期选择器](#)

[1.3 窗口的动画效果](#)

[实例014 页面自动滚动](#)

[实例015 打开窗口特殊效果](#)

[实例016 动态显示窗口](#)

[实例017 慢慢放大的窗口](#)

[实例018 下降式浏览器](#)

[实例019 旋转的窗口](#)

[实例020 移动的窗口](#)

[实例021 震动的窗口](#)

[实例022 弹出广告窗口](#)

#### [1.4 窗口控制](#)

[实例023 窗口始终在最上面](#)

[实例024 窗口的最小化、最大化](#)

[实例025 频道方式窗口](#)

[实例026 全屏显示](#)

[实例027 设置窗口大小和位置](#)

[实例028 刷新当前页](#)

[实例029 自动最大化](#)

[实例030 自定义导航控制面板](#)

[实例031 根据用户分辨率自动调整窗口](#)

#### [1.5 窗口的其他效果](#)

[实例032 打开窗口时显示对话框](#)

[实例033 使窗口背景透明](#)

[实例034 立体窗口](#)

[实例035 动态标题栏](#)

[实例036 固定大小的窗口](#)

#### [1.6 框架的应用](#)

[实例037 框架集的嵌套](#)

[实例038 在网页中应用浮动框架](#)

[实例039 创建空白框架](#)

[实例040 居中显示框架页](#)

#### [1.7 无边框窗口](#)

[实例041 全屏显示无边框有滚动条的窗口](#)

[实例042 应用CSS+DIV实现指定尺寸无边框无滚动条窗口](#)

[实例043 应用JavaScript实现指定尺寸无边框无滚动条窗口](#)

## [1.8 水平导航条应用](#)

[实例044 图片按钮导航条](#)

[实例045 导航条的动画效果](#)

[实例046 不用图片实现质感导航条](#)

## [1.9 下拉菜单式导航条](#)

[实例047 二级导航菜单](#)

[实例048 半透明背景的下拉菜单](#)

[实例049 展开式导航条](#)

[实例050 用层制作下拉菜单1](#)

[实例051 用层制作下拉菜单2](#)

## [1.10 侧导航条设计](#)

[实例052 自动隐藏式菜单](#)

[实例053 收缩式导航菜单](#)

[实例054 树状导航菜单](#)

## [第2章 表单及表单元素](#)

### [2.1 文本框/编辑框/隐藏域组件](#)

[实例055 获取文本框/编辑框/隐藏域的值](#)

[实例056 自动计算金额](#)

[实例057 设置文本框的只读属性](#)

[实例058 限制多行文本域输入的字符个数](#)

[实例059 自动选择文本框/编辑框中的文字](#)

[实例060 按下回车键时自动切换焦点](#)

### [2.2 下拉列表/菜单](#)

[实例061 获取下拉列表/菜单的值](#)

[实例062 遍历多选择下拉列表](#)

[实例063 在下拉列表中进行多选移除](#)

[实例064 将数组中的数据添加到下拉菜单中](#)

[实例065 应用下拉菜单选择所要联机的网站](#)

[实例066 多级级联菜单](#)

[实例067 可以输入文字的下拉菜单](#)

[实例068 根据下拉菜单的值显示不同控件](#)

[实例069 分级下拉列表](#)

### [2.3 单选按钮组](#)

[实例070 不提交表单获取单选按钮的值](#)

[实例071 选中单选按钮后显示其他表单元素](#)

[实例072 通过单选按钮控制其他表单元素是否可用](#)

### [2.4 复选框](#)

[实例073 不提交表单获取复选框的值](#)

[实例074 控制复选框的全选或反选](#)

[实例075 只有一个复选框时控制复选框的全选或反选](#)

### [2.5 密码域](#)

[实例076 让您的密码域更安全](#)

[实例077 不提交表单自动检测密码域是否相同](#)

### [2.6 表单应用](#)

[实例078 通过JavaScript控制表单的提交与重置](#)

[实例079 带记忆功能的表单](#)

[实例080 防止表单重复提交](#)

[实例081 自动提交表单](#)

[实例082 通过for循环获取表单元素的中文名称](#)

[实例083 可以提交到不同处理页的表单](#)

## 第3章 实用JavaScript函数

### 3.1 数据验证

实例084 通过正则表达式验证日期

实例085 验证输入的日期格式是否正确

实例086 检查表单元素是否为空

实例087 验证E-mail是否正确

实例088 通过正则表达式验证电话号码

实例089 验证输入的字符串是否为汉字

实例090 验证身份证号码

实例091 验证用户名和密码

实例092 验证车牌号码

实例093 验证网站地址

实例094 验证数量和金额

实例095 验证字符串是否以指定字符开头

实例096 限制输入字符串的长度

### 3.2 字符串处理

实例097 小写金额转换为大写金额

实例098 去除字符串中的空格

实例099 转换输入文本中的回车和空格

实例100 将数字字符串格式化为指定长度

实例101 把一个长数字分位显示

实例102 将RGB格式的颜色值转换为十六进制格式

实例103 将IP地址转换为对应的数值

实例104 从指定URL中提取文件名

### 3.3 随机函数

实例105 随机产生指定位数的验证码

实例106 生成随机字符串

### 3.4 日期与时间

实例107 显示长日期格式的系统日期

实例108 实时显示系统时间

## 第4章 日期和时间

### 4.1 日期时间显示

实例109 在标题栏中显示日期时间

实例110 使用toLocaleString(\_)方法获取本地时间

实例111 全中文显示日期

实例112 在状态栏中显示日期时间

实例113 使用数组显示星期

实例114 在表格中显示时间

实例115 退出页面时显示停留时间

实例116 显示用户在页面的停留时间

实例117 显示最后修改时间

### 4.2 日期时间算法

实例118 计算某一天是星期几

实例119 判断指定年份是否为闰年

实例120 计算从出生到现在度过的时间

实例121 返回两个日期之间的间隔小时

实例122 倒计时

实例123 访问时间限制

实例124 计步器

### 4.3 日期时间特效

实例125 分时间候

实例126 节日提示

实例127 倒影时钟

实例128 带阴影的时钟

[实例129 动态石英钟](#)

[实例130 生日提醒器](#)

[实例131 数字时钟](#)

[实例132 各地时间表](#)

[实例133 位于页面顶层的时钟](#)

## [第5章 文字特效](#)

### [5.1 文字的颜色变换](#)

[实例134 简单文字变色](#)

[实例135 文字变色](#)

[实例136 变换的文字](#)

[实例137 描边文字](#)

[实例138 霓虹灯文字](#)

[实例139 追逐点亮的文字](#)

[实例140 荧光文字](#)

[实例141 发光文字的闪烁效果](#)

[实例142 文字的舞台灯光效果](#)

[实例143 制作彩色渐变文字动画](#)

### [5.2 改变文字大小](#)

[实例144 文字伸缩](#)

[实例145 动荡的文字](#)

[实例146 文字伸展](#)

[实例147 文字逐个放大](#)

[实例148 自动改变大小](#)

[实例149 选择字体的大小](#)

### [5.3 文字显示效果](#)

[实例150 文字渐隐渐现](#)

[实例151 文字虚幻变化](#)

[实例152 文字虚幻抖动](#)

#### [5.4 指定文字位置](#)

[实例153 将文字置于工作区左上角](#)

[实例154 右上角文字](#)

[实例155 鼠标移动文字](#)

#### [5.5 文字动态移动](#)

[实例156 文字自动滚屏](#)

[实例157 文字上下滚动](#)

#### [5.6 文字立体效果](#)

[实例158 文字的旋转](#)

[实例159 立体旋转文字](#)

[实例160 文字3D效果](#)

#### [5.7 文字动画效果](#)

[实例161 反弹文字](#)

[实例162 飞舞的文字](#)

[实例163 飞翔的文字](#)

[实例164 平面旋转的文字](#)

[实例165 输出文字](#)

[实例166 文字打字效果](#)

[实例167 文字抖动](#)

[实例168 指向文字时飞出星形标记](#)

[实例169 文字的抛出效果](#)

#### [5.8 文字特殊效果](#)

[实例170 波浪文字](#)

[实例171 梦幻文字效果](#)

[实例172 屏风文字](#)

[实例173 文字阴影](#)

## 5.9 其他

实例174 随机显示文字

实例175 文字加密及解密

## 第6章 超级链接特效

### 6.1 超级链接样式

实例176 改变超级链接字体样式

实例177 当鼠标移动到超级链接时改变超级链接颜色

实例178 改变超级链接背景色

### 6.2 超级链接控制

实例179 建立E-mail超级链接

实例180 获取页面中的全部超级链接

实例181 将网站设为首页

实例182 单击鼠标右键自动链接到指定网站

实例183 单击超级链接将本页加入收藏夹

实例184 访问指定的链接地址

实例185 返回默认主页

实例186 导航链接

### 6.3 超级链接特效

实例187 快速闪动页面中的超级链接

实例188 滚动的超级链接提示信息

实例189 公告栏中显示超级链接

实例190 显示超级链接站点相关信息

实例191 显示超级链接的提示信息

实例192 半透明背景的超级链接提示

## 第7章 操作表格

### 7.1 表格的颜色效果

实例193 闪烁的表格边框

[实例194 单元格边框变色](#)

[实例195 选中的行变色](#)

## [7.2 对单元格的焦点进行控制](#)

[实例196 选定表格中的单元格](#)

[实例197 左右移动单元格的信息](#)

[实例198 通过键盘使单元格焦点任意移动](#)

## [7.3 对表格的行、列进行修改](#)

[实例199 动态制作表格](#)

[实例200 动态生成行或列](#)

[实例201 删除表中的行](#)

## [7.4 单元格的相关操作](#)

[实例202 隐藏及显示单元格](#)

[实例203 编辑单元格中的文本信息](#)

[实例204 合并单元格](#)

[实例205 在表格中添加行及单元格](#)

[实例206 删除表中的单元格](#)

## [7.5 表格的特殊效果](#)

[实例207 透明表格](#)

[实例208 限制表格的宽度](#)

[实例209 表格的标题](#)

[实例210 表格的外阴影](#)

[实例211 立体表格](#)

[实例212 虚线边框表格](#)

[实例213 表格作为分割线](#)

[实例214 表格向下展开](#)

## [第8章 图形图像与多媒体](#)

### [8.1 图片大小](#)

[实例215 打开自定义大小的图片](#)

[实例216 图片放大缩小](#)

[实例217 通过鼠标滚轮放大缩小图片](#)

## [8.2 图片与鼠标相关操作](#)

[实例218 跟随鼠标移动的图片](#)

[实例219 可以左右拖动的图片](#)

[实例220 随意拖动图片](#)

[实例221 当鼠标经过图片时显示图片](#)

[实例222 改变图片获取焦点时的状态](#)

[实例223 抖动的图片](#)

[实例224 鼠标移动放大图片](#)

## [8.3 图片与时间相关操作](#)

[实例225 定时隐藏图片](#)

[实例226 根据时间变换页面背景](#)

[实例227 使图片不停闪烁](#)

[实例228 上下跳动的图片](#)

[实例229 图片左右晃动](#)

[实例230 飘舞的变形图片](#)

## [8.4 图片的动画效果](#)

[实例231 图片翻转效果](#)

[实例232 水波倒影特效](#)

[实例233 图片渐隐渐现](#)

[实例234 图片的探照灯效果](#)

[实例235 雷达扫描图片特效](#)

[实例236 在页面中旋转的图片](#)

[实例237 改变形状的图片](#)

[实例238 图片在页面浮动](#)

[实例239 随机变化的网页背景](#)

## [8.5 选择头像](#)

[实例240 在列表中选择图片](#)

[实例241 在弹出的新窗口中选择图片](#)

## [8.6 在页面中播放图片](#)

[实例242 幻灯片式播放图片](#)

[实例243 无间断的图片循环滚动效果](#)

## [8.7 图片的其他效果](#)

[实例244 导航地图](#)

## [8.8 播放音乐](#)

[实例245 为网页设置背景音乐](#)

[实例246 随机播放背景音乐](#)

[实例247 MIDI音乐选择](#)

## [8.9 插入Flash动画](#)

[实例248 插入Flash动画](#)

[实例249 插入背景透明的Flash动画](#)

## [8.10 播放视频文件](#)

[实例250 播放AVI文件](#)

[实例251 自制视频播放器](#)

# [第9章 页面特效](#)

## [9.1 页面背景效果](#)

[实例252 背景固定居中](#)

[实例253 背景图片纵向重复显示](#)

[实例254 通过按钮变换背景颜色](#)

[实例255 背景自动变色](#)

[实例256 百叶窗](#)

[实例257 渐隐渐现的背景颜色](#)

[实例258 页面缩小](#)

[实例259 页面上下打开效果](#)

[实例260 页面左右打开效果](#)

[实例261 页面溶解效果](#)

## [9.2 特殊页面](#)

[实例262 页首页尾切换](#)

[实例263 调用下载页面](#)

[实例264 程序加载页面](#)

[实例265 颜色拾取器](#)

## [9.3 广告页面](#)

[实例266 图片总置于顶端](#)

[实例267 随机显示广告](#)

[实例268 广告随滚动条漂移](#)

## [9.4 页面动画效果](#)

[实例269 下雪](#)

[实例270 飘落的枫叶](#)

[实例271 下雨](#)

[实例272 背景的烟花效果](#)

[实例273 变色的圆圈](#)

[实例274 滚动的光环](#)

[实例275 星空极速飞入效果](#)

[实例276 闪烁的星星](#)

## [9.5 其他](#)

[实例277 带密码的网页](#)

[实例278 页面左右滚动](#)

[实例279 动态移动的层](#)

[实例280 在页面中显示十字光标](#)

## 第10章 状态栏特效

### 10.1 状态栏文字动画

实例281 使状态栏中的文字不停闪烁

实例282 文字从右向左依次弹出

实例283 文字从中间向两边展开

实例284 文字跑马灯特效

实例285 文字依次显示后快速收缩

实例286 文字的展开与收缩

实例287 状态栏中的文字依次弹出

### 10.2 其他

实例288 在状态栏显示固定自定义信息

实例289 在状态栏显示鼠标坐标

实例290 在状态栏显示特定的超级链接信息

## 第11章 报表与打印

### 11.1 Web打印

实例291 调用IE自身的打印功能实现打印

实例292 打印指定框架中的内容

实例293 利用WebBrowser打印

实例294 设置页眉页脚

### 11.2 利用Word打印报表

实例295 将页面中的表格导出到Word并打印

实例296 打开指定的Word文档并打印

实例297 在JSP中利用Word自动打印指定格式的会议记录

实例298 在ASP中利用Word自动打印指定格式的会议记录

实例299 在PHP中调用Word自动打印指定格式的会议记录

### 11.3 利用Excel打印报表

实例300 将Web页面中的数据导出到Excel

[实例301 将Web页面中的数据导出到Excel并自动打印](#)

#### [11.4 利用CSS样式打印](#)

[实例302 利用CSS样式打印页面中的指定内容](#)

[实例303 利用CSS样式分页打印](#)

#### [11.5 套打邮寄产品单](#)

[实例304 打印汇款单](#)

[实例305 打印快递单](#)

[实例306 打印信封](#)

### [第12章 网站安全](#)

#### [12.1 禁止用户复制网页内容](#)

[实例307 禁止用户复制网页内容（方法一）](#)

[实例308 禁止用户复制网页内容（方法二）](#)

[实例309 禁止网页另存为](#)

#### [12.2 禁止用户刷新屏幕](#)

[实例310 屏蔽IE主菜单](#)

[实例311 屏蔽键盘相关事件](#)

[实例312 屏蔽鼠标右键](#)

#### [12.3 登录页面](#)

[实例313 具有浏览器检测功能的登录页面](#)

[实例314 防止SQL注入的登录页面](#)

[实例315 带验证码的登录页面](#)

#### [12.4 其他](#)

[实例316 使用Script Encoder加密工具加密](#)

### [第13章 HTML/CSS样式](#)

#### [13.1 页面效果](#)

[实例317 统一站内网页风格](#)

[实例318 设置超级链接文字的样式](#)

[实例319 网页换肤](#)

[实例320 滚动文字](#)

[实例321 制作渐变背景](#)

### [13.2 表格样式](#)

[实例322 只有外边框的表格](#)

[实例323 彩色外边框的表格](#)

[实例324 控制表格指定外边框不显示](#)

[实例325 背景颜色渐变的表格](#)

[实例326 表格隔行变色](#)

### [13.3 鼠标及滚动条样式](#)

[实例327 显示自定义鼠标形状](#)

[实例328 动画光标](#)

[实例329 制作彩色滚动条](#)

### [13.4 文字及列表样式](#)

[实例330 应用删除线样式标记商品特价](#)

[实例331 在文字上方标注说明标记](#)

[实例332 指定图标的列表项](#)

### [13.5 文字滤镜特效](#)

[实例333 文字的发光效果](#)

[实例334 文字的阴影效果](#)

[实例335 文字的渐变阴影效果](#)

[实例336 文字的图案填充效果](#)

[实例337 文字的探照灯效果](#)

[实例338 文字的闪烁效果](#)

[实例339 文字的空心效果](#)

[实例340 文字的浮雕效果](#)

[实例341 文字的阳文效果](#)

[实例342 文字的雪雕效果](#)

[实例343 火焰字](#)

[实例344 文字扭曲动画](#)

### [13.6 图片滤镜特效](#)

[实例345 图片的半透明效果](#)

[实例346 图片的模糊效果](#)

[实例347 图片的水波纹特效](#)

[实例348 图片的灰度效果](#)

[实例349 图片的动态说明文字](#)

## [第14章 JavaScript与XML](#)

### [14.1 读取XML文件](#)

[实例350 使用XML DOM对象读取XML文件](#)

[实例351 使用XMLHttpRequest对象读取XML文件](#)

### [14.2 显示XML文档](#)

[实例352 使用CSS显示XML文档](#)

[实例353 使用XSL显示XML文档](#)

[实例354 使用IE XML数据岛输出XML文档](#)

### [14.3 操作XML](#)

[实例355 用JavaScript控制XML文档的分页显示](#)

[实例356 通过操作XML数据岛实现添加、删除留言信息](#)

## [第15章 JavaScript与ASP结合](#)

### [15.1 弹出窗口/对话框控制](#)

[实例357 关闭弹出窗口时刷新父窗口](#)

[实例358 打开新窗口显示详细信息](#)

[实例359 弹出网页模式对话框](#)

[实例360 日期选择器](#)

[实例361 弹出提示对话框并重定向网页](#)

[实例362 打开指定大小的新窗口并居中显示](#)

## [15.2 弹出确认对话框](#)

[实例363 删除数据前弹出确认对话框](#)

[实例364 安全退出前弹出确认对话框](#)

## [15.3 其他](#)

[实例365 远程获取其他网页情报](#)

[实例366 树状导航菜单](#)

[实例367 使用XML实现不刷新页面查询数据](#)

## [第16章 JavaScript与JSP结合](#)

### [16.1 窗口与对话框](#)

[实例368 弹出提示对话框并重定向网页](#)

[实例369 打开新窗口显示详细信息](#)

[实例370 打开指定大小的新窗口并居中显示](#)

[实例371 弹出网页模式对话框](#)

[实例372 关闭弹出窗口时刷新父窗口](#)

### [16.2 无刷新技术](#)

[实例373 无刷新的用户名检测](#)

[实例374 无刷新的级联下拉列表](#)

[实例375 使用XML实现不刷新页面查询数据](#)

### [16.3 调用与控制](#)

[实例376 将查询结果导出到Word](#)

[实例377 调用Excel](#)

[实例378 调用PowerPoint](#)

### [16.4 其他](#)

[实例379 树状导航菜单](#)

## [第17章 JavaScript与ASP.NET结合](#)

### [17.1 窗口与对话框](#)

[实例380 使用JavaScript脚本弹出对话框](#)

[实例381 打开新窗口显示详细信息](#)

[实例382 关闭弹出窗口刷新父窗口](#)

[实例383 弹出网页模式对话框](#)

[实例384 弹出全屏显示网页](#)

## [17.2 其他](#)

[实例385 自动隐藏式菜单](#)

## [第18章 JavaScript与PHP结合](#)

### [18.1 窗口与对话框](#)

[实例386 弹出提示对话框并重定向网页](#)

[实例387 关闭弹出窗口时自动刷新父窗口](#)

[实例388 在弹出的网页模式对话框中选择个性头像](#)

[实例389 时间选择器](#)

[实例390 弹出提示对话框并重定向网页](#)

[实例391 删除数据前弹出确认对话框](#)

### [18.2 其他](#)

[实例392 树状导航菜单](#)

## [第19章 综合应用](#)

### [19.1 计算器](#)

[实例393 简单计算器](#)

[实例394 复杂计算器](#)

### [19.2 日历](#)

[实例395 精美日历](#)

[实例396 带农历的日历](#)

### [19.3 购物车](#)

[实例397 添加至购物车](#)

[实例398 查看购物车](#)

[实例399 修改商品购买数量](#)

[实例400 从购物车中移去指定商品](#)

[实例401 清空购物车](#)

## [第20章 jQuery应用开发](#)

### [20.1 jQuery基础开发](#)

[实例402 获取文本输入框中输入的值](#)

[实例403 动态修改<div>标记内容](#)

[实例404 通过类名选择器选择元素并更改样式](#)

[实例405 隐藏超级链接地址](#)

[实例406 为版权列表设置样式](#)

[实例407 为表单的直接子元素input换肤](#)

[实例408 改变指定元素的背景颜色为淡蓝色](#)

[实例409 筛选页面中div元素的同辈元素](#)

[实例410 获取页面上隐藏和显示的input元素值](#)

[实例411 获取和设置元素的文本内容与HTML内容](#)

[实例412 为多行列表框设置并获取值](#)

### [20.2 jQuery表格应用](#)

[实例413 匹配表单中相应的元素](#)

[实例414 带表头的双色表格](#)

[实例415 应用内容过滤器匹配指定单元格](#)

[实例416 隔行换色并且鼠标指向行变色的表格](#)

### [20.3 jQuery在实际中的应用](#)

[实例417 自动隐藏式菜单](#)

[实例418 伸缩式导航菜单](#)

[实例419 实现图片传送带](#)

[实例420 打造自己的开心农场](#)

## [附录](#)

[版权](#)

[光盘下载链接](#)

# JavaScript网页特效范例宝典

■张鑫 薛莉颖 高茹 编著

人民邮电出版社

北京

# 前言

随着网络的迅速发展，Web应用已不再是高不可攀的技术，但要制作出页面友好、美观、具有良好交互性的网站并非是一件容易的事。它不仅需要网站开发者熟练掌握服务器端开发技术，而且需要掌握客户端脚本语言。

JavaScript是比较流行的一种制作网页特效的脚本语言，它由客户端浏览器解释执行，可以应用在ASP、JSP、PHP和ASP.NET网站中。目前非常热门的jQuery就是以JavaScript为基础的。由此可见，熟练掌握并应用JavaScript对于网站开发人员来说是非常重要的。

总之，网站开发是一门创造性较强的活动，因其涉及面广，开发者往往需要学习、研究各方面的技术和问题。技术水平的提高与开发时间成正比，因此需要长时间经验的积累和磨炼。编程也是一项需要相互学习、相互交流的工作，在交流过程中，不但可以分享他人的编程经验、体会，更会产生创新的灵感，达到事半功倍的效果。

## 本书内容

本书精选了420个典型实例及应用特效实例，所选实例主要覆盖开发中的热点问题和关键问题。全书按实际应用进行分类，可以使读者在短时间内掌握更多的有用的技术，快速提高编程水平。书中所选内容均来源于实际项目的开发，有的实例是作者开发实践的积累，有的实例来源于公司的开发项目，还有的来自读者的提问。这些实例所涉及的问题都是非常实用的。

通过对这些实例进行详细分析和讲解，可以让读者迅速掌握网站设计中的开发经验技巧，迅速提高网站设计的综合水平。全书共 20 章，涵盖了窗口/框架与导航条设计、表单及表单元素、实用 JavaScript 函数、日期和时间、文字特效、超级链接特效、操作表格、图形图像与多媒体、页面特效、状态栏特效、报表与打印、网站安全、HTML/CSS 样式、JavaScript 与 XML、JavaScript 与 ASP 结合、JavaScript 与 JSP 结合、JavaScript 与 ASP.NET 结合、JavaScript 与 PHP 结合、综合应用、jQuery 应用开发等方面的内容。

在实例讲解上，全书采用了统一的编排方式，每个实例都包括“实例说明”、“技术要点”、“实现过程”和“举一反三”4 个部分。在“实例说明”中，以图文结合的方式给出了实例的功能说明及运行效果。在“技术要点”中给出了实例的重点、难点技术和相关编程技巧。在“实现过程”中介绍了该实例的设计过程和主要程序代码。在“举一反三”中给出了相关实例的扩展应用。

### 本书特色

- 所有实例内容都以解决开发者在编程中遇到的实际问题和开发中应该掌握的技术为中心，每个实例都可以独立解决某一方面的问题。有的可以解决工作中的难题，有的可以提高工作效率，有的可以提升工作价值。

- 所选实例具有极强的扩展性，能够给读者以启发，使读者举一反三、实现非常实用的效果。

- 所选实例具有广泛的代表性，所有实例都提供了源代码，读者可以快速使用。

- 为便于查找实例中的编程技术和技巧，本书附录提供了程序快速索引功能，该索引按字母顺序列出了本书中使用相关技术和技巧的实例号。

● 本书光盘中提供了本书实例的“源码速查电子搜索引擎”，读者可以快速检索所需技术和代码。

### 本书的约定

● 书中每个实例的标题栏都给出了程序的特色和所在光盘中的路径，读者可根据需要学习和使用。

● 书中涉及的数据库实例，在对应文件夹中均提供了用于保存数据库文件的文件夹。

● 书中有可能多个实例用到了同一主要技术，为节省篇幅，相关技术的讲解只在一个实例中进行介绍，读者可通过书后的技术索引了解相关技术。

● 因篇幅限制，本书实例只给出了关键代码，其他代码参见光盘中实例的源程序。

● 使用本书实例光盘前，请仔细阅读光盘中的“光盘使用说明”。

### 本书的服务

本书由明日科技组织编写，参加编写的有王小科、王国辉、崔佳音、张鑫、周佳星、辛洪郁、赛奎春、高春艳、杨丽、陈英、刘佳、刘丽艳、刘红艳、孙雨婷等。由于作者水平有限，疏漏之处在所难免，请广大读者批评指正。

如果读者在使用本书时遇到问题，可以访问明日科技网站，我们将通过明日科技网站全面为读者提供网上服务和支持。对于读者使用本书时遇到的错误和问题，我们承诺将在5个工作日内回复。

服务网站：[www.mingribook.com](http://www.mingribook.com)

服务信箱：[mingrisoft@mingrisoft.com](mailto:mingrisoft@mingrisoft.com)

服务电话：4006751066

作者

2014年7月

# 第1章 窗口/框架与导航条 设计

窗口控制

弹出网页对话框

窗口的动画及其他效果

框架的应用

无边框窗口

水平导航条应用

下拉菜单式导航条

## 1.1 弹出窗口控制

本节所说的弹出窗口是指通过 `window.open()` 方法打开的新窗口。这种弹出窗口在网站中经常会用到，如众所周知的弹出式网站公告或广告等，还有打开新窗口显示公告的详细内容等。下面将通过具体实例介绍如何控制弹出窗口。

### 实例001 打开新窗口显示广告信息

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\01\001

实例说明

广告是网站最大的盈利手段，任何网站都不可能将其省略，那么，如何才能既可以从广告中获利，又能使用户流畅地阅读网站信息呢？本实例将介绍如何制作一个单独的广告窗体，在用户登录网站的时候自动弹出，然后再由用户将其关闭。运行本实例，将弹出一个新窗口显示广告信息，运行结果如图1.1所示。



图1.1 打开新窗口显示广告信息

### 技术要点

本实例主要应用 JavaScript 的 window 对象，该对象应用极为普遍，但主要应用在 HTML 中打开窗口。用户浏览器决定窗口的样子，设计者左右不了其窗口的大小及样式，但 JavaScript 给了程序这种控制权。在 JavaScript 中，可以使用 window 对象实现对窗口的控制，该对象的常用方法如表 1.1 所示。

下面将对本实例中应用的 open() 方法进行详细介绍。

表 1.1 window 对象的常用方法

方 法	说 明
alert()	弹出一个警告对话框
confirm()	在确认对话框中显示指定的字符串
prompt()	弹出一个提示对话框
close()	关闭被引用的窗口
focus()	将被引用的窗口放在所有打开窗口的前面
open()	打开新浏览器窗口并且显示由 URL 或名字引用的文档，并设置创建窗口的属性
resizeTo(x,y)	设置窗口的大小
resizeBy(offsetx,offsety)	按照指定的位移量设置窗口的大小

应用 JavaScript 的 open() 方法可以打开浏览器窗口。程序开发人员可以基于特定的条件创建带有被装入其中的特定文档的新窗口，也可以指定新窗口的大小以及窗口中可用的选项，并且可以为引用它的窗口指派名字。

使用window对象打开窗口的语法格式如下。

```
WindowVar=window.open(url,windowname[,location]);
```

参数说明如下。

- WindowVar: 当前打开窗口的句柄。如果 open() 方法成功，则 WindowVar 的值为一个window对象的句柄，否则WindowVar的值是一个空值。

- url: 目标窗口的URL。如果URL 是一个空字符串，则浏览器将打开一个空白窗口，允许用write() 方法创建动态HTML。

- windowname: window对象的名称。

- location: 对窗口属性进行设置，其可选参数如表1.2 所示。

表1.2 对窗口属性进行设置的可选参数

属 性	说 明
width	窗口的宽度
height	窗口的高度
scrollbars	是否显示滚动条
resizable	设定窗口大小是否固定
toolbar	浏览器工具条, 包括后退及前进按钮等
menubar	菜单条, 一般包括有文件、编辑及其他一些条目
location	定位区, 也叫地址栏, 是可以输入 URL 的浏览器文本区
direction	更新信息的按钮

## 实现过程

(1) 首先创建用于弹出广告信息的页面, 在该页面中添加所要宣传的广告信息。

(2) 在网站首页写上如下代码, 每次进入网站首页时将弹出广告窗口, 代码如下。

```
<script language="javascript">
<!--
window.open("new.htm", "new", "height=141,width=693,top=10,
left=20");
-->
</script>
```

## 举一反三

根据本实例, 读者可以:

开发大型门户网站中弹出广告窗口;

开发商务网站中弹出公告或广告窗口;

开发信息资讯网站中弹出公告或广告窗口。

## [实例002 定时打开窗口](#)

本实例是一个简化操作的程序

实例位置: 光盘\mingrisoft\01\002

## 实例说明

在开发网站时，常常需要打开窗口，而打开窗口有很多种方式，本实例主要讲解如何定时打开窗口。运行本实例，双击 index.htm 浏览该页面，5s 后将打开一个指定大小的窗口，通过该窗口可以显示相关数据信息或是一个网站内的宣传广告等。程序运行结果如图1.2所示。



图1.2 定时打开窗口

## 技术要点

本实例主要应用 window 对象 setTimeout() 方法实现定时打开窗口，下面将对 setTimeout() 方法进行详细介绍。

setTimeout() 方法的语法格式如下。

setTimeout (function, milliseconds)

参数说明如下。

- function: 要调用的JavaScript 自定义函数名称。
- milliseconds: 设路超时时间 (以ms 为单位)。

功能: 超过超时时间后, 调用函数。此值可以用clearTimeout () 函数清除。

注意: setTimeout () 方法在超时时间后只调用一次函数, 而 setInterval () 方法是按一定时间重复调用指定的函数。

说明: 1000ms等于1s。

实现过程

(1) 制作用于打开窗口时显示的数据信息页面, 在该页面中加入相关所要显示的数据信息。

(2) 应用function函数创建自定义pp () 函数, 同时应用 setTimeout () 方法实现定时打开窗口, 代码如下。

```
<script language="javascript">
<!--
function pp ()
{
window.open ("new.htm", "new", "height=190,width=775,top=30,
left=30");
}
setTimeout ("pp()", 5000);
-->
</script>
```

举一反三

根据本实例, 读者可以:

开发新闻系统中的即时新闻、热点新闻等;

开发电子商务系统中的商品展示；  
开发在线论坛中的主题信息浏览。

## 实例003 通过按钮创建窗口

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\003

实例说明

在设计网页时，经常会在主页面中调用或创建其他页面，并对调用的页面指定其大小、位路 and 显示样式。实例在运行“index.htm”文件后，将打开“通过按钮创建窗口”页面，在该页面中单击“创建新窗口”按钮，将按指定的大小及位路打开“明日美食留言簿”页面。运行结果如图1.3所示。



图1.3 通过按钮创建窗口

技术要点

本实例主要是用window对象的open()方法来创建和调用其他窗口。

window对象open()方法的介绍可参见实例001。

## 实现过程

(1) 在页面中添加一个按钮，并在onClick（单击）事件中调用自定义函数pp()，代码如下。

```
<form name="form1" method="post" action="">  
  <input type="button" name="Button" value="创建新窗口"  
onClick="pp()">  
</form>
```

(2) 编写用于实现创建新窗口的JavaScript代码。

```
<script language="javascript">  
function pp() {  
  window.open('trans.htm','','toolbar,menubar,scrollbars,  
resizable,  
status,location,directories,copyhistory,height=400,width=  
500');  
}  
</script>
```

## 举一反三

根据本实例，读者可以：

在页面中动态打开其他网页；

打开一个没有状态栏的网页。

## [实例004 自动关闭的广告窗口](#)

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\01\004

实例说明

进入网站后，很多网站都会弹出新窗口（如广告等），多数窗口需要浏览者自行关闭，为了方便浏览者对页面中信息的浏览，笔者在这里运用了一种新的方法来解决弹出窗口自动关闭问题。

当用户浏览网站时，无需关闭弹出的新窗口，在页面运行超过一定的时间之后，该窗口便自动关闭，这大大方便了浏览者。运行本实例，将弹出一个新窗口显示广告信息，如图 1.4所示。页面运行5s后，弹出的广告窗口将自动关闭。



图1.4 自动关闭的广告窗口

### 技术要点

本实例主要应用 window 对象的 setTimeout()方法和 close()方法实现。window 对象的setTimeout()方法用于延迟执行某一操作。setTimeout()方法的相关语法格式请读者参见实例002。

### 实现过程

(1) 在打开广告窗口的页面中添加如下代码实现打开新窗口的功能。

```
<script language="javascript">
```

```
<!--  
    window.open("new.htm", "new", "height=230, width=250, top=10,  
left=20");  
-->  
</script>
```

(2) 在弹出的广告窗口中通过设置window对象的setTimeout()方法，实现窗口自动关闭，代码如下。

```
<body onload="window.setTimeout('window.close()', 5000)">
```

举一反三

根据本实例，读者可以：

在编写商务网站时应用此技术；

在开发信息资源网站应用此技术；

在编写教育信息网站应用此技术。

## 实例005 控制弹出窗口居中显示

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\005

实例说明

使用过 JavaScript 的 window.open()方法打开新窗口的读者都知道，在使用 window.open()方法打开新窗口时，新弹出的窗口默认是在屏幕左上端显示的，这样如果弹出的窗口比较小，一般不会引起浏览者的注意，如果让弹出的窗口居中显示就会很直观了。本实例将介绍如何控制弹出窗口的居中显示。运行本实例，单击页面下方的“TOP”超级链接，即可弹出居中显示的管理员登录窗口，如图1.5所示。



图1.5 控制弹出窗口居中显示

### 技术要点

本实例首先应用JavaScript中的window对象的open()方法，打开指定大小的新窗口，然后通过screen对象，获取屏幕的分辨率，再根据获取的值通过window对象的moveTo()方法，将新窗口移动到屏幕居中位路。由于window对象的open()方法在实例001中已经介绍，下面只介绍window对象的moveTo()方法和screen对象。

moveTo()方法将窗口移动到指定坐标(x, y)处，其语法格式如下。

```
window.moveTo(x, y)
```

其中，参数x, y表示窗口移动到的位路处的坐标值。

注意：moveTo()方法是Navigator和IE都支持的方法，它不属于W3C标准的DOM。

screen对象是JavaScript中的屏幕对象，反映了当前用户的屏幕设路。该对象的常用属性如表1.3所示。

表1.3 screen对象的常用属性

属 性	说 明
width	用户整个屏幕的水平尺寸，以像素为单位
height	用户整个屏幕的垂直尺寸，以像素为单位
pixelDepth	显示器的每个像素的位数
colorDepth	返回当前颜色设置所用的位数，1 代表黑白，8 代表 256 色，16 代表增强色，24/32 代表真彩色。8 位颜色支持 256 种颜色，16 位颜色（通常叫做“增强色”）支持大概 64000 种颜色，而 24 位颜色（通常叫做“真彩色”）支持大概 1600 万种颜色
availHeight	返回窗口内容区域的垂直尺寸，以像素为单位
availWidth	返回窗口内容区域的水平尺寸，以像素为单位

## 实现过程

(1) 在页面的适当位路添加控制窗口弹出的超级链接，本例中采用的是图片热点超级链接，关键代码如下。

```
<map name="Map">
  <area shape="rect" coords="82, 17, 125, 39"
href="#"onClick="manage()">
  <area shape="circle" coords="49, 28, 14">
</map>
```

(2) 编写自定义的JavaScript函数manage()，用于弹出新窗口并控制其居中显示，代码如下。

```
<script language="javascript">
function manage()
{
  var
hdc=window.open('Login_M.htm', '', 'width=322,height=206');
  width=screen.width;
  height=screen.height;
  hdc.moveTo((width-322)/2, (height-206)/2);
}
</script>>
```

(3) 最后设计弹出窗口页面Login\_M.htm，代码如下。

```

<script language="javascript">
function mycheck(myform)
{
if (myform.Manager.value=="")
{alert("请输入管理员! ");myform.Manager.focus();return;}
if(myform.PWD.value=="")
{alert("请输入密码! ");myform.PWD.focus();return;}
myform.submit();
}
</script>
<form method="post" action="" name="myform">
<table width="100%" border="0" cellspacing="-2"
cellpadding="-2">
<tr>
<td width="20" height="65">&nbsp;</td>
<td width="82">&nbsp;</td>
<td width="200">&nbsp;</td>
<td width="20">&nbsp;</td>
</tr>
<tr>
<td height="34">&nbsp;</td>
<td align="center" class="word">&nbsp;&nbsp;&nbsp;管理员: </td>
<td>
<input name="Manager" type="text" class="txt_grey"
id="Manager3"onKeyDown="if(event.keyCode==13) myform.PWD.
focus();" maxlength="20">
</td>

```

```

<td>&nbsp;</td>
</tr>
<tr>
<td height="34">&nbsp;</td>
<td align="center" class="word">&nbsp;&nbsp;&nbsp;密
&nbsp;&nbsp;&nbsp;码: </td>
<td>
<input name="PWD" type="password" class="txt_grey"
id="PWD" onKeyDown="if(event.keyCode==13)
mycheck(myform)"maxlength="20"></td>
<td>&nbsp;</td>
</tr>
</table>
</form>

```

举一反三

根据本实例，读者可以：

将弹出的用户登录窗口居中显示；

弹出居中显示的广告窗口；

将弹出的显示详细信息页居中显示。

## [实例006 弹出的窗口之Cookie控制](#)

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\01\006

实例说明

在很多网站中都应用了弹出窗口显示广告信息，每次访问网站都会弹出相同的广告窗口，久而久之就会让人产生厌倦感。为了解决该

问题，本实例将介绍一种只有第一次访问网站时才弹出广告窗口的方法。运行本实例，将弹出如图 1.6 所示的广告窗口，关闭浏览器后再次运行本实例，广告窗口将不再自动弹出。

### 技术要点

Cookie 是网站在访问者硬盘上存储的一些定制的信息段。通过浏览器，网页可以实现对Cookie的存储、获取和删除。Cookie的目的只有一个，即记录访问者的个体信息。在开始使用Cookie前，读者需要知道下面的规则。

浏览器可以存储的总Cookie数量不能超过300个，每个服务器不得超过20个（对于整个服务器，而不仅仅是用户自己的网页或网站）。存储容量也限制在每个 Cookie 4KB，所以不要试图在一个Cookie中存储过多的信息。默认情况下，一个Cookie可以在整个浏览器的运行期间存在；当用户退出浏览器后，Cookie 内容也就会消失。为了让一个Cookie 的持续时间超过一个浏览周期，可以设路失效日期。



图1.6 弹出的窗口之Cookie控制

注意：document.cookie 对象并不会直接连接到用户的硬盘上，它只是映射所有存储在那里的Cookie。

### 实现过程

(1) 建立一个预打开的placard.htm文件，通常该文件中放广告或公告信息，本例中放的是公告信息。

(2) 在需要弹出公告窗口的页面中，判断客户端浏览器中是否存在指定的Cookie，如果不存在，则弹出新窗口显示公告信息，否则不弹出公告窗口，代码如下。

```
<Script Language="JavaScript">
function openWindow() {
    window.open("placard.htm", "", "width=352,height=193")
}
```

```

function GetCookie(name) {
    var search = name+"=";
    var returnvalue = "";
    var offset,end;
    if(document.cookie.length>0) {
        offset = document.cookie.indexOf(search);
        if(offset !=-1) {
            offset+= search.length;
            end = document.cookie.indexOf(";", offset);
            if(end ==-1) end = document.cookie.length;
            returnvalue =
unescape(document.cookie.substring(offset, end));
        }
    }
    return returnvalue;
}

function LoadPop() {
    if(GetCookie("pop")== "")
    {
        openWindow();
        var today = new Date()
var time="Sunday, 1-jan-"+today.getFullYear()+1+" 23:59:59
GMC";
        document.cookie="pop=yes;expires="+time;
    }
}

</Script>

```

(3) 在需要弹出广告窗口页面的 onLoad 事件中调用弹出广告窗口的函数，程序代码如下。

```
<body onLoad="LoadPop()">
```

技巧：当浏览者将弹出的窗口关闭之后，再次进入该网站时，此窗口将不会再出现。如果想让该窗口再次出现，可以选择“工具”→“Internet 选项”，在弹出的对话框中“常规”选项卡中选中“删除Cookies”复选框，单击“确定”按钮即可。

举一反三

根据本实例，读者可以：

开发游戏网站中的系统公告；

开发新闻资讯类网站中的最新公告；

开发大型门户类网站中的弹出广告。

## [实例007 为弹出的窗口加入关闭按钮](#)

本实例是一个简化操作的程序

实例位置：光盘\mingrisoft\01\007

实例说明

使用Windows操作系统的用户都知道，单击IE窗口中的“关闭”按钮时，可以关闭窗口，但是这样有时并不是很直观。为了方便用户，可以在弹出的窗口中加入“关闭”按钮。运行本实例，单击“最新服务”超级链接，将弹出最新服务详细信息页面，如图 1.7 所示。在该图的右下角有一个“关闭”按钮，单击该按钮可以将当前窗口关闭。

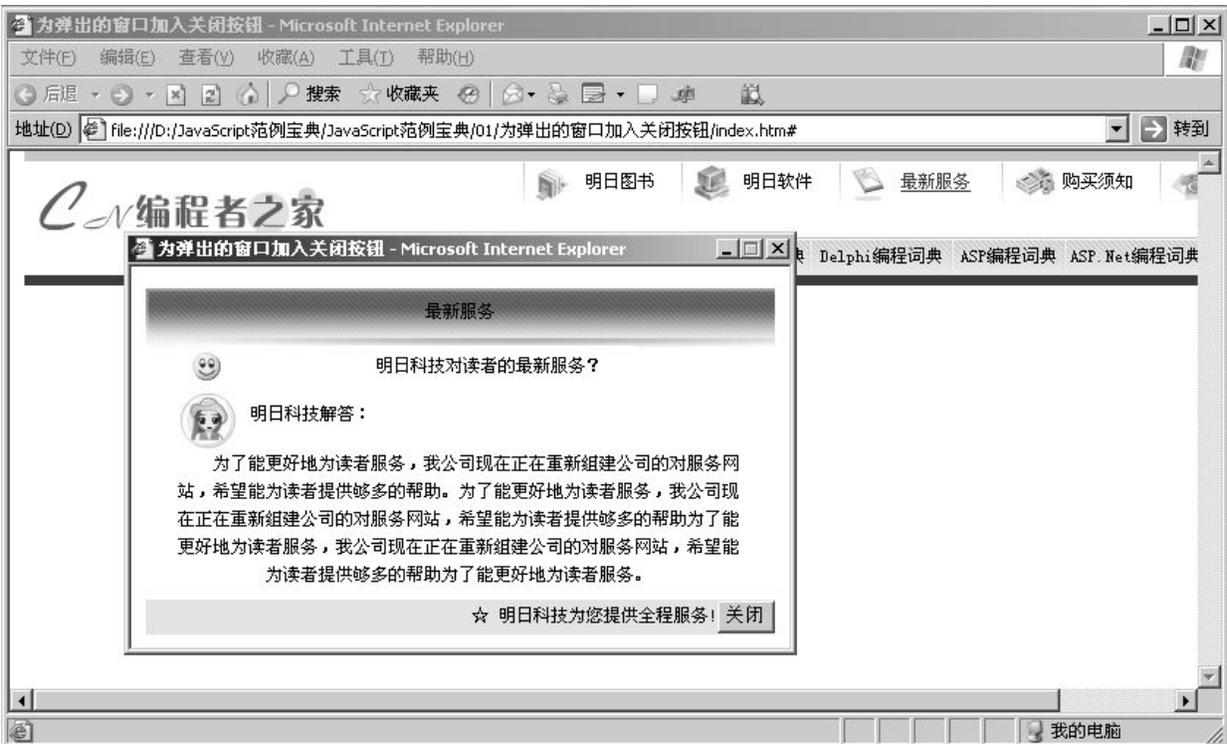


图1.7 为弹出的窗口加入关闭按钮

### 技术要点

本实例主要应用window对象的close()方法实现。close()方法的语法如下。

```
window.close();
```

功能：window对象的close()方法用于自动关闭浏览器窗口。

注意：应用window对象的close()方法关闭弹出的窗口时，和单击IE标题栏中的“关闭”按钮是一样的，但是如果关闭IE的主窗口，系统就会弹出“是否关闭此窗口”的确认对话框，只有单击“是”按钮后才可以关闭IE主窗口。

### 实现过程

(1) 创建提供“最新服务”超级链接的文件，注意该超级链接需要执行的操作应该是应用window.open()方法打开新窗口，代码如下。

```
<a href="#"  
onClick="window.open(' thinking.htm', '', 'width=450,height=300'  
)">最新服务</a>
```

(2) 在最新服务窗口中加入关闭按钮，关键代码如下。

```
<input name="Button" type="button" class="btn_grey"  
value="关闭" onClick="window.close();">
```

举一反三

根据本实例，读者可以：

实现制作办公自动化系统中的添加员工信息页面；

实现打开新窗口显示新闻详细信息页面；

在弹出的网站公告窗口中加入关闭按钮。

## 实例008 关闭弹出窗口时刷新父窗口

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\01\008

实例说明

关闭弹出窗口时刷新父窗口也可以说是关闭子窗口时自动刷新父窗口中的信息，即用户通过window对象的open()方法打开一个新窗口（子窗口），当用户在该窗口中进行关闭操作后，关闭子窗口的同时系统会自动刷新父窗口来实现实时更新页面。例如，运行本实例，在会议记录中，单击页面中的“会议记录”超级链接，将弹出会议记录页面，该页面通过单击“关闭”按钮将自动关闭，同时系统会自动刷新父窗口，如图1.8所示。

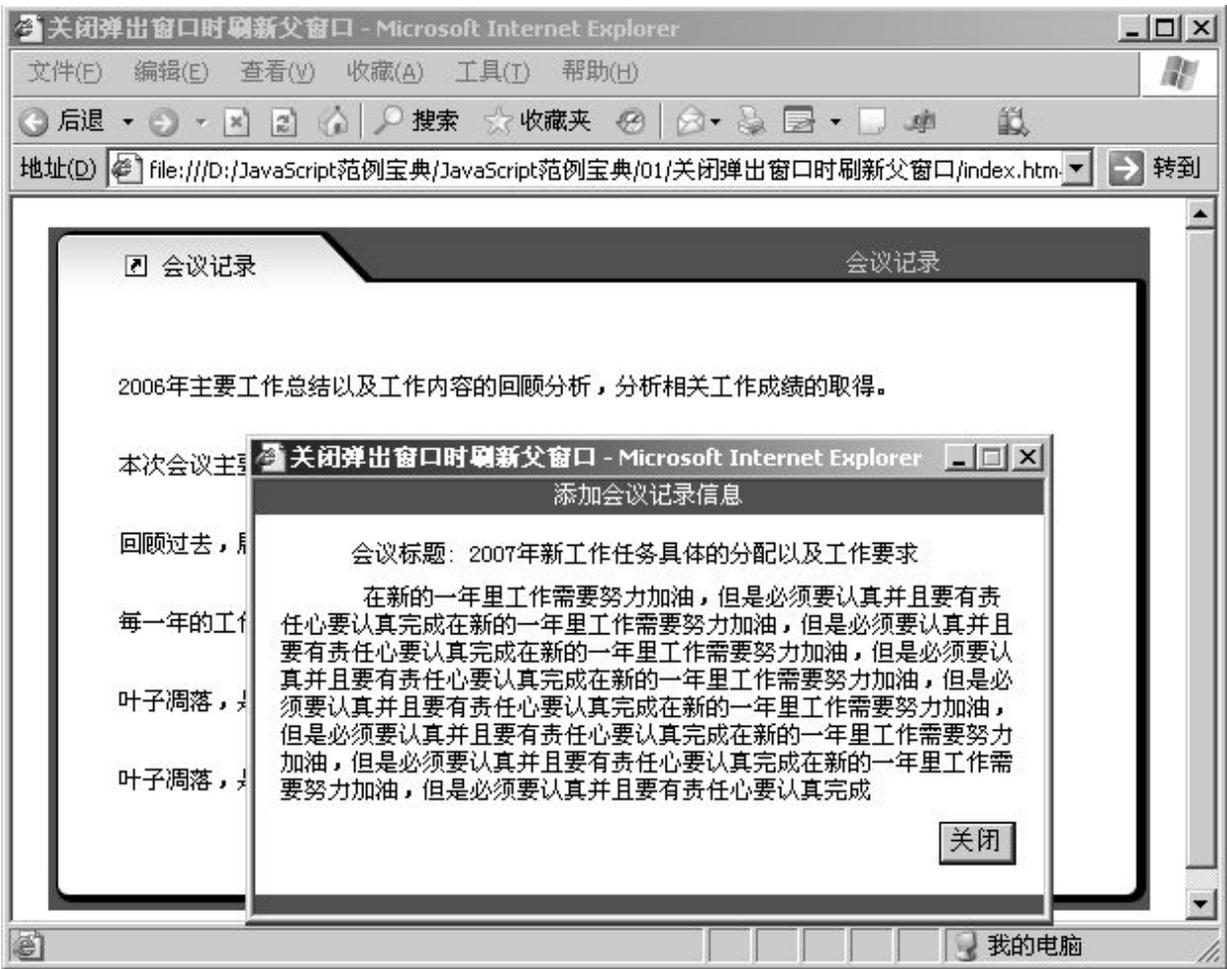


图1.8 关闭弹出窗口时刷新父窗口

### 技术要点

本实例主要应用`window.open()`语句打开新窗口,并在新窗口中应用`opener`属性,该属性返回一个引用,用于指定打开本窗口的窗口对象。

### 语法:

`window.opener`

`window.opener.方法`

`window.opener.属性`

功能: 返回的是一个窗口对象。`opener`属性与打开该窗口的父窗口相联系,当访问子窗口中`opener`属性时,返回的是父窗口,通过该

属性，可以使用父窗口对象中的方法和属性。

注意：如果窗口不是由其他窗口打开的，在Netscape中这个属性返回null，在IE中返回“未定义”（undefined）。undefined在一定程度上等于null。需要说明的是：undefined不是JavaScript常数，如果读者企图使用undefined，那就真的返回“未定义”了。

reload()方法相当单击浏览器上的“刷新”（IE浏览器）按钮或“Reload”（Netscape浏览器）按钮。

### 实现过程

(1) 制作用于显示会议信息列表的会议管理页面，在该页面中加入空的超级链接，并在其单击事件中加入JavaScript脚本，实现打开一个指定大小的新窗口，代码如下。

```
<a href="#"
onClick="Javascript:window.open('new.htm','','width=400,height=220')">
会议记录</a>
```

(2) 制作会议记录详细信息页面，在该页面中通过“关闭”按钮的onClick事件调用自定义函数Mycheck()，从而实现关闭弹出窗口时刷新父窗口，代码如下。

```
<input type="submit" name="Submit" value="关闭"
onclick="Mycheck();">
<script language="javascript">
function Mycheck()
{
alert("关闭子窗口!");
window.opener.location.reload();
//刷新父窗口中的网页
window.close();
```

```
//关闭当前窗窗口  
}
```

```
</script>
```

举一反三

根据本实例，读者可以开发：

办公自动化系统中，打开新的页面添加员工信息后，关闭该页面时需要刷新父窗口；

在电子商务网站中后台管理添加商品信息、修改商品信息和删除商品信息时，都需要刷新父窗口。

## 实例009 关闭IE主窗口时，不弹出询问对话框

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\009

实例说明

通常情况下，当使用JavaScript的window.close()语句关闭IE主窗口时，都会弹出一个“你查看的网页正在试图关闭窗口。是否关闭此窗口？”的询问对话框。在制作网络应用程序时，这种情况是我们不想看到的，那么如何才能屏蔽该对话框呢？本实例将介绍解决该问题的方法。运行本实例，如图 1.9所示，单击“关闭系统”超级链接后，该 IE窗口将被直接关闭。



图1.9 关闭IE主窗口时，不弹出询问对话框

### 技术要点

本实例主要应用 JavaScript的 window对象的 opener属性将要关闭的 IE窗口的打开窗口设路为 null，然后再通过 window 对象的 close( )方法将该窗口关闭，这样就不会弹出询问对话框了。

### 实现过程

在实例首页中，添加一个用于关闭窗口的超级链接，在该超级链接的 onClick 事件中添加控制窗口关闭的代码，具体代码如下。

```
<a href="#"  
onClick="window.opener=null;window.close();">• 关闭系统</a>
```

### 举一反三

根据本实例，读者可以：

开发进销存管理系统中的退出系统；

开发办公自动化管理系统中的关闭系统；

开发宽带影院中的打开全屏显示的主窗口。

## 1.2 弹出网页对话框

这里所说的网页对话框是指通过指定的脚本代码打开一个新的窗口，并且该窗口可以有返回值，网页对话框可以分为网页模式对话框和网页非模式对话框。下面将通过具体实例介绍网页对话框的应用。

### 实例010 弹出网页模式对话框

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\01\010

实例说明

在开发程序网站时，有时需要弹出网页模式对话框。例如本实例将弹出一个指定大小的网页模式对话框。运行结果如图1.10所示。



图1.10 弹出网页模式对话框

技术要点

本实例主要应用window对象的showModalDialog()方法，该方法用于弹出网页（模式）对话框，其语法格式如下。

```
variant = object.showModalDialog(sURL [, vArguments [, sFeatures]])
```

参数说明如下。

- sURL：指定URL 文件地址。
- vArguments：用于向网页对话框传递参数，传递参数的类型不限制，对于字符串类型，最大为4096个字符，但也可以传递对象，例如index.htm。
- sFeatures：可选项，窗口对话框的设路参数，主要参数如表1.4 所示。

表1.4 参数说明

参 数	说 明
dialogWidth:number	可选项，用于设置对话框的宽度
dialogHeight:number	可选项，用于设置对话框的高度
dialogTop:number	可选项，用于设置对话框窗口相对于桌面左上角的 TOP 位置
dialogLeft:number	可选项，用于设置对话框窗口相对于桌面左侧的 Left 位置
center:{yes no 1 0}	可选项，用于指定是否将对话框在桌面上居中，yes 1 为居中显示；no 0 为不居中显示，默认值为 yes
help: {yes no 1 0}	可选项，用于指定对话框窗口中是否显示上下文敏感的帮助图标，默认值是 yes
scroll: {yes no 1 0}	可选项，用于指定对话框是否出现滚动条
resizable: {yes no 1 0}	可选项，用于指定对话框窗口大小是否可变，默认值是 no
status: {yes no 1 0}	可选项，用于指定对话框是否显示状态栏

### 实现过程

(1) 制作外出登记的页面，在该页面中添加相应的表单或表单元素，在“登记”图片按钮的onClick事件中调用自定义JavaScript函数opendialog()，代码如下。

```

```

(2) 编写自定义JavaScript函数opendialog(), 用于打开网页对话框, 代码如下。

```
<script language="javascript">
function opendialog()
{
var
someValue=window.showModalDialog("new.htm","", "dialogWidth=640px;dialogHeight=423px;status=no;help=no;scrollbars=no")
}
</script>
```

(3) 制作用于添加外出登记的页面 new.htm, 通过该页面完成外出登记信息的添加。完整代码如下。

```
<body>
<table width="632" height="398" border="0"
cellpadding="0" cellspacing="0">
<tr>
<td width="611" height="398" valign="middle"
background="Images/waichuan_add.gif"><table width="525"
height="318" border="0" align="center" cellpadding="0"
cellspacing="0">
<tr>
<td height="34" colspan="8"><table width="597"
height="27" border="0" cellpadding="0"
cellspacing="0">
<tr>
<td width="61"><div align="center"
class="style1">姓名</div></td>
```

```

        <td width="192"><label>
            <input name="name1" type="text" id="name1">
        </label></td>
        <td width="69"><div align="center"
class="style1">所属部门</div></td>
        <td width="275"><label>
            <input name="department" type="text"
            id="department">
        </label></td></tr>
    </table></td></tr>
<tr>
    <td width="78" height="249"><span class="style1">
外出原因</span></td>
    <td colspan="7"><label>
        <textarea name="content" cols="70" rows="12"
wrap="physical" id="content"></textarea>
    </label></td></tr>
<tr>
    <td height="22" class="style1">外出时间</td>
    <td width="32" height="22" class="STYLE2"><label>
</label></td>
    <td width="127" height="22" class="STYLE2"><input
name="time1" type="text" id="time1" size="20"></td>
    <td width="7" class="STYLE2">&nbsp;</td>
    <td width="26" height="22" class="STYLE2"></td>
    <td width="19" height="22" class="style1">至</td>

```

```
<td width="126" class="STYLE2"><label>
    <input name="time2" type="text" id="time2"
size="20">
</label></td>
    <td width="182" class="STYLE2">&nbsp;</td>
</tr>
<tr>
    <td height="13" colspan="8"
valign="bottom">&nbsp;</td>
</tr></table></td>
</tr>
</table>
</body>
```

### 举一反三

根据本实例，读者可以：

在编写人力资源管理系统时，为员工指定所属部门；

在编写校园管理系统时，弹出网页对话框选择班级或专业；

在客房预订系统中，弹出网页对话框选择客户信息。

## 实例011 弹出全屏显示的网页模式对话框

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\01\011

### 实例说明

在设计网页时，对于一些只有指定用户才可以看到的信息，可以将其以全屏显示的网页（模式）对话框展示给用户，这样可以保证用

户看后能及时将其关闭。例如，在明日科技编程词典网站中的“购买须知”就是只有登录用户才可以查看。用户登录后，“购买须知”超级链接方可使用，单击“购买须知”超级链接，即可进入如图1.11所示的“购买须知”页面，这时的“购买须知”页面是以全屏网页（模式）对话框形式进行显示的，用户只有关闭该页面后，才可以查看网站的其他内容。



图1.11 弹出全屏显示的网页模式对话框

### 技术要点

本实例主要应用screen对象的width、height属性和window对象的showModalDialog()方法实现，在实例010中已经介绍了window对象showModalDialog()方法，这里不再赘述。其实还有一种方法也可以打开网页对话框，即showModelessDialog()方法。

使用showModalDialog()与showModelessDialog()方法的区别在于，showModalDialog()打开的网页对话框为模式窗口，落在父窗口上，必须关闭才能访问父窗口；而showModelessDialog()方法打开的对话框是无模式窗口，打开后不必关闭也可以访问父窗口或其他窗口。

### 实现过程

(1) 创建提供“购买须知”超级链接的页面，在该页面中添加空的超级链接，在该超级链接的onClick事件中调用自定义JavaScript函数opendialog()。关键代码如下。

```
<a href="#" onClick="opendialog()">购买须知</a>
```

(2) 编写自定义JavaScript函数opendialog()，用于打开全屏显示的网页对话框，程序代码如下。

```
<script language="javascript">
function opendialog()
{
var width=screen.width;
var height=screen.height;
window.showModalDialog("new.htm","", "dialogWidth="+width+
"px;dialogHeight="+height+"px;status=no;help=no;scrollbars=no")
}
</script>
```

(3) 创建用于全屏显示的页面，在该页面中添加所要显示的信息。程序代码如下。

```
<table width="96%" cellpadding="0" cellspacing="0"
bordercolor="#FFFFFF" border="1" bordercolordark="#FFFFFF"
bordercolorlight="#999999" style="padding:5px">
```

```

<tr bgcolor="#EEEEEE">
  <td width="9%" height="27" class="style2">软件版本 </td>
  <td width="16%" align="center" class="style2">功能</td>
  <td width="65%" align="center" class="style2">享受服务
</td>
  <td width="10%" align="center" class="style2">价格(元)
</td>
</tr>
<tr>
  <td height="25" align="center" class="style2">标准版</td>
  <td align="center" class="style2">详细功能</td>
  <td class="style2">一年两次升级</td>
  <td align="center" class="style2">128</td>
</tr>
<tr>
  <td height="25" align="center" class="style2">开发版</td>
  <td align="center" class="style2">详细功能</td>
  <td class="style2">每月升级, 每月赠送最新软件, 编程锦囊。
全年提供30项目源程序的有限服务支持。</td>
  <td align="center" class="style2">568</td>
</tr>
<tr>
  <td height="25" align="center" class="style2">专业版</td>
  <td align="center" class="style2">详细功能</td>
  <td class="style2">每月升级, 每月赠送最新软件, 编程锦囊。
全年提供60项目源程序的有限服务支持。</td>
  <td align="center" class="style2">1998</td>

```

```
</tr>
<tr>
<td height="25" align="center" class="style2">企业版</td>
<td align="center" class="style2">详细功能</td>
<td class="style2">每月升级，每月赠送最新软件，编程锦囊。
提供一年全程技术支持服务。</td>
<td align="center" class="style2">6998</td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

实现弹出的网站后台管理员登录窗口；

实现当用户执行非法操作后，弹出一个大大的警告框。

## 实例012 网页拾色器

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\01\012

实例说明

在动态网站中，经常会用到要求用户可以在客户端应用自定义颜色来显示某些信息的情况，如留言文字的颜色。这可以在网页中加入一个网页拾色器实现。运行本实例，单击页面中的颜色块，将打开网页拾色器，如图1.12所示。当鼠标光标移动到指定色块时，将显示该色块的颜色值；当用户单击指定色块时，就会关闭该网页对话框，并将选择的颜色设路为颜色块的背景色。



图1.12 网页拾色器

### 技术要点

在本实例中仅使用216种浏览器安全的颜色，即所谓Netscape色块。这216种颜色分别代表0、51、102、153和204这5个颜色值以及每一种原色（即红、绿、蓝）。这些十进制数值对应的十六进制数分别为：0x00、0x33、0x66、0x99、0xCC和0xFF。在HTML的颜色属性中，黑色是#000000，纯红色是#FF0000，纯绿色就是#00FF00，纯蓝色是#0000FF，而白色是#FFFFFF。在实现网页拾色器时，需要应用JavaScript的数组。创建数组可以有以下3种方法。

(1) 无参数调用。语法格式如下。

```
var h = new Array();
```

(2) 指定数组前n个元素的值。语法格式如下。

```
var h = new Array(arglist);
```

其中，参数arglist是一个用逗号隔开的值表，这些值用于给Variant所包含的数组的各元素赋值。如果不提供参数，则创建一个长度为0的数组。

(3) 指定具有的元素个数。语法格式如下。

```
var h = new Array(n);
```

其中，参数n是指定数组的长度。在JavaScript中，数组的第一个元素的下标值为0，所以n的值为数组的最大下标值加1。

注意：没有作为数组声明的Variant也可以表示数组。除了长度固定的字符串以及用户定义类型之外，Variant 变量可以表示任何类型的数组。尽管一个包含数组 Variant 和一个元素为Variant类型的数组在概念上有所不同，但对数组元素的访问方式是相同的。

### 实现过程

(1) 创建需要调用网页拾色器的页面，在该页面中添加一个文本框，将文本框设为只读，同时将文本框的背景颜色设为黑色，并在该文本框的onClick事件中调用自定义的JavaScript函数colorpick()，代码如下。

```
<input name="color" type="text" id="color" size="3"
readonly="yes" style="background-
color:#000000"onClick="colorpick(this);">
```

(2) 编写自定义的JavaScript函数colorpick()，用于打开网页对话框调用网页拾色器，并用网页拾色器的返回值设给文本框的背景颜色，代码如下。

```
<script language="javascript">
function colorpick(field)
{
    var rtn =
window.showModalDialog("color.htm","", "dialogWidth:220px;di
alogHeight:200px;status:no;help:no;scrolling=no;
scrollbars=no");
    if(rtn!=null)
```

```
field.style.background=rtn;
return;
}
</script>
```

(3) 制作网页拾色器页面color.htm。关键代码如下。

```
<script language="JavaScript">
var h = new Array(6)
h[0] = "FF";
h[1] = "CC";
h[2] = "99";
h[3] = "66";
h[4] = "33";
h[5] = "00";
function action(RGB)
{
    parent.window.returnValue="#" + RGB;
    window.close();
}
function Mcell(R, G, B)
{
    document.write(' <td bgcolor="#" + R + G + B + "' >');
    document.write(' <a href="#" onClick="action(\'' +
(R + G + B) + '\')"' >');
    document.write(' <img border=0 height=12 width=12 \' "'
alt='\ #' + R + G + B + '\') >');
    document.write(' </a>');
    document.write(' </td>');
```

```

}
function Mtr(R, B)
{
    document.write('<tr>');
    for (var i = 0; i < 6;++i)
    {
        Mcell(R, h[i], B);
    }
    document.write('</tr>')
}
function Mtable(B)
{
    document.write('<table cellpadding=0 cellspacing=0
border=0>');
    for (var i = 0; i < 6;++i) {
        Mtr(h[i], B);
    }
    document.write('</table>');
}
function Mcube() {
    document.write('<table cellpadding=0 cellspacing=0
border=0><tr>');//
    for (var i = 0; i < 6;++i)
    {
        if(i%3==0) {
            document.write('<tr>');
        }
    }
}

```

```
document.write(' <td bgcolor="#FFFFFF">');  
Mtable(h[i])  
document.write(' </td>');  
}  
if(i%3==0) {  
document.write(' </tr>');  
}  
document.write(' </tr></table>');  
}  
Mcube()  
</script>
```

举一反三

根据本实例，读者可以：

实现具有UBB功能的在线论坛；

实现网站聊天室；

实现网页换肤。

## [实例013 日期选择器](#)

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\01\013

实例说明

在动态网站的人机交互页面中，经常会用到要求用户输入日期的情况，如果让用户手动输入，不但操作不方便，而且日期格式不好控制。为了解决此问题，可以在页面中加入日期选择器。运行本实例，单击“日期时间图片”超级链接，将弹出具有日期选择功能的网页对

对话框，如图1.13所示。当用户单击需要的日期后，日期选择网页对话框将自动关闭，并将选择结果填入“请假时间”文本框中。

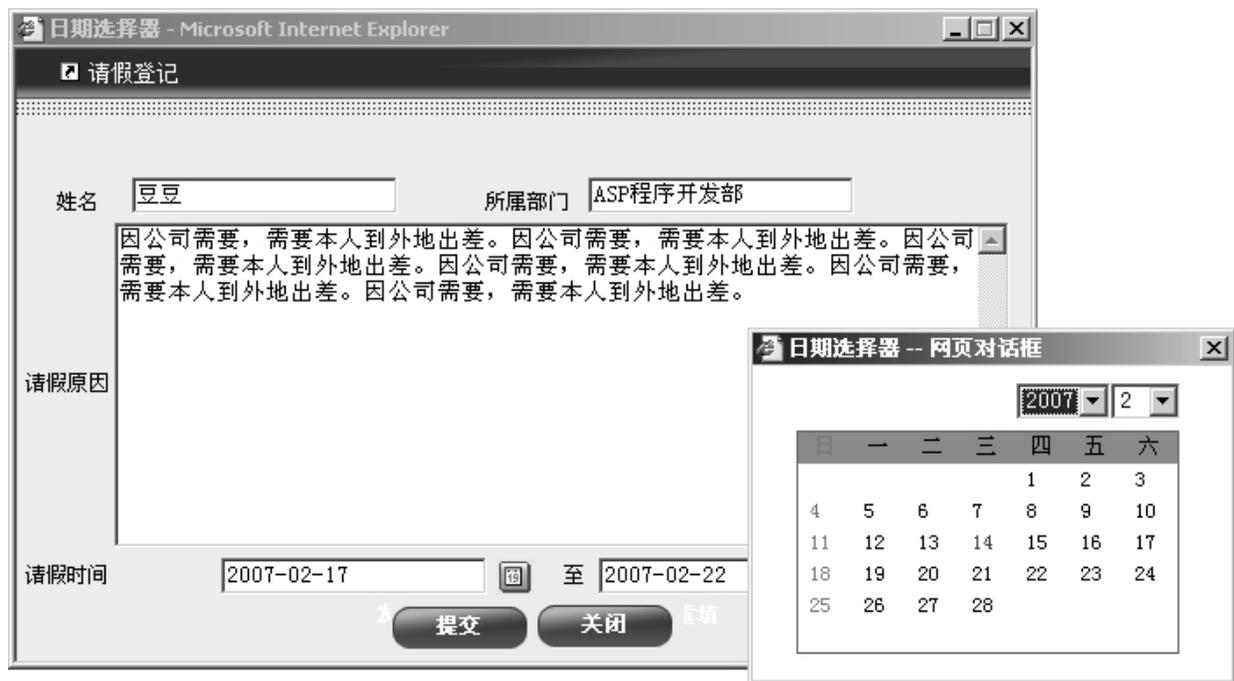


图1.13 日期选择器

### 技术要点

本实例主要应用window 对象的showModalDialog()方法实现，关于showModalDialog()方法的详细介绍请读者参见实例010。

### 实现过程

(1) 创建需要调用日期选择器的页面，在该页面中添加一个标识日期时间的图片，在该图片的onClick事件中调用自定义的JavaScript函数loadCalendar()，代码如下。

```

```

(2) 编写自定义的JavaScript函数loadCalendar()，用于打开网页对话框调用日期选择器，代码如下。

```
<script language="javascript">
function loadCalendar(field)
```

```

    {
        var rtn =
window.showModalDialog("calendar.htm","", "dialogWidth:280px
;dialogHeight:250px;status:no;help:no;scrolling=
no;scrollbars=no");
    if(rtn!=null)
    field.value=rtn;
    return;
    }
</script>

```

(3) 制作日期选择器页面calendar.htm。首先，添加两个下拉列表，用于选择要显示的年份和月份，然后制作一个7行7列的表格，其中最顶行用于标记星期，其他行显示日期。关键代码如下。

```

<body onLoad="load(document.calendar)" topmargin="0">
<center>
<form name="calendar">
<table border="0" cellpadding="0" cellspacing="0">
<tr><td colspan="3" height="24"></td></tr>
<tr>
<td width="205" align="right" VALIGN="MIDDLE"
nowrap="nowrap">
<select name="yearList"
onChange="display_month(this.form)"></select></td>
<td width="65" nowrap="nowrap" align="left">
<select name="monthList" size="1"
onChange="display_month(this.form)">
<option value="1">1</option>

```

```

<option value="2">2</option>
<option value="3">3</option>
<option value="4">4</option>
<option value="5">5</option>
<option value="6">6</option>
<option value="7">7</option>
<option value="8">8</option>
<option value="9">9</option>
<option value="10">10</option>
<option value="11">11</option>
<option value="12">12</option></select></td>
<td width="10"></td></tr>
<tr><td colspan="3" height="6"></td></tr>
<tr><td colspan="3"><table border="0" cellpadding="1"
cellspacing="0" align="center">
<tr>
<td bgcolor="#82664F"><table border="0" cellpadding="0"
cellspacing="0">
<tr bgcolor="#82664F" height="18">
<td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF"><font color="#FF0000">日</font></td>
<td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">一</td>
<td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">二</td>
<td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">三</td>

```

```

        <td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">四</td>
        <td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">五</td>
        <td width="31" align="center" nowrap="nowrap"
BGCOLOR="#0099FF">六</td></tr>
        <tr bgcolor="#FFFFFF" height="18">
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouseOver="this.
style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>

```

```

    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr>
    <tr bgcolor="#FFFFFF" height="18">
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>

```

```

        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr>
        <tr bgcolor="#FFFFFF" height="18">

```

```
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouseOver="this.
style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
<td align="center"><input type="text" size="2"
name="dayBox" readOnly
```

```
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr>
    <tr bgcolor="#FFFFFF" height="18">
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
```

```

Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr>
    <tr bgcolor="#FFFFFF" height="18">
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse

```

```

Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
    <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr>

```

```

<tr bgcolor="#FFFFFF" height="18">
  <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
  <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
  <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
  <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
  <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>

```

```

        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouseOver="this.
style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td>
        <td align="center"><input type="text" size="2"
name="dayBox" readOnly
onClick="javascript:CheckDate(this.value);" onMouse
Over="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' "></td></tr></table>
</td></tr>
</table></td></tr></table></form></center></body>

```

(4) 最后，编写实现日期选择功能的JavaScript代码，程序代码如下。

```

<script language=javascript>
var monthNames = new Array ( "",
"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12" );
var endDay = new Array ( 0, 31, 28, 31, 30, 31, 30, 31,
31, 30, 31, 30, 31 );
var dayNow = 0;
var monthNow = 0;
var yearNow = 0;
function load ( form )
{
set_month_year_now ();
var found= false;
for ( varmonth=0;month<form.monthList.length;month++)

```

```

if ( form.monthList[month].text==monthNames[monthNow] ) {
form.monthList[month].selected= true;
found= true;}
if ( !found) {error ();
return;}
var found= false;
for ( var year=0;year<form.yearList.length;year++)
if ( form.yearList[year].text==yearNow ) {
form.yearList[year].selected= true;
found= true;}
if ( !found) {
error ();
return;}
display_month ( form );}
functionpreceding_month ( form ) {
varmonth_selected= form.monthList.selectedIndex;
var year_selected= form.yearList.selectedIndex;
if ( !month_selected&& !year_selected ) {error ();
return;}
if (month_selected>0 )
month_selected--;
else {
month_selected=11;
year_selected--;}
form.monthList[month_selected].selected= true;
form.yearList[year_selected].selected= true;
display_month ( form );}

```

```

function following_month ( form ) {
var month_selected= form.monthList.selectedIndex;
var year_selected= form.yearList.selectedIndex;
if (month_selected>= ( form.monthList.length-1 )&&
year_selected>= ( form.yearList.length-1 ) ) {error ();
return;}
if (month_selected<11 )
month_selected++;
else {
month_selected=0;
year_selected++;}
form.monthList[month_selected].selected= true;
form.yearList[year_selected].selected= true;
display_month ( form );}
function set_month_year_now () {
var form=document.calendar;
var now=newDate ();
monthNow=now.getMonth ()+1;
yearNow= now.getYear ();
dayNow= now.getDate ();
yearNow= ( yearNow<100 ) ?yearNow+1900 :yearNow;
var count=0
for (var i= yearNow-103; i<yearNow+50; i++) {
eval ("form.yearList.options[count]=newOption(' "+i+"', ' "
+i+"')");
count++;}
form.yearList.selectedIndex=103;

```

```

form.yearList.length= count;}
function error () {
alert ( "超出范围!" );}
functiondisplay_month ( form ) {
varmonth= form.monthList.selectedIndex+1;
var year=parseInt ( form.yearList.options[
form.yearList.selectedIndex].text );
var start_day= start_day_in_month ( year,month );
var count=0;
for ( var row=0; row<6; row++) {
for ( var col=0; col<7; col++){
if ( row==0&& col< ( start_day-1 ) )
var day= "";
else if ( count < endDay[month] )
day =++count;
else
day = "";
form.dayBox[(row*7)+col].style.display = "";
form.dayBox[(row*7)+col].style.color = "black";
if (day == "") {
form.dayBox[(row*7)+col].style.display = "none";
} else {
form.dayBox[(row*7)+col].value = day;
if (col%7 == 0) form.dayBox[(row*7)+col].style.color =
"red";
if (yearNow == year && monthNow == month && dayNow ==
day) form.dayBox[(row*7)+col].style.color = "blue";;}

```

```

}
}
}
function start_day_in_month ( year, month )
{
var day, daynum, ndays, mnum;
sday = start_day_in_year ( year );
endDay[2] = ( year %4 ) ? 28 : 29;
if ( month == 1 )
daynum = sday;
else {
ndays = sday;
for ( mnum=2; mnum<month+1; mnum++)
ndays = ndays+endDay[mnum-1];
daynum = ndays %7;}
daynum = (!daynum) ? 7 : daynum;
return (daynum);}
function start_day_in_year ( year )
{
var y, m, d;
var n;
y = year-1; m = 13; d = 1;
n = d+2 * m+( Math.floor ( ( 0.6+(m+1) ) ) +y );
n = n+Math.floor ( ((y/4)-Math.floor ( (y/100)
)+Math.floor ( ( y/400 ) ) )+2 ;
n = Math.floor ( ( (n/7-Math.floor ( (n/7) ) ) * 7+0.5 )
);

```

```

return (n+1);}
function CheckDate(strDay)
{
var docFrm = document.calendar;
var choice_daynum = 0;
var current_daynum = 0;
var day_temp;
if (strDay != "")
{
var strY = docFrm.yearList.value;
var strM = docFrm.monthList.value;
var curr_y = new String(yearNow);
var curr_m = new String(monthNow);
var curr_d = new String(dayNow);
if (curr_m.length == 1) curr_m = "0"+curr_m;
if (curr_d.length == 1) curr_d = "0"+curr_d;
current_daynum = new Number(curr_y+curr_m+curr_d) ;
if (strM.length == 1) strM = "0"+strM;
if (strDay.length == 1) strDay = "0"+strDay;
choice_daynum = new Number(strY+strM+strDay);
parent.window.returnValue= strY+"-"+strM+"-"+strDay; //

```

将选择的日期传递到父窗口中

```

parent.window.close();
}
return false;}

```

```
</script>
```

举一反三

根据本实例，读者可以：

开发电子商务管理系统中的订单查询模块；

开发企业进销存管理系统中的数据查询模块；

开发用户注册模块中的生日选择功能。

## 1.3 窗口的动画效果

在浏览网页时，许多窗口在打开时，都显示了各种各样的动画效果，这样可以在浏览页面时，给人一种动态、新颖的效果。下面将通过几个典型实例进行详细讲解。

### 实例014 页面自动滚动

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\01\014

实例说明

本实例实现在打开页面，当页面出现纵向滚动条时，页面中的内容将从上向下进行滚动，当滚动到页面最底端时停止。运行结果如图1.14所示。



图1.14 页面自动滚动

技术要点

本实例主要用window对象的scroll()方法指定窗口的当前位路。  
下面对scroll()方法进行详细说明。

scroll()方法的语法格式：

```
scroll(x, y);
```

参数说明如下。

- x: 屏幕的横向坐标。
- y: 屏幕的纵向坐标。

功能：指定窗口滚动坐标的位路。

实现过程

编写用于实现页面自动滚动的JavaScript代码，代码如下。

```
<script language="JavaScript">  
var position = 0;  
function scroller() {  
    if (true) {  
        position++;  
        scroll(0, position);  
        clearTimeout(timer);  
        var timer = setTimeout("scroller()", 10);  
    }  
}  
scroller();  
</script>
```

举一反三

根据本实例，读者可以：

使页面中的文字上下循环滚动；

使页面中的文字左右循环滚动。

## 实例015 打开窗口特殊效果

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\015

实例说明

本实例在打开 index.htm 文件后，将弹出“打开窗口特殊效果”页面，在该页面中单击超级链接将打开“明日美食留言簿”页面，在屏幕的左上角会弹出一个窗口，并动态改变窗口的宽度和高度，当窗口的高度和屏幕的高度一致时，继续添加窗口的宽度，直到与屏幕大小相同为止。运行结果如图1.15、图1.16所示。

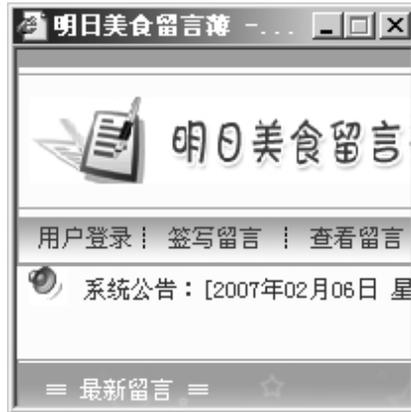


图1.15 打开窗口时的效果



图1.16 放大后的窗口效果

### 技术要点

本实例主要是用 window 对象的 open() 方法来打开一个已有的窗口，用 screen 对象的 availHeight 属性来获取屏幕可工作区域的高度，用 moveTo() 和 resizeTo() 方法来指定窗口的位路及大小，并用 resizeBy() 方法使窗口逐渐变大，直到窗口大小与屏幕的工作区大小相同。下面对 window 对象的 moveTo()、resizeTo() 和 resizeBy() 方法分别进行介绍。

#### (1) moveTo() 方法

moveTo() 方法的语法格式：

```
window.moveTo(x, y)
```

参数说明如下。

- x: 窗口左上角的x 坐标。
- y: 窗口左上角的y坐标。

功能：将窗口移动到指定坐标(x, y)处。

#### (2) resizeTo() 方法。

resizeTo() 方法的语法格式：

`window.resizeTo(x, y)`

参数说明如下。

- x: 窗口的水平宽度。
- y: 窗口的垂直宽度。

功能: 将当前窗口改变成(x, y)大小, x、y分别为宽度和高度。

(3) `resizeBy()`方法。

`resizeBy()`方法的语法格式:

`window.resizeBy(x, y)`

参数说明如下。

- x: 放大或缩小的水平宽度。
- y: 放大或缩小的垂直宽度。

功能: 将当前窗口改变到指定的大小(x, y), 当x、y的值大于0时为扩大, 小于0时为缩小。

实现过程

(1) 在页面中添加一个链接标记[<a>](#), 用于调用自定义函数`go1()`, 代码如下。

```
<a href="javascript:go1()">打开明日美食留言簿</a>
```

(2) 编写用于实现打开窗口特殊效果的JavaScript代码。

自定义函数`go1()`, 用于打开指定的窗口, 并设置其位置和大小, 代码如下。

```
<script language=JavaScript>
```

```
var winheight, winsize, x;
```

```
function go1() {
```

```
    winheight=100;
```

```
    winsize=100;
```

```
    x=5;
```

```
    win2=window.open("melody.htm", "", "scrollbars='no'");
```

```
win2.moveTo(0,0);  
win2.resizeTo(100,100);  
go2();  
}
```

自定义函数go2(), 用于动态改变窗口的大小, 代码如下。

```
function go2()  
{  
  if (winheight>=screen.availHeight-3)  
    x=0  
  win2.resizeBy(5,x)  
  winheight+=5  
  winsize+=5  
  if (winsize>=screen.width-5) {  
    winheight=100  
    winsize=100  
    x=5  
    return  
  }  
  setTimeout("go2()", 50)  
}  
</script>
```

举一反三

根据本实例, 读者可以:

指定窗口显示的位置及大小;

用按钮动态改变窗口的大小。

## 实例016 动态显示窗口

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\01\016

实例说明

本实例实现在窗口打开时，将窗口设为最小的宽度，并使窗口的高度逐渐增加，当窗口的高度与屏幕的高度一致时，使窗口的宽度不断增加，直到与屏幕的宽度相一致。运行结果如图1.17、图1.18所示。



图1.17 窗口高度不断增加



图1.18 窗口变大后的效果

### 技术要点

本实例主要应用了screen对象的availWidth和availHeight属性来获得当前屏幕的宽度和高度，并用resizeTo()方法来自动增加窗口的高度和宽度。

resizeTo()方法已在实例015中进行详细讲解，这里不再赘述。

### 实现过程

(1) 编写用于实现动态显示窗口的JavaScript代码。

```
<script language="JavaScript">
function expandingWindow()
{
    var heightspeed = 2;
    var widthspeed = 7;
    var leftdist = 0;
    var topdist = 0;
    var winwidth = window.screen.availWidth-leftdist;
    var winheight = window.screen.availHeight-topdist;
    for (sizeheight = 1; sizeheight < winheight;
sizeheight+= heightspeed) {
        self.resizeTo("1", sizeheight);
    }
    for (sizewidth = 1; sizewidth < winwidth; sizewidth+=
widthspeed) {
        self.resizeTo(sizewidth, sizeheight);
    }
}
</script>
```

(2) 在<body>标记的窗口载入 (onload) 事件中调用自定义函数 expandingWindow()。代码如下：

```
<body onload="expandingWindow()">
```

举一反三

根据本实例，读者可以：

设置打开窗口的大小与屏幕大小相同；

使窗口出现下拉式打开效果。

## 实例017 慢慢放大的窗口

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\01\017

实例说明

本实例实现在窗口打开时，将窗口按照指定的大小放在屏幕的中间，并使窗口不断放大，直到窗口大小与屏幕大小相同为止。运行结果如图1.19、图1.20所示。



图1.19 窗口放大前的效果

技术要点

本实例主要应用screen对象的availWidth和availHeight属性来获得屏幕工作区域的宽度和高度，并用window对象的resizeTo( )方法来指定窗口的位路，用window对象的moveTo( )方法使窗口不断变大。

window对象的resizeTo( )和moveTo( )方法的介绍可参见实例015中的技术要点。

## 实现过程

(1) 编写用于实现慢慢放大的窗口的JavaScript代码。



图1.20 窗口慢慢放大的效果

自定义变量，用于获取屏幕的大小，并将窗口放在屏幕指定的位  
置。

```
<script language="JavaScript">
var sun=400;
var screenW=window.screen.availWidth;
var screenH=window.screen.availHeight;
window.resizeTo((screenW-sun), (screenH-sun));
var initialW=screenW-sun;
var initialH=screenH-sun;
var x=((screenW)-initialW)/2;
var y=((screenH)-initialH)/2;
```

```

window.moveTo(x, y);
var timer;
自定义函数blow(), 用于使窗口在屏幕中逐渐变大。
function blow() {
    if (initialH<(window.screen.height-30)) {
        initialW=initialW+2;
        initialH=initialH+2;
        window.resizeTo(initialW, initialH);
        timer=setTimeout("blow()", 50);
        x=x-1;
        y=y-1;
        window.moveTo(x, y);
    }
    else {window.clearTimeout(timer)}
}
</script>

```

(2) 在<body>标记的窗口载入 (onload) 事件中调用自定义函数 blow(), 代码如下。

```
<body onLoad=" blow()">
```

举一反三

根据本实例，读者可以：

将窗口慢慢缩小；

使窗口在右下角慢慢放大。

## [实例018 下降式浏览器](#)

本实例是一个定义窗口的程序

实例位置：光盘\mingrisoft\01\018

## 实例说明

本实例是在窗口打开时，将整个窗口放在屏幕的最上面，使窗口无法在屏幕中进行显示，然后动态地使窗口进行下移，直到窗口显示在屏幕的左上角为止。运行结果如图1.21所示。



图1.21 下降式浏览器

## 技术要点

本实例主要应用了 screen 对象的 availHesht 属性来获得当前屏幕工作区的高度，并用window对象的moveBy()方法使窗口自动下移。下面对window对象的moveBy()方法进行详细介绍。

moveBy()方法的语法格式：

```
window.moveBy(x, y)
```

参数说明如下。

- x：水平位移量。

● y: 垂直位移量。

功能: 按照指定位移量设置窗口的大小。

实现过程

(1) 编写用于实现下降式浏览器的JavaScript代码。

```
<script language="JavaScript">
function drop() {
    var heig=window.screen.availHeight;
    if(self.moveBy) {
        self.moveBy (0, -heig);
        for(i = Math.floor(heig/3); i > 0; i--) {
            self.moveBy(0, 3);
        }
    }
}
</script>
```

(2) 在<body>标记的窗口载入 (onload) 事件中调用自定义函数 drop()。

```
<body onLoad="drop()">
```

举一反三

根据本实例, 读者可以:

制作一个从左页慢慢移出的浏览器;

制作一个飞出的浏览器。

## 实例019 旋转的窗口

这是一个可以启发思维的实例

实例位置: 光盘\mingrisoft\01\019

## 实例说明

本实例是在窗口打开时，将窗口的左上角的位置以圆形的轨迹进行移动，使窗口产生旋转的效果。运行结果如图1.22所示。



图1.22 旋转的窗口

## 技术要点

本实例主要使用公式  $r * \text{Math}.\cos((i * \text{Math}.\text{PI}) / 180)$  和  $r * \text{Math}.\sin((i * \text{Math}.\text{PI}) / 180)$  来计算出圆形轨迹，其中， $r$ 表示圆的半径， $i$ 表示圆周（在这里 $i$ 的值为 $0 \sim 360$ ，通过 $i$ 值可以计算出圆各个点的位路），然后使用window对象的`moveTo()`方法使窗口进行旋转。

`moveTo()`方法的介绍可参见实例015中的技术要点。

## 实现过程

编写用于实现旋转窗口的JavaScript代码。

```
<script language="javascript">
```

```
var a=275;
```

```
var b=275;
```

```
var pp;
```

```
var r=60;
```

```
var x=y=0;
window.resizeTo(200,200);
function eddy(i) {
    var ob=document.all("tt");
    x = r*Math.cos((i*Math.PI)/180)+a;
    y = r*Math.sin((i*Math.PI)/180)+b;
    window.moveTo(x,y);
    i=i+1;
    if (r>100) {window.clearTimeout(pp);}
    else{
        if (i>360) {i=0; }
        pp=setTimeout("eddy("+i+")",10);
    }
}
eddy(0);
</script>
```

举一反三

根据本实例，读者可以：

使窗口以不规则的状态进行移动；

使窗口以阶梯状进行向下移动。

## 实例020 移动的窗口

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\01\020

实例说明

本实例实现在窗口打开时，将窗口放在屏幕的左上角，并将窗口从左到右以随机的角度进行移动，当窗口的外边框碰到屏幕四边时，窗口将进行反弹。运行结果如图1.23、图1.24所示。



图1.23 窗口移动前的效果



图1.24 窗口移动后的效果

### 技术要点

本实例的主要思想是用screen对象的availWidth和availHeight属性来获得当前屏幕的宽度和高度，然后判断窗口的4个边是否碰到屏幕的4个边界，如果碰到，则进行反弹。

resizeTo() 和moveTo() 方法的介绍可参见实例015。

### 实现过程

编写用于实现移动窗口的JavaScript代码。

```
<script language="JavaScript">  
window.resizeTo(300,300)  
window.moveTo(0,0)  
inter=setInterval("go()", 1);
```

```
var aa=0
var bb=0
var a=0
var b=0
function go()
{
    try{
        if (aa==0)
            a=a+2;
        if (a>screen.availWidth-300)
            aa=1;
        if (aa==1)
            a=a-2;
        if (a==0)
            aa=0;
        if (bb==0)
            b=b+2;
        if (b>screen.availHeight-300)
            bb=1;
        if (bb==1)
            b=b-2;
        if (b==0)
            bb=0;
        window.moveTo(a, b);
    }
    catch(e) {}
}
```

</script>

举一反三

根据本实例，读者可以：

使窗口在屏幕中随意走动；

使窗口出现下拉式打开效果。

## 实例021 震动的窗口

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\021

实例说明

本实例实现的是窗口震动的效果，在窗口中可以单击“左右震动”、“上下震动”和“整屏震动”3个按钮，使窗口有3种不同的震动效果。运行结果如图1.25所示。



图1.25 震动的窗口

技术要点

实现本实例主要使用了window对象的moveBy( )方法，该方法的详细介绍可参见实例018中的技术要点。

## 实现过程

(1) 在页面中添加3个按钮，并调用各自的单击事件，代码如下。

```
<input name="button1" type="button" id="button1" value="左右震动" onclick="flap_x(5)">
```

```
<input name="button2" type="button" id="button2" value="上下震动" onclick="flap_y(5)">
```

```
<input name="button3" type="button" id="button3" value="整屏震动" onclick="flap_xy(5)">
```

(2) 编写用于实现震动窗口的JavaScript代码。

自定义函数flap\_xy()，用于实现窗口的整屏震动，具体代码如下。

```
<SCRIPT language=javascript>
function flap_xy(n)
{
    if (self.moveBy) {
        for (i = 10; i > 0; i--) {
            for (j = n; j > 0; j--) {
                self.moveBy(0, i);
                self.moveBy(i, 0);
                self.moveBy(0, -i);
                self.moveBy(-i, 0);
            }
        }
    }
}
}
```

自定义函数flap\_x(), 用于实现窗口的左右震动, 具体代码如下。

```
function flap_x(n) {
    if (self.moveBy) {
        for (i = 10; i > 0; i--) {
            for (j = n; j > 0; j--) {
                self.moveBy(i, 0);
                self.moveBy(-i, 0);
            }
        }
    }
}
```

自定义函数flap\_y(), 用于实现窗口的上下震动, 具体代码如下。

```
function flap_y(n) {
    if (self.moveBy) {
        for (i = 10; i > 0; i--) {
            for (j = n; j > 0; j--) {
                self.moveBy(0, i);
                self.moveBy(0, -i);
            }
        }
    }
}
</SCRIPT>
```

举一反三

根据本实例, 读者可以:

使窗口按一定的轨迹进行移动。

## 实例022 弹出广告窗口

本实例可以美化界面

实例位置：光盘\mingrisoft\01\022

实例说明

在浏览网页时，当打开一个特定的网页时，会在屏幕的右下角慢慢浮出一个广告窗口，这样可以起到一个宣传的作用，本实例通过打开一个主页面，在屏幕的右下角自动以上移的方式弹出一个广告窗口。运行结果如图1.26所示。

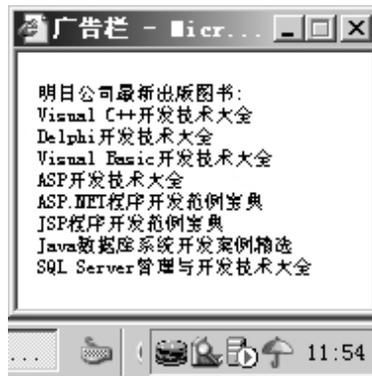


图1.26 弹出广告窗口

技术要点

本实例主要实现在主页面打开时，用 window 对象的 open() 方法调用子窗口，并设路子窗口的大小与位路，再打开子窗口后，用 window对象的moveTo() 方法将子窗口逐渐上移。

moveTo () 方法的介绍可参见实例015。

实现过程

(1) 主窗口的相关代码。

编写用于实现调用广告窗口的JavaScript代码。

```
<script language="javascript">
```

```

var newformW=180;
var newformH=130;
self.focus();
function pp()
{
    var T=window.screen.width-newformW;
    var L=window.screen.height+newformH;
    var
name=window.open("ad.htm","", "width="+newformW+", height="+n
ewformH
    +", top="+T+", left="+L);
}
pp();
</script>

```

(2) 广告窗口的相关代码。

编写用于实现广告窗口从右下角上移的JavaScript代码。

```

<script language="javascript">
var W=document.body.clientWidth;
var H=document.body.clientHeight;
var L=window.screen.width-W;
var T=window.screen.height;
self.moveTo(L, T);
function drop(n) {
    if (n>(T-H-60)) {
        self.moveTo(L-12, --n);
        var tim=setTimeout("drop("+n+")", 1);
    }
}

```

```
    else {clearTimeout (tim)}  
  }  
  drop(T);  
</script>
```

举一反三

根据本实例，读者可以：

关闭窗口时，弹出广告；

关闭广告窗口时，使主窗口显示其他网页。

## 1.4 窗口控制

在制作网站时，对窗口的控制是十分重要的，例如可以根据屏幕分辨率的大小来控制页面的大小，也可以根据需要将页面进行最大化显示或对页面的大小进行控制。下面将通过几个典型实例进行详细讲解。

### 实例023 窗口始终在最上面

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\01\023

实例说明

浏览网页时，经常会看到一些广告总是浮动于所有窗口的最上面，这样使浏览者不得不观看广告中的内容。本实例将指定一个页面总在窗口的最上面。运行结果如图1.27所示。

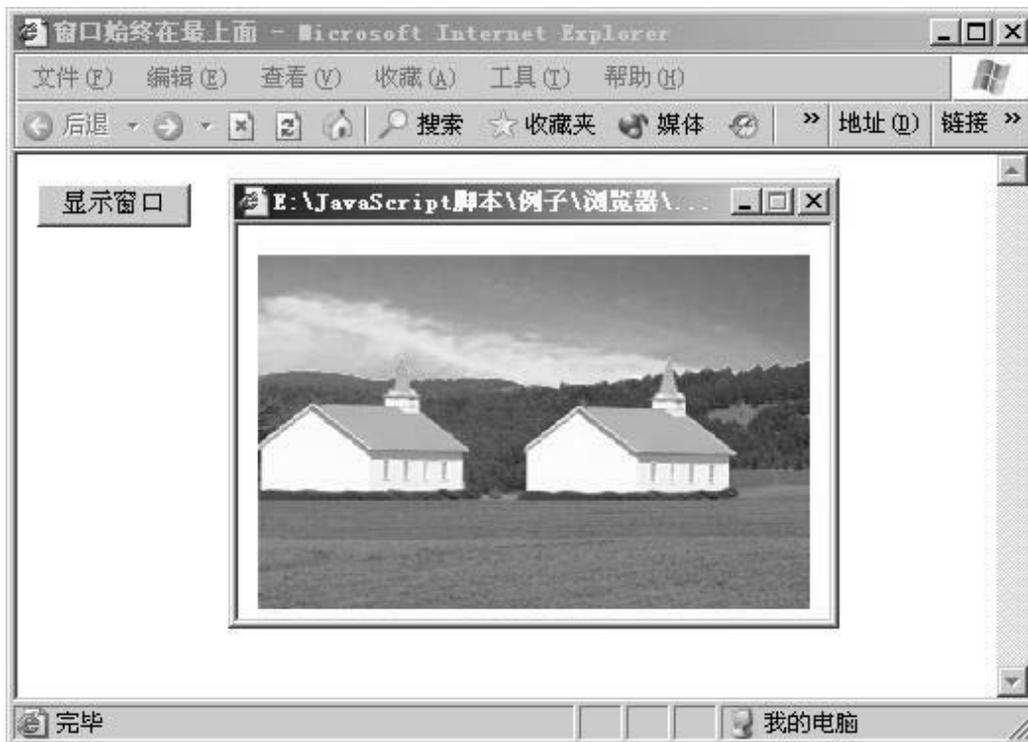


图1.27 窗口始终在最上面

### 技术要点

本实例主要应用了document对象的focus()方法来获得窗口的焦点。用window对象的open()方法所创建的窗口进行操作，将其返回值（窗口对象）赋给一个变量，然后用该变量对创建的窗口进行操作。下面对window对象的open()方法进行详细介绍。

window对象open()方法的语法格式：

```
WindowVar=window.open(url,windowname[,location]);
```

参数说明如下。

- URL：用于打开一个新的浏览器窗口，并在新窗口中装入一个指定的URL 地址。
- windowname：打开一个新的浏览器窗口时，指定窗口的名称。
- location：对窗口属性进行设置，其可选参数如表1.5 所示。

表1.5 location参数的取值

参 数 值	描 述
toolbar	指定窗口是否有标准工具栏。当该选项的值为 1 或 yes 时，表示有标准工具栏；当该选项的值为 0 或 no 时，表示没有标准工具栏
location	指定窗口是否有地址工具栏，选项的值及含义与 toolbar 相同
directories	指定窗口是否有链接工具栏，选项的值及含义与 toolbar 相同
status	指定窗口是否有状态栏，选项的值及含义与 toolbar 相同
menubar	指定窗口是否有菜单，选项的值及含义与 toolbar 相同
scrollbar	指定当前窗口文档大于窗口时是否有滚动条，选项的值及含义与 toolbar 相同
resizable	指定窗口是否可改变大小，选项的值及含义与 toolbar 相同

续表

参 数 值	描 述
alwaysLowered	指定窗口隐藏在所有窗口之后，选项的值及含义与 toolbar 相同
alwaysRaised	指定窗口浮在所有窗口之上，选项的值及含义与 toolbar 相同
dependent	指定打开的窗口为当前窗口的一个子窗口，并随着父窗口的关闭而关闭，选项的值及含义与 toolbar 相同
hotkeys	在没有菜单栏的新窗口中设置安全退出的热键，选项的值及含义与 toolbar 相同
innerHeight	设定窗口中文档的像素高度
innerWidth	设定窗口中文档的像素宽度
screenX	设定窗口距离屏幕左边界的像素长度
screenY	设定窗口距离屏幕上边界的像素长度
titleBar	指明标题栏是否在新窗口中可见，选项的值及含义与 toolbar 相同
z-lock	指明当窗口被激活时，不能浮在其他窗口之上，选项的值及含义与 toolbar 相同

功能：打开一个新的窗口，并在窗口中装载指定URL地址的网页。

实现过程

(1) 在页面中添加一个按钮，用于显示窗口，代码如下。

```
<input type="button" name="Button" value="显示窗口"
onClick="newform()">
```

(2) 编写用于实现选中表格中单元格的JavaScript代码。

自定义函数newform()，用于创建一个窗口，具体代码如下。

```
<script language="JavaScript">
var name;
function newform()
{
```

```
name=window.open("", "", "width=300,height=200");
name.document.write('<input name="imageField"
type="image" src="6.jpg"
width="280" height="180" border="0">');
show(name);
}
```

自定义函数show(), 使窗口不断获得焦点, 使其在最上面, 具体代码如下。

```
function show(name) {
    try{
        name.document.focus();
        setTimeout("show(name)", 1);
    }
    catch(e) {}
}
</script>
```

举一反三

根据本实例, 读者可以:

实现用window对象的open()方法调用其他的窗口;  
指定新窗口的大小及位置。

## 实例024 窗口的最小化、最大化

本实例是一个提高规范化的实例

实例位置: 光盘\mingrisoft\01\024

实例说明

在浏览网页时，有些网页会自动对窗口的显示状况进行调试。本实例是通过页面中的“最小化”和“最大化”按钮控制窗口的最小化和最大化。运行结果如图1.28所示。



图1.28 窗口的最大化、最小化

### 技术要点

本实例主要应用到了<object>标记和<param>标记。<object>标记里的classid属性是告诉浏览器插件的类型。用<object>标记在页面中插入Activex控件或其他对象之后，有时候需要向该对象或者控件传递参数，这就要使用<param>标记。该标记没有相应的结束标志</param>，下面对<param>标记进行详细说明。

<param>标记的一般格式为：

```
<param name=* value=* valuetype=* type=*>
```

参数说明如下。

- name：是参数的名字。
- value：指定参数的值。
- valuetype：指定怎样表示参数的值。
- type：指定媒体类型。

### 实现过程

(1) 在页面中添加两个按钮和一个层，代码如下。

```
<input name="botton1" type=button id="botton1"
onClick=hh1.Click() value=最小化>
<input name="botton2" type=button id="botton22"
onClick=hh2.Click() value=最大化>
<div id="aa"></div>
```

(2) 编写用于实现窗口最大化、最小化的JavaScript代码，具体代码如下。

```
<script language="JavaScript">
var s="";
s=s+'<object id=hh1 classid="clsid:ADB880A6-D8FF-11CF-
9377-00AA003B7A11">';
s=s+'<param name="Command" value="Minimize"></object>';
s=s+'<object id=hh2 classid="clsid:ADB880A6-D8FF-11CF-
9377-00AA003B7A11">';
s=s+'<param name="Command" value="Maximize"></object>';
aa.innerHTML=s;
</script>
```

举一反三

根据本实例，读者可以：

使窗口最大化后，最大化按钮显示还原。

## 实例025 频道方式窗口

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\01\025

实例说明

在浏览网页时，经常会看到一些网页以频道的方式进行打开。本实例在打开“index.htm”文件后，将弹出“频道方式窗口”页面，在该页面中单击“频道方式窗口”按钮，将以频道方式打开“明日美食留言簿”页面。运行结果如图1.29所示。



图1.29 频道方式窗口

### 技术要点

本实例主要应用 window 对象的 open() 方法中的 channelmode 参数值来使创建窗口以频道方式进行显示。

window对象open()方法中参数的详细说明可参见实例023。

### 实现过程

(1) 在页面中添加一个按钮，代码如下。

```
<form name="form1" method="post" action="">
```

```
<input type="button" name="Submit" value="频道方式窗口"
onClick="show()">
</form>
```

(2) 编写用于实现频道方式窗口的JavaScript代码。

```
<script language="javascript">
function show() {
window.open("trans.htm","", "channelmode, scrollbars");
}
</script>
```

举一反三

根据本实例，读者可以：

去掉窗口的状态栏；

去掉窗口中的工具栏。

## 实例026 全屏显示

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\026

实例说明

在制作网站时，有时根据网页内容的需要，要将网页进行全屏显示。本实例在打开页面后，单击页面中的“全屏显示”按钮，将页面进行全屏显示。运行结果如图1.30、图1.31所示。



图1.30 窗口打开时的效果



图1.31 窗口全屏显示效果

技术要点

本实例主要应用window对象的open( )方法中的fullscreen 参数，将当前窗口重新创建，并进行全屏显示。这样做可以避免在打开其他窗口时，也使其全屏显示。

另外，本实例还使用了document对象的location属性，该属性用于指定当前窗口。

### 实现过程

(1) 在页面中添加两个按钮，将其放路在层中，并用层的align属性的center值使按钮居中于页面，代码如下。

```
<div align="center">
  <input name="button1" type="button" id="button1"
onClick="fullscreen()"
value="全屏显示">
  <input name="button2" type="button" id="button2"
value="关闭窗口" onClick
="window.close()">
</div>
```

(2) 编写用于实现全屏显示的JavaScript代码。

```
<script language="javascript">
var pp;
function fullscreen() {
  pp=window.open('Index.htm', '', 'fullscreen=yes');
}
</script>
```

### 举一反三

根据本实例，读者可以：

制作一个与F11（全屏）键相同的功能。

## 实例027 设置窗口大小和位置

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\027

实例说明

本实例实现在窗口中设置要打开页面的高度和宽度，以及在当前屏幕中的位路。运行结果如图1.32所示。

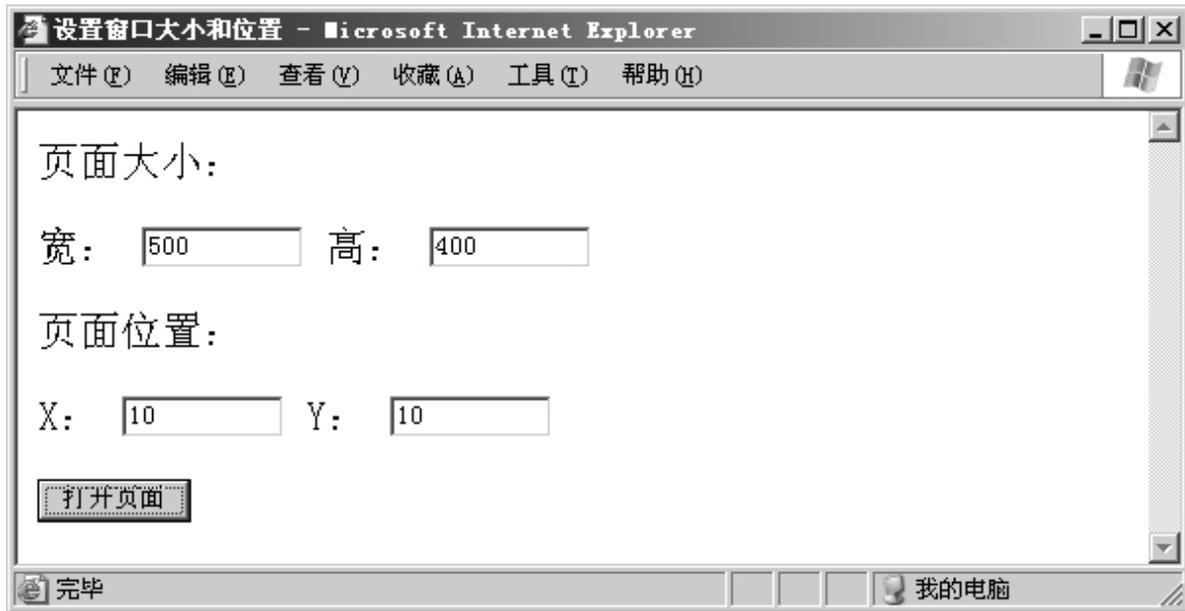


图1.32 设置窗口大小和位置

技术要点

本实例主要通过用户输入的值来动态改变 window 对象的 open( )方法中的一些参数值，从而实现动态设置窗口的大小和位置。

window对象的open( )方法的详细介绍可参见实例001。

实现过程

(1) 在页面中添加相关的组件，用于输入要创建窗口的位路及大小，代码如下。

```
<form name="form1" method="post" action="">
```

```
<p>页面大小: </p>
```

```
<p>宽:
  <input name="text1" type="text" id="text1"
value="500" size="10">
  高:
  <input name="text2" type="text" id="text2"
value="400" size="10">
</p>
<p>页页位置: </p>
<p>X:
  <input name="text3" type="text" id="text3" value="10"
size="10">
Y:
  <input name="text4" type="text" id="text4" value="10"
size="10">
</p>
  <p>
    <input type="button" name="Submit" value="打开页面"
onclick="place()">
  </p>
</form>
```

(2) 编写用于实现设路窗口大小和位路的JavaScript代码。

```
<script language="javascript">
var ss="";
function place()
{
  ss="width="+document.form1.text1.value+",height="+docum
ent.form1.text2.value
```

```
+", top="+document.form1.text3.value+", left="+document.f  
orm1.text4.value;  
var newform=window.open("melody.htm","", ss);  
}  
</script>
```

举一反三

根据本实例，读者可以：

用window对象的open( )方法任意打开指定路径下的HTML文件。

## 实例028 刷新当前页

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\01\028

实例说明

在浏览页面时，有很多窗口都有自带的刷新功能，本实例将实现对当前窗口的刷新操作，只要单击页面中的“刷新”按钮即可。运行结果如图1.33所示。

技术要点

本实例主要应用location对象的reload()方法，对当前页进行刷新。

注意：reload()方法不可以用在循环语句中。

实现过程

(1) 在页面中添加一个按钮，并在单击事件中调用自定义函数ref()，代码如下。

```
<form name="form1" method="post" action="">  
  <input type="button" name="Submit" value="刷新"  
  onclick="ref()">
```

</form>



图1.33 刷新当前页

(2) 编写用于实现刷新当前页的JavaScript代码。

```
<script language="javascript">  
function ref()  
{  
    window.location.reload();  
}  
</script>
```

举一反三

根据本实例，读者可以：

制作一个按时间对当前页进行刷新的页面。

## [实例029 自动最大化](#)

本实例可以提高工作效率

实例位置：光盘\mingrisoft\01\029

实例说明

在打开 IE 浏览器时，IE 浏览器的大小是上一次关闭窗口默认值，但有些窗口只能以最大化形式进行显示。本实例可以在显示窗口时，使其最大化显示，运行结果如图1.34所示。

#### 技术要点

本实例主要应用screen对象的availWidth和availHeight属性来获取屏幕工作区的宽度和高度，并用resizeTo()方法将当前窗口的大小设为屏幕大小，再用moveTo()方法将窗口放在屏幕的左上角。

self获取的是当前窗口的对象，也可以用this代表当前窗口的对象。

resizeTo()和moveTo()方法的介绍可参见实例015。

#### 实现过程

编写用于实现自动最大化的JavaScript代码。

```
<script language="JavaScript">  
self.moveTo(0,0);  
self.resizeTo(screen.availWidth,screen.availHeight);  
</script>
```



图1.34 自动最大化

举一反三

根据本实例，读者可以：

将指定大小的窗口放在屏幕的中间；

制作一个在屏幕中移动的窗口。

## 实例030 自定义导航控制面板

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\01\030

实例说明

在开发网络程序时，有时需要创建一个自定义导航控制面板。通过该面板可以直接进行相关功能的操作。例如，本实例在电子商城网

站中添加了一个自定义导航控制面板，通过该面板可以将当前打开的IE窗口关闭，并且在关闭窗口时，将给予相关提示信息；应用“刷新”按钮将完成对当前网页的刷新操作；单击“打印”按钮可以将当前指定的网页内容打印出来；单击网页中的“前进”和“后退”按钮可以实现不同页面间的切换显示。运行结果如图1.35所示。

### 技术要点

本实例主要应用 history对象实现导航控制面板的功能。下面对 history对象进行介绍。该对象主要用来存储一个最近所访问网页的 URL地址的列表。

history对象的属性或方法的语法格式如下。

```
[window.]history.property|method([parameters])
```



图1.35 自定义导航控制面板

history对象的常用属性以及说明如表1.6所示。

表1.6 history对象的常用属性

属 性	描 述
length	历史列表的长度, 用于判断列表中的入口数目
current	当前文档的 URL
next	历史列表的下一个 URL
previous	历史列表的前一个 URL

history对象的常用方法以及说明如表1.7所示。

表1.7 history对象的常用方法

方 法	描 述
back()	退回前一页
forward()	重新进入下一页
go()	进入指定的网页

### 实现过程

(1) 在需要自定义导航控制面板的页面中插入如下程序代码, 即可实现所需要的相应功能, 代码如下。

```
<script language="javascript">
function goto()
{
    if(window.history.length>0)
        window.history.go(window.history.length-1);
}
</script>
<form action="#" method="get">
    <div align="center">
        <input type="button" value="打印"
onclick="if(window.print) window.print();">
        <input type="button" value="前进"
onclick="if(window.history.forward)
```

```
window.history.forward();">
    <input type="button" value="后退" onclick="goto()">
    <input type="button" value="刷新"
onclick="if(window.location.reload)
window.location.reload();">
    <input type="button" value="关闭"
onclick="if(window.close) window.close();">
    </div>
</form>
```

(2) 在单击超级链接所进入的页面后，插入以下程序代码，同时也可以实现相应功能，代码如下。

```
<script language="javascript">
var i=1;
function goto()
{
    while (true)
    {
        if ((window.history.length-i)==-1) {
            break;
        }
        else{i=i+1;}
    }
    if(window.history.length>0)
        window.history.go(window.history.length-i);
}
</script>
<form action="#" method="get">
```



计算机分辨率的不同而影响页面的整体效果。本实例将根据不同计算机分辨率的大小来自动调整窗口。实例运行结果如图1.36所示。

### 技术要点

本实例主要应用window对象的resizeTo()方法设置窗口的大小。下面将对该方法进行详细介绍。

resizeTo()方法的语法格式：

window.resizeTo(x, y)

参数说明如下。

- x: 窗口的水平宽度。

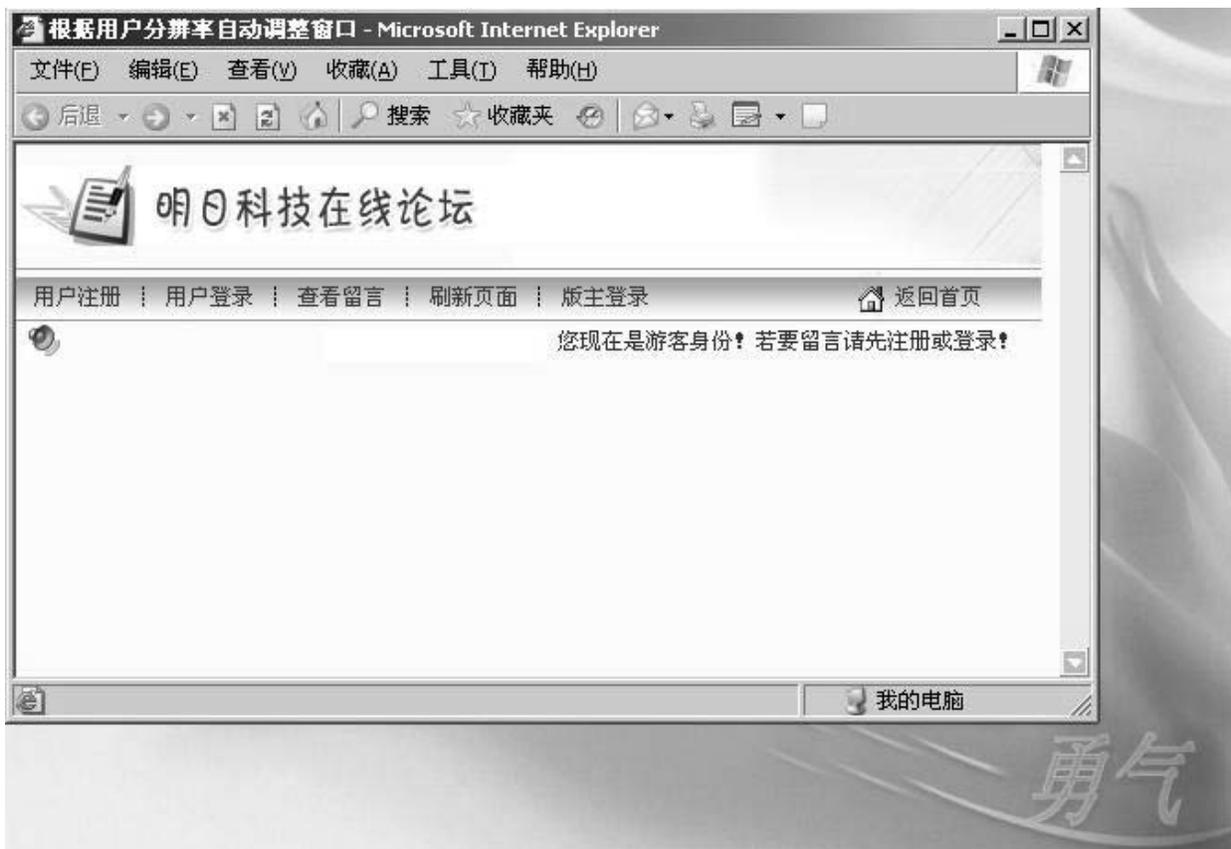


图1.36 根据用户分辨率自动调整窗口

- y: 窗口的垂直宽度。
- 功能: 将当前窗口改变成(x, y)大小, x、y分别为宽度和高度。

## 实现过程

(1) 应用JavaScript脚本创建size()函数，该函数主要用于自动调整窗口大小，代码如下。

```
<script language="javascript">  
function size()  
{  
window.resizeTo(screen.width-100,screen.height-100);  
}  
</script>
```

(2) 应用onload()事件调用自定义函数size()，代码如下。

```
<body onload="size();">
```

### 举一反三

根据本实例，读者可以：

开发宠物网站程序时，应用到此技术；

开发供求信息管理系统时，应用到此技术；

开发新闻网站时，应用到此技术；

开发网上论坛中的用户注册窗口及用户登录窗口时应用到此技术。

## 1.5 窗口的其他效果

在浏览网页时，窗口会以多种方式进行显示，例如，在打开窗口前进行提示、使窗口透明等。下面将通过几个典型实例进行详细讲解。

### 实例032 打开窗口时显示对话框

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\01\032

实例说明

在浏览一些网页时，经常会弹出一些询问式的对话框，问浏览者是否要打开网页。本实例将通过一个简单的语句来实现这一功能。运行结果如图1.37所示。



图1.37 打开窗口时显示对话框

技术要点

本实例主要应用了window对象的confirm()方法和close()方法。下面对这两个方法进行详细说明。

confirm()方法又被称为确认对话框，由两个按钮组成（在中文平台中一般显示为“确定”和“取消”，英文平台为“OK”和“Cancel”）。该方法最重要的是具有返回值，当单击“确定”（“OK”）按钮时，返回值为true；当单击“取消”（“Cancel”）按钮时，返回值为false。设计者可以通过该对话框的返回值来决定脚本如何操作。

close()方法用于关闭一个浏览器窗口。

实现过程

编写用于实现打开窗口时显示对话框的JavaScript代码。

```
<script language="JavaScript">  
if (confirm("欢迎进入明日科技网站!")==false) {  
    window.close();  
}  
</script>
```

举一反三

根据本实例，读者可以：

在打开网页时，显示对话框，是否打开一些其他相关网页；  
在关闭子窗口时，显示对话框，是否关闭其父窗口。

## [实例033 使窗口背景透明](#)

本实例可以提高工作效率

实例位置：光盘\mingrisoft\01\033

实例说明

在制作网站时，有些网站的部分内容可能与其他的网站相同，这样就可以将其他网站的内容直接放路在该网站中。本实例是将一个页面以框架的形式放到另一个页面中，并将页面设为透明状态。运行结果如图1.38所示。



图1.38 使窗口背景透明

#### 技术要点

本实例主要应用的是浮动框架标记<iframe>。用<iframe>标记可以将其他页中的内容显示在当前页面中。下面对<iframe>标记的相关属性进行说明。

<iframe>标记的属性如表1.8所示。

表1.8 <iframe>标记的属性

属 性	描 述
src	浮动框架中显示页面源文件的路径
width	浮动框架的宽度
height	浮动框架的高度
name	浮动框架的名称
align	浮动框架的排列方式, left 表示居左, center 表示居中, right 表居右
frameborder	设置框架边框显示
scrolling	设置框架滚动条显示
marginwidth	设置框架边缘宽度
marginheight	设置框架边缘高度
allowtransparency	设置框架背景透明

### 实现过程

(1) 在页面中添加3个文本框, 并设路相应的名称, 代码如下。

```
<table valign="center" align="center" width="518"
height="318" border="1"
cellpadding="0" cellspacing="0">
  <tr>
    <th width="514" height="316" valign="middle"
scope="col">
      <div id="div1" align="justify"></div>
    </th>
  </tr>
</table>
```

(2) 编写用于实现动态制作表格的JavaScript代码。

```
<script language="JavaScript">
div1.innerHTML='<IFRAME frameborder="0" scrolling="no"
src="melody.htm" width=506
height=360 allowtransparency></iframe>';
</script>
```

### 举一反三

根据本实例，读者可以：  
在页面中动态生成一个登录界面；  
动态替换表格的内容。

## 实例034 立体窗口

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\01\034

实例说明

在浏览网页时，经常会看到带有特殊效果的页面，本实例将通过对窗口样式的设路，使窗口具有立体效果。运行结果如图1.39所示。



图1.39 立体窗口

技术要点

本实例主要应用了<body>标记的style样式的borderWidth（边框的宽度）、borderColor（边框的颜色）和borderStyle（边框的样式）属性，对页面的边框进行设路。下面对borderStyle属性值进行详细地介绍。

borderStyle属性（边框样式）值如表1.9所示。

表1.9 borderStyle属性值

边框样式属性值	描 述
none	无边框
hidder	隐藏边框, IE 不支持
dotted	边框由点组成

续表

边框样式属性值	描 述
dashed	边框由短线组成
solid	边框是实线
double	边框是双实线
groove	边框带有立体感的沟槽
ridge	边框成脊形
inset	边框内嵌一个立体边框
outset	边框外嵌一个立体边框

注意：在对页面边框的样式进行设置时，要将页面中最上面的两行代码删除。代码如下。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
```

实现过程

编写用于实现立体窗口的JavaScript代码。

```
<script language="JavaScript">
document.body.style.borderWidth=5;
document.body.style.borderColor="#CCCCFF";
document.body.style.borderStyle="groove";
</script>
```

举一反三

根据本实例，读者可以：

使页面边框以不同的颜色进行闪动；

使页面边框具有凸出效果。

## 实例035 动态标题栏

这是一个人性化的实例

实例位置：光盘\mingrisoft\01\035

实例说明

在浏览网页时，经常会看到一些IE浏览器中标题栏的信息不停地闪动或变换。本实例将在打开页面时，对标题栏里的文字进行不断的变换。运行结果如图1.40、图1.41所示。



图1.40 标题栏文字改变前的效果



图1.41 标题栏文字改变后的效果

## 技术要点

本实例主要用document对象的title属性来修改页面标题栏中的信息，并用window对象的setTimeout()方法对title属性中的信息以指定间隔时间进行修改。下面对setTimeout()方法进行详细介绍。

setTimeout()方法的语法格式如下：

```
setTimeout (function, milliseconds)
```

参数说明如下。

- function: 要调用的JavaScript 自定义函数名称。
- milliseconds: 设路超时时间（以毫秒为单位）。

功能：经过超时时间后，调用函数。此值可以用clearTimeout()函数清除。

注意：setTimeout()方法在超时时间后只调用一次函数，而setInterval()方法是按一定时间重复调用指定的函数。

## 实现过程

编写用于实现动态标题栏的JavaScript代码。

```
<script language="JavaScript">
var n=0;
function title() {
    n++;
    if (n==3) {n=1}
    if (n==1) {document.title=' ☆★动态标题栏★☆' }
    if (n==2) {document.title=' ★☆明日科技网站☆★' }
    setTimeout("title()", 350);
}
title();
</script>
```

举一反三

根据本实例，读者可以：  
在打开网页时，动态修改标题栏中的信息。

## 实例036 固定大小的窗口

本实例可以美化界面

实例位置：光盘\mingrisoft\01\036

实例说明

在浏览网页时，经常会看到一些窗口根据页面的内容来固定窗口的大小。本实例在打开窗口后，窗口的大小将不允许用户修改。运行结果如图1.42所示。



图1.42 固定大小的窗口

技术要点

本实例实现了在打开窗口后通过 window 对象的 `resizeTo()` 方法指定窗口的大小，再通过 `<body>` 标记的 `onresize` 事件（窗口缩放事件）在改变窗口大小的同时恢复其窗口的指定大小。

window 对象的 `resizeTo()` 方法的介绍可参见实例 015 中的技术要点。

### 实现过程

编写用于实现固定大小的窗口的 JavaScript 代码。

```
<script language="JavaScript">
function fastness() {
    window.resizeTo(600, 440);
}
document.body.onresize=fastness;
document.body.onload=fastness;
</script>
```

### 举一反三

根据本实例，读者可以：

实现弹出固定大小的广告窗口。

## 1.6 框架的应用

所谓框架就是网页的各部分为相互独立的网页，又由一个网页将这些分开的网页组成一个完整的网页，显示在浏览者的浏览器中，重复出现的内容被固定下来，每次浏览者发出对页面的请求时，只下载发生变化的框架页面，其他子页面保持不变。

使用框架可以将容器窗口划分为若干个子窗口，在每一个子窗口中可以分别显示不同的网页。使用框架可以非常方便地完成导航工作。下面将通过具体实例详细介绍框架的应用。

### 实例037 框架集的嵌套

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\037

实例说明

在进行网络程序开发时，在网页中使用一个框架集是非常简单的。但如果要在同一个网页中应用更多的框架集时，应该怎样进行框架集的嵌套呢？本实例将解决这一问题。运行结果如图1.43所示。

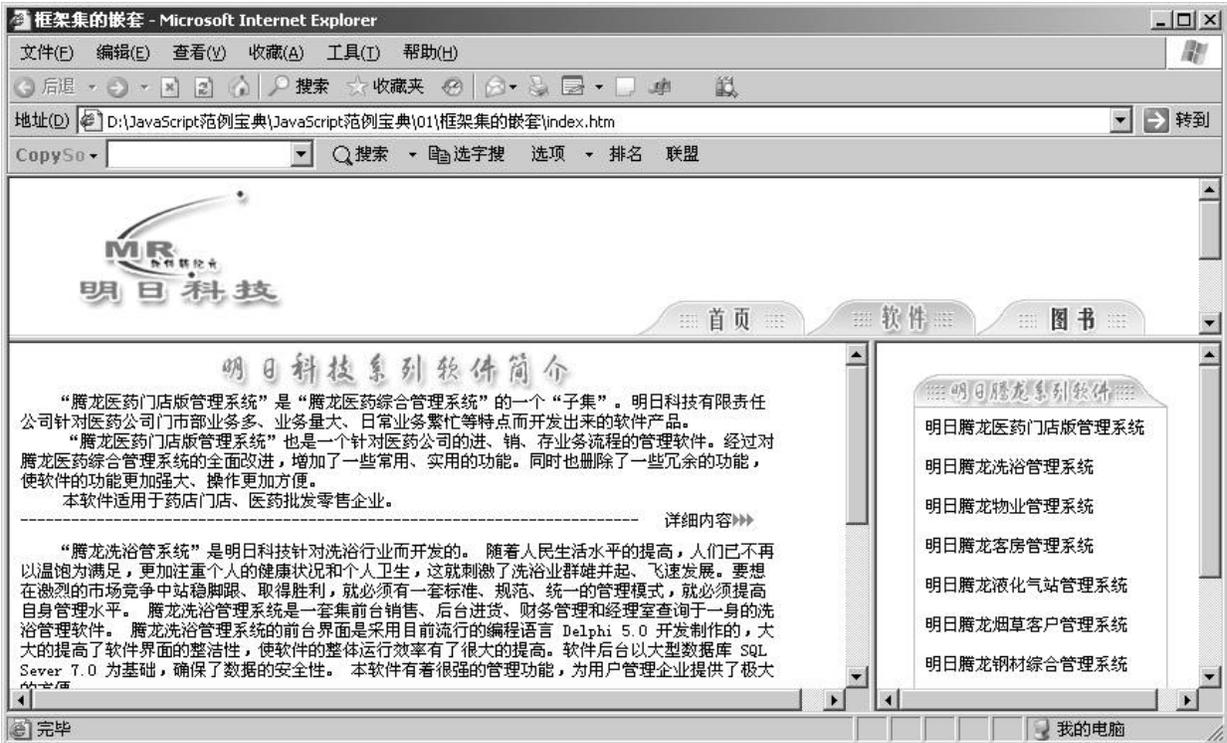


图1.43 框架集的嵌套

### 技术要点

本实例主要介绍框架集的嵌套应用，下面将对框架集进行详细介绍。

框架集主要包含如何组织各个框架的信息，可以通过<frameset>标记来定义框架集。框架是按照行和列来组织的，可以使用<frameset>标记的下列属性对框架的结构进行设路。

#### (1) 左右分割窗口属性COLS。

在水平方向上将浏览器分割成多个窗口，这可以通过框架的左右分割窗口属性COLS实现。其语法格式如下：

```
<frameset cols="value, value, ……">
    <frame>
    <frame>
</frameset>
```

参数属性说明如下。

● value: 用于指定各个框架的列宽, 取值有3种形式: 像素、百分比(%)和相对尺寸(\*)。

例如, 通过框架将浏览器窗口划分为3列, 其中, 第1列占浏览器窗口宽度的20%, 第2列为120像素, 第3列为浏览器窗口剩余部分的框架, 代码如下。

```
<frameset cols="20%, 120, *" >
  <frame>
  <frame>
</frameset>
```

技巧: 如果将COLS属性设置为“\*, \*, \*”, 则表示将窗口划分成3个等宽的框架; 如果将 COLS 属性设置为“\*, 2\*, 3\*”, 则表示左边的框架占窗口宽度的 1/6, 中间的框架占窗口宽度的1/3, 右边的框架占窗口宽度的1/2。

(2) 上下分割窗口属性ROWS。

在垂直方向上将浏览器分割成多个窗口, 这可以通过框架的上下分割窗口属性 ROWS 实现。其语法格式如下。

```
<frameset rows="value, value, ……">
  <frame>
  <frame>
</frameset>
```

参数说明如下。

● value: 用于指定各个框架的行高, 取值有3种形式: 像素、百分比(%)和相对尺寸(\*), 设置方法与COLS属性类似。例如, 通过框架将浏览器窗口划分为3行, 其中的第1行占浏览器窗口宽度的20%, 第2行为120像素, 第3行为浏览器窗口剩余部分的框架。代码如下。

```
<frameset rows="20%, 120, *" >
  <frame>
```

<frame>

</frameset>

(3) 框架边框显示属性FRAMEBORDER。

该属性用于指定框架周围是否显示边框，取值为1（显示边框，默认值）或0（不显示边框）。

(4) FRAMESPACING。

该属性用于指定框架之间的间隔，以像素为单位。如果不设置该属性，则框架之间没有间隔。

(5) 指定边框宽度属性BORDER。

该属性用于指定边框的宽度，只有FRAMEBORDER属性为1时才有效。

实现过程

在需要进行布局的页面中写入以下程序代码，将 src 属性值修改为需要链接的文件即可。代码如下。

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=gb2312"/>
<title>框架集的嵌套</title>
</head>
<!--创建嵌套框架集-->
<frameset rows="30%,*">
  <frame src="software_top.htm">
  <frameset cols="50%,50%">
    <frame src="software_main.htm">
    <frame src="software_left.htm" scrolling="auto">
  </frameset>
</frameset>
```

```
</frameset>
```

```
<body>
```

```
</body>
```

```
</html>
```

举一反三

根据本实例，读者可以：

开发在线论坛程序时，应用该技术；

开发新闻网站程序时，应用该技术；

开发办公自动化管理系统时，应用该技术。

## 实例038 在网页中应用浮动框架

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\01\038

实例说明

在开发网络程序时，经常会应用框架，例如网络考试管理系统中就涉及到了浮动框架。通过浮动框架将网站中各部分独立的网页重新组成一个完整的网页，显示在浏览器中。运行本实例，如图1.44所示。网页的左侧是一个独立的网页，右侧也是一个独立的网页。



图1.44 在网页中应用浮动框架

### 技术要点

本例主要应用浮动框架将各部分独立的网页重新组成一个完整的网页，下面将对浮动框架进行详细介绍。

浮动框架<iframe>是一种特殊的框架页面，在浏览器窗口中可以嵌套子窗口，在其中显示子页面的内容。其语法格式如下。

```
<IFRAME src="文件" height="数值" width="数值" name="框架名称" scrolling="值" frameborder="值">
```

```
</IFRAME >
```

下面是浮动框架属性的详细介绍。

#### 1. 浮动框架的文件路径属性SRC

语法：

```
<IFRAME SRC= "file_name" >
```

参数说明如下。

- file\_name：指明浮动框架文件的文件名或者其他超级链接的网址。

## 2. 浮动框架的名称属性NAME

语法:

```
<IFRAME SRC= "file_name" NAME= "frame_name" >
```

参数说明如下。

- frame\_name: 定义的浮动框架名称。

## 3. 浮动框架的对齐属性ALIGN

语法:

```
<IFRAME SRC= "file_name" ALIGN= "left/center/right" >
```

参数说明如下。

- left: 居左对齐。
- center: 居中对齐。
- right: 居右对齐。

## 4. 浮动框架的宽度和高度属性WIDTH、HEIGHT

语法:

```
<IFRAME SRC= "file_name" WIDTH= "value" HEIGHT= "value" >
```

参数说明如下。

- WIDTH: 浮动框架的宽度。
- HEIGHT: 浮动框架的高度。

## 5. 浮动框架滚动条显示属性SCROLLING

语法:

```
<IFRAME SRC= "file_name" SCROLLING= "value" >
```

参数说明如下。

value有3个取值。

- YES: 显示滚动条。
- NO: 不显示滚动条。
- AUTO: 根据窗口内容决定是否有滚动条。

## 6. 浮动框架边框属性FRAMEBORDER

语法:

```
<IFRAME SRC= "file_name" FRAMEBORDER= "value" >
```

参数说明如下。

● value: 值为yes 代表显示框架边框, 值为no 代表隐藏框架边框。

## 7. 浮动框架边缘的宽度和高度属性MARGINWIDTH、MARGINHEIGHT

语法:

```
<IFRAME SRC= "file_name" MARGINWIDTH= "value"
```

```
MARGINHEIGHT= "value" >
```

参数说明如下。

● MARGINWIDTH: 设定浮动框架左右边缘与边框的宽度。

● MARGINHEIGHT: 设定浮动框架上下边缘与边框的高度。

实现过程

在需要重新组建的网页中写入浮动框架的程序代码, 并将 src 属性值设为需要组建的网页文件。代码如下。

```
<table cellpadding="0" cellspacing="0" border="0"
width="100%" height="100%">
```

```
<tr>
```

```
<td width="6%"></td>
```

```
<td height="100%"><iframe src="adm_Mainleft.htm"
frameborder="0" width="100%" height="100%" name="main1"
align="middle" scrolling="auto">对不起, 你的浏览器不支持框
架! </iframe></td>
```

```
<td width="7%">
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</td></tr>
<tr>
<td height="36">&nbsp;</td>
</tr>
</table>
</td>
<td width="1%">

</td>
<td width="490" height="300">
<iframe src="adm_Mainright.htm" frameborder="0"
width="100%" height="100%" name="mainr" scrolling="auto">对不起，你的浏览器不支持框架！</iframe>
</td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

开发在线论坛程序时，应用到该技术；

开发网络游戏程序时，应用到该技术；

开发物流信息网时，应用到该技术。

## [实例039 创建空白框架](#)

本实例可以提高工作效率

实例位置：光盘\mingrisoft\01\039

## 实例说明

在进行网络程序开发时常常需要应用框架，所谓框架就是网页的各部分为相互独立的网页，又由一个网页将这些分开的网页组成一个完整的网页，显示在浏览者的浏览器中，此时重复出现的内容将被固定下来。那么如何在一个网页中创建空白的框架呢？本实例将解决这一问题。实例运行结果如图1.45所示。



图1.45 创建空白框架

## 技术要点

本实例主要应用框架的src属性给框架写一个通用的HTML文档。下面将对Frame框架进行详细介绍。

使用Frame框架可以设置框架的属性，包括框架的名称、框架是否包含滚动条以及在框架中显示的网页等。其语法格式如下。

```
<Frame src="文件" name="框架名称" scrolling="值"  
[noresize] frameborder="数值">
```

下面对框架属性进行详细介绍。

### 1. 框架文件的路径属性SRC

通过SRC属性来定义框架装载文件的路径。

语法：

```
<FRAME SRC= "file_name" >
```

参数说明如下。

- file\_name: 指明框架文件的文件名或者其他超级链接的地址。

### 2. 框架的名称属性NAME

通过NAME属性定义框架的名称，所起的名称将应用于页面的链接和脚本描述。

语法：

```
<FRAME SRC= "file_name" NAME= "frame_name" >
```

### 3. 框架边框显示属性FRAMEBORDER

可以设路是否显示框架边框。框架边框的显示情况继承框架集边框属性的设定。

语法：

```
<FRAME SRC= "file_name" FRAMEBORDER= "value" >
```

参数说明如下。

- value: 值为yes 代表显示框架边框，值为no 代表隐藏框架边框。

### 4. 框架滚动条显示属性SCROLLING

通过SCROLLING属性设路是否在框架内显示滚动条。

语法：

```
<FRAME SRC= "file_name" SCROLLING= "value" >
```

参数说明如下。

value: 有3个取值。

- YES: 显示滚动条。
- NO: 不显示滚动条。
- AUTO: 根据窗口内容决定是否显示滚动条。

5. 框架边缘的宽度和高度属性MARGINWIDTH、MARGINHEIGHT  
通过设置这两个属性可以调整框架页面内容与边框的距离。

语法:

```
<FRAME SRC= "file_name" MARGINWIDTH= "value"  
MARGINHEIGHT= "value" >
```

参数说明如下。

- MARGINWIDTH: 设定框架左右边缘与边框的宽度。
- MARGINHEIGHT: 设定框架上下边缘与边框的高度。

6. 框架尺寸调整属性NORESIZE

利用该属性可以控制框架的尺寸是否可以调整。

语法:

```
<FRAME SRC= "file_name" NORESIZE>  
NORESIZE: 禁止改变框架的尺寸。
```

7. 不支持框架标记<NOFRAMES>

某些版本的浏览器是不支持框架结构的, 如果遇到这种情况, 就可以使用<NOFRAMES>和</NOFRAMES>标记再声明一对文件主体标记<BODY>和</BODY>, 代表在无法接受框架结构时唯一显示的页面。

实现过程

在需要添加空白框架的页面中添加以下程序代码, 即可完成空白框架的创建。代码如下:

```
<html>  
<head>
```

```
<script type="text/javascript">
<!--
function sun()
{
return "<html></html>";
}
//-->
</script>
<meta http-equiv="Content-Type" content="text/html;
charset=gb2312"/>
<title>创建空白框架</title>
</head>
<frameset>
<frame name="Frame1" src="ss.htm">
<frame name="Frame2" src="javascript:parent.sun()">
</frameset>
</html>
```

举一反三

根据本实例，读者可以：

开发在线论坛程序时，应用到该技术；

开发在线点播系统时，应用到该技术；

在开发宠物网站时，应用到该技术。

## [实例040 居中显示框架页](#)

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\01\040

## 实例说明

在建设Web网站时不得不考虑的一个问题就是：如何让不同分辨率的用户都能看到网页的最佳效果。目前，虽然许多用户的浏览器的分辨率还是800×600的，但是也有一部分已经是1024×768的了，所以在设计页面时，经常要兼顾两者。那么就应该这样设计页面：将网页在800×600的分辨率下设计，然后将页面居中显示，这样的网页在1024×768下显示时，将不会给人很难看的感觉。

将页面设路为居中显示对于用表格布局的页面来说很简单，只需设路表格水平居中显示即可，但如果是框架页面就没有那么简单了，这就需要应用框架嵌套技术。例如，在明日科技网站的软件首页就是应用框架嵌套技术居中显示页面的，如图1.46所示。

## 技术要点

本实例主要应用框架集实现框架页的居中显示，框架集的详细介绍请读者参见实例037。

## 实现过程

通过对框架集（<frameset>标记）的属性设路，将指定的框架页进行居中显示，程序代码如下。

```
<html>
<head>
<title>居中显示框架页</title>
<meta http-equiv="Content-Type" content="text/html;
charset=gb2312">
</head>
<frameset rows="*" cols="1*,800,1*" frameborder="NO"
framespacing="0" border="0">
    <frame src="blank.htm" name="BLFrame" scrolling="NO"
noresize>
```

```
<frameset rows="157,*" cols="*" frameborder="NO"
border="0" framespacing="0">
  <frame src="software_top.htm" name="topFrame"
scrolling="NO" noresize >
    <frameset rows="*" cols="225,*" framespacing="0">
      <frame src="software_left.htm" name="leftFrame"
scrolling="auto" noresize>
        <frame src="software_main.htm" name="mainFrame"
frameborder="no" scrolling="auto">
          </frameset>
        </frameset>
      <frame src="blank.htm" name="BRFrame" scrolling="NO"
noresize>
        </frameset>
      </frameset>
    <noframes>
      <body>
        </body>
      </noframes>
    </html>
```

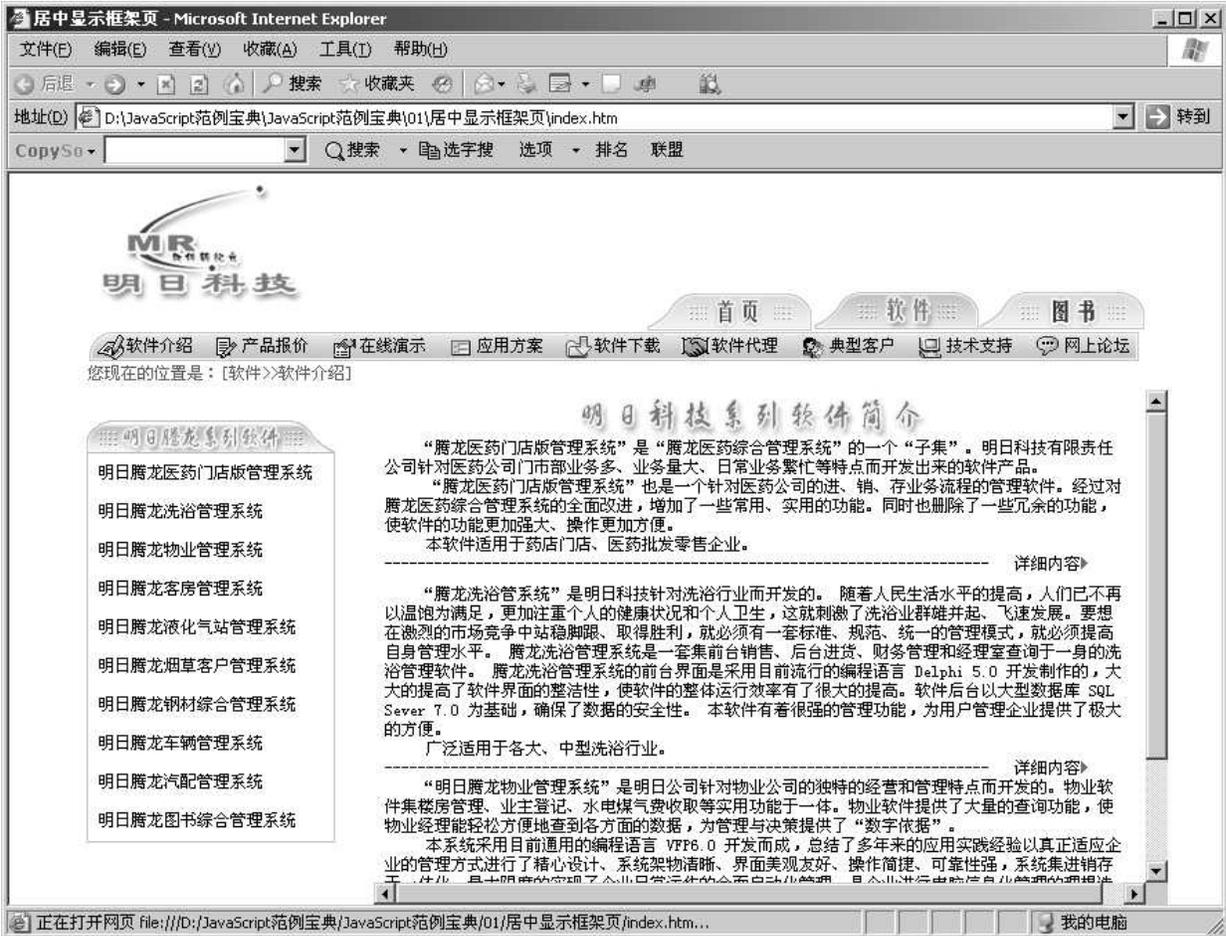


图1.46 居中显示框架页

举一反三

根据本实例，读者可以：

在开发网络图书程序时，应用到此技术；

在开发聊天程序时，应用到此技术；

在开发公司软件网站时，应用到此技术。

## 1.7 无边框窗口

所谓无边框窗口是指不包括IE浏览器窗口所固有的标题栏及灰色窗口边框的窗口，无边框窗口在某些特定的网站中起着举足轻重的作用。下面将通过几个具体实例介绍如何实现无边框窗口。

### 实例041 全屏显示无边框有滚动条的窗口

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\041

实例说明

通常用户在IE浏览器中浏览网页时，浏览器窗口都是包括标题栏、菜单栏和状态栏等固定内容的。虽然通过IE主菜单可以将浏览器的菜单栏和状态栏隐藏，但是标题栏却不能隐藏，即使将网页全屏显示也不能将其去除。有时为了网站的整体效果或其他目的，不得不使用全屏显示的无边框有滚动条的窗口。例如，在皮皮宠物网站中，为了给用户更真实的感觉，就可以将整个网站的主页面设路为全屏显示模式，如图1.47所示。从图1.47中可以看出作者还为用户设计了“退出”超级链接，可以方便浏览者随时离开宠物网站，这也是使用全屏显示时需要读者注意的，不然浏览者会因找不到出口而烦恼。



图1.47 全屏显示无边框有滚动条的窗口

### 技术要点

实现全屏显示无边框有滚动条窗口主要通过 window 对象的 open() 方法实现。这就需要借助一个中转页实现，即当该中转页运行时调用 open() 方法打开一个全屏显示的窗口，并关闭打开中转页的窗口，这就需要应用 window 对象 close() 方法关闭窗口，关闭窗口时不弹出确认对话框。

说明：使用window对象的open()方法中的“fullscreen=1”参数，可以将打开的新窗口全屏显示。

### 实现过程

(1) 新建一个空的 HTML 页面，该页面中可以没有任何页面布局元素或内容，文件名为index.htm。在该页面中添加代码如下。

```
<script language="javascript">
<!--
winClose();
function winClose()
{
    window.opener=null;
    window.close();
    window.open("main.htm","","fullscreen=1");
}
-->
</script>
```

(2) 创建main.htm文件，该文件才是网站的真正首页，前面的index.htm页面只是起到一个页面跳转的作用。该页面读者可根据实际情况设计，这里不再进行详细介绍。

### 举一反三

根据本实例，读者可以：

应用该方法制作全屏显示的网络考试页面，这样可以防止考生刷新屏幕；

制作图书馆管理系统的主界面；

开发游戏网站。

## 实例042 应用CSS+DIV实现指定尺寸无边框无滚动条窗口

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\042

实例说明

对于一个页面风格清新自然的网站来说，如果弹出的窗口带有不适合页面风格的灰色边框及死板的标题栏，那么这势必会影响网站的整体效果。本实例将向读者介绍一种更为完善的无边框窗口的实现方法，那就是应用 CSS+DIV 实现。运行本实例，单击“用户登录”超级链接即可弹出无边框的用户登录窗口，如图1.48所示。



图1.48 应用CSS+DIV实现指定尺寸无边框无滚动条窗口

技术要点

本实例主要通过页面中加入DIV层，并在页面中控制层的位路及其显示和隐藏来实现。在页面中加入层的语法如下：

```
<DIV id="value" align="value" class="value" style="value">
    ""
</DIV>
```

参数说明如下。

● ID: <div>标记的 ID 也可以说是它的名字, 常和 CSS 样式结合, 实现对网页中任何元素的控制。

● align: 用于控制<div>标记中的元素的对齐方式, 其value 值可以是left、center 和right, 分别用于设置元素的居左、居中和居右对齐。

● class: 用于设置<div>标记中的元素的样式, 其value 值为 CSS 样式中的class 选择符。

● style: 用于设置<div>标记中的元素的样式, 其 value 值为 CSS 属性值, 各属性值间应用分号分隔。该属性最常用的功能之一就是进行<div>标记的定位, 其对应的属性为position, 该属性有两个可选值: relative和absolute, relative表示<div>标记的位置是相对于它所在的窗口的, absolute 表示<div>标记的位置是绝对的, 可以通过表 1.10 所示的属性进行设置。

表1.10 属性说明

属 性	说 明
left	相对于窗口左边的位置，单位为像素（pixels）
top	相对于窗口上边的位置，单位为像素（pixels）
width	<div>标记的宽度。所有在<div>标记里的文字或 HTML 元素都包含在里面
height	<div>标记的高度。该属性很少使用，除非想要对层进行分割
clip:rect(top,right,bottom,left)	给出层的可见部分。该属性可使得<div>标记显示为一个可以定义得很准确的方块区域
visibility	隐藏或显示<div>标记中的元素，其值为 visible, hidden, inherit
z-index	<div>标记的立体位置，值越大<div>标记的位置越高
background-color	<div>标记的背景颜色
layer-background-color	Netscape 的<div>标记的背景颜色
background-image	<div>标记的背景图像
layer-background-image	Netscape 的<div>标记的背景图像

style属性的另一个常用功能是控制<div>标记的display属性，用于设置元素的浮动特征，当display被设置为block（块）时，容器中所有元素都将会被当作一个单独的块放入到页面中；将display设置为inline，将使其行为和元素inline一样，即使是普通的块元素也会被组合成像<span>那样的输出流输出到页面上；将display设置为none，<div>元素就像从页面中被移走一样，它下面的所有元素都会自动跟上填充。

说明：如果在网页中，单独使用<div>而不加任何CSS-P（Cascading Style Sheets Positioning的简写，是CSS的一个扩展，它可以控制任何东西在网上的位置），它在网页中的效果与使用<p>标记是一样的。

### 实现过程

（1）在要弹出无边框窗口页面的最底部加入一个DIV层，其name属性的值为User，并通过其 style 属性控制层的大小和层的隐藏，该层中的内容为无边框窗口要显示的内容。User 层的完成代码如下。

```
<!--应用层设计用户登录表单页面开始-->
<div id="User" style="position:absolute;width:240px;
height:139px;display:none;">
```

```

<table width="240" height="139" border="0" align="center"
cellpadding="-2" cellspacing="-2">
  <tr>
    <td height="139" align="center"><form name="form_U"
method="post" action="#">
      <table width="220" height="115" bgcolor="#FFFFFF"
border="0" align="center" cellpadding="-2"
cellspacing="-2" class="tableBorder">
        <tr align="center" valign="middle">
          <td height="24" colspan="2"
background="Images/bg_login.gif"><font
color="#505875">=== 用户登录 === </font> </td>
        </tr>
        <tr>
          <td width="61" height="27" align="right"
valign="middle">用户名: </td>
          <td width="157" valign="middle"><input name="UID"
type="text" maxlength="20"></td>
        </tr>
        <tr>
          <td height="27" align="right" valign="middle">密
&nbsp;码: </td>
          <td valign="middle"><input name="PWD"
type="password" maxlength="20"
onKeyDown="if(event.keyCode==13)mycheck(form_U,'用户
名')"></td>
        </tr>
      </table>
    </td>
  </tr>

```

```

        <tr align="center" valign="middle">
            <td height="27" colspan="2"><input name="Submit"
            type="button" class="btn_grey" value="登录"
            onClick="mycheck(form_U,'用户名')">
                &nbsp;
            <input name="Submit2" type="button"
            class="btn_grey" value="关闭"
            onClick="javascript:User.style.display='none';"></td>
        </tr>
    </table>
</form>
</td>
</tr>
</table>
</div>
<!--应用层设计用户登录表单页面结束-->

```

注意：不要忘记在层中加入控制层隐藏的按钮（即关闭按钮），该按钮执行的操作是将User层的display样式设置为none。

（2）在页面中加入“用户登录”超级链接，该超级链接执行的操作是调用自定义JavaScript函数Myopen()显示User层。关键代码如下。

```
<a href="#" onClick="Myopen(User)">用户登录</a>
```

（3）编写自定义的JavaScript函数Myopen()，用于控制层的居中显示，代码如下。

```

<script language="javascript">
function Myopen(divID)
{

```

```
//根据传递的参数确定显示的层
divID.style.display='block';
divID.style.left=(document.body.clientWidth-240)/2;
divID.style.top=(document.body.clientHeight-139)/2;
}
</script>
```

举一反三

根据本实例，读者可以：

制作在线影院中的会员登录页面；

制作在线论坛中的版主登录页面；

开发游戏网站。

## 实例043 应用JavaScript实现指定尺寸无边框无滚动条窗口

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\043

实例说明

通常情况下，网页中弹出的窗口都是包括标题栏和有边框的，这样的窗口有时会影响页面的整体效果。根据实际情况将弹出的窗口设置为无边框的窗口，会让页面更加引人注目，同时也可统一网站风格。例如，明日实业网站的管理员登录窗口就是无边框窗口，运行本实例，单击页面下方的“TOP”超级链接，即可弹出无边框的居中显示的管理员登录窗口，如图1.49所示。这样可以使页面看起来更和谐统一。需要说明的是该实例只适合IE 6.0 及以下版本，对于已经安装更新版本（如更新版本为SP1）的浏览器弹出的窗口不能指定尺寸，只能全屏显示。



图1.49 应用JavaScript实现指定尺寸无边框无滚动条窗口

### 技术要点

本实例主要应用window对象的open()方法，弹出一个最大化的窗口，再通过window对象的resizeTo()方法和moveTo()方法控制窗口的尺寸和显示位路。由于window对象的open()方法在实例001中已经介绍，下面对window对象的resizeTo()方法进行详细介绍。

功能：将当前窗口改变成(x, y)大小。

语法格式：

```
window.resizeTo(x, y)
```

参数说明如下。

- x：表示水平宽度，单位为像素。
- y：表示垂直高度，单位为像素。

注意：window对象的resizeTo()方法在IE 6.0 以上版本中，只能改变以一般状态显示的窗口的尺寸，不能改变最大化窗口（用window对象的open()方法中设置了fullscreen=1参数的窗口）的尺寸。

### 实现过程

(1) 在实例的 index.htm 页面中，添加控制窗口弹出的超级链接，本例中采用的是图片热点超级链接，该超级链接执行的操作是调用自定义的JavaScript函数manage()。manage()函数实现了在网页中弹出一个以最大化方式显示的窗口，该窗口的内容是login.htm页的内容。关键代码如下。

```

<script language="javascript">
function manage()
{
var
w=window.open(' login.htm', '', ' fullscreen=1, scrollbars=0' );
}
</script>
<map name="Map">
<area shape="rect" coords="73, 13, 135, 43" href="#"
onClick="manage()">
</map>
```

(2) 制作弹出的窗口页面login.htm，在该页面中添加控制窗口位路和大小的JavaScript代码。代码如下。

```
<script language="javascript">
self.resizeTo(321, 203);
width=screen.width;
```

```
height=screen.height;  
self.moveTo((width-240)/2, (height-139)/2);  
</script>
```

(3) 在 login.htm页面的<body>标记中还需要加入“scroll=no”，否则弹出的窗口会带有滚动条。代码如下。

```
<body scroll=no>
```

举一反三

根据本实例，读者可以：

开发宽带影院中的在线视听窗口或歌词查看窗口；

开发游戏网站中的用户登录窗口。

## 1.8 水平导航条应用

网站导航条对于每个网站都是必不可少的，它相当于城市中的路标，可以指引浏览者找到自己热衷的内容。水平导航条有很多种，如本节中介绍的带图标的文字导航条、Flash导航条、图片按钮导航条、质感效果的导航条等。下面将介绍网站中常用的几种导航条。

### 实例044 图片按钮导航条

这是一个应用较为广泛的实例

实例位置：光盘\mingrisoft\01\044

实例说明

网站网页形式的多样性是吸引浏览者的一个重要途径，增强页面的立体感是网站多样性的一种表现，设计按钮形式的导航条使导航效果更加明显，给人以更加真实的感觉。当浏览者单击该链接按钮时，给浏览者一种直接触摸按钮的感觉。运行本实例，如图1.50所示。



图1.50 图片按钮导航条

## 技术要点

本实例主要应用图片热点超级链接实现，通过HTML的<map>标记可以为图片添加热点。为图片设置热点超级链接的语法格式如下。

```
<img src= “file_name” usemap= “#MapName” >  
<mapname= “MapName” >  
<area shape= “value” coords= “坐标” href= “URL” alt= “描述文字” >  
”  
</map>
```

标记<map>的属性如表1.11所示。

表1.11 <map>标记的属性说明

属 性	说 明
name	图片热点的名称
shape	定义图片热点区域的名称
coords	设定区域坐标
href	设定区域的链接地址
alt	设定区域链接的描述文字

在<map>标记中，根据属性shape的取值不同，相应坐标的设定也不同。下面介绍属性shape的3种取值以及相应坐标的设定。

- 设定属性shape 的属性值为rect。

属性shape取值为rect，表示矩形区域，属性coords的坐标形式为“x1, y1, x2, y2”。其中，x1、y1代表矩形左上角的x坐标和y坐标，x2、y2代表矩形右下角的x坐标和y坐标。

- 设定属性shape 的属性值为circle。

属性shape取值为circle，表示圆形区域，属性coords的坐标形式为“x, y, r”。其中，x、y为圆心坐标，r为圆的半径。

- 设定属性shape 的属性值为poly。

属性shape取值为poly，表示多边形区域，属性coords的坐标形式为“x1, y1, x2, y2, ..., xn, yn”。其中，xn、yn代表构成多边形每一点的坐标值，n的取值为1, 2, 3, ..., n，多边形有几边就有几对x、y坐标。

注意：应在<body>区域中的任一位置定义<map name=mapname></map>标签，name 是图片热点的名称。

### 实现过程

(1) 在图像制作软件中设计一张带有超级链接按钮的图片。

(2) 在Dreamweaver的菜单中，选择“插入”→“图像”命令，将插入一张图片，使用鼠标左键单击选中此图片，此时在“属性”面板中可以看到此图片的相关属性。初次使用“属性”面板时其默认为“简化”形式，单击右下方的向下箭头可打开其扩展属性模式。

(3) 选择图片后，在其“属性”面板中选择“矩形热点工具”，为图片的每个按钮添加热点超级链接。选中热点区域，在其“属性”面板中的“超级链接”文本框中输入超级链接的地址，本实例中设路为空链接“#”，依次应用每个热点的onClick事件，调用JavaScript脚本弹出相应提示对话框。代码如下。

```
<map name="Map">
  <area shape="rect" coords="292,17,382,57" href="#"
  onclick="javascript:alert('单击该按钮返回到商城首页')">
  <area shape="rect" coords="410,19,500,59" href="#"
  onclick="javascript:alert('单击按钮进入到新品上市页面')">
  <area shape="rect" coords="525,19,616,60" href="#"
  onclick="javascript:alert('单击按钮进入到特价商品页面')">
  <area shape="rect" coords="640,18,734,59" href="#"
  onclick="javascript:alert('单击按钮进入到购物推车页面')">
```

```
<area shape="rect" coords="756,19,848,59" href="#"
onclick="javascript:alert('单击按钮进入到订单查询页面')">
</map>
```

举一反三

根据本实例，读者可以：

在游戏网站中实现图形化的的导航条；

实现公司门户网站中的导航条；

实现个人主页中的导航条。

## 实例045 导航条的动画效果

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\01\045

实例说明

导航条是网站设计不可缺少的元素之一，它能正确地引导浏览者查找需要的资料，成为浏览者的网站路标。同时网页导航条的设计风格也能影响页面的整体效果，对于一个静态元素居多的页面，可以为导航条添加动态效果，这样可以使整个网站不至于太呆板，增强网页的欣赏价值。例如，在明日实业网站中，为导航条设计了动画效果，这样只要用户将鼠标移动到任意一个导航按钮上时，该按钮都会突出显示，鼠标移走后，又恢复为原来位路，如图1.51所示。



图1.51 导航条的动画效果

技术要点

本实例主要是通过Image对象的鼠标事件控制Image对象src属性的值实现的。

在JavaScript里提供了对图像进行处理的专用对象Image来装入文档的图形。Image对象类型与其他对象的差别在于，其允许通过构造器显示和创建新的Image对象类型，创造和预装入的图形之前并非是Web页面的组成部分。Image对象存在浏览器的缓冲区中，用于替换已经显示的图像。

用构造器创建图形的语法格式如下。

```
ObjImg=new image() //创建了一个新的Image对象，并将其赋予变量ObjImg
```

```
ObjImg.src="图片文件相对路径" //设置Image对象的src属性
```

在利用Image对象编程之前，首先利用IMG标志显示属性。在HTML中，要显示图片可以用如下语句来实现。

```
<IMG src="pictureName.gif" name="ImgName">
```

在JavaScript中可以用下面的语句访问Image对象。

```
document.Img.src=" menu_01.gif";
```

实现过程

(1) 准备14张图片，其中7张是鼠标移出时显示的图片，图片的名称为“menu\_0”+1~7的数字+“.gif”，另7张是鼠标移入时显示的图片，图片的名称为“menu\_0”+1~7的数字+“\_over.gif”。

注意：如果图片menu\_01.gif代表的是公司首页的图片，则menu\_01\_over.gif代表的图片一定也是公司首页的图片。

(2) 将过程(1)所准备的7张鼠标移出时显示的图片按顺序插入到页面中的适当位路，并设路其鼠标事件 onMouseMove 和 onMouseout 执行的操作，这里分别调用两个不同的自定义JavaScript函数，关键代码如下。

```
<td width="95" align="center">
```

```

</td>
<td width="95" align="center">

</td>
<td width="94" align="center">

</td>
<td width="94" align="center">

</td>
<td width="94" align="center">

</td>
<td width="94" align="center">

    </td>
    <td width="95" align="center">
        
    </td>

```

(3) 编写自定义的JavaScript函数move()和out()。move()用于设路鼠标移入导航按钮上时显示的图片，out()用于设路鼠标移出导航按钮上时显示的图片，代码如下。

```

<script language="javascript">
//鼠标移动效果
var A_Img=new Image();
function move(image,num)
{
    image.src='Images/top/menu_0'+num+'_over.gif';
}
function out(image,num)
{
    image.src='Images/top/menu_0'+num+'.gif';
}
</script>

```

举一反三

根据本实例，读者可以：

开发企业门户网站的导航条；

开发在线影院的导航条；

开发电子商务网站的导航条。

## 实例046 不用图片实现质感导航条

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\01\046

实例说明

很多网站都采用了质感比较强的导航条，用以增强网页的视觉冲击力，大多数的情况下这些导航条的背景都是通过图像制作软件制作的质感图片实现的，应用这种方法制作的网站，如果想改变网页的主体色调会比较麻烦，需要再次修改图片。下面介绍一种只需修改网页代码就可改变导航条背景色的方法，那就是应用 CSS 样式实现。运行本实例，如图 1.52 所示，导航条背景的质感效果就是通过CSS样式实现的。



图1.52 不用图片实现质感导航条

技术要点

本实例主要通过CSS的波浪变形效果滤镜Wave实现。Wave滤镜的基本语法如下。

```
{filter:wave(add=add, freq=freq, lightstrength=strength, phase=phase, strength=strength)}
```

参数说明如下。

- add: 表示是否要把对象按照波形样式打乱，值为true 或 false。
- freq: 表示波形的频率，也就是指定在对象上一共需要产生多少个完整的波纹。
- lightstrength: 表示可以对波纹增强光影效果，范围是0~100。
- phase: 用来设置正弦波的偏移量。

● strength: 代表振幅的大小。

### 实现过程

(1) 在网页导航条的位置添加一个单线边框的表格，并将表格的背景颜色设置为页面的主色调，同时添加导航文字。代码如下。

```
<tablewidth="100%"height="27" border="1"cellpadding="-2"
cellspacing="-2"
bordercolor="#FFFFFF"bordercolordark="#FFFFFF"bordercolor
light="#006666"
bgcolor="#336666">
  <tr align="center">
    <tdvalign="bottom"><a href="#" class="shadow">公司首页
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">公司简介
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">产品介绍
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">技术支持
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">公司荣誉
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">网站地图
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">招贤纳士
</a></td>
    <tdvalign="bottom"><a href="#" class="shadow">与我联系
</a></td>
  </tr>
```

</table>

(2) 为过程(1)中制作的导航条添加质感效果,即在<table>标记中添加style属性,该属性值为Wave滤镜的代码,具体代码如下。

```
style="filter:wave(add=false,freq=1,lightstrength=50,phase=50,strength=10)"
```

举一反三

根据本实例,读者可以:

为图片添加特殊的效果;

实现网站的自动换肤功能。

## 1.9 下拉菜单式导航条

下拉菜单式导航条可以使整个页面看起来更规范，下拉菜单式导航条可以分为二级导航菜单、半透明背景的下拉菜单、弹出式下拉菜单等。下面通过具体实例详细介绍实现下拉菜单式导航条的方法。

### 实例047 二级导航菜单

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\01\047

实例说明

网站中的导航条在整个网站中充当着网站路标的作用，如果一个网站的导航条层次混乱，或是分类不清晰，就会给网站的浏览者带来诸多不便，因此设计一个层次分明的网站导航对于一个网站及为重要。例如，企业进销存管理系统中的菜单大致可以分为“基础信息”、“采购管理”、“库存管理”等，但是在基础信息中还要分为“客户信息管理”、“商品信息管理”、“供应商信息管理”等，这就需要建立二级导航菜单。运行本实例，将鼠标移动到一级菜单“基础信息”超级链接上时，在其下方将显示与其对应的二级导航菜单，如图1.53所示。



图1.53 二级导航菜单

技术要点

实现二级导航菜单主要应用JavaScript的switch语句确定要显示的二级菜单的内容，switch语句的语法格式如下：

```
switch (expression)
{
case label :
statement;
break;
case label :
statement;
break;
...
default: statement;
}
```

说明：

①JavaScript 执行switch 语句时，首先计算switch后括号内表达式的值。当此表达式的值与某个case后面的常数表达式的值相等时，就执行此case后的语句。如果所有case后的常数表达式的值都不等于此表达式的值，就执行default后面的语句。

②当执行某个 case 后的语句时如果遇到 break 语句，就结束这条 switch 语句的执行，转去执行这条switch之后的语句。break语句也可以省略，但这时程序会一直执行到这条switch语句的结束标记，即右大括号“}”。

注意：通常情况下，都应该在switch语句的每个分支后面加上break语句，使JavaScript只执行匹配的分支。

实现过程

(1) 在网页中适当的位路添加一级导航菜单，本实例中的一级导航菜单是由一系列空的超级链接组成的，这些空的超级链接执行的操

作是调用自定义的JavaScript函数Fsubmenu()显示对应的二级菜单，在调用时需要传递一个标记主菜单项的参数，关键代码如下：

```
<td width="90%" class="shadow">
<a href="#" onClick="Fsubmenu('基础')">基础信息</a>
|<a href="#" onClick="Fsubmenu('采购')">采购管理</a>
|<a href="#" onClick="Fsubmenu('库存')">库存管理</a>
|<a href="#" onClick="Fsubmenu('销售')">商品销售</a>
|<a href="#" onClick="Fsubmenu('查询')">查询统计</a>
|<a href="#" onClick="Fsubmenu('往来')">往来管理</a>
|<a href="#" onClick="Fsubmenu('系统')">系统设置</a>
</td>
```

(2) 在网页中显示二级菜单的位路添加一个名为“submenu”的DIV层，代码如下。

```
<div id="submenu" class="word_yellow">&nbsp;&nbsp;&nbsp;</div>
```

(3) 编写自定义 JavaScript函数Fsubmenu()，用于根据鼠标移动到某一级菜单时传递的参数值在页面的相应位路显示对应的二级菜单，代码如下。

```
<script language="javascript">
functionFsubmenu(value)
{
    switch (value)
    {
        case "基础":
            submenu.innerHTML="<a href=' #' target=' mainF' >客户
            信息管理</a>|<a href=' #' target=' mainF' >商品信息管理
            </a>|<a href=' #' target=' mainF' >供应商信息管理</a>|
            <a href=' #' target=' mainF' >客户信息查询</a>|<a href=' #'
```

```
target=' mainF' >商品信息查询</a>|<a href=' #'
target=' mainF' >供应商信息查询</a>";
    break;
    case "采购":
        submenu.innerHTML="<a href=' #' target=' mainF' >商品
采购</a>|<a href=' #' target=' mainF' >采购查询</a>";
        break;
    case "库存":
        submenu.innerHTML="<a href=' #' target=' mainF' >商品
入库</a>|<a href=' #' target=' mainF' >商品入库退货</a>|<a
href=' #' target=' mainF' >库存查询</a>|<a href=' #'
target=' mainF' >价格调整</a>";
        break;
    case "销售":
        submenu.innerHTML="<a href=' #' target=' mainF' >商品
销售</a>|<a href=' #' target=' mainF' >商品销售退货</a>|<a
href=' #' target=' mainF' >";
        break;
    case "查询":
        submenu.innerHTML="<a href=' #' target=' mainF' >销售
信息查询</a>|<a href=' #' target=' mainF' >销售退货查询
</a>|<a href=' #' target=' mainF' >商品入库查询</a>|<a
href=' #' target=' mainF' >入库退货查询</a>|<a href=' #'
target=' mainF' >商品销售排行</a>|<a href=' #'
target=' mainF' >年销售额分析</a>";
        break;
    case "往来":
```

```

        submenu.innerHTML="<a href=' #' target=' mainF' >商品
销售结账</a>|<a href=' #' target=' mainF' >入库退货结账
</a>|<a href=' #' target=' mainF' >销售结账查询</a>|<a
href=' #' target=' mainF' >商品入库结账</a>|<a href=' #'
target=' mainF' >销售退货结账</a>|<a href=' #'
target=' mainF' >入库结账查询</a>";
        break;
    case "系统":
        submenu.innerHTML="<a href=' #' target=' mainF' >添加
操作员</a>|<a href=' #' target=' mainF' >更改操作员密码
</a>|<a href=' #' target=' mainF' >删除操作员</a>";
        break;
    }
}
</script>

```

举一反三

根据本实例，读者可以：

开发企业网站后台管理模块的导航条；

开发电子商务网站后台管理模块的导航条。

## [实例048 半透明背景的下拉菜单](#)

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\048

实例说明

网页设计中采用半透明的效果给人以亦真亦幻的感觉，使整个网站更具吸引力。因此本实例将图书馆管理系统的下拉菜单设路为半透

明效果。运行本实例，将鼠标移动到“系统设置”等导航链接上时，在其下方将显示出半透明的下拉式菜单，透过此下拉菜单仍可以看到页面上的内容，例如将鼠标移动到“系统设置”超级链接上时，其下方会显示出半透明背景的下拉菜单，如图1.54所示。



图1.54 半透明背景的下拉菜单

### 技术要点

实现半透明背景的下拉菜单，首先需要在页面中实现下拉菜单，然后再通过设置下拉菜单的 CSS 样式实现半透明效果。实现下拉菜单的半透明效果可以应用 CSS 样式的透明效果滤镜alpha实现。透明效果滤镜alpha的语法如下。

```
{filter:alpha(opacity=opacity, finishopacity=finishopacity, style=style, startx=startx, starty=starty, finishx=finishx, finishy=finishy)}
```

### 功能：

alpha属性是把一个目标元素与背景混合。设计者可以指定数据来控制混合的程度。这种“与背景混合”通俗地说是一个元素的透明度。通过指定坐标可以指定点、线、面的透明度。

参数说明如下。

● opacity: 代表透明度水准; 默认的范围是0~100, 其实是百分比的形式, 即0 代表完全透明, 100代表完全不透明。

● finishopacity: 可选, 如果想要设置渐变的透明效果, 可以使用该参数指定结束时的透明度。范围也是0~100。

● style: 指定透明区域的形状特征, 其中, 0 代表统一形状, 1代表线形, 2 代表放射状, 3代表长方形。

● startx: 代表渐变透明效果的开始x坐标。

● starty: 代表渐变透明效果的开始y坐标。

● finishx: 代表渐变透明效果结束x坐标。

● finishy: 代表渐变透明效果结束y坐标。

#### 实现过程

(1) 编写实现下拉菜单的 JavaScript 代码, 该代码中主要包括控制下拉菜单的显示和隐藏的自定义函数。由于该段代码属于通用型代码, 所以将其保存在一个单独的 .JS文件中, 名称为menu. JS, 在使用该代码的页面中只需加入如下代码即可。

```
<script src="JS/menu. JS"></script>
```

menu. JS文件的关键代码如下。

```
var menuOffX=0
```

```
//菜单距链接文字最左端距离
```

```
var menuOffY=18
```

```
//菜单距链接文字顶端距离
```

```
var fo_shadows=new Array()
```

```
var linkset=new Array()
```

```
var
```

```
IE4=document.all&&navigator.userAgent.indexOf("Opera")==-1
```

```
var netscape6=document.getElementById&&!document.all
```

```
var netscape4=document.layers
```

```

//显示下拉菜单的方法
function showmenu(e, vmenu, mod)
{
    if
(!document.all&&!document.getElementById&&!document.layers)
        return
    which=vmenu
    clearhidemenu()
    IE_clearshadow()
    menuobj=IE4?document.all.popmenu :netscape6?
document.getElementById("popmenu") :netscape4?
document.popmenu : ""
    menuobj.thestyle=(IE4||netscape6)?menuobj.style
:menuobj
    if (IE4||netscape6)
        menuobj.innerHTML=which
    else{
        menuobj.document.write(' <layer name="other"
bgColor="#E6E6E6"width="165"
onmouseover="clearhidemenu()"
onmouseout="hidemenu()">' +which+' </layer>')
        menuobj.document.close()
    }
    menuobj.contentwidth=(IE4||netscape6)?
menuobj.offsetWidth :menuobj.document.other.document.width
    menuobj.contentheight=(IE4||netscape6)?
menuobj.offsetHeight

```

```

:menuobj. document. other. document. height
    eventX=IE4?event.clientX :netscape6?e.clientX : e.x
    eventY=IE4?event.clientY :netscape6?e.clientY : e.y
    var rightedge=IE4?document.body.clientWidth-eventX
:window. innerWidth-eventX
    varbottomedge=IE4?document.body.clientHeight-eventY
:window. innerHeight-eventY
    if (rightedge<menuobj.contentwidth)
        menuobj.thestyle.left=IE4?
document.body.scrollLeft+eventX-
menuobj.contentwidth+menuOffX : netscape6?
    window.pageXOffset+eventX-menuobj.contentwidth :
eventX-menuobj.contentwidth
    else
        menuobj.thestyle.left=IE4?
IE_x(event.srcElement)+menuOffX :netscape6?
window.pageXOffset+eventX : eventX
    if (bottomedge<menuobj.contentheight&&mod!=0)
        menuobj.thestyle.top=IE4?
document.body.scrollTop+eventY-menuobj.contentheight-
event.offsetY+menuOffY-23 :
    netscape6?window.pageYOffset+eventY-
menuobj.contentheight-10 : eventY-menuobj.contentheight
    else
        menuobj.thestyle.top=IE4?
IE_y(event.srcElement)+menuOffY :netscape6?
window.pageYOffset+eventY+10 :eventY

```

```
    menuobj.thestyle.visibility="visible"
    IE_dropshadow(menuobj, "#999999", 3)
    return false
}
function IE_y(e)
{
    var t=e.offsetTop;
    while(e=e.offsetParent)
    {
        t+=e.offsetTop;
    }
    return t;
}
function IE_x(e)
{
    var l=e.offsetLeft;
    while(e=e.offsetParent) {
        l+=e.offsetLeft;
    }
    return l;
}
function IE_dropshadow(el, color, size)
{
    var i;
    for (i=size; i>0; i--){
        var rect=document.createElement('div');
        var rs= rect.style
```

```

    rs.position= 'absolute' ;
    rs.left= (el.style.posLeft+i)+'px' ;
    rs.top= (el.style.posTop+i)+'px' ;
    rs.width= el.offsetWidth+'px' ;
    rs.height= el.offsetHeight+'px' ;
    rs.zIndex= el.style.zIndex-i;
    rs.backgroundColor= color;
    var opacity=1-i/(i+1);
    rs.filter= 'alpha(opacity='+ (100 *opacity)+' )';
    fo_shadows[fo_shadows.length]= rect;
}
}
function IE_clearshadow()
{
    for(var i=0;i<fo_shadows.length;i++){
        if (fo_shadows[i])
            fo_shadows[i].style.display="none"
    }
    fo_shadows=newArray();
}
//隐藏菜单的方法
functionhidemenu()
{
    if (window.menuobj)
        menuobj.thestyle.visibility=(IE4||netscape6)?
"hidden" : "hide"
    IE_clearshadow()
}

```

```

}
function dynamichide(e)
{
    if (IE4&&!menuobj.contains(e.toElement))
        hidemenu()
    else if (netscape6&&e.currentTarget!= e.relatedTarget&&
!contains_netscape6(e.currentTarget, e.relatedTarget))
        hidemenu()
}
function delayhidemenu()
{
    if (IE4||netscape6||netscape4)
        delayhide=setTimeout("hidemenu()", 500)
}
function clearhidemenu()
{
    if (window.delayhide)
        clearTimeout(delayhide)
}
function highlightmenu(e, state)
{
    if (document.all)
        source_el=event.srcElement
    else if (document.getElementById)
        source_el=e.target
    if (source_el.className=="menuitems") {
        source_el.id=(state=="on")? "mouseoverstyle" : ""
    }
}

```

```

    }
    else{
        while(source_el.id!="popmenu")
        {
            source_el=document.getElementById?
            source_el.parentNode : source_el.parentElement
            if (source_el.className=="menuitems"){
                source_el.id=(state=="on")? "mouseoverstyle" : ""
            }
        }
    }
}
//鼠标移入时设置菜单背景
function overbg(tdbg)
{
    tdbg.style.background='url(Images/item_over.gif)'
    tdbg.style.border='#9CA6C6 1px solid'
}
//鼠标移出时设置菜单背景
function outbg(tdbg)
{
    tdbg.style.background='url(Images/item_out.gif)'
    tdbg.style.border=''
}
var sysmenu='<tablewidth=80><tr><td
id=libraryonMouseOver=overbg(library)onMouseOut=outbg(library
)><ahref=library.do?

```

```

        action=libraryQuery>图书馆信息</a></td></tr>\
        <tr><td id=manager onMouseOver=overbg(manager)
onMouseOut=outbg(manager)><a
        href=manager.do?action=managerQuery>管理员设置</a></td>
</tr>\
        <tr><td id=paraonMouseOver=overbg(para)
onMouseOut=outbg(para)><a href=parameter.do?
action=parameterQuery>参数设
        置</a></td></tr>\
        <tr><td
id=bookcaseonMouseOver=overbg(bookcase) onMouseOut=outbg(bookc
ase)><a href=bookCase.do?action=bookcase
        Query>书架设置</a></td></tr>\
</table>'
'''
var sortmenu=' <tablewidth=100><tr><td id=readerSort
onMouseOver=overbg(readerSort) onMouseOut=outbg(readerSort)>
<a
        href=borrow.do?action=readerBorrowSort>读者借阅排行榜</a>
</td></tr>\
        <tr><td id=bookSort onMouseOver=overbg(bookSort)
onMouseOut=outbg(bookSort)><a href=borrow.do?action=bookBorrow
        Sort>图书借阅排行榜</a></td></tr>\
</table>'

```

(2) 在页面中加入导航链接，同时设路其鼠标的onMouseover和onMouseout事件，分别用于控制鼠标移入导航链接上时显示下拉菜单和鼠标移出导航链接时隐藏下拉菜单。关键代码如下。

```
<a onmouseover=showmenu(event, sysmenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >系统设置</a> !
```

```
<a onmouseover=showmenu(event, readermenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >读者管理</a> !
```

```
<a onmouseover=showmenu(event, bookmenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >图书管理</a> !
```

```
<a onmouseover=showmenu(event, borrowmenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >图书借还</a> !
```

```
<a onmouseover=showmenu(event, querymenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >系统查询</a> !
```

```
<a onmouseover=showmenu(event, sortmenu)
onmouseout=delayhidemenu() class='navlink'
style="CURSOR:hand" >排行榜</a>
```

(3) 在CSS样式表文件中加入 `menuskin`类选择符, 用于设路下拉菜单的半透明效果, 代码如下。

```
.menuskin
{
BORDER:#6666661px solid;VISIBILITY:hidden;
POSITION: absolute;
/*BACKGROUND-COLOR:#EFEFEF;*/
background-image:url("../Images/item_out.gif");
background-repeat : repeat-y;
```

```
Filter:Alpha(Opacity=85);  
}
```

(4) 在页面中加入DIV层，名称为“popmenu”，用于引用menuskinner类选择符设置下拉菜单的半透明效果。代码如下：

```
<div class=menuskinner id=popmenu  
onmouseover="clearhidemenu();highlightmenu(event,'on')"  
onmouseout="highlightmenu(event,'off');dynamichide(ev  
ent)"  
style="Z-index:100;position:absolute;"></div>
```

举一反三

根据本实例，读者可以：

开发办公自动化管理系统的导航菜单；

开发物资管理系统的导航菜单；

开发网站后台管理页面的导航菜单。

## 实例049 展开式导航条

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\049

实例说明

对于一个企业网站来说，“联系我们”超级链接是必不可少的。为了使其更加引人注目，可以将其制作成具有展开式动画效果的导航条。运行本实例，读者可以看到页面右侧的“联系我们”导航条是慢慢展开的，并且每次刷新页面时，都会以同样的动画效果展开，展开效果如图1.55、图1.56所示。



图1.55 展开式导航条



图1.56 展开后的导航条

### 技术要点

本例主要通过自定义JavaScript函数控制图片（Image）对象的height属性值实现。为了实现动画效果，在JavaScript函数中还需要应用setTimeout方法延迟执行改变图片高度的操作。

注意：调整setTimeout()方法的运行间隔时间的值，可以控制展开的速度，该值以毫秒为单位。

### 实现过程

(1) 在网页中要显示“联系我们”导航条的位置插入一张全部展开后的“联系我们”导航图片，在需要设置超级链接的文字上添加图片热点并设置相应的超级链接。

(2) 将图片的height属性值设置为0，name属性设置为our。

(3) 编写自定义的JavaScript函数ourmove()，用于实现图片的展开效果。ourmove()函数的代码如下：

```
<script language="javascript">  
function ourmove()
```

```
{
  if(our.height<163)
  {
    our.height=our.height+9
    setTimeout(ourmove,1)
  }
}
</script>
```

(4) 在当前页面的最底部，也就是</body>标记之前，添加如下代码调用 ourmove()，实现导航条的展开式动画效果。

```
<script language="javascript">
ourmove();
</script>
```

注意：一定要将调用ourmove()函数的代码放在</body>标记之前，否则页面的其他内容就要等到动画显示完毕才可以显示，这样就达不到预期的效果了。

举一反三

根据本实例，读者可以：

开发企业门户网站的“联系我们”导航条；

开发网站中的二级导航菜单。

## [实例050 用层制作下拉菜单1](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\01\050

实例说明

本实例通过层制作了一个下拉菜单，当用鼠标指向下拉菜单时，会根据指定的菜单在下面以下拉的方式显示下拉列表，并可以通过单击下拉列表中的选项进入指定的网页，运行结果如图1.57、图1.58、图1.59所示。



图1.57 下拉菜单的初始效果



图1.58 弹出下拉菜单的菜单项



图1.59 在下拉菜单中选中菜单项

### 技术要点

本实例主要应用CSS样式中的display属性来控制层是否显示，当层显示时，用CSS样式中的rect()方法来对层以y轴逐渐增大的方式进行局部显示。下面对display属性和rect()方法进行详细说明。

display属性的语法格式。

```
display:block | none | inline | compact | marker |  
inline-table | list-item | run-in | table | table-caption |  
table-cell | table-column | table-column-group | table-footer-  
group | table-header-group | table-row | table-row-group
```

display属性的参数取值如表1.12所示。

表1.12 display属性的参数说明

参 数 值	描 述
block	默认值。用该值为对象之后添加新行
none	隐藏对象。与 visibility 属性的 hidden 值不同，其不为被隐藏的对象保留其物理空间
inline	内联对象的默认值。用该值将从对象中删除行
compact	分配对象为块对象或基于内容之上的内联对象
marker	指定内容在容器对象之前或之后。要使用此参数，对象必须和:after 及:before 伪元素一起使用
inline-table	将表格显示为无前后换行的内联对象或内联容器
list-item	将块对象指定为列表项目，并可以添加可选项目标志
run-in	分配对象为块对象或基于内容之上的内联对象
table	将对象作为块元素级的表格显示
table-caption	将对象作为表格标题显示
table-cell	将对象作为表格单元格显示
table-column	将对象作为表格列显示
table-column-group	将对象作为表格列组显示
table-header-group	将对象作为表格标题组显示
table-footer-group	将对象作为表格脚注组显示
table-row	将对象作为表格行显示
table-row-group	将对象作为表格行组显示

功能：设路或检索对象是否及如何显示。

rect方法的语法格式：

clip : auto | rect( number number number number )

参数说明如下。

- auto：对象无剪切。

- rect (number number number number)：依据上-右-下-左的顺序提供自对象左上角为 (0,0) 坐标计算的4个偏移数值，其中的任一数值都可用auto替换，即此边不剪切。

功能：检索或设路对象的可视区域。区域外的部分是透明的，必须将 position 的值设为absolute，此属性方可使用。

实现过程

(1) 在页面中添加CSS样式，用于改变下拉菜单的颜色及位路。  
代码如下。

```
<style type="text/css">
```

```

    .menubar {position:absolute;top:10px;width:100px;height:20
px;cursor:default;
    border-width:1px;border-
style:outset;color:#99FFFF;background:#669900}
    .menu {position:absolute;top:32px;width:140px;display:none
;border-width:2px;
    border-style:outset;border-color:white sliver sliver
white;background:#333399;
    padding:15px}
    .menu A {text-decoration:none;color:#99FFFF;}
    .menu A:hover {color: #FFFFFF;}
</style>

```

(2) 在页面中添加一个表格，用于控制下拉菜单的位路及大小。  
代码如下。

```

<table border="0" width="400" cellspacing="0"
cellpadding="0" style="font-size:15px">
<tr>
<td width="20%">
<div align="center" id="Tdiv_1" class="menubar"
onmouseover="divControl(1)"
onmouseout="divControl(0)">
    教育网站
</div>
</td>
<td width="20%">
<div align="center" id="Tdiv_2" class="menubar"
onmouseover="divControl(1)"

```

```

onmouseout="divControl(0)">
    电脑丛书网站
</div>
</td>
<td width="20%">
    <div align="center" id="Tdiv_3" class="menubar"
onmouseover="divControl(1)"
onmouseout="divControl(0)">
        新出图书
    </div>
</td>
<td width="20%">
    <div align="center" id="Tdiv_4" class="menubar"
onmouseover="divControl(1)"
onmouseout="divControl(0)">
        其他网站
    </div>
</td>
</tr>
</tr>
<tr>
<td width="20%">
    <div align="left" id="Div1" class="menu"
onmouseover="keepstyle(this)"
onmouseout="hidediv(this)">
        <a href="#">重庆XX大学</a><br>
        <a href="#">长春XX大学</a><br>

```

```

<a href="#">吉林XX大学</a>
</div>
</td>
<td width="20%">
<div align="left" id="Div2" class="menu"
onmouseover="keepstyle(this)"
onmouseout="hidediv(this)">
<a href="http://www.mingrisoft.com">明日科技VB图书</a>
<br>
<a href="http://www.mingrisoft.com">明日科技Delphi图书
</a><br>
<a href="http://www.mingrisoft.com">明日科技Java图书</a>
</div>
</td>
<td width="20%">
<div align="left" id="Div3" class="menu"
onmouseover="keepstyle(this)"
onmouseout="hidediv(this)">
<a href="http://www.mingrisoft.com">Delphi XX图书</a><br>
<a href="http://www.mingrisoft.com">VB XX图书</a><br>
<a href="http://www.mingrisoft.com">Java XX图书</a>
</div>
</td>
<td width="20%">
<div align="left" id="Div4" class="menu"
onmouseover="keepstyle(this)"
onmouseout="hidediv(this)">

```

```
<a href="http://www.mingrisoft.com">明日科技主页</a><br>
<a href="http://www.mingrisoft.com">明日科技图书网</a>
<br>
<a href="http://www.mingrisoft.com">明日技术支持网</a>
</div>
</td>
</table>
```

(3) 编写用于实现下拉菜单的JavaScript代码。

自定义函数divControl(), 判断是否显示相对应的下拉列表。

```
<script language="Javascript">
function divControl(show) {
    window.event.cancelBubble=true;
    var objID=event.srcElement.id;
    var index=objID.indexOf("_");
    var mainID=objID.substring(0,index);
    var numID=objID.substring(index+1);
    if(mainID=="Tdiv") {
        if(show==1) {eval("showdiv"+"Div"+numID+"");}
        else {eval("hidediv"+"Div"+numID+"");}
    }
}
```

自定义函数displayMenu(), 在显示下拉菜单时, 以下拉方式显示下拉菜单。

```
var nbottom=0, speed=2;
function displayMenu(obj) {
    obj.style.clip="rect(0 100%"+nbottom+"%0)";
    nbottom+=speed;
```

```

    if(nbottom<=100) {
        timerID=setTimeout("displayMenu("+obj.id+")",1");
    }
    else clearTimeout(timerID);
}

```

自定义函数showdiv(), 显示下拉列表的下拉选项。

```

function showdiv(obj) {
    obj.style.display="block";
    obj.style.clip="rect(0 0 0 0)";
    nbottom=5;
    displayMenu(obj);
}

```

自定义函数hidediv(), 隐藏下拉列表的下拉选项。

```

functionhidediv(obj) { //
    nbottom=0;
    obj.style.display="none";
}

```

function keepstyle(obj) { //在下拉菜单中移动时, 保持下拉列表的样式

```

    obj.style.display="block";
}

```

</script>

举一反三

根据本实例, 读者可以:

开发可以隐藏的导航条。

## [实例051 用层制作下拉菜单2](#)

本实例可以美化界面

实例位置：光盘\mingrisoft\01\051

实例说明

本实例在打开窗口时，在页面中会出现一个主菜单，当鼠标指向主菜单时，主菜单中的菜单项会以奇偶的形式从左右两边向中间移动，并且将透明的菜单项逐渐显示为不透明状态，运行结果如图1.60、图1.61所示。

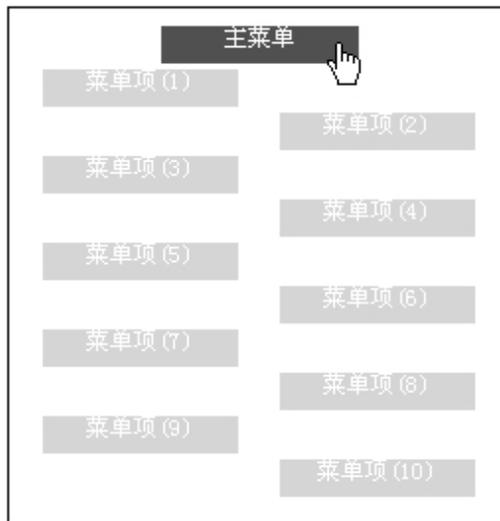


图1.60 下拉菜单的动态效果



图1.61 选择下拉菜单中的菜单项

技术要点

本实例主要应用CSS样式的opacity和pixelLeft属性来控制层的透明度及横向位路。下面对opacity和pixelLeft属性进行详细说明。

● opacity: 代表透明水准, 默认的范围是0~100, 0 代表完全透明, 100 代表完全不透明。

● pixelLeft: 用于改变层的水平位路。只有在 position 属性设为 absolute 时, 该属性才会有效。

### 实现过程

(1) 在页面中添加N个层, 用来显示下拉菜单中的各个选项。代码如下。

```
<div style="position:absolute; left:0px; top:-30px"><div align=center id="menu01" style="position:absolute; left:150px;top:33px;width:100px; height:19px; z-index:1; background-color: #006633; layer-background-color: #006600; border: 1px none#000000;padding-top:3;cursor:hand;font-size:9pt"onmouseover=' stopTim();moveIn()' onmouseout=' runTim()'><font color=white>主菜单</div>  
<div align=center id="div1" style="position:absolute; left:50px; top:55px;width:99px;height:19px; z-index:2;background-color:#00CC66; layer-background-color:#0099CC;border:1pxnone#000000; filter:alpha(opacity=0);padding-top:4; cursor:hand; font-size:9pt; color:#FFFFFF;" onmouseover="this.style.backgroundColor='#006699';stopTim()" onmouseout="this.style.backgroundColor='#00CC66';runTim()">菜单项(1)</div>
```

```

    <div align=center id="div2" style="position:absolute;
left:250px; top:77px;width:99px;height:19px; z-
index:3;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(2)</div>
    <div align=center id="div3" style="position:absolute;
left:50px; top:99px;width:99px;height:19px; z-
index:4;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
    filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(3)</div>
    <div align=center id="div4" style="position:absolute;
left:250px; top:121px;width:99px;height:19px;z-
index:5;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"

```

```
    onmouseover="this.style.backgroundColor=
' #006699' ;stopTim()"
    onmouseout="this.style.backgroundColor=
' #00CC66' ;runTim()">菜单项(4)</div>
    <div align=center id="div5" style="position:absolute;
left:50px; top:143px;width:99px;height:19px;z-
index:6;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
' #006699' ;stopTim()"
    onmouseout="this.style.backgroundColor=
' #00CC66' ;runTim()">菜单项(5)</div>
    <div align=center id="div6" style="position:absolute;
left:250px; top:165px;width:99px;height:19px;z-
index:6;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
' #006699' ;stopTim()"
    onmouseout="this.style.backgroundColor=
' #00CC66' ;runTim()">菜单项(6)</div>
    <div align=center id="div7" style="position:absolute;
left:50px; top:187px;width:99px;height:19px; z-
index:6;background-color:#00CC66; layer-background-
```

```
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(7)</div>
    <divalign=center id="div8" style="position:absolute;
left:250px; top:209px;width:99px;height:19px;z-
index:6;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(8)</div>
    <divalign=center id="div9" style="position:absolute;
left:50px; top:231px;width:99px;height:19px; z-
index:6;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(9)</div>
```

```

    <div align=center id="div10" style="position:absolute;
left:250px; top:253px;width:99px;height:19px;z-
index:6;background-color:#00CC66; layer-background-
color:#0099CC;border:1pxnone#000000;
filter:alpha(opacity=0);padding-top:3;cursor:hand;font-
size:9pt"
    onmouseover="this.style.backgroundColor=
'#006699';stopTim()"
    onmouseout="this.style.backgroundColor=
'#00CC66';runTim()">菜单项(10)<span class="style2"></span>
</div>
</div>

```

(2) 编写用于实现下拉菜单的 JavaScript 代码。

自定义函数 runTim(), 用于在鼠标指向主菜单时显示菜单项。

```

<script language="JavaScript">
varTimer,timIn,timOut;
var leftLine=50;
function runTim() {
window.Timer = setTimeout('moveOut()',300);
}

```

自定义函数 stopTim(), 用于在鼠标移出主菜单时隐藏菜单项。

```

function stopTim() {
    clearTimeout(window.Timer);
}

```

自定义函数 moveIn(), 将菜单项以奇数的形式从左右两边向中间移动, 并使各菜单项的透明度不断加大。

```

function moveIn() {

```

```
    if( leftLine != 150)  {
        div1.style.pixelLeft+=20;div1.filters.alpha.opacity+=
20;
        div2.style.pixelLeft-=
20;div2.filters.alpha.opacity+=20;
        div3.style.pixelLeft+=20;div3.filters.alpha.opacity+=
20;
        div4.style.pixelLeft-=
20;div4.filters.alpha.opacity+=20;
        div5.style.pixelLeft+=20;div5.filters.alpha.opacity+=
20;
        div6.style.pixelLeft-=
20;div6.filters.alpha.opacity+=20;
        div7.style.pixelLeft+=20;div7.filters.alpha.opacity+=
20;
        div8.style.pixelLeft-=
20;div8.filters.alpha.opacity+=20;
        div9.style.pixelLeft+=20;div9.filters.alpha.opacity+=
20;
        div10.style.pixelLeft-=
20;div10.filters.alpha.opacity+=20;
        leftLine+=20;
    }
    else{
        clearTimeout(window.timIn);
        return false;
    }
}
```

```
    timerIn=window.setTimeout('moveIn()',1);  
}
```

自定义函数 `moveOut()`，将菜单项以奇数的形式从中间向左右两边移动，并使各菜单项的透明度不断减小。

```
functionmoveOut() {  
    clearTimeout(window.timIn);  
    if (leftLine != 50) {  
        div1.style.pixelLeft-= 20;div1.filters.alpha.opacity-  
= 20;  
        div2.style.pixelLeft+=20;div2.filters.alpha.opacity-  
=20;  
        div3.style.pixelLeft-= 20;div3.filters.alpha.opacity-  
= 20;  
        div4.style.pixelLeft+=20;div4.filters.alpha.opacity-  
=20;  
        div5.style.pixelLeft-= 20;div5.filters.alpha.opacity-  
= 20;  
        div6.style.pixelLeft+=20;div6.filters.alpha.opacity-  
=20;  
        div7.style.pixelLeft-= 20;div7.filters.alpha.opacity-  
= 20;  
        div8.style.pixelLeft+=20;div8.filters.alpha.opacity-  
=20;  
        div9.style.pixelLeft-= 20;div9.filters.alpha.opacity-  
= 20;  
        div10.style.pixelLeft+=20;div10.filters.alpha.opacity-  
-= 20;
```

```
    leftLine-=20;
}
else{
    clearTimeout(window.timOut);
    return false;
}
timerOut=window.setTimeout("moveOut()",1);
}
</script>
```

举一反三

根据本实例，读者可以：

当鼠标指向下拉菜单时，使下拉菜单中的选项一个一个地显示。

## 1.10 侧导航条设计

在网站设计时有时为了页面的整体效果可以将网站导航条放路在侧面，也称为侧导航条，下面将通过具体实例介绍几种不同样式的侧导航条。

### 实例052 自动隐藏式菜单

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\01\052

实例说明

在一些个性化的网站中，经常会有自动隐藏式的菜单。该类菜单不仅使页面美观别致，而且节省页面空间。当鼠标移到菜单标签上时，该菜单将自动展开；当鼠标离开菜单标签时，该菜单将自动隐藏。本实例制作的自动隐藏式菜单运行结果如图1.62、图1.63所示。



图1.62 自动隐藏菜单

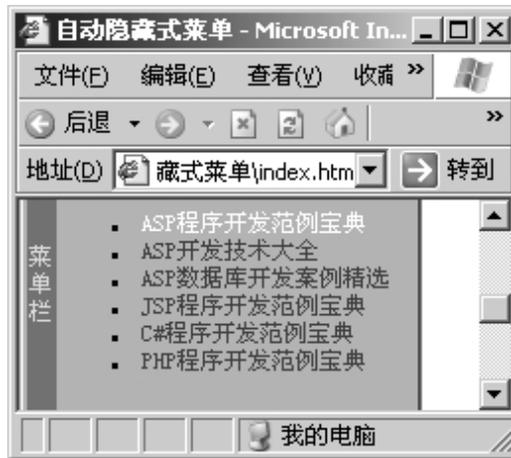


图1.63 菜单展开

### 技术要点

本实例主要应用window对象的setTimeout()方法和clearTimeout()方法实现的。下面将详细介绍这两种方法。

#### (1) setTimeout()方法。

该方法设在指定的若干毫秒后要调用一次的函数。

语法：

```
setTimeout (code, delay)
```

说明：在延迟delay参数指定的毫秒数后执行code字符串指定的代码。在Netscape 4和IE 5及以后的版本中，code参数可以用函数引用来代替字符串，此时，延迟delay毫秒后执行函数。该方法返回一个内部ID值，该值可以传递给clearTimeout()方法来取消延迟的代码执行。

本实例应用该方法的代码如下：

```
c = setTimeout("MenuHide()", 1)
```

#### (2) clearTimeout()方法。

该方法用于取消在指定的若干毫秒后要调用一次的函数。

本实例应用该方法的代码如下。

```
clearTimeout(d)
```

实现过程

(1) 应用JavaScript脚本创建自定义函数实现菜单的显示与隐藏, 代码如下。

```
<script language="javascript">
a=null
b=null
c=null
d=null
Netscape4 = (document.layers)?1:0
IE4 = (document.all)?1:0;
functionMenuHide()
//菜单隐藏
{
  if(Netscape4)
  {
    clearTimeout(d)
    if(menu.left>menuW*-1+20+10 )
    {
      menu.left-=10
      c= setTimeout("MenuHide()", 1)
    }
    else if(menu.left>menuW*-1+20 )
    {
      menu.left--
      c= setTimeout("MenuHide()", 1)
    }
  }
}
else
```

```

{
  clearTimeout(d)
  if(menu.pixelLeft>menuW*-1+20+10 )
  {
    menu.pixelLeft-= 10
    c= setTimeout("MenuHide()", 1)
  }
  else if(menu.pixelLeft>menuW*-1+20 )
  {
    menu.pixelLeft--
    c= setTimeout("MenuHide()", 1)
  }
}
}
functionMenuOut()
//菜单显示
{
  if(Netscape4)
  {
    clearTimeout(c)
    if(menu.left<-10)
    {
      menu.left+=4
      d= setTimeout("MenuOut()", 1)
    }
    else if(menu.left<0)
    {

```

```

        menu.left++
        d= setTimeout("MenuOut()", 1)
    }
}
else
{
    clearTimeout(c)
    if(menu.pixelLeft<-10)
    {
        menu.pixelLeft+=2
        d= setTimeout("MenuOut()", 1)
    }
    else if(menu.pixelLeft<0 )
    {
        menu.pixelLeft++
        d= setTimeout("MenuOut()", 1)
    }
}
}
function fireOver()
{
    clearTimeout(b)
    a= setTimeout("MenuOut()", 10)
}
function fireOut()
{
    clearTimeout(a)

```

```

        b= setTimeout("MenuHide()", 10)
    }
function Init()
{
    if(Netscape4)
    {
        menu=document.D1
        menuW=menu.document.Width
        menu.left=menu.document.Width*-1+20
        menu.onmouseover=menuOut
        menu.onmouseout=menuIn
menu.visibility= "visible"
    }
    else if(IE4)
    {
        menu=dviID.style
        menuW=dviID.offsetWidth
        dviID.style.pixelLeft=dviID.offsetWidth*-1+20
        dviID.onmouseover= fireOver
        dviID.onmouseout= fireOut
        dviID.style.visibility= "visible"
    }
}
</script>

```

(2) 设给<body>标记的onload事件调用Init()函数, 代码如下:

```

<bodyonload="Init()">
<div id="dviID" style="left:0px; top:0px">

```

```
<tableborder="0"width="198"height="500">
  <tr>
    <tdwidth="13" rowspan="2" valign="middle"
bgColor="#2e8b57" style="width:13px">
      <font color="#00ff99">
        菜<br>
        单
        栏</font>
      </td>
      <tdwidth="175" colspan="2" rowspan="2" valign="middle">
        <ul>
          <br/><br/>
          <li><a href="javascript:void(0)">ASP程序开发范例宝典
</a></li>
          <li><a href="javascript:void(0)">ASP开发技术大全</a>
</li>
          <li><a href="javascript:void(0)">ASP数据库开发案例精
选</a></li>
          <li><a href="javascript:void(0)">JSP程序开发范例宝典
</a></li>
          <li><a href="javascript:void(0)">C#程序开发范例宝典
</a></li>
          <li><a href="javascript:void(0)">PHP程序开发范例宝典
</a></li>
        </ul>
      </td>
    </tr>
```

```
<tr>
</tr>
</table>
</div>
</body>
```

举一反三

根据本实例，读者可以：

开发图片资源网站导航栏；

开发任意网站页面导航模块。

## 实例053 收缩式导航菜单

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\01\053

实例说明

在网站中不仅可以设路导航条，而且还可以设路导航菜单。由于菜单内容比较多，同一页面显示比较杂乱，所以很多设计者都采用了收缩式的菜单形式。例如，本实例就应用了收缩式导航菜单，运行本实例，当浏览者单击“网站管理”超级链接时，页面将展开“网站管理”的子菜单，如图1.64所示。当浏览者再次单击“网站管理”超级链接时，展开的收缩式导航菜单又缩回去，页面中不再显示菜单中的内容。



图1.64 收缩式导航菜单

### 技术要点

本实例主要利用显示隐藏表格的功能来实现收缩式导航菜单的功能，单击导航超级链接，显示当前菜单的内容，隐藏上一个显示的菜单，在隐藏菜单时，让其有规律地隐藏进而实现展开或收缩的动画效果。

### 实现过程

(1) 显示菜单的自定义函数的代码如下。

```
<SCRIPT language=javascript>
function show(obj,maxg,obj2)
{
    if(obj.style.pixelHeight<maxg)
    {
        obj.style.pixelHeight+=maxg/10;
        obj.filters.alpha.opacity+=20;
        obj2.background="images/title_hide.gif";
        if(obj.style.pixelHeight==maxg/10)
```

```

    obj.style.display='block';
    myObj=obj;
    mymaxg=maxg;
    myObj2=obj2;
    setTimeout('show(myObj, mymaxg, myObj2)', '5');
  }
}
</SCRIPT>

```

(2) 隐藏菜单的自定义函数的代码如下。

```

<SCRIPT language=javascript>
functionhide(obj, maxg, obj2)
{
  if(obj.style.pixelHeight>0)
  {
    if(obj.style.pixelHeight==maxg/5)
    obj.style.display='none';
    obj.style.pixelHeight-=maxg/5;
    obj.filters.alpha.opacity-=10;
    obj2.background="images/title_show.gif";
    myObj=obj;
    mymaxg=maxg
    myObj2=obj2;
    setTimeout('hide(myObj, mymaxg, myObj2)', '5');
  }
  else if(whichContinue)whichContinue.click();
}
</SCRIPT>

```

(3) 单击菜单时，隐藏前一个菜单，显示当前菜单的代码如下。

```
<SCRIPT language=javascript>
function chang(obj,maxg,obj2)
{
    if(obj.style.pixelHeight)
    {
        hide(obj,maxg,obj2);
        nopen='';
        whichcontinue='';
    }else if(nopen)
    {
        whichContinue=obj2;
        nopen.click();
    }
}
Else
{
    show(obj,maxg,obj2);
    nopen=obj2;
    whichContinue='';
}
}
</SCRIPT>
```

(4) 调用函数的方法，代码如下。

```
<TD class=list_title id=list1
onmouseover="this.typename='list_title2';"
onclick=chang(menu1,60,list1);
onmouseout="this.typename='list_title';"
```

background="images/title\_hide.gif" height=25><SPAN>网站管理  
</SPAN></TD>

举一反三

根据本实例，读者可以：

开发电子商务网站的后台管理模块；

开发新闻网的后台管理模块；

开发企业办公自动化系统的导航条。

## 实例054 树状导航菜单

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\01\054

实例说明

对于一个导航文字很多，可以对导航内容进行分类的网站来说，可以将页面中的导航文字以树视图的形式显示，这样不仅可以有效节约页面，而且也可以方便用户查看。运行本实例，如图1.65所示，在企业进销存管理系统将系统功能以树状导航菜单的形式列出，在页面的左侧“企业进销存导航”中可以看到该系统内所包含的具体功能。在默认情况下，所有功能节点都是折叠，单击节点名称可以展开指定节点；再单击该节点名称可以将该节点折叠。

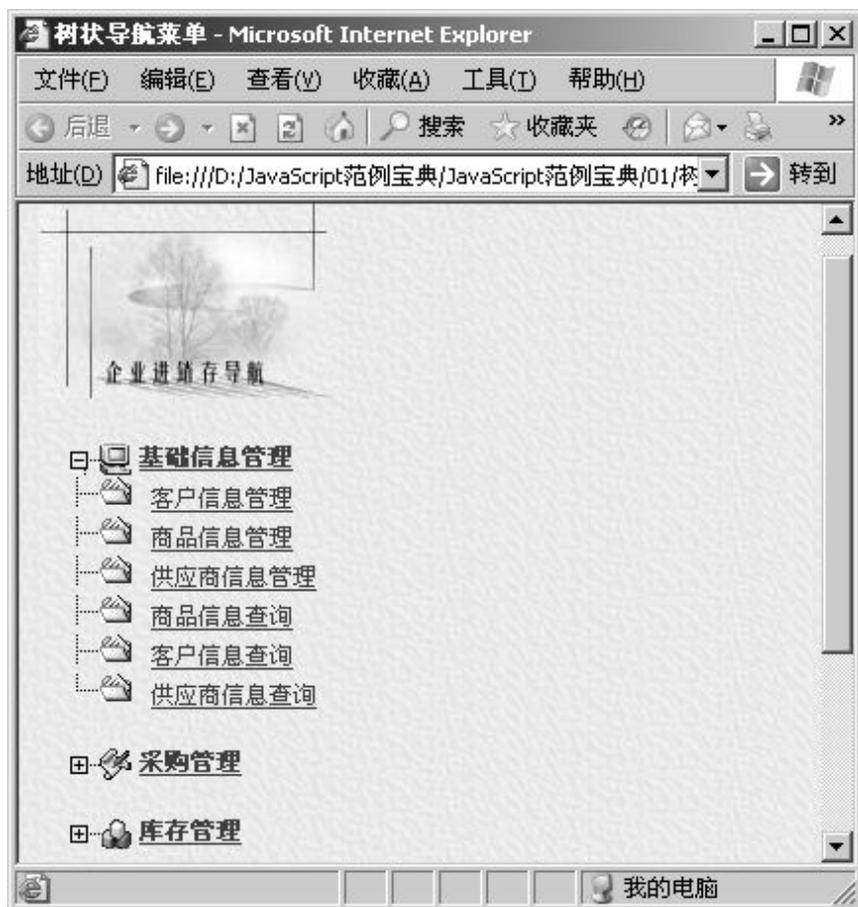


图1.65 树状导航菜单

### 技术要点

本实例主要通过页面中加入DIV层，并通过页面控制层的显示和隐藏来实现树状导航菜单。有关层的相关介绍请读者参见实例042。

### 实现过程

(1) 应用JavaScript 定义一个show\_div( )函数，主要用于显示一个层（menu 层）中的内容，用来实现模拟树形视图。关键代码如下。

```
<script language="JavaScript">
function show_div(menu)
{
var Imgname;
```

```

if (document.all.item(menu).style.display== "none")
{document.all.item(menu).style.display= "block";
Imgname="images/Img"+menu+"_2.gif"
document.all.item("Img"+menu).src=Imgname;
}
else
{document.all.item(menu).style.display= "none";
Imgname="images/Img"+menu+"_1.gif"
document.all.item("Img"+menu).src=Imgname;
}
}
</script>

```

(2) 插入导航图片和导航文字，关键代码如下。

```

<P> <Ahref="#"onClick="show_div('menu1')">
<imgname="Imgmenu1" src="images/Imgmenu1_1.gif"BORDER="0"
ALIGN="ABSMIDDLE"width="39"height="16">
<span class="l"><b>基础信息管理</b></span>
<br>
</a>
<DIV ID="menu1" style="display:none">

<ahref="#"class="l">客户信息管理</a><br>

<ahref="#"class="l">商品信息管理</a><br>

<ahref="#"class="l">供应商信息管理</a><br>


```

```

<a href="#" class="1">商品信息查询</a><br>
  
<a href="#" class="1">客户信息查询</a><br>
  
<a href="#" class="1">供应商信息查询</a><br>
<br>
</DIV>
<p> <a href="#" onClick="show_div(' menu2' )" >
<span class="1">
<b>采购管理</b></span></a>
  <br>
  <DIV ID=menu2 STYLE="display:None"> <a href="#" class="1">商品采购</a><br>
  
<a href="#" class="1">采购查询</a><br>
<br>
</DIV>
<p>
<A HREF="#" onClick="show_div(' menu3' )" >
<IMG name="Imgmenu3" SRC="images/Imgmenu3_1.gif" BORDER="0"
  ALIGN="ABSMIDDLE" width="39" height="16"><span class="1">
<b>库存管理</b>
</span></A>
  <br>
  <DIV ID=menu3 STYLE="display:None">

```

```

<a href="#" class="1">商品入库</a><br>
  
  <a href="#" class="1">商品入库退货</a><br>
    
    <a href="#" class="1">库存查询</a><br>
      
      <a href="#" class="1">价格调整</a><br>
    <br>
  </div>
<p> <A HREF="#" onClick="show_div('menu4')">
  <IMG name="Imgmenu4" SRC="images/Imgmenu4_1.gif"
BORDER="0"
  ALIGN="ABSMIDDLE" width="39" height="16"><span
class="1"><b>商品销售</b>
</span>
</A>
  <br>
<DIV ID=menu4 STYLE="display:none">
  
  <a href="#" class="1">商品销售</a><br>
    
    <a href="#" class="1">销售退货</a><br>
  <br>
</div>
<p> <A HREF="#" onClick="show_div('menu5')">
```

```
<IMG name="Imgmenu5" SRC="images/Imgmenu5_1.gif"
BORDER="0"
ALIGN="ABSMIDDLE" width="39" height="16"><b class="1">
查询统计</b></A>
<br>
<DIV ID=menu5 STYLE="display:None">
    
<a href="#" class="1">销售信息查询</a><br>
    
<a href="#" class="1">商品入库查询</a><br>
    
<a href="#" class="1">商品销售排行</a><br>
<br>
</div>
```

(3) 读者可以根据实现情况自行添加所需节点，方法同上。

### 举一反三

根据本实例，读者可以：

开发在线论坛中论坛管理页面中的应用目录树结构显示主题及回复信息；

在企业内部邮件发送时，根据部门选择收件人；

在电子商城网站中，对商品信息进行分类显示时应用树状导航菜单。

## 第2章 表单及表单元素

文本框/编辑框/隐藏域组件

下拉列表/菜单

单选按钮组

复选框

密码域

表单应用

## 2.1 文本框/编辑框/隐藏域组件

在制作网页过程中，经常要对文本框、编辑框、隐藏域等组件的属性进行设置，以方便用户操作。本节将介绍文本框/编辑框/隐藏域组件的应用。

### 实例055 获取文本框/编辑框/隐藏域的值

本实例是一个获取控件值的程序

实例位置：光盘\mingrisoft\02\055

实例说明

用户在浏览网页时，经常需要填写一些动态表单。例如，在互联网上发表自己的文章时，需要填写作者名称、文章标题以及文章内容。当用户提交表单时，程序需要获取表单内容，并对表单内容进行验证或者存储。本实例将介绍如何获取表单中的文本框、编辑框以及隐藏域的值。程序运行结果如图2.1所示。

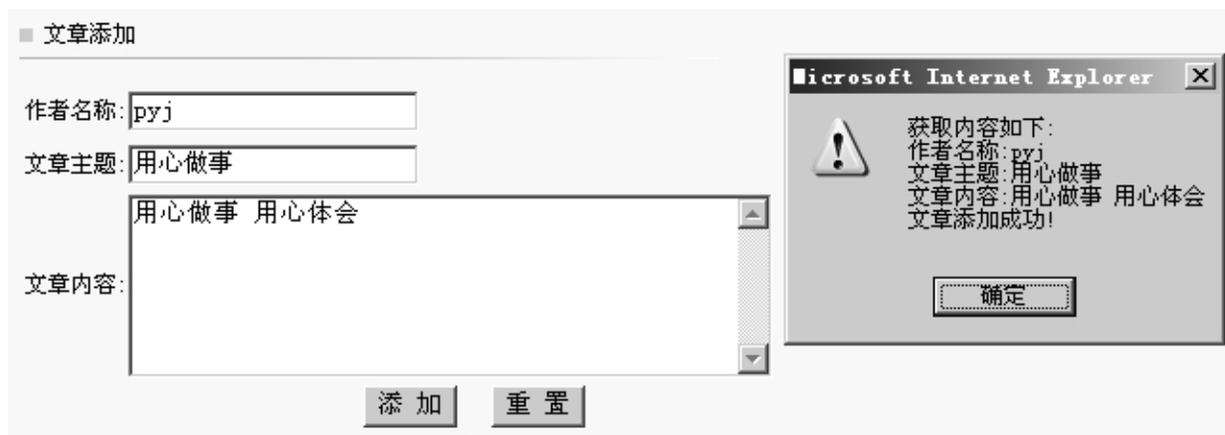


图2.1 获取文本框/编辑框/隐藏域的值

技术要点

本实例主要应用 JavaScript 脚本的 document 对象从客户端获得各项信息，并调用 window对象的 alert()方法（用于弹出一个警告对话框）动态显示获取到的信息。document 对象指向HTML 文档，文档是内容与网页中可访问元素的组合。下面介绍 document 对象的常用属性和方法。

- document.forms[]属性：表示文档中所有表单元素的数组，方括号内的索引值指向数组中的某个元素。

- document.images[]属性：表示使用<img>标记插入到文档中的图像数组。

- document.title 属性：获取网页的标题文本。

- document.write()方法：将指定字符串（可以包括 HTML 标记语言）写入到 HTML 文档中。

- document.write()方法：将打开一个新的输出流，它将清除当前网页的内容，而只显示指定字符串的内容。

### 实现过程

获取文本框/编辑框/隐藏域的值的关键代码如下。

```
<script type="text/javascript">
functionMycheck() {
    var checkstr="获取内容如下:\n";
    if (document.form1.文章作者.value != "") {
        checkstr+="作者名称:"+document.form1.文章作者.value+"\n";
    }
    if (document.form1.文章主题.value != "") {
        checkstr+="文章主题:"+document.form1.文章主题.value+"\n";
    }
}
```

```
    if (document.form1.文章内容.value != "") {
        checkstr+="文章内容:"+document.form1.文章内容.value+"\n";
    }
    if (document.form1.隐藏域.value != "") {
        checkstr+=document.form1.隐藏域.value;
    }
    if (checkstr != "") {
        alert(checkstr);
        return false;
    }
    else return
        return true;
}
</script>
<form name="form1"onSubmit="Mycheck()">
```

举一反三

根据本实例，读者可以：

实现编写企业门户网站中的加盟申请页面；

实现编写电子商务网站中的会员申请页面。

## [实例056 自动计算金额](#)

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\056

实例说明

在实际应用中，为了提高操作效率以及保证计算的准确性，可以使用 JavaScript 语言编写程序用于自动计算金额。例如，在签订合同页面，根据用户填写的签订合同年限，自动计算用户应缴纳的保证金额，其中，每年缴纳的保证金额是固定的。运行本实例，用户在输入合同年限后，程序将自动计算保证金额，并将其显示在文本框内。程序运行结果如图2.2所示。

### 技术要点

本实例主要应用JavaScript的document对象获取文本框以及隐藏域的值，并将计算后的数值赋值给文本框，使该文本框立即显示计算后得到的保证金总额。关于document对象的介绍请参见实例055，这里不再赘述。

### 实现过程

通过以下步骤在页面中实现自动计算金额。

(1) 自定义一个 JavaScript 函数。该函数首先获取用户在文本框中填写的合同年限数值和隐藏域的保证金额/年，然后将获取到的两个数值进行计算得到保证金总额，并将此计算金额赋值给相应文本框，其关键代码如下。



图2.2 自动计算金额

```
<script type="text/javascript">
function count() {
    var amount, price, money;
    amount=document.form1.txt_year.value;
    price=document.form1.txt_by.value;
    money=amount*price;
    document.form1.txt_bail.value=money;
}
</script>
```

(2) 在“合同年限”文本框的onBlur事件（即在文本框失去焦点时触发的事件）中调用自定义的JavaScript函数count()，代码如下。

```
<tr>
    <td height="22" align="right">合同年限:</td>
```

```
<td height="22" align="left"><input name="txt_year"
type="text" id="txt_year" size="10" onBlur="count()">年
<input name="txt_by" type="hidden" id="txt_by"
value="2000"></td>
</tr>
<tr>
<td height="22" align="right">保证金:</td>
<td height="22" align="left"><input name="txt_bail"
type="text" id="txt_bail" readonly="true"></td>
</tr>
```

举一反三

根据本实例，读者可以：

实现开发电子商务网站中的购物车程序；

实现开发企业进销存系统中的商品销售程序。

## 实例057 设置文本框的只读属性

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\057

实例说明

在网站后台管理系统中，管理员经常要查看用户基本信息，用户基本信息是指用户在网站前台填写的注册或者申请等信息。虽然管理员具有查看用户基本信息的权限，但是不可以修改用户所提交的信息内容。在设计页面时，设置显示用户信息的文本框为只读属性，即可以查看但不可修改文本框中的文本。运行本实例，当鼠标光标放在文本框上或者在文本框内使用键盘按键时，都将弹出提示信息。程序运行结果如图2.3所示。



图2.3 设置文本框的只读属性

### 技术要点

为了防止用户修改文本框内的文本，可以设置文本框的readOnly属性值为true，这时将禁止用户更改或删除文本框中的内容。

在自定义的JavaScript函数中，应用document对象获取文本框的readOnly属性值，如果为true（即该文本框为只读属性），则调用alert()方法弹出一个警告对话框显示相应信息。

### 实现过程

(1) 在

```
<input name="UserName" type="text" id="UserName" value="future" readonly="true" onKeyDown="Mycheck()" onMouseup="Mycheck()">
```

(2) 自定义的JavaScript函数Mycheck()的关键代码如下。

```
<script type="text/javascript">
functionMycheck() {
    if (document.form1.UserName.readOnly) {
        alert("用户名文本框为只读属性!");return false;
    }
    else return
    return true;
}
</script>
```

举一反三

根据本实例，读者可以：

实现开发电子商务网站中的查看会员基本信息页面；

实现开发在线论坛中的查看注册用户信息页面。

## 实例058 限制多行文本域输入的字符个数

本实例是一个提高程序安全性的程序

实例位置：光盘\mingrisoft\02\058

实例说明

在网站建设过程中，数据库的设计起着关键作用，为了节省存储数据的空间，提高数据库操作效率，在创建数据表时，通常要对字段大小进行限制。在页面中，为了保证程序的可靠性，根据数据表的结构要对用户输入的信息量进行控制，保证与数据表中的字段属性保持一致。运行本实例，当用户输入内容超过限制的字符个数时，将弹出提示信息。程序运行结果如图2.4所示。

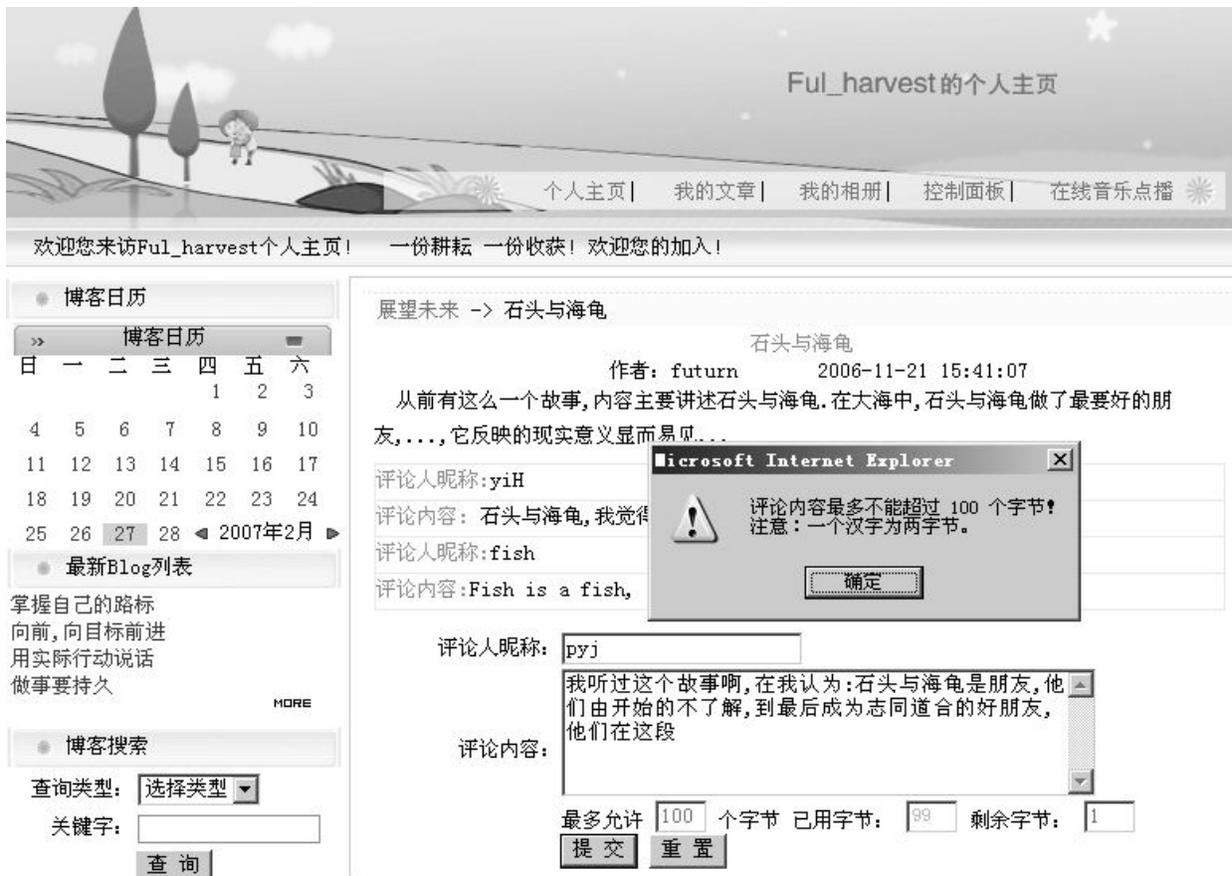


图2.4 限制多行文本域输入的字符个数

### 技术要点

本实例主要应用JavaScript语言中的if...else条件判断语句和for循环语句, 以及应用String对象的charCodeAt()方法和substring()方法, 下面分别进行介绍。

(1) 对变量或表达式进行判定时, 可以使用if...else条件判断语句, 其语法格式如下。

```

if(条件表达式) {
    语句序列1           //条件满足时执行
}else{语句序列2       //条件不满足时执行
}

```

执行上述语句时，首先计算“条件表达式（任意的逻辑表达式）”的值，如果值为 true，则执行“语句序列1”，执行完毕后结束该if语句；否则执行“语句序列2”，执行后同样结束该if语句。

上述if语句是典型的二路分支结构，其中的else部分可以省略，而且“语句序列”为单一语句时，其两边的大括号可以省略。

例如：

```
if(document.question.value=="") alert("请填写密码提示问题！");
```

if语句是一种使用很灵活的语句，可以根据需要对if语句进行嵌套使用。所谓嵌套就是将一个if语句当成另一个if语句的执行语句来使用。if语句的嵌套语法格式如下。

```
if (条件表达式) {  
    if (条件表达式) {  
        语句序列1  
    }else {  
        语句序列2  
    }  
}else {  
    if (条件表达式) {  
        语句序列3  
    }else {  
        语句序列4  
    }  
}
```

(2) for循环语句是JavaScript语言中应用比较广泛的条件语句。for循环语句通常使用一个变量作为计数器来执行循环，这个变量称为循环变量。

for循环语句的语法格式如下。

```
for(循环变量赋初值;循环条件;循环变量递增/减值) {  
    循环体;  
}
```

for循环语句的相关参数说明如表2.1所示。

表2.1 for循环语句的参数说明

参 数	描 述
循环变量赋初值	一条初始化语句，用来对循环变量进行初始化赋值
循环条件	一个包含比较运算符的表达式，用来限定循环变量的边限。如果循环变量超过了该边限，则停止该循环语句的执行
循环变量递增/减值	用来指定循环变量的递增/减值

for循环语句可以使用break语句来中止循环语句的执行。

(3) 调用 string 对象的 charCodeAt() 方法将返回一个整数，该整数表示字符串对象中指定位置处的字符的Unicode编码。

(4) 调用string对象的substring()方法将返回字符串中指定区间的子字符串。

### 实现过程

通过以下步骤实现限制多行文本域输入的字符个数。

(1) 自定义一个JavaScript脚本函数。该函数首先使用if...else条件判断语句和for循环语句判断文本域内输入的字符个数是否超过允许的最多字符个数，如果输入的字符个数超过允许的最多字符个数，则调用 alert()方法弹出一个警告对话框，并立即退出循环；然后将已用字符数和剩余字符数返回给页面。其关键代码如下。

```
<SCRIPT language=JavaScript>  
<!--  
varLastCount=0;  
functionCountStrByte(Message, Total, Used, Remain) { //字节统  
计
```

```

varByteCount=0;
varStrValue  =Message.value;
varStrLength=Message.value.length;
varMaxValue  =Total.value;
if (LastCount !=StrLength) { //在此判断，减少循环次数
    for (i=0;i<StrLength;i++) {
        ByteCount  = (StrValue.charCodeAt(i)<=256) ?
ByteCount+1 :ByteCount+2;
        if (ByteCount>MaxValue) {
            Message.value=StrValue.substring(0,i);
            alert("评论内容最多不能超过 "+MaxValue+"个字节！
\n注意：一个汉字为两字节。");
            ByteCount=MaxValue;
            break;
        }
    }
}
Used.value=ByteCount;
Remain.value=MaxValue-ByteCount;
LastCount=StrLength;
}
}
//-->
</SCRIPT>

```

(2) 在<textarea>标记的 onKeyDowns 事件和 onKeyUp 事件中分别调用自定义 JavaScript脚本函数CountStrByte(), 其关键代码如下。

```
<textarea name="评论内容" cols="45" rows="5" id="评论内容" onKeyDown="CountStrByte(this.form.评论内容,this.form.total, this.form.used, this.form.remain);" onKeyUp="CountStrByte(this.form.评论内容,this.form.total, this.form.used, this.form.remain); >
</textarea><br/>
```

最多允许

```
<input name="total" type="text" disabled
class="textbox" id="total" value="100" size="3">
```

个字节 已用字节:

```
<input name="used" type="text" disabled class="textbox"
id="used" value="0" size="3">
```

剩余字节:

```
<input name="remain" type="text" disabled
class="textbox" id="remain" value="100" size="3">
```

举一反三

根据本实例，读者可以：

实现开发中介信息网站中的信息发布页面；

实现开发餐饮网站中的留言信息页面。

## [实例059 自动选择文本框/编辑框中的文字](#)

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\059

实例说明

在一些实际应用中，经常要对大量的文字信息进行编辑。为了节省对文字编辑的时间或是引起用户的注意，可以自动选择编辑区域内

的文字内容，方便用户进行替换、读取等操作。运行本实例，当单击文本框/编辑框（即当页面中的文本框/编辑框获得焦点）时，文本框/编辑框中的文字将自动全部选中。程序运行结果如图2.5所示。



图2.5 自动选择文本框/编辑框中的文字

### 技术要点

本实例主要应用 JavaScript 的 document 对象来设置文本框/编辑框的 select() 属性，select() 属性用于选择文本框/编辑框中的所有文本。

### 实现过程

(1) 实现自动选择文本框/编辑框中的文字的自定义 JavaScript 函数，其关键代码如下。

```
<script type="text/javascript">
function Myselect_txt() {
    if (document.form1.title.focus) {
        document.form1.title.select();
    }
}
function Myselect_txtarea() {
```

```
    if (document.form1.content.focus) {  
        document.form1.content.select();  
    }  
</script>
```

(2) 分别在<input>标记和<textarea>标记的onClick事件中调用自定义JavaScript函数，代码如下。

```
<input name="title" type="text" size="50" value="今日新闻  
头条" onClick="Myselect_txt()">  
<textarea name="content" cols="50" rows="6"  
onClick="Myselect_txtarea()">今日,据相关方面报道,...  
</textarea>
```

举一反三

根据本实例，读者可以：

实现开发电子商务网站中的查看商品信息页面；

实现开发中介网站中的查看用户求职信息页面。

## [实例060 按下回车键时自动切换焦点](#)

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\02\060

实例说明

在设计表单时，为了方便用户填写表单，可以设置按下回车键自动切换到下一个控件的焦点，而不是直接提交表单。运行本实例，当用户填写完一项信息后，按下回车键时将焦点自动切换到下一个文本框中。程序运行结果如图2.6所示。



图2.6 按下回车键时自动切换焦点

### 技术要点

本实例主要应用JavaScript的event对象的keyCode只读属性。event对象的确切属性依赖于发生的事件。下面给出event对象的部分属性，如表2.2所示。

表2.2 Event对象的部分属性

属 性	描 述
altKet	值为 true 表示按下 Alt 键，值为 false 表示没有按下 Alt 键
button	对于特定的鼠标事件，表示按下的鼠标按钮
ctrlKey	值为 true 表示按下 Ctrl 键，值为 false 表示没有按下 Ctrl 键
keyCode	表示按下按键的数字代号
type	事件的名称

### 实现过程

实现按下回车键时自动切换焦点的关键代码如下。

```
<script type="text/javascript">
function Myenter(str) {
    if (event.keyCode == 13) {
        str.focus();}
}
</script>
<input name="用户名称" type="text" class="wenbenkuang"
id="用户名称" maxlength="50" onKeyPress="Myenter(form1.密码)"/>
```

举一反三

根据本实例，读者可以：

开发在线论坛的用户注册页面；

开发客房预订管理系统的用户信息登记页面。

## 2.2 下拉列表/菜单

在开发网络程序时，对下拉列表的应用非常广泛，通过下拉列表用户不仅可以查看多种选项，在页面中还可以使用多个下拉列表制作多级菜单，为用户提供更多信息。本节将介绍下拉列表的应用。

### 实例061 获取下拉列表/菜单的值

本实例是一个获取控件值的程序

实例位置：光盘\mingrisoft\02\061

实例说明

在设计网页时，可以使用下拉列表/菜单提供多个选项，不仅为用户提供方便，程序人员还可以根据获取到的下拉列表/菜单值将用户提交的信息进行分类存储，保证对数据库信息的有效维护。运行本实例，单击“添加”按钮，在弹出的窗口中将显示用户在下拉列表中选择选项名称。程序运行结果如图2.7所示。

技术要点

本实例主要应用JavaScript的document对象获取下拉列表/菜单的值，并调用alert()方法弹出一个警告对话框，显示获取的信息内容。

实现过程

实现获取下拉列表/菜单的值的的关键代码如下。



图2.7 获取下拉列表/菜单的值

```

<script type="text/javascript">
functionMycheck() {
    var val1, val2, val3;
    val1=document.form1.address.value;
    val2=document.form1.class1.value;
    val3=document.form1.huxing.value;
    alert("获取下拉列表值:\n地段:"+val1+"\n房源类别:"+val2+"\n房源户型:"+val3);
}
</script>

```

```

<form name="form1"onSubmit="Mycheck()">

```

举一反三

根据本实例，读者可以：

开发旅游网站中的用户旅游申请页面；

开发在线考试系统中的考生登记页面。

## [实例062 遍历多选择下拉列表](#)

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\062

实例说明

在使用下拉列表时，用户并不局限在只能选择一个选项，还可以在下拉列表中选择多个选项。运行本实例，用户可以按住Ctrl键并使用鼠标单击所要选择的选项，或者按住Shift键并使用鼠标选择所要选择的区段，实现在下拉列表中选择多个选项，选择的内容将显示在编辑框中。程序运行结果如图2.8所示。



图2.8 遍历多选择下拉列表

技术要点

本实例首先设置下拉列表的multiple属性，即允许用户选择多个选项，然后应用JavaScript脚本获取下拉列表的 length 属性值（即选项个数）。在 for 循环语句中判断 select 元素对象的selected属性值，如果值为true（即此选项已被选中），则将此选项的value值赋给一指定变量，并将此变量值赋予页面中的编辑框。

实现过程

（1）为页面中的下拉列表设置multiple属性，并在下拉框的onchange事件中调用自定义的JavaScript函数updateorder()，其关键代码如下。

```
<select name="mcusep" size="4" multiple class="wenbenkuang"
onchange="updateorder()">
```

```
    <optionvalue="JavaScript程序设计">JavaScript程序设计
</option>
    <optionvalue="JavaScript经典教程">JavaScript经典教程
</option>
    <optionvalue="JavaScript范例宝典">JavaScript范例宝典
</option>
    <optionvalue="JavaScript与CSS样式">JavaScript与CSS样式
</option>
    <optionvalue="JavaScript与HTML">JavaScript与
HTML</option>
</select>
```

(2) 自定义的JavaScript脚本函数updateorder()的关键代码如下。

```
<script language="javascript">
function updateorder()
{
    var orderstring="";
    var n=document.form1.mcusep.length;
    for (i=0;i<n;++i)
    {
        if (document.form1.mcusep.options[i].selected)
        {
            orderstring+=document.form1.mcusep.options[i].value
            +"\n";
        }
    }
    document.form1.textarea.value=orderstring;
}
```

```
}  
</script>
```

举一反三

根据本实例，读者可以：

开发企业商务系统中的企业信息添加页面；

编写动态网站中的友情链接管理页面。

## 实例063 在下拉列表中进行多选移除

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\02\063

实例说明

在开发动态网站程序时，经常会遇到将下拉列表中的选项进行多选移除或者多选移入。例如，在为员工分配工作地区时，为了避免工作冲突，分给员工A负责的地区就不可以再分给员工B了，同理，分给员工B负责的地区就不可以再分给员工A了。运行本实例，选择“任务地区1”下拉列表中的选项，并单击“>>”按钮，此选项将出现在“任务地区2”下拉列表中；选择“任务地区2”下拉列表中的选项，并单击“<<”按钮，此选项将出现在“任务地区1”下拉列表中。程序运行结果如图2.9所示。



图2.9 在下拉列表中进行多选移除

技术要点

本实例主要应用JavaScript的while循环语句，判断如果select元素的selectedIndex属性值不为-1，则获取在下拉列表中的选中项的索引值和对应的文本，然后应用select元素对象的add()方法将此选项添加到另一个下拉列表中，并应用select元素对象的remove()方法移除当前下拉列表中的此选项。

### 实现过程

(1) 在页面中分别添加两个下拉列表和普通按钮，在按钮的onClick事件中调用自定义的JavaScript脚本函数，代码如下。

```
<selectname="sel_place1" size="6"multiple
class="wenbenkuang" id="sel_place1" style="width:100px ">
  <option value="sel1">甘肃省</option>
  <option value="sel2">辽宁省</option>
  <option value="sel3">黑龙江省</option>
  <option value="sel4">吉林省</option>
  <option value="sel5">浙江省</option>
</select>
<inputname="sure1" type="button"
id="sure1"onClick="allsel(document.form1.sel_place2,document.
form1.sel_place1);"value="<<">
  <inputname="sure2" type="button"
id="sure2"onClick="allsel(document.form1.sel_place1,document.
form1.sel_place2);"value=">>">
  <selectname="sel_place2" size="6"multiple
class="wenbenkuang" id="sel_place2" style="width:100px ">
</select>
```

(2) 自定义的JavaScript函数的关键代码如下。

```
<script language="javascript">
```

```
function allsel(n1,n2)
{
    while(n1.selectedIndex!=-1)
    {
        var indx=n1.selectedIndex;
        var t=n1.options[indx].text;
        n2.options.add(new Option(t));
        n1.remove(indx);
    }
}
</script>
```

举一反三

根据本实例，读者可以：

开发员工管理系统中的员工管理权限页面；

开发电子商务网站中的商品销售渠道页面；

编写物流信息网中的货物运输方式页面。

## 实例064 将数组中的数据添加到下拉菜单中

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\02\064

实例说明

在开发动态网站时，可以将一些固定不变的数据存储在数组中，然后将数组中的数据显示到下拉菜单中以供用户选择，这样可以加快程序运行速度，方便用户浏览网页。运行本实例，单击下拉菜单的控制按钮，将显示出该下拉菜单中的全部选项。程序运行结果如图2.10所示。



图2.10 将数组中的数据添加到下拉菜单中

### 技术要点

本实例主要应用JavaScript的构造函数Array()和构造函数Option()。下面分别进行介绍。

(1) 使用构造函数Array()和运算符new可以创建Array数组对象，并且可以使用多种方式创建数组，详细内容请参见实例012。

(2) 使用构造函数Option()可以动态地创建Option对象，语法如下。

```
new Option(text,value,defaultSelected,selected)
```

参数说明如下。

- text: 表示一个字符串，指定Option对象的text属性。
- value: 表示一个字符串，指定Option对象的value属性。
- defaultSelected: 为布尔值，指定Option对象的defaultSelected属性。
- selected: 为布尔值，指定Option对象的selected属性。

### 实现过程

(1) 在<script>...</script>标记之间，首先定义一个数组，并为数组赋值，然后自定义一个JavaScript函数，将数组中的数据添加到表单的下拉菜单中，其关键代码如下。

```
<script type="text/javascript">  
var counts;  
counts=0;
```

```
arr=newArray("JavaScript与ASP","JavaScript与
JSP","JavaScript与ASP.NET","JavaScript与PHP");
counts=arr.length;
functionMyselect(){
    var i;
    for (i=0;i< counts; i++) {
        document.form1.sel.options[i]=newOption(arr[i],i);
    }
}
</script>>
```

(2) 在<select>标记的onFocus事件中调用自定义的JavaScript函数Myselect(), 代码如下。

```
<select name="sel" id="sel" onFocus="Myselect()">
```

举一反三

根据本实例, 读者可以:

开发企业进销存管理系统中的商品分类信息页面;

开发客户关系管理系统中的客户管理页面。

## [实例065 应用下拉菜单选择所要联机的网站](#)

本实例是一个应用广泛的程序

实例位置: 光盘\mingrisoft\02\065

实例说明

在进行网站程序设计时, 大多数网站都设有友情链接这一模块, 通常情况下, 链接的网站全部显示在页面中, 浏览者可通过单击网站名称或地址的超级链接进入所要访问的网站。有时为了节约页面也可以通过下拉菜单的方法来实现该功能。运行本实例, 当用户选择下拉

菜单中所要联机的网站后（如图 2.11 所示），单击“GO！”按钮可以立即链接到指定的网站。此实例还附加是否以“新窗口访问”选项，浏览者可根据需要确定是否选择。

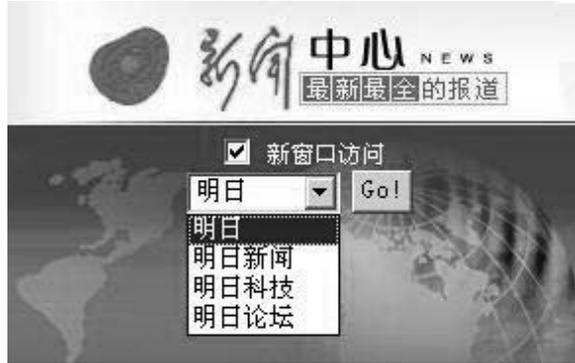


图2.11 应用下拉菜单选择所要联机的网站

### 技术要点

本实例将下拉菜单中各选项的 value 属性设置为对应联机网站的网址，并在自定义的 JavaScript 函数 goURL() 中获取用户选择的下拉列表选项值，然后使用 window 对象的 location 属性指定链接的 URL 地址。

### 实现过程

(1) 自定义一个 JavaScript 函数，该函数首先判断用户是否选择在“新窗口访问”，如果勾选此项，则使用 window 对象的 open() 方法打开一个新窗口浏览页面，否则在本窗口中打开新页面，其关键代码如下。

```
<script language="javascript">
<!--
function goURL() {
var thebox=document.form1
if(thebox.fig.checked) {
    if(!window.newwindow) {
        newwindow=window.open("")
```

```

        newwindow.location=
        thebox.example.options[thebox.example.selectedIndex].
value
    }
}
else
window.location=
thebox.example.options[thebox.example.selectedIndex].valu
e
}
//-->
</script>

```

(2) 在标记的onClick事件中调用自定义JavaScript函数goURL(), 代码如下。

```

<form name="form1">
    <input type="checkbox" name="fig" value="on">
        <span class="style1">新窗口访问</span><br>
        <select name="example" onchange="">
            <option value="http://www.mingrisoft.com">明日
</option>
            <option value="http://www.mingrisoft.com">明日新闻
</option>
            <option value="http://www.mingrisoft.com">明日科技
</option>
            <option
value="http://www.mingrisoft.com/board/index.asp">明日
论坛</option>

```

```
</select>  
<inputname="button"  
type="button"onclick="goURL()"value="Go!">  
</form>
```

举一反三

根据本实例，读者可以：

在动态网站中应用本实例实现网站导航；

开发图书管理系统中按图书分类搜索图书的页面。

## 实例066 多级级联菜单

本实例是一个提高工作效率的程序

实例位置：光盘\mingrisoft\02\066

实例说明

在实际应用中，经常将事物进行分类，分类可以有二级或者多级。在设计页面时，可以使用多个下拉菜单分别显示不同级别的分类信息，即实现多级级联菜单。运行本实例，当选择商品的“第一级分类”下拉菜单时，商品的“第二级分类”下拉菜单的内容会随即发生变化。程序运行结果如图2.12所示。

所属类别:	家用电器 ▾	电视机 ▾
商品名称:	P_L音乐手机	电视机 电冰箱
会员价:	2000	
提供厂商:	长春	
商品简介:	新品	
商品数量:	300	
<input type="button" value="添加"/> <input type="button" value="重置"/>		

图2.12 多级级联菜单

### 技术要点

本实例首先在<script>标记中定义两个数组，分别用于记录商品一级分类和二级分类的信息，然后自定义一个 JavaScript 函数，该函数的功能是当触发商品“第一级分类”下拉菜单中的 onChange 事件时，先清空商品的“第二级分类”下拉菜单中的选项内容，然后再将对应信息装载到商品的“第二级分类”下拉菜单中。

### 实现过程

实现多级级联菜单的关键代码如下。

```
<script type="text/javascript">
var arr2=newArray(4);
arr2["数码设备"]= newArray("数码相机","打印机");
arr2["家用电器"]= newArray("电视机","电冰箱");
arr2["礼品工艺"]= newArray("鲜花","彩带");
function removeinfo(classMenu) { //将下拉框各选项清空
    for (var i=0; i< classMenu.options.length; i++) {
        classMenu.options[i]=null;
    }
}
```

```

    }
}
function changeMenu(classList, classMenu) {
    removeinfo(classMenu)
    for (var i=0; i< classList.length; i++){
        classMenu[i]=newOption(classList[i], classList[i]);
    }
}
</script>
<select name="类别" id="类别"
onChange="changeMenu(arr2[document.form1.类别.
options[document.form1.类别.selectedIndex].text], document.form1.分类);">
    <option selected>数码设备</option>
    <option>家用电器</option>
    <option>礼品工艺</option>
</select>
<select name="分类" id="分类">
    <option>数码相机</option>
    <option>打印机</option>
</select>

```

举一反三

根据本实例，读者可以：

开发中介信息网站的房屋求购页面；

开发简历信息网中的个人简历添加页面。

## [实例067 可以输入文字的下拉菜单](#)

本实例是一个技巧性较强的程序

实例位置：光盘\mingrisoft\02\067

实例说明

网页中下拉菜单的选项是固定的，这样就存在一些弊端，如果用户想选择的项目不在下拉菜单的所列选项中，该怎么办呢？如果能在下拉菜单中输入文字，这个问题就解决了。运行本实例，在“等级名称”下拉菜单中，只包含了“新进成员”、“游侠”、“斑竹”3个选项，如果要添加的用户是上面3个选项以外的等级，就可以在下拉菜单中直接输入等级名称。程序运行结果如图2.13所示。

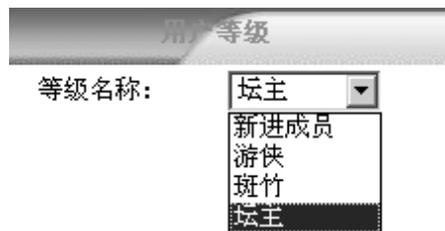


图2.13 可以输入文字的下拉菜单

技术要点

由于下拉菜单的选项是固定的，所以要实现在下拉菜单中输入文字，首先需要捕获用户的键盘事件，然后根据键盘输入的数据动态添加下拉菜单的选项，同时还需要判断用户按下的是退格键还是空格键，如果是退格键，则需要从已经添加的文字中删除一个文字，如果是空格键，则需要已经在添加的文字后面添加一个空格字符。

本实例主要应用JavaScript来设置元素的innerText属性，应用String对象的fromCharCode()方法以及select元素对象option数组的remove()方法。下面分别进行介绍。

- innerText 属性：用于修改起始标记与结束标记之间的文本。
- fromCharCode 方法：返回指定数值对应的字符，即将数值转换为字符。
- remove 方法：删除option数组中对应的选项。

## 实现过程

(1) 在页面中添加“等级名称”下拉菜单，其中包括“新进成员”、“游侠”、“斑竹”3个选项，代码如下。

```
<select name="sel_grand"
id="sel_grand" onkeypress="add_opt(3)" onkeyup="edit_opt(3)">
  <option selected>新进成员</option>
  <option>游侠</option>
  <option>斑竹</option>
</select>
```

(2) 在<select>标记的onkeypress事件中调用自定义的JavaScript函数add\_opt(opt)，用于在下拉菜单中添加新的选项，并设置该选项为选中状态，代码如下。

```
<script language="javascript">
function add_opt(opt) {
//opt表示现有可选项的数目
  form1.sel_grand.options[opt]=newOption(ok=
(form1.sel_grand.options[opt]))?
  form1.sel_grand.options[opt].innerText+String.fromCharCode(event.keyCode):
  String.fromCharCode(event.keyCode),ok)
  form1.sel_grand.selectedIndex=opt;
}
</script>
```

(3) 在<select>标记的onkeyup事件中调用自定义的JavaScript函数edit\_opt(opt)，用于处理当用户按下退格键时，将已经写到下拉菜单中的文字删除，当用户按下空格键时，在下拉菜单中添加一个空格符号，代码如下。

```
<script language="javascript">
function edit_opt(opt) {
//opt表示现有可选项的数目
  if(form1.sel_grand.options[opt]) {
    if(event.keyCode==8) {
      var str=form1.sel_grand.options[opt].innerText;
      var len=str.length;
      form1.sel_grand.options[opt].innerText=str.substrin
g(0, len-1);
      if(form1.sel_grand.options[opt].innerText=="") {
        form1.sel_grand.remove(opt);
        form1.bsel_grandbranch.selectedIndex=1;
      };
    }
    if(event.keyCode==32) {
      form1.sel_grand.options[opt].innerText+=" ";
    }
  }
}
</script>
```

举一反三

根据本实例，读者可以：

开发企业进销存管理系统中的商品添加页面；

编写动态网站中的在线信息收集页面。

## [实例068 根据下拉菜单的值显示不同控件](#)

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\068

实例说明

在开发Web应用程序时，经常涉及到多条件查询的问题。为了方便用户，可以根据用户在下拉菜单中选择的条件，显示不同的控件。运行本实例，在“统计类型”下拉菜单中选择“按月统计”时，在下方将显示关于月份的复选框，如图2.14所示。在选择“按年统计”时，在下方则显示关于年份的单选框，如图2.15所示。

图2.14 选择“按月统计”

图2.15 选择“按年统计”

技术要点

本实例主要通过JavaScript的document对象设置表格元素的display属性，来实现控件的隐藏或者显示。当display属性值为none时，表示元素隐藏，display属性值为block时，表示元素显示。

实现过程

(1) 根据下拉菜单的值显示不同控件的自定义JavaScript脚本函数，代码如下。

```
<script type="text/javascript">  
function changeMenu(MenuText) {  
    if(MenuText == "按月统计") {
```

```

        document.all.NumID.style.display="block";
        document.all.DateID.style.display="none";
    }
    if(MenuText == "按年统计"){
        document.all.DateID.style.display="block";
        document.all.NumID.style.display="none";
    }
}
</script>

```

(2) 在<select>标记的 onChange 事件中调用 JavaScript脚本函数 changeMenu(MenuText)，并设置控件在默认状态下是不可见的，代码如下。

```

<select name="type" id="type"
onChange="changeMenu(document.form1.type.options[document.form1.type.selectedIndex].text)">
    <option selected>选择统计类型</option>
    <option value="1">按月统计</option>
    <option value="2">按年统计</option>
</select>
<tr align="left" bgcolor="#E3E3E3" id="NumID"
style="display:none">
    ...略
<tr align="left" bgcolor="#E3E3E3" id="DateID"
style="display:none">
    ...略

```

举一反三

根据本实例，读者可以：

开发校友录网站中的班级查看页面；  
开发企业门户网站中的站内信息添加页面。

## 实例069 分级下拉列表

本实例是一个方便用户浏览和操作的程序

实例位置：光盘\mingrisoft\02\069

实例说明

在制作网页时，如果信息分类的内容很多，可以在下拉列表中将  
其分级显示，使用户更明楚所查看的选项。运行本实例，可以看到下  
拉列表中的选项是分级显示的。程序运行结果如图2.16所示。

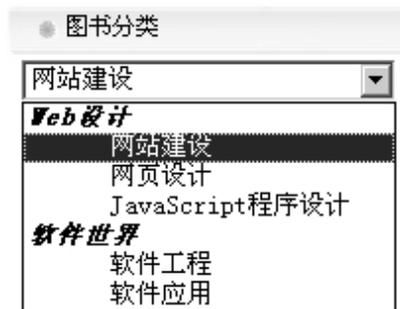


图2.16 分级下拉列表

技术要点

本实例主要应用了<optgroup>标记，并设置其label属性。  
<optgroup>标记主要用于对select元素中的选项进行逻辑分组，在  
<optgroup>标记中指定的文本是非可选项，一般通过替换文本与可选  
项区分开。

注意：<optgroup>标记与</optgroup>应成对使用。

实现过程

实现分级下拉列表的关键代码如下。

```
<selectname="book_class" id="book_class">  
  <optgroup label="Web设计">
```

```
<option>网站建设</option>
<option>网页设计</option>
<option>JavaScript程序设计</option>
</optgroup>
<optgroup label="软件世界">
  <option>软件工程</option>
  <option>软件应用</option>
</optgroup>
</select>
```

举一反三

根据本实例，读者可以：

在省份城市下拉列表中应用该方法。

## 2.3 单选按钮组

单选按钮组是由一组单选按钮组成的，并且在一个单选按钮组中，每次只能有一个单选按钮被选中。单选按钮组也是HTML页面中常用的表单元素之一。下面通过几个具体实例介绍单选按钮组的应用。

### 实例070 不提交表单获取单选按钮的值

本实例是一个获取控件值的程序

实例位置：光盘\mingrisoft\02\070

实例说明

在编写信息添加模块时，一般都需要对用户输入信息的合法性进行判断。判断用户输入信息的合法性有两种实现方法：一种是在提交表单以前，应用 JavaScript实现；另一种是在表单提交以后通过 VBScript 实现。通常情况下采用前者。在判断用户输入的证件号码是否正确时，需要根据前面选择的证件类型进行判断，因为不同的证件其号码位数是不同的，例如，身份证一般为 15位或 18位，而学生证一般为 8位或 12位。运行程序，当用户选择证件类型为“学生证”后，在“证件号码”文本框中输入学生证号码，当光标移出该文本框时，程序将判断用户输入的学生证号码是否正确，如果不正确将给予提示，如图2.17所示。

本实例主要通过一组单选按钮组成的单选按钮组实现，单选按钮组每次只能有一个单选按钮被选中。单选按钮的属性和事件如表2.3所示。



### 注册帮助

- 用户名：为用户登录留言簿的通行证，可使用英文字母、数字或英文字母、数字、下划线的组合，长度控制在3-20个字符之内。
- 真实姓名：请输入真实的姓名，该项为隐藏项，用户可以放心输入。
- Email：请填写有效的Email地址，以便与您联系。

用户名： \*

真实姓名： \*

证件类型： 身份证  学生证

证件号码：

联系电话：

Email：

个人主页：



图2.17 不提交表单获取单选按钮的值

### 技术要点

表2.3 单选按钮的属性及事件

属 性	说 明	事 件	说 明
checked	表示单选按钮被选中	onBlur	失去焦点时的动作
name	对象属性访问名称	onClick	单击时的动作
type	对象类型	onFocus	获得焦点时的动作
value	对象值		

### 实现过程

实现不提交表单获取单选按钮的值的的关键代码如下。

```
<script type="text/javascript">
function getVal() {
    varCardTypeValue;
    for (i=0;i<form1.CardType.length;i++) {
        if (form1.CardType[i].checked) {
            CardTypeValue=form1.CardType[i].value;
            break; //使用该语句可以减少不必要的循环次数
        }
    }
}
```

```
}
if(CardTypeValue=="身份证"){
    if(form1.pcard.value.length!=15&&
form1.pcard.value.length!=18){
        alert("你输入的证件号码有
误!");form1.CardType.focus();return;
    }
}else{
    if(CardTypeValue=="学生证"){
        if(form1.pcard.value.length!=8&&
form1.pcard.value.length!=12){
            alert("你输入的证件号码有
误!");form1.CardType.focus();return;
        }
    }
}
}
</script>
```

```
<inputname="pcard" type="text" id="Tel"onBlur="getVal()">
```

举一反三

根据本实例，读者可以：

开发信息网站中的招聘/求职信息添加页面；

开发手机销售网站中的购买订单页面。

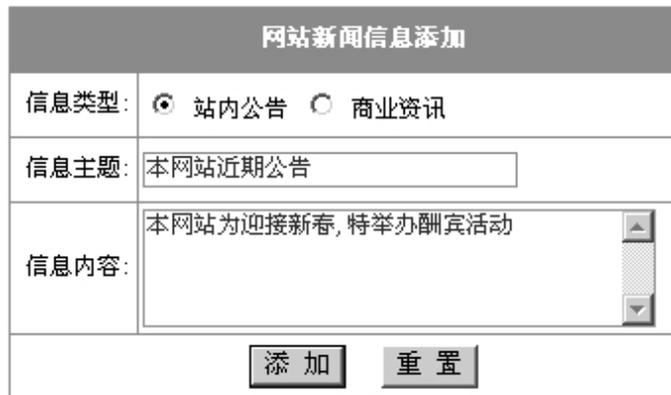
## [实例071 选中单选按钮后显示其他表单元素](#)

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\071

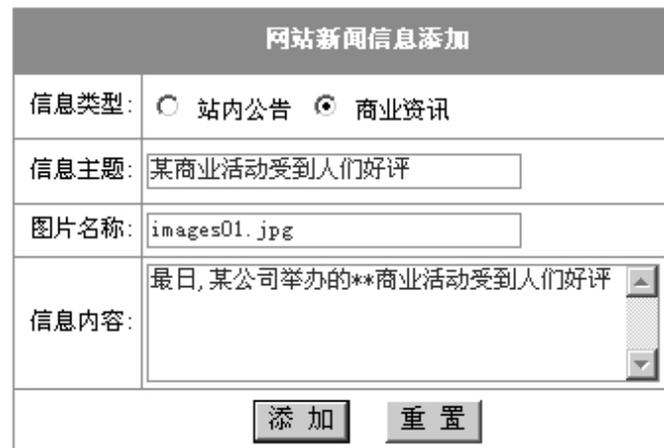
### 实例说明

在设计网站新闻信息添加页面时，根据选择的信息类型的不同，需要显示不同的表单元素。运行本实例，选中“站内公告”单选按钮时，其下方显示“信息主题”文本框和“信息内容”编辑框，如图2.18所示。选中“商业资讯”单选按钮时，其下方将增添一个“图片名称”文本框，如图2.19所示。



网站新闻信息添加	
信息类型:	<input checked="" type="radio"/> 站内公告 <input type="radio"/> 商业资讯
信息主题:	<input type="text" value="本网站近期公告"/>
信息内容:	<input type="text" value="本网站为迎接新春,特举办酬宾活动"/>
<input type="button" value="添加"/> <input type="button" value="重置"/>	

图2.18 选中“站内公告”单选按钮



网站新闻信息添加	
信息类型:	<input type="radio"/> 站内公告 <input checked="" type="radio"/> 商业资讯
信息主题:	<input type="text" value="某商业活动受到人们好评"/>
图片名称:	<input type="text" value="images01.jpg"/>
信息内容:	<input type="text" value="最日,某公司举办的**商业活动受到人们好评"/>
<input type="button" value="添加"/> <input type="button" value="重置"/>	

图2.19 选中“商业资讯”单选按钮

### 技术要点

本实例主要应用JavaScript来设置表格的display样式，以实现表格的显示和隐藏。通过设置表格的id属性，可实现对表格的整体控

制。通过将表格的display样式设置为“none”，可以控制表格的隐藏；将表格的display样式设置为“block”，可以显示表格。

实现过程

(1) 设置表格id属性值，并设置其在默认状态下是不可见的。

```
<tr bgcolor="#FFFFFF" id="img" style="display:none">
```

(2) 在选中“站内公告”单选按钮时，触发其onClick事件，使“图片名称”文本框不可见。

```
<input name="infotype" type="radio" value="1"
onClick="img.style.display='none';" checked>
```

(3) 在选择“商业资讯”单选按钮时，触发其onClick事件，使“图片名称”文本框可见。

```
<input type="radio" name="infotype" value="2"
onClick="img.style.display='block';">
```

举一反三

根据本实例，读者可以：

开发动态网站管理员级别设置页面；

● 开发中介信息网站中的多条件信息搜索页面。

## [实例072 通过单选按钮控制其他表单元素是否可用](#)

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\02\072

实例说明

在开发动态网站时，根据用户选择的选项不同，可以设置表单中的元素是否可用，以保证提交信息的准确性。运行本实例，选中“普通主题”单选按钮时，用户可以使用“主题分类”下拉列表，如图 2.20 所示。选中“版面公告”单选按钮时，“主题分类”下拉列表呈

现灰色状态，即在发表版面公告时不允许选择主题分类，如图2.21所示。

图2.20 选中“普通主题”单选按钮

图2.21 选中“版面公告”单选按钮

### 技术要点

本实例主要应用JavaScript来设置下拉列表的disabled属性。当元素的disabled属性为true时表示该元素不可用，控件元素呈现灰色状态。

### 实现过程

(1) 设置下拉列表的id属性，并设置下拉列表在默认情况下是可用的。

```
<select name="t_class" id="t_class">  
  <option>网页设计</option>  
  <option>网站建设</option>  
</select>
```

(2) 当选中“普通主题”单选按钮时，触发其onClick事件，刷新本页面。

```
<input name="t_type" type="radio" value="1"  
onClick="window.location='index.htm';" checked>
```

(3) 当选中“版面公告”单选按钮时，触发其onClick事件，设置下拉列表的disabled属性值为true。

```
<input type="radio" name="t_type" value="2"  
onClick="t_class.disabled=' true' ;">
```

举一反三

根据本实例，读者可以：

开发企业员工信息管理系统中的员工职称信息页面；

开发物业管理系统中的管理员权限页面。

## 2.4 复选框

在设计网页时，根据用户需求，可以使用复选框为用户提供多个选项，这样用户针对某些问题时可以选择一个以上的选项。复选框显示一个带有标识的小方格，当用户单击时，在其中会显示一个选择标志，或者将已有标志取消。本节将介绍复选框的应用。

### 实例073 不提交表单获取复选框的值

本实例是一个获取控件值的程序

实例位置：光盘\mingrisoft\02\073

实例说明

在制作网页时，在不提交表单的情况下也可以获取复选框的值。运行本实例，选择复选框后，在“处理信息显示”文本框内单击鼠标使文本框获得焦点，将即时显示用户选中的复选框值。程序运行结果如图2.22所示。

○ 订单处理	
订单处理项：	<input checked="" type="checkbox"/> 已收款 <input type="checkbox"/> 已发货 <input checked="" type="checkbox"/> 已确认
处理信息显示：	<input type="text" value="已收款已确认"/>

图2.22 不提交表单获取复选框的值

技术要点

本实例主要应用JavaScript的for循环语句来判断复选框的checked属性值，如果checked属性值为true（即说明复选框已被选中），则记录复选框值，并将所有选中的复选框值赋给“处理信息显示”文本框。

## 实现过程

(1) 自定义一个JavaScript函数，该函数使用for循环语句，将选中的复选框值赋给变量，并将变量值赋给文本框，其关键代码如下。

```
<script type="text/javascript">
function Mycheck() {
    var str="";
    for (var i=0; i < document.form1.chk_type.length; i++) {
        if (document.form1.chk_type[i].checked) {
            str+=document.form1.chk_type[i].value;
        }
        document.form1.txt_info.value=str;
    }
}
</script>
```

(2) 在

```
<input name="txt_info" type="text" id="txt_info"
size="28" onFocus="Mycheck()">
```

### 举一反三

根据本实例，读者可以：

开发办公自动化管理系统中的管理员信息添加页面；

开发在线投票页面。

## [实例074 控制复选框的全选或反选](#)

本实例是一个应用较为广泛的程序

实例位置：光盘\mingrisoft\02\074

## 实例说明

在查看数据信息时，经常需要对多条数据进行操作。例如，删除数据时，通过控制复选框的全选或反选，可以批量删除数据等。运行本实例，选中“[全选/反选]”复选框，表单中所有的复选框都将处于选中状态，如图2.23所示。

商品信息查看				
选择	所属类别	商品名称	会员价	数量
<input checked="" type="checkbox"/>	手机	P_L音乐手机	1980	200
<input checked="" type="checkbox"/>	玻璃制品	迷你水杯	49	500
<input checked="" type="checkbox"/>	音响	CXO音响	2070	200
<input checked="" type="checkbox"/>	休闲装	休闲上衣	195	500
<input checked="" type="checkbox"/>	[全选/反选]			

图2.23 控制复选框的全选或反选

## 技术要点

本实例应用的自定义JavaScript函数首先判断表单中包含复选框的数量，如果数量大于0，则使用for循环语句设置所有复选框的checked属性值为true，如果“[全选/反选]”复选框处于未选中状态，则设置所有复选框的checked属性值为false。

## 实现过程

控制复选框的全选或反选的关键代码如下。

```
<script type="text/javascript">
function CheckAll(elementsA,elementsB)
{
    var len = elementsA;
    if(len.length > 0)
    {
        for(i=0;i<len.length;i++)
        {
```

```
        elementsA[i].checked = true;
    }
    if(elementsB.checked ==false)
    {
        for(j=0;j<len.length;j++)
        {
            elementsA[j].checked = false;
        }
    }
}
}
</script>
<input name="chk_all" type="checkbox" id="chk_all"
value="checkbox"
onClick="CheckAll(this.form.chk_id,this.form.chk_all)">
```

举一反三

根据本实例，读者可以：

开发新闻信息网站中的信息查看页面；

开发博客网站中的文章评论查看页面。

## 实例075 只有一个复选框时控制复选框的全选或反选

本实例是一个提高操作技能的程序

实例位置：光盘\mingrisoft\02\075

实例说明

在网页中，如果显示的信息数量只有一条，即只有一个需要控制的复选框时，使用上一实例的 JavaScript 函数控制复选框的全选或

反选就会出现错误。本实例将实现只有一个复选框时控制复选框的全选或反选的功能。程序运行结果如图2.24所示。

文章信息查看			
选择	文章类别	作者名称	文章标题
<input checked="" type="checkbox"/>	展望未来	future	掌握自己的路标
<input checked="" type="checkbox"/>	[全选/反选]		

图2.24 只有一个复选框时控制复选框的全选或反选

### 技术要点

本实例主要通过 JavaScript 的 if 条件判断语句确定复选框的数量是否大于 0，如果数值不大于 0，说明需要控制的复选框只有一个（即返回的 length 属性值为 null），则可以直接设置复选框的 checked 属性值为 true 或者 false。

### 实现过程

实现只有一个复选框时控制复选框的全选或反选的关键代码如下。

```
<script type="text/javascript">
function CheckAll(elementsA, elementsB)
{
    var len = elementsA;
    if(len.length > 0) {}
    else
    {
        len.checked = true;
        if(elementsB.checked == false)
        {
            len.checked = false;
        }
    }
}
```

```
}
```

```
</script>
```

举一反三

根据本实例，读者可以：

开发网络考试系统中的试题查看页面；

开发在线美食网站后台管理系统中的美食查看页面。

## 2.5 密码域

在开发动态网站时，网站前台用户的注册页面、登录页面或者网站后台管理员登录页面都要用到密码域。在页面中不仅可以获取密码域的值，还可以对密码域进行设置使其内容更安全。本节将介绍密码域的应用。

### 实例076 让您的密码域更安全

本实例是一个提高安全性的程序

实例位置：光盘\mingrisoft\02\076

实例说明

对于网络应用程序或网站来说，安全最为重要。其中，密码域的安全也非常重要。虽然在密码域中已经将所输入的字符以掩码形式显示了，但是它并没有实现真正保密，因为用户可以通过复制该密码域中的内容，并将复制的密码粘贴到其他文档中，查看密码的“真实面目”。为实现密码的真正安全，可以将密码域的复制功能屏蔽，同时改变密码域的掩码符号。运行本实例，输入密码并选中所输入的密码，再单击鼠标右键时，可以发现原来的复制菜单项变为灰色即为不可用状态，并且复制快捷键“Ctrl+C”也不可用了。程序运行结果如图2.25所示。



图2.25 让您的密码域更安全

### 技术要点

本实例主要通过控制密码域的 oncopy、oncut、onpaste 事件来实现密码域的内容禁止复制的功能，并通过改变其style样式属性来实现改变密码域中掩码的样式。

注意：在修改掩码的样式时一定要选择Windows系统自带的字体样式，如果设置的字体样式不存在，密码将会以原形显示。

### 实现过程

(1) 在页面中添加密码域，代码如下。

```
<input name="txt_passwd" type="password" class="textbox" id="txt_passwd" size="12" maxlength="50">
```

(2) 添加代码，禁止用户复制、剪切和粘贴密码，代码如下：

```
<input name="txt_passwd" type="password" class="textbox" id="txt_passwd" size="12" maxlength="50" oncopy="return False" oncut="return False" onpaste="return False">
```

(3) 改变密码域的掩码样式，将其“font-family”设置为“Wingdings”，最终代码如下。

```
<input name="txt_passwd" type="password" class="textbox"
id="txt_passwd" size="12" maxlength="50" oncopy="return
False" oncut="return False" onpaste="return False"
style="font-family:Wingdings;">
```

举一反三

根据本实例，读者可以：

开发电子商务网站中的用户登录页面；

开发客户管理系统的管理员登录页面；

开发在线论坛的用户登录页面。

## 实例077 不提交表单自动检测密码域是否相同

本实例是一个提高安全性的程序

实例位置：光盘\mingrisoft\02\077

实例说明

在制作用户注册信息页面时，要求用户输入确认密码，以确定用户输入密码是否准确。运行本实例，如果用户在“确认密码”域输入的信息与“密码”域输入的信息不同，将弹出提示框，要求用户再次确认输入的密码。程序运行结果如图2.26所示。



图2.26 不提交表单自动检测密码域是否相同

技术要点

本实例主要通过JavaScript的if条件语句判断两次输入的密码是否相同，如果不同，则调用alert()方法弹出一个警告提示框。

## 实现过程

实现不提交表单自动检测密码域是否相同的关键代码如下。

```
<input name="pwd2" type="password" class="wenbenkuang"
id="pwd2" onBlur="if(this.value!=this.form.pwd1.value)
alert('您两次输入的密码不一致!');" size="18">
```

## 举一反三

根据本实例，读者可以：

开发餐饮网站中的会员注册页面；

开发汽车网站中的VIP会员注册页面。

## 2.6 表单应用

表单是实现动态网页的一种主要的外在形式，使用表单可以收集客户端提交的有关信息。表单是实现网站互动功能的重要组成部分。下面通过几个具体的实例介绍表单的应用。

### 实例078 通过JavaScript控制表单的提交与重置

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\02\078

实例说明

用户填写表单时，可以单击“提交”按钮提交表单，也可以直接按回车键提交表单，同时还可以单击“重置”按钮将所填内容清空。为了防止用户因疏忽而按错键，可以使用 JavaScript 来控制表单的提交与重置。程序运行结果如图2.27所示。



图2.27 通过JavaScript控制表单的提交与重置

技术要点

本实例主要通过JavaScript来调用表单的submit()提交方法和reset()重置方法。页面中的按钮为普通按钮，即标记的类型

为button。

实现过程

实现通过JavaScript脚本控制表单的提交与重置的关键代码如下。

```
<input name="add" type="button" class="button" id="add" value="添加" onClick="document.form1.submit();">
```

```
<input type="button" name="Submit2" value="重置" class="button" onClick="document.form1.reset();">
```

举一反三

根据本实例，读者可以：

开发宽带点播网站中的音频和影视文件上传页面；

开发供求信息网站中的信息添加页面。

## 实例079 带记忆功能的表单

本实例是一个提高操作效率、人性化的程序

实例位置：光盘\mingrisoft\02\079

实例说明

在一般情况下，用户提交表单后，将会跳转到另一个页面，同时表单中的内容也会清空。而有时为了简化操作步骤，需要保留历史信息，即当用户再返回原来页面时，还可以看到刚才所填写的信息。运行本实例，当用户刷新或者单击“确定”按钮提交表单后再退回到原来页面时，“计量单位”和“供应商”文本框中的内容将保持不变。程序运行结果如图2.28所示。

添加商品信息			
商品名称:	<input type="text"/>		
商品规格:	<input type="text"/>	计量单位:	<input type="text" value="台"/>
供应商:	<input type="text" value="长春市经济开发区**公司"/>		
出厂日期:	<input type="text"/>		
	<input type="button" value="确定"/>	<input type="button" value="重置"/>	

图2.28 带记忆功能的表单

### 技术要点

本实例主要通过通过在CSS样式定义中，设置behavior确定对象的行为，并设置<meta>元信息标记中的name属性和content属性来实现保留历史信息的功能。<meta>标记用来在HTML文件中模拟HTTP协议的响应头报文，是实现元数据的主要标记，可以用于鉴别作者、标注内容摘要和关键字、设定页面字符集、刷新页面等。在HTML文档头部可以包括任意数量的<meta>标记。

### 实现过程

实现带记忆功能的表单的关键代码如下。

```
<html>
<head>
<meta name="save" content="history">
<title>带记忆功能的表单</title>
<style type="text/css">
.saveHistory{
behavior:url(#default#savehistory);
}
</style>
</head>
<body style="font-size:12px">
```

```
<input name="txt_unit" type="text" class="saveHistory"
id="txt_unit">
<input name="txt_co" type="text" class="saveHistory"
id="txt_co" size="35">
</body>
</html>
```

举一反三

根据本实例，读者可以：

开发网上书店系统中的图书添加页面；

开发简历管理系统中的简历录入页面。

## 实例080 防止表单重复提交

本实例是一个提高操作安全性的程序

实例位置：光盘\mingrisoft\02\080

实例说明

用户在提交表单时，如果单击提交按钮后，表单没有立即提交，用户则会多次单击提交按钮，即可能会多次提交表单内容，这样会造成意想不到的后果，使应用程序无法运行。运行本实例，当用户单击“确定”按钮后，提交表单的同时该按钮将被禁用，防止表单重复提交。程序运行结果如图2.29所示。

加盟商存款信息

加盟商编号：	<input type="text" value="LN200703050916"/>
存款金额	<input type="text" value="50000"/> 元
<input type="button" value="确定"/>	

图2.29 防止表单重复提交

技术要点

本实例首先设置<input>标记的 disabled 属性值，然后再调用表单的 submit() 方法。页面中的按钮为普通按钮，即<input>标记的类型为button。

实现过程

防止表单重复提交的关键代码如下。

```
<input type="button" name="Submit2" value="确定"  
onclick="this.disabled=true; this.form.submit();">
```

举一反三

根据本实例，读者可以：

开发电子商务网站中的商品入库信息页面；

开发客户关系管理系统中的客户签订合同页面。

## 实例081 自动提交表单

本实例是一个提高操作效率、人性化的程序

实例位置：光盘\mingrisoft\02\081

实例说明

根据实际需要，有时需要设置自动提交表单。例如，网络系统中的考试页面，当考生填写表单的时间到达考试时限，程序将自动提交表单，表示考试结束。运行本实例，用户单击“开始考试”超级链接时，表示考试开始，当剩余时间为0时，表单将自动提交。程序运行结果如图2.30所示。

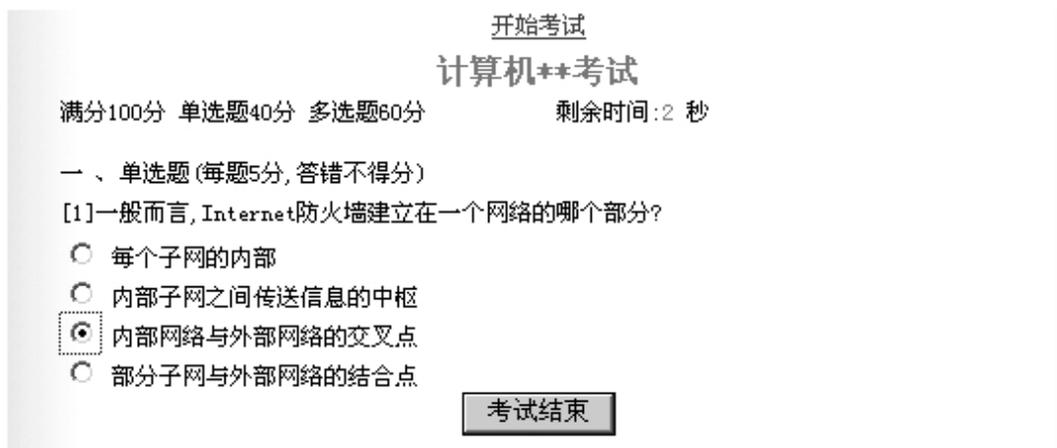


图2.30 自动提交表单

### 技术要点

本实例主要应用JavaScript语言window对象的setTimeout()方法。SetTimeout()方法的作用是根据设定的时间间隔执行一次指定的表达式，语法如下。

setTimeout(表达式, 延时时间[, 参数])

延时时间是以毫秒为单位的(即1000ms=1s)。

通过循环调用 setTimeout()方法, 可以实现多次执行指定程序, 从而判断用户用时是否已经达到规定的时限, 如果达到规定的时限, 则调用表单的submit()方法提交表单内容。

### 实现过程

实现自动提交表单的关键代码如下。

```
<a href="#" onClick="Time_limit()">开始考试</a>
<form name="form1">
  剩余时间:<font color="#FF0000" id="showtime"></font> 秒
  <input name="sec_nums" type="hidden" id="sec_nums">
  <div id="show" style="display:none "><input name="sure"
type="submit" id="sure" value="考试结束"></div>
  <div id="hidden"><input name="sure2" type="submit"
id="sure2" value="考试结束" disabled></div>
```

```
</form>
<script type="text/javascript">
var sec=5;
functionTime_limit() {
    document.all.show.style.display="block";
    document.all.hidden.style.display="none";
    thistime=document.form1.sec_nums.value;
    document.all.showtime.innerHTML=thistime;
    if (thistime<=0) {
        document.form1.submit();
    }
    document.form1.sec_nums.value=thistime-1;
    var timer=window.setTimeout("Time_limit()",1000);
}
document.all.form1.sec_nums.value=sec;
</script>
</html>
```

举一反三

根据本实例，读者可以：

开发电子商务网站中的订单提交页面；

编写游戏网站中的在线问答页面。

## [实例082 通过for循环获取表单元素的中文名称](#)

本实例是一个提高基础技能的程序

实例位置：光盘\mingrisoft\02\082

实例说明

当页面中包含很多表单元素时，可以使用JavaScript脚本在for循环语句内获取表单元素的中文名称，并验证表单控件元素是否为空值。运行本实例，当用户提交表单时，如果表单中控件元素值为空字符串，则会弹出警告提示框。程序运行结果如图2.31所示。



图2.31 通过for循环获取表单元素的中文名称

### 技术要点

本实例主要通过JavaScript的document对象来获取表单控件元素数组elements[]的值，并应用for循环语句以及if条件语句来判断各表单控件元素的值是否为空，如果为空将弹出带中文提示信息的对话框。

### 实现过程

(1) 在页面中添加表单，并设置文本框对应的name属性值和title属性值，代码如下。

```
<form name="form1">  
  <tr align="left" bgcolor="#FFFFFF"><td width="183"  
height="22" align="left">销售编号:</td>  
  <td width="298" height="22"><input name="txt_snum"  
type="text" id="txt_snum" title="销售编号"></td></tr>
```

```

<tr align="left" bgcolor="#FFFFFF"><td width="183"
height="22" align="left">退货数量:</td>
    <td width="298" height="22"><input name="txt_nums"
type="text" id="txt_nums" title="退货数量"></td></tr>
    <tr align="left" bgcolor="#FFFFFF"><td height="22"
align="left">退货日期:</td>
    <td height="22"><input name="txt_date" type="text"
id="txt_date" title="退货日期"></td></tr>
    <tr align="left" bgcolor="#FFFFFF"><td height="22"
align="left">加盟商应收退款是否对账:</td>
    <td height="22"><input name="txt_fig"
type="radio" value="1" checked>是
    <input type="radio" name="txt_fig" value="0">否 </td></tr>
    <tr align="center" bgcolor="#FFFFFF">
<td height="22" colspan="2"><input
type="submit" name="Submit" value=" 商品销售退
货 "onClick="Mycheck(this.form)"></td></tr>
</form>

```

(2) 自定义的JavaScript函数Mycheck(form)的关键代码如下。

```

<script language="javascript">
functionMycheck(form) {
    for(i=0;i<form.length;i++) {
        if(form.elements[i].value=="") {
            alert(form.elements[i].title+"不能为空!");return
false;}
    }
}

```

</script>

举一反三

根据本实例，读者可以：

开发物流信息网站中的货物详细信息页面；

开发游戏网站中的软件下载页面。

## 实例083 可以提交到不同处理页的表单

本实例是一个提高操作效率的程序

实例位置：光盘\mingrisoft\02\083

实例说明

在实际应用中，经常需要根据用户填写内容的不同，将表单提交到不同的处理页。运行本实例，根据在“提交页面”文本框中输入的页面名称，将表单提交到相应的处理页。程序运行结果如图2.32所示。

图2.32 可以提交到不同处理页的表单

技术要点

本实例主要通过JavaScript来设置表单的action属性（确定提交到的处理页面），然后调用表单的submit()方法（表示执行提交表单的命令）。

实现过程

实现可以提交到不同处理页的表单的关键代码如下。

```
<script type="text/javascript">
```

```
functionMycheck() {  
    var txt=document.form1.txt_action.value;  
    document.form1.action=txt;  
    document.form1.submit();  
}  
</script>  
<input type="button" name="Submit" value="确定"  
onClick="Mycheck();">
```

举一反三

根据本实例，读者可以：

开发根据文件类别上传文件的页面；

开发在线论坛中的用户等级设置页面。

# 第3章 实用JavaScript函数

数据验证

字符串处理

随机函数

日期与时间

## 3.1 数据验证

在动态网站中，为了确保用户输入信息的有效性 & 准确性，必须对用户输入的数据加以验证，保存数据的准确及安全。

### 实例084 通过正则表达式验证日期

本实例是一个提高录入准确性的程序

实例位置：光盘\mingrisoft\03\084

实例说明

为了确保用户输入信息的有效性，应该检查用户输入信息的合法性。在客户端应用JavaScript判断输入信息的合法性可以提高程序的执行速度。例如，本例在提交表单执行查询语句之前，先判断输入日期是否合法，就可以减少服务器的工作量。运行本实例，在“从”字后面的文本框中输入“2007-2-1”，在“到”字后面的文本框中输入“2007-2-29”后，单击“查询”按钮后，将弹出如图3.1所示的对话框，原因是2007年的2月份没有29日。



图3.1 通过正则表达式验证日期

技术要点

在 JavaScript 中使用正则表达式主要是通过正则表达式对象实现的，使用正则表达式对象的语法格式如下。

语法1:

```
re =/pattern/[flags]
```

语法2:

```
re = new RegExp("pattern", ["flags"])
```

功能：本对象包含正则表达式模式以及表明如何应用模式的标志。

参数说明如下。

- re: 必选项，将要赋值为正则表达式模式的变量名。
- pattern: 必选项，要使用的正则表达式模式。如果使用语法1，用“/”字符分隔模式。如果用语法2，用引号将模式引起来。
- flags: 可选项，用于指定匹配标志，如果使用语法2 要用引号将flag 引起来。标志可以组合使用，常用的标记有g（全文查找出现的所有pattern）、i（忽略大小写）、m（多行查找）。

使用test方法`regexp.test(str)`可以返回一个boolean值，它指出在被查找的字符串中是否存在模式。

注意：在JavaScript中调用正则表达式时，正则表达式以正斜线“/”开头和结尾。

验证日期格式是否正确时，需要注意以下几点。

- (1) 年份由4位数字组成。
- (2) 月份由1~12的数字组成。
- (3) 1、3、5、7、8、10、12月份为31天；4、6、9、11月份为30天。
- (4) 闰年的2月份为29天，非闰年的2月份为28天。
- (5) 日期分隔符可以是“/”或“-”。

验证日期格式的正则表达式如下。

```

/^((((1[6-9]|[2-9]\d)\d{2}) (\/|\-) (0?[13578]|1[02])
(\/|\-) (0?[1-9]| [12]\d|3[01])) | (((1[6-9]|[2-9]\d)\d{2})
(\/|\-) (0?[13456789]|1[012]) (\/|\-) (0?[1-9]| [12]\d|30)) |
(((1[6-9]|[2-9]\d)\d{2}) (\/|\-)0?2(\/|\-) (0?[1-9]|1\d|2[0-
8])) | (((1[6-9]|[2-9]\d) (0[48] | [2468] [048] | [13579] [26]) | ((16|
[2468] [048] | [3579] [26])00))-0?2-29-))$/

```

## 实现过程

(1) 应用JavaScript编写验证输入的日期格式是否正确的函数CheckDate(), 该函数只有一个参数INDate, 用于指定要判断的日期字符串, 返回值为true或false, 代码如下:

```

<script language="javascript">
functionCheckDate(str) {
    //在JavaScript中, 正则表达式只能使用"/"开头和结束, 不能
    使用双引号
    varExpression=/^((((1[6-9]|[2-9]\d)\d{2}) (\/|\-) (0?
    [13578]|1[02]) (\/|\-) (0?[1-9]| [12]\d|3[01])) | (((1[6-9]| [2-
    9]\d)\d{2}) (\/|\-) (0?[13456789]|1[012]) (\/|\-) (0?[1-9]|
    [12]\d|30)) | (((1[6-9]|[2-9]\d)\d{2}) (\/|\-)0?2(\/|\-) (0?[1-
    9]|1\d|2[0-8])) | (((1[6-9]|[2-9]\d) (0[48] | [2468] [048] |
    [13579] [26]) | ((16| [2468] [048] | [3579] [26])00))-0?2-29-))$/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }else{
        return false;
    }
}
}

```

```
</script>
```

(2) 编写自定义的JavaScript函数check(), 用于在提交表单前判断用户输入是否合法, 在该函数中再调用CheckDate()函数判断输入的日期是否正确, 如果不正确则给予提示, 否则提交表单。关键代码如下。

```
<script language="javascript">
function check(myform) {
    if(myform.sDate.value=="") {
        alert("请输入开始日期");myform.sDate.focus();return;
    }
    if(CheckDate(myform.sDate.value)) {
        alert("您输入的开始日期不正确! \n请注意日期格式 (如:
2007/07/17或2007-07-17) 或闰
年! ");myform.sDate.focus();return;
    }
    if(myform.eDate.value=="") {
        alert("请输入截止日期");myform.eDate.focus();return;
    }
    if(CheckDate(myform.eDate.value)) {
        alert("您输入的截止日期不正确! \n请注意日期格式 (如:
2007/07/17或2007-07-17) 或闰
年! ");myform.eDate.focus();return;
    }
    myform.submit();
}
</script>
```

举一反三

根据本实例，读者可以：

在会员或用户注册页面中，判断输入的生日是否正确；

在日期型数据查询时，判断输入的日期是否正确。

## 实例085 验证输入的日期格式是否正确

本实例是一个提高录入准确性的程序

实例位置：光盘\mingrisoft\03\085

实例说明

在实例084中已经介绍了如何通过正则表达式验证用户输入的日期格式是否正确，由于该方法的正则表达式比较复杂，本节将介绍如何通过普通的 JavaScript 代码验证日期格式。运行本实例，在“从”字后面的文本框中输入“2007-02-29”，在“到”字后面的文本框中输入“2007-03-30”后，单击“查询”按钮后，将弹出如图 3.2 所示的对话框，原因是 2007 年的 2 月份没有 29 日。



图3.2 验证输入的日期格式是否正确

技术要点

应用JavaScript编写验证输入的日期格式是否正确的代码时，需要注意以下几点。

(1) 首先需要从输入的字符串中提取出年份、月份和日，并判断输入的年份、月份和日是否是大于0的数字，并且将月份和日中小于10

的数字格式化成长度为2的字符串（在其前面填充“0”）。

（2）将提取并格式化后的年份、月份和日重新组合，使其组合成为“YYYY-MM-DD”格式的字符串，并判断新组合后的字符串的长度是否为10。

（3）判断输入的年份是否为闰年，继而判断2月份的天数。

说明：闰年的条件是能被4整除，但不能被100整除或者能被400整除。

（4）根据1、3、5、7、8、10、12月份为31天，其他月份为30天的原则判断除2月份以外的月份的日是否正确。

（5）判断月份是否大于12。

（6）日期分隔符可以是“/”或“-”。

#### 实现过程

（1）应用JavaScript编写验证输入的日期格式是否正确的函数CheckDate()，该函数只有一个参数INDate，用于指定要判断的日期字符串，返回值为true或false，代码如下。

```
<script language="javascript">
//判断输入的日期是否正确
functionCheckDate(INDate) {
if (INDate=="") {return true;}
if(INDate.indexOf('-',0)!=-1) {    separate="-"}
else{
    if(INDate.indexOf('/',0)!=-1) {separate="/"}
    else {return true;}
}
area=INDate.indexOf(separate,0)
//获取年份
subYY=INDate.substr(0,area)
```

```

    if(isNaN(subYY) || subYY<=0) {
        return true;
    }
    //转换月份
    subMM=INDate.substr(area+1, INDate.indexOf(separate, area
+1)-(area+1))
    if(isNaN(subMM) || subMM<=0) {
        return true;
    }
    if(subMM.length<2) {subMM="0"+subMM}
    //转换日
    area=INDate.lastIndexOf(separate)
    subDD=INDate.substr(area+1, INDate.length-area-1)
    if(isNaN(subDD) || subDD<=0) {
        return true;
    }
    if(eval(subDD)<10) {subDD="0"+eval(subDD)}
    NewDate=subYY+"-"+subMM+"-"+subDD
    if(NewDate.length!=10) {return true;}
        if(NewDate.substr(4, 1)!="-") {return true;}
        if(NewDate.substr(7, 1)!="-") {return true;}
    varMM=NewDate.substr(5, 2);
    varDD=NewDate.substr(8, 2);
    if((subYY%4==0&& subYY%100!=0) || subYY%400==0) {//判断是
否为闰年
        if(parseInt(MM)==2) {
            if(DD>29) {return true;}

```

```

    }
}else{
    if(parseInt(MM)==2){
        if(DD>28){return true;}
    }
}
varmm=newArray(1,3,5,7,8,10,12);//判断每月中最大天数
for(i=0;i<mm.length;i++){
    if(parseInt(MM)==mm[i]){
        if(parseInt(DD)>31){
            return true;
        }else{
            return false;
        }
    }
}
if(parseInt(DD)>30){return true;}
if(parseInt(MM)>12){return true;}
return false;
}
</script>

```

(2) 编写自定义的JavaScript函数check(), 用于在提交表单前判断用户输入是否合法, 在该函数中再调用CheckDate()函数判断输入的日期是否正确, 如果不正确则给予提示, 否则提交表单。关键代码如下。

```

<script language="javascript">
function check(myform){

```

```
if(myform.sDate.value=="") {
    alert("请输入开始日期");myform.sDate.focus();return;
}
if(CheckDate(myform.sDate.value)) {
    alert("您输入的开始日期不正确! \n请注意日期格式(如:
2007/07/17或2007-07-17) 或闰
年!");myform.sDate.focus();return;
}
if(myform.eDate.value=="") {
    alert("请输入截止日期");myform.eDate.focus();return;
}
if(CheckDate(myform.eDate.value)) {
    alert("您输入的截止日期不正确! \n请注意日期格式(如:
2007/07/17或2007-07-17) 或闰
年!");myform.eDate.focus();return;
}
myform.submit();
}
</script>
```

举一反三

根据本实例，读者可以：

在会员或用户注册页面中，判断输入的生日是否正确；

在日期型数据查询时，判断输入的日期是否正确。

## [实例086 检查表单元素是否为空](#)

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\086

## 实例说明

在动态网站开发时，经常会遇到要求数据表中所有字段均不为空的情况，这就要求程序员在开发网站时，对用户输入的空数据加以控制，也就是要求表单中全部输入均不允许为空。在本实例中将介绍一个简单、快捷的方法：利用循环语句进行判断。运行本实例，单击“保存”按钮时，将弹出如图3.3所示的对话框。



图3.3 检查表单元素是否为空

## 技术要点

实现检查表单元素是否为空主要是通过for语句和form对象的相关属性进行判断的。下面详细介绍form对象的常用属性。

创建form对象的语法如下。

```
<FORM ACTION="url" METHOD="GET|POST" NAME="formname"
TARGET="target"></FORM>
```

form对象的常用属性如下。

- name: form对象的名字。
- method: 用于定义表单结果从浏览器传送到服务器的方法，一般有两种方法：GET和POST。
- action: 用来定义表单处理程序的目标地址（相对地址或绝对地址，一般为相对地址）。
- target: 设路表单提交后，显示处理结果的窗口。
- length: 表单中表单元素的个数。
- elements: 包含表单中所有表单元素的数组。

说明：在JavaScript中form表单的elements属性的value属性代表指定表单元素的值；name属性代表指定表单元素的名称；title属性代表表单元素的标题。

### 实现过程

(1) 应用JavaScript编写检查表单元素是否为空的函数check()，该函数只有一个参数Form，用于指定要进行检查的表单对象，无返回值，代码如下。

```
//检查表单元素是否为空
function check(Form) {
    for(i=0;i<Form.length;i++) {
        if(Form.elements[i].value== "") {           //Form的属性
elements的首字母e要小写
            alert(Form.elements[i].title+"不能为空!");
            Form.elements[i].focus();
            return;
        }
    }
}
```

```
    }  
    Form.submit();  
}
```

(2) 在页面中添加所需的表单及其表单元素，在“保存”按钮的onClick事件中调用check()函数判断表单元素是否为空，注意一定要将当前表单作为参数传递到check()函数中。关键代码如下。

```
<form name="form1" method="post" action="">  
    留言人：  
    <input name="author" type="text" id="author" size="30"  
title="留言人">  
    E-mail：  
    <input name="email" type="text" id="email" size="72"  
title="E-mail地址">  
    留言内容：  
    <textarea name="content" cols="70"  
rows="10" class="wenbenkuang"  
title="留言内容" id="content"></textarea>  
    <input name="Submit" type="button"  
class="btn_grey" value="保存"  
onClick="check(form1)">  
    <input name="Submit2"  
type="reset" class="btn_grey" value="重置">  
</form>
```

举一反三

根据本实例，读者可以：

开发在线论坛或博客中的添加留言信息模块；

开发网站后台管理模块中的信息添加和修改模块；

开发用户或会员注册模块。

## 实例087 验证E-mail是否正确

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\087

实例说明

在数据录入页面中，经常需要对输入的 E-mail 地址进行判断，这样可以提高数据的有效性。本实例将介绍如何通过正则表达式验证 E-mail地址是否正确。运行本实例，在E-mail文本框中输入E-mail地址“wgh717”，单击“注册”按钮，将弹出如图3.4所示的对话框，提示用户输入的E-mail地址不正确。



图3.4 验证E-mail是否正确

技术要点

在JavaScript中，验证E-mail地址是否合法，可以通过正则表达式实现。验证E-mail地址的正则表达式如下。

```
/\w+([-+.' ]\w+)*@\w+([-.\ ]\w+)*.\w+([-.\ ]\w+)*
```

在上面的正则表达式中，应用了正则表达式的字符类及重复字符。下面给出正则表达式中常用字符类及重复字符的用法。

正则表达式的常用字符类及其匹配情况如表3.1所示。

表3.1 正则表达式的常用字符类

字符类	匹 配
[...]	位于中括号内的任意字符
[^...]	除括号内字符之外的任意字符
.	除换行符和其他 Unicode 行终止符之外的任意字符
\w	任何 ASCII 单字符，可以是“a-zA-Z0-9”之内的任意字符
\W	任何 ASCII 非单字符，必须是除“a-zA-Z0-9”之内任意字符以外的字符
\s	任何 Unicode 空白符
\S	任何非 Unicode 空白符
\d	任何 ASCII 数字，可以是0~9的任意数字
\D	除了 ASCII 数字之外的任何字符
[b]	退格直接量

正则表达式的常用重复字符及其描述如表3.2所示。

表3.2 正则表达式的常用重复字符及其描述

字 符	描 述
{n,m}	匹配前一项至少 n 次，但不能超过 m 次
{n,}	匹配前一项 n 次，或更多次
{n}	匹配前一项恰好 n 次
?	匹配前一项 0 次或 1 次
+	匹配前一项 1 次或多次
*	匹配前一项 0 次或多次

注意：在JavaScript中调用正则表达式时，正则表达式以正斜线“/”开头和结尾。

实现过程

(1) 使用JavaScript编写一个用于检测E-mail地址是否正确的函数checkemail(), 该函数只有一个参数str, 用于获取输入的E-mail地址, 返回值为true或false。代码如下。

```
<script language="javascript">
function checkemail(str){
    //在JavaScript中, 正则表达式只能使用"/"开头和结束, 不能使用双引号
    varExpression=/\w+([-+.' ]\w+)*@\w+([-.\ ]\w+)*\.\w+([-.\ ]\w+)*;/;
    var objExp=newRegExp(Expression);
    if(objExp.test(str)==true){
        return true;
    }else{
        return false;
    }
}
</script>
```

(2) 调用 checkemail() 函数判断 E-mail 地址是否正确, 并显示相应的提示信息。关键代码如下。

```
<script language="javascript">
function check(myform){
    if(myform.email.value==""){
        alert("请输入E-mail地址!");myform.email.focus();return;
    }
    if(!checkemail(myform.email.value)){
```

```
    alert("您输入的E-mail地址不正  
确!");myform.email.focus();return;  
}  
myform.submit();  
}  
</script>
```

举一反三

根据本实例，读者可以：

开发会员注册模块中判断E-mail格式是否正确；

开发在线订单模块中判断E-mail格式是否正确。

## 实例088 通过正则表达式验证电话号码

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\088

实例说明

在网络应用程序的数据录入页面中，经常需要对输入的电话号码进行判断，在 JavaScript中判断电话号码可以通过正则表达式实现。运行本实例，在“家庭电话”文本框中输入联系电话号码“0431-49766”，单击“确定保存”按钮，将返回如图3.5所示的页面，提示用户输入的电话号码不正确。



图3.5 通过正则表达式验证电话号码

### 技术要点

一个正则表达式就是由普通字符（例如字符a~z）以及特殊字符（称为元字符）组成的文字模式。该模式描述在查找文字主体时匹配的一个或多个字符串。正则表达式作为一个模板，将某个字符模式与所搜索的字符串进行匹配。

验证电话号码的正则表达式如下。

```
/(\d{3}-)?\d{8}|(\d{4}-)(\d{7})/
```

### 实现过程

(1) 使用JavaScript编写一个用于检测电话是否正确的函数 `checktel()`，该函数只有一个参数 `tel`，用于获取输入的联系电话号码，返回值为 `true` 或 `false`。代码如下。

```
<script language="javascript">
```

```

function checktel(tel) {
    var str=tel;
    //在JavaScript中，正则表达式只能使用"/"开头和结束，不能
    使用双引号
    varExpression=/(\d{3}-)?\d{8}|(\d{4}-)(\d{7})/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }else{
        return false;
    }
}
</script>

```

(2) 调用Mycheck()函数判断家庭电话是否正确，并显示相应的提示信息。关键代码如下。

```

<script language="javascript">
functionMycheck(myform) {
    if(myform.familytel.value=="") {
        alert("请输入家庭电
话!!");myform.familytel.focus();return;
    }
    if(!checktel(myform.familytel.value)) {
        alert("您输入的家庭电话不正
确!");myform.familytel.focus();return false;
    }
    form1.submit();
}

```

</script>

举一反三

根据本实例，读者可以：

实现用户注册模块中的用户电话号码的判断；

实现企业客户管理系统中的添加客户信息时判断联系电话。

## 实例089 验证输入的字符串是否为汉字

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\089

实例说明

在动态网站的数据录入页面中，经常需要对用户输入的真实姓名进行判断，例如，某博客网站要求输入的真实姓名为汉字，所以如果用户输入的内容不为汉字将被视为不合法。本实例中的真实姓名被限制为只能输入汉字。运行本实例，在“真实姓名”文本框中输入“hui”后，单击“确定保存”按钮，将弹出如图3.6所示的对话框，提示用户输入的真实姓名不正确。



图3.6 验证输入的字符串是否为汉字

### 技术要点

实现验证输入的字符串是否为汉字的方法有很多种，例如可以通过正则表达式进行判断，也可以根据字符的ASCII码进行判断。其中，应用正则表达式判断比较方便、快捷。实现验证输入的字符串是否为汉字的正则表达式如下：

```
/[^\u4E00-\u9FA5]/
```

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。

### 实现过程

(1) 使用JavaScript编写一个用于检测用户输入的真实姓名是否为汉字的函数checkrealname()，该函数只有一个参数realname，用于获取输入的真实姓名，返回值为true或false。代码如下。

```
<script language="javascript">
```

```

function checkrealname(realname) {
    var str=realname;
    //在JavaScript中，正则表达式只能使用"/"开头和结束，不能
    //使用双引号
    varExpression=/[^\u4E00-\u9FA5]/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }else{
        return false;
    }
}
</script>

```

(2) 调用check()函数判断用户输入的真实姓名是否为汉字，并显示相应的提示信息。关键代码如下。

```

<script language="javascript">
function check(myform) {
    if(myform.realname.value=="") {
        alert("请输入真实姓名!");myform.realname.focus();return;
    }
    if(checkrealname(myform.realname.value)) {
        alert("您输入的真实姓名不正确!");myform.realname.focus();return;
    }
    myform.submit();
}

```

</script>

(3) 在页面的合适位路添加“确定保存”按钮，在该按钮的onClick事件中调用check()函数验证输入的字符串是否为汉字，代码如下。

```
<input name="Button" type="button" class="btn_grey" value="确定保存" onClick="check(form1)">
```

举一反三

根据本实例，读者可以：

限制在线订单中的联系人姓名必须为汉字；

限制进销存管理系统中的经手人、联系人字段必须为汉字。

## [实例090 验证身份证号码](#)

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\090

实例说明

在动态网站的会员注册页面中，经常需要对用户输入的身份证号码进行判断，因此笔者编写了一个验证身份证号码的函数，在需要时进行调用即可。例如本实例就应用了该函数，运行本实例，在“身份证号”文本框中输入“220104198007”后，单击“确定保存”按钮，将弹出如图3.7所示的对话框，提示用户输入的身份证号码不正确。



图3.7 验证身份证号码

### 技术要点

要实现验证身份证号码首先需要了解身份证号码的编码规则，现在有效的身份证有两种编码规则，分别是15位居民身份证号码和18位居民身份证号码，下面分别给出这两种编码规则。

15位的居民身份证的号码为。

aa bb cc yymmdd nnn

其中，aa代表省份编码，bb代表省地区编码，cc代表地区级县（市）编码，yymmdd代表出生年月日，其中，yy用年份的后两位表示；nnn代表同一地区出生年月日相同的人的序号，该序号以户主为准，子女依照户主序号确定，奇数=男，偶数=女，且女=男+1。

18位的居民身份证号码为。

aa bb cc yyyymmdd nnn r

其中，aa bb cc 所代表的含义与15 位的相同，yyyymmdd 代表出生年月日，年份为4 位；nnn代表同一地区出生年月日相同的人的序号；r为校验位，数字是1~10，因为不能用10（否则就成了19位数），所以用X代表10。

验证身份证号码的正则表达式如下：

```
/^\d{17}[\d|X]|\d{15}$/
```

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。

### 实现过程

（1）使用JavaScript编写一个用于验证身份证号码是否合法的函数checkeNO()，该函数只有一个参数NO，用于获取输入的身份证号码，返回值为true或false。代码如下。

```
<script language="javascript">  
function checkeNO(NO) {  
    var str=NO;  
    //在JavaScript中，正则表达式只能使用"/"开头和结束，不能  
    使用双引号  
    var Expression=/^\d{17}[\d|X]|\d{15}$/;  
    var objExp=new RegExp(Expression);  
    if(objExp.test(str)==true) {  
        return true;  
    }else{  
        return false;  
    }  
}  
</script>
```

(2) 调用check()函数判断用户输入的身份证号码是否合法，如果不合法，将提示用户输入的身份证号码不正确。关键代码如下。

```
<script language="javascript">
function check(myform) {
    if(myform.number.value=="") {
        alert("请输入身份证号
码!");myform.number.focus();return;
    }
    if(!checkNO(myform.number.value)) {
        alert("您输入的身份证号码不正
确!");myform.number.focus();return;
    }
    myform.submit();
}
</script>
```

(3) 在页面的合适位路添加“确定保存”按钮，在该按钮的onClick事件中调用check()函数验证身份证号码是否合法，代码如下。

```
<input name="Button" type="button" class="btn_grey"
value="确定保存"
onClick="check(form1)">
```

举一反三

根据本实例，读者可以：

在办公自动化管理系统中判断员工的身份证号是否合法；

根据正确的身份证号自动计算员工的年龄及出生日期。

## 实例091 验证用户名和密码

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\091

实例说明

在动态网站的用户注册页面中，经常需要对用户输入的用户名和密码的位数及成员组成进行判断，这样可以规范用户的注册信息。例如本实例，要求用户名由3~10位的字母、数字和下划线组成，密码由6~20位的字母、数字、下划线和点“.”组成并且首字符为字母，这时就需要对用户输入的用户名和密码进行判断，因此笔者编写了两个函数，分别用于验证输入的用户名和密码是否合法。运行本实例，在“用户名”文本框中输入“无语”后，单击“确定保存”按钮，将弹出如图3.8所示的对话框，提示用户输入的用户名不合法。

技术要点

实现验证用户名是否由3~10位的字母、数字和下划线组成的正则表达式如下。

```
/^(\w){3,10}$/
```

实现验证密码是否由6~20位的字母、数字、下划线和点“.”组成的正则表达式如下。

```
/^[A-Za-z]{1}([A-Za-z0-9]|[. _]){5,19}$/
```

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。



图3.8 验证用户名和密码

### 实现过程

(1) 使用JavaScript 编写一个用于验证用户名是否合法的函数 `checkusername ()`，该函数只有一个参数 `username`，用于获取输入的用户名，返回值为 `true` 或 `false`。代码如下。

```
<script language="javascript">
function checkusername(username) {
    var str=username;
    //在JavaScript中，正则表达式只能使用"/"开头和结束，不能
    //使用双引号
    varExpression=/^(\w){3,10}$/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }
}
```

```

    }else{
        return false;
    }
}
</script>

```

(2) 使用 JavaScript 编写一个用于验证密码是否合法的函数 checkePWD(), 该函数只有一个参数PWD, 用于获取输入的密码, 返回值为true或false。代码如下。

```

<script language="javascript">
function checkePWD(PWD) {
    var str=PWD;
    //在JavaScript中, 正则表达式只能使用"/"开头和结束, 不能
    //使用双引号
    varExpression=/^[A-Za-z]{1}([A-Za-z0-9]|[. _]){5,19}$/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }else{
        return false;
    }
}
</script>

```

(3) 编写自定义的JavaScript函数check(), 用于判断用户输入的用户名和密码是否合法, 如果不合法, 将给予提示信息。关键代码如下。

```

<script language="javascript">
function check(myform) {

```

```
    if(myform.username.value=="") {
        alert("请输入用户名!");myform.username.focus();return;
    }
    if(!checkusername(myform.username.value)) {
        alert("您输入的用户名不合法!");myform.username.focus();return;
    }
    if(myform.PWD.value=="") {
        alert("请输入密码!");myform.PWD.focus();return;
    }
    if(!checkPWD(myform.PWD.value)) {
        alert("您输入的密码不合法!");myform.PWD.focus();return;
    }
    if(myform.PWD1.value=="") {
        alert("请确认密码!");myform.PWD1.focus();return;
    }
    if(myform.PWD1.value!=myform.PWD.value) {
        alert("您两次输入的密码不一致，请重新输入!");myform.PWD.focus();return;
    }
    myform.submit();
}
</script>
```

举一反三

根据本实例，读者可以：

判断用户输入的单据号码是否正确。

## 实例092 验证车牌号码

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\092

实例说明

在开发动态网站时，经常需要对用户输入的信息进行验证。例如，在物流信息网的车辆信息发布页面中，就需对输入的车牌号码进行判断。下面将通过具体实例介绍如何编写验证车牌号码是否正确的函数。运行本实例，在“车牌号码”文本框中输入“JA-0123”后，单击“提交”按钮，将弹出如图3.9所示的对话框，提示输入的车牌号码不正确。



图3.9 验证车牌号码

技术要点

要实现验证车牌号码是否正确，首先需要了解车牌号码的编码规则，下面将给出验证国内非军用车车牌号码时应注意的几点规则。

- (1) 第1个字符必须为汉字。
- (2) 第2个字符必须为字母（a~z或A~Z）。
- (3) 第3位为分隔符“-”。
- (4) 分隔符后面由5位数字或字母组成。

根据上面的规则得出验证车牌号码是否正确的正则表达式如下。

```
/^[\\u4E00-\\u9FA5]?[a-zA-Z]-\\w{5}$/
```

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。

#### 实现过程

(1) 使用JavaScript编写一个用于验证车牌号码是否正确的函数checkNo()，该函数只有一个参数str，用于获取输入的车牌号码，返回值为true或false。代码如下。

```
<script language="javascript">
function checkNo(str) {
    varExpression=/^[\\u4E00-\\u9FA5]?[a-zA-Z]-\\w{5}$/;
    varobjExp=newRegExp(Expression);
    return objExp.test(str)
}
</script>
```

(2) 编写check()函数判断用户输入的车牌号码是否正确，如果不正确，将提示用户输入的车牌号码不正确。关键代码如下。

```
<script language="javascript">
function check(form) {
    if(form.carNo.value=="") {
        alert("请输入车牌号码!");form.carNo.focus();return;
    }
}
```

```

    }
    if(!checkNo(form.carNo.value)){
        alert("您输入的车牌号码不正
确!");form.carNo.focus();return;
    }
    form.submit();
}
</script>

```

(3) 在页面的合适位路添加“提交”按钮，在该按钮的onClick事件中调用check()函数验证车牌号码是否合法，代码如下。

```

<input name="Button" type="button" class="btn_grey"
value="提交"
onClick="check(form1)">

```

举一反三

根据本实例，读者可以：

判断用户输入的单据号码是否正确。

## [实例093 验证网站地址](#)

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\093

实例说明

在动态网站的用户注册页面中，为了保证输入信息的有效性，经常需要对用户输入的网址地址进行判断，因此笔者编写了一个验证网站地址是否合法的函数。例如本实例就要求对用户输入的个人主页地址进行判断。运行本实例，在“个人主页”文本框中输入

“http://wgh”后，单击“确定保存”按钮，将弹出如图3.10所示的对话框，提示输入的个人主页地址不合法。



图3.10 验证网站地址

### 技术要点

实现验证网站地址是否合法的正则表达式如下。

```
http(s)?://([\w-]+\.)+[\w-]+(/[\w-./?%&=]*)?
```

注意：由于在JavaScript中不能直接输出“/”字符，需要应用转义字符“\”输出一个字符“/”，因此转换后的正则表达式如下。

```
/http(s)?:\:\/\/([\w-]+\.)+[\w-]+(\//[\w-.\//?%&=]*)?/
```

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。

### 实现过程

(1) 使用JavaScript编写一个用于验证网站地址是否合法的函数 `checkurl()`，该函数只有一个参数 `url`，用于获取输入的网站地址，

返回值为true或false。代码如下。

```
<script language="javascript">
function checkurl(url) {
    var str=url;
    //判断url地址的正则表达式为:http(s)?://([\w-]+\.)+
    [\w-]+(/[\w-./?%&=]*)?
    //下面的代码中应用了转义字符"\输出一个字符"/"
    varExpression=/http(s)?:\\/\\/([\w-]+\.)+[\w-]+
    (\\/[\w-.\\/?%&=]*)?\/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(str)==true) {
        return true;
    }else{
        return false;
    }
}
</script>
```

(2) 编写check()函数判断用户输入的网站地址是否合法, 如果不合法, 将给予提示信息。关键代码如下。

```
<script language="javascript">
function check(myform) {
    varhomepage=myform.homepage.value;
    if(homepage=="") {
        alert("请输入个人主页地址!");
        myform.homepage.focus(); return;
    }else{
        if(!checkurl(homepage)) {
```

```
        alert("您输入的个人主页地址不合法!");
        myform.homepage.focus();
        return;
    }
}
myform.submit();
}
```

</script>

(3) 在页面的合适位置添加“确定保存”按钮，在该按钮的onClick事件中调用check()函数验证网站地址是否合法，代码如下：

```
<input name="Button" type="button" class="btn_grey"
value="确定保存"
onClick="check(form1)">
```

举一反三

根据本实例，读者可以：

过滤员工信息表中指定姓氏的员工信息；

验证用户输入的邮箱地址是否合法。

## [实例094 验证数量和金额](#)

本实例是一个提高录入准确性的程序

实例位置：光盘\mingrisoft\03\094

实例说明

在动态网站开发时，经常会遇到限制用户输入数据类型的问题。例如，“数量”文本框通常只能输入正整数，而“金额”文本框又只能输入数值型数据。这就要求程序员在开发网站时，对用户输入的数值型数据进行判断。例如本实例的采购信息登录页面，就要求对单价

和数量进行控制。运行本实例，在“单价”文本框中输入“-”时，系统将弹出如图 3.11 所示的对话框，提示用户输入错误。

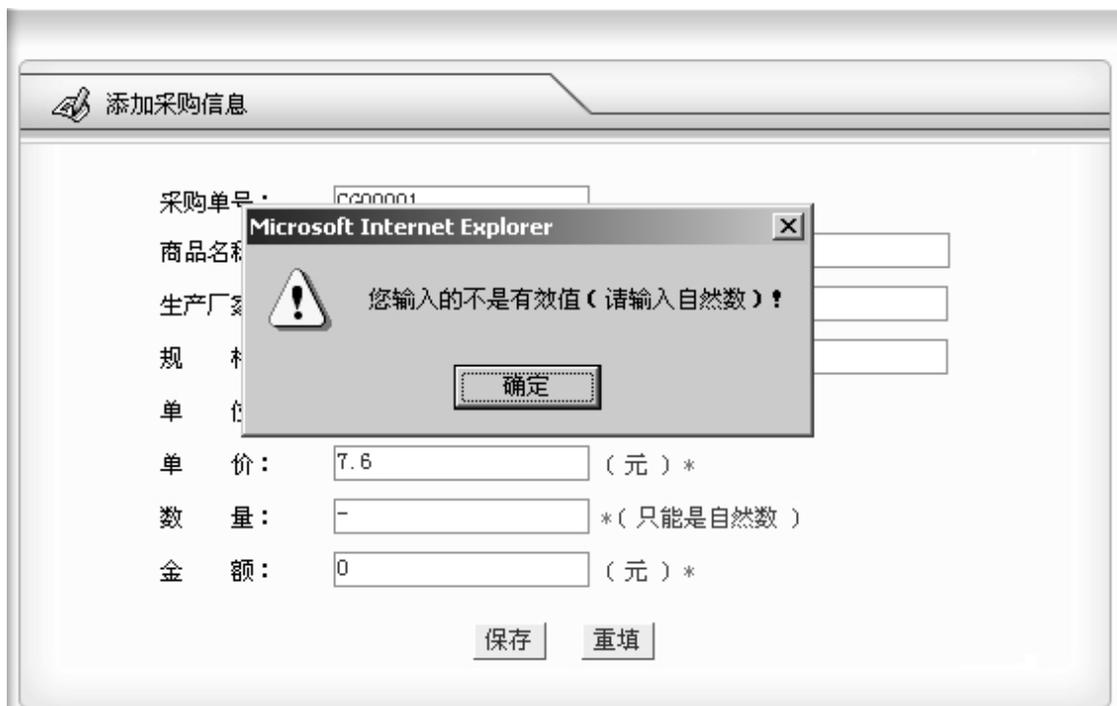


图3.11 验证数量和金额

### 技术要点

要验证输入的金额和数量（只能是正整数）是否合法首先需要知道它们各有何特点，下面分别说明。

- 金额：通常情况下，金额为数值型数据，包括小数，如7.6。
- 数量：通常情况下，数量一般为正整数，即只能由0~9 之间的数字组成，如770。

根据上面的分析，可以总结出判断数量和金额的方法。

判断金额可以使用JavaScript的全局函数isNaN()，其关键代码如下。

```
isNaN(form1.dj.value)
```

判断数量可以使用正则表达式，其关键代码如下。

```
str=form1.sl.value;
```

```
varExpression=/^[1-9]+(\d*$)/;
var objExp=newRegExp(Expression);
if(objExp.test(str)==false){
    alert("您输入的不是有效值（请输入自然数）！");
    form1.s1.focus();form1.s1.select();return;
}
```

### 实现过程

(1) 应用JavaScript编写验证输入的单价（数值型数据）是否合法的函数checkPrice()。代码如下。

```
<script language="javascript">
function checkPrice(){
if (form1.dj.value==0&& form1.dj.value==""){
    alert("请输入单
价！");form1.dj.focus();form1.dj.select();return;}
if(isNaN(form1.dj.value)){
    alert("您输入的单价不是有效
值！");form1.dj.focus();form1.dj.select();return;
}
}
</script>
```

(2) 应用JavaScript编写验证输入的数量（数值型数据）是否合法的函数checkNum()。代码如下。

```
<script language="javascript">
function checkNum(){
if (form1.s1.value==0&& form1.s1.value==""){
    alert("请输入数
量！");form1.s1.focus();form1.s1.select();return;
}
```

```

}
str=form1.sl.value;
varExpression=/^[1-9]+(\d*$)/;
var objExp=newRegExp(Expression);
if(objExp.test(str)==false){
    alert("您输入的不是有效值（请输入自然数）！");
    form1.sl.focus();form1.sl.select();return;
}
}
</script>

```

(3) 在网页的适当位路添加用于获取单价、数量和金额的文本框，并为单价和数量文本添加onkeyup事件执行的操作，用于判断输入的单价和数量是否合法。关键代码如下。

```

<tr valign="baseline">
<tdheight="27"align="center"nowrap>单 价： </td>
<td><font color="#00CCFF">
<input
type="text"name="dj"value="0"onKeyUp="checkPrice()">
</font><span class="style1">( 元 )</span><font
color="#FF0000"> *</font></td>
</tr>
<tr valign="baseline">
<tdheight="27"align="center"nowrap>数 量： </td>
<td><font color="#00CCFF">
<input
type="text"name="sl"value="1"onKeyUp="checkNum()"onBlur="ao
toPay()">

```

```

</font><font color="#FF0000">*</font><span
class="style1">( 只能是自然数 )</span></td>
</tr>
<tr valign="baseline">
<tdheight="27"align="center"nowrap>金 额: </td>
<td><input type="text" name="je" value="0" readonly="yes">
<span class="style1">( 元 )</span><font
color="#FF0000"> *</font></td>
</tr>

```

举一反三

根据本实例，读者可以：

验证用户输入的其他数据类型。

## 实例095 验证字符串是否以指定字符开头

本实例是一个实用、方便的程序

实例位置：光盘\mingrisoft\03\095

实例说明

在制作网页时，有时需要判断输入的字符串是否以指定的字符开头。本实例将介绍如何通过正则表达式验证输入字符串是否以指定的字符开头。运行本实例，在“请输入字符串的起始字符”文本框中输入“220”，在“请输入要判断的字符串”文本框中输入

“220104198007”，单击“判断”按钮，将弹出如图3.12所示的对话框。

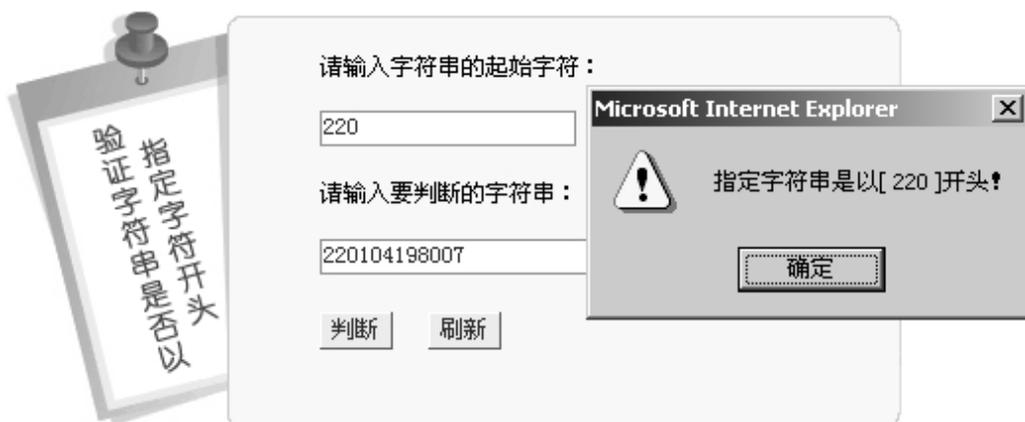


图3.12 验证字符串是否以指定字符开头

### 技术要点

验证字符串是否以指定的字符开头可以使用正则表达式的锚元素“^”实现，例如验证字符串是否以字符a开头的正则表达式如下。

```
/^a/
```

正则表达式的锚元素“^”用于匹配字符串的开头，在多行检索中，匹配一行的开头。

由于本实例中的起始字符是通过变量传递的，所以需要使用下面的代码实现。

```
eval("/^"+startStr+"/")
```

在上面的代码中eval()方法用于计算JavaScript代码串，并返回结果。

说明：关于在JavaScript中调用正则表达式的详细介绍请参考实例084。

### 实现过程

(1) 应用JavaScript编写验证字符串是否以指定的字符开头的函数checkOccur()，该函数有两个参数 startStr（用于指定字符串的起始字符）和 inStr（用于指定要判断的字符串），返回值为true或false。代码如下：

```
<script language="javascript">
```

```
function checkOccur(startStr, inStr) {
    var Expression=eval("/^"+startStr+"/");
    var objExp=new RegExp(Expression);
    return objExp.test(inStr);
}
</script>
```

(2) 编写自定义的JavaScript函数check(), 用于在提交表单前判断用户输入是否合法, 在该函数中再调用 checkOccur() 函数验证字符串是否以指定的字符开头, 并给予提示。关键代码如下:

```
<script language="javascript">
function check(myform) {
    var startStr=myform.startStr.value;
    var inStr=myform.inStr.value;
    if(startStr=="") {
        alert("请输入字符串的起始字符!");
        myform.startStr.focus();return;
    }
    if(inStr=="") {
        alert("请输入要判断的字符串!");
        myform.inStr.focus();return;
    }else{
        if(checkOccur(startStr, inStr)) {
            alert("指定字符串是以[ "+startStr+" ]开头!");
            return;
        }else{
            alert("指定字符串不是以[ "+startStr+" ]开头!");
            return;
        }
    }
}
```

```
    }  
  }  
}  
</script>
```

(3) 在页面的合适位路添加“判断”按钮，在该按钮的onClick事件中调用check()函数验证身份证号码是否合法，代码如下：

```
<input name="Submit" type="button" class="btn_grey"  
onClick="check(form1);" value="判断">
```

举一反三

根据本实例，读者可以：

实现在线论坛或留言簿中限制留言主题的字数；

实现新闻网中限制新闻标题的字数。

## 实例096 限制输入字符串的长度

本实例是一个提高录入规范性的程序

实例位置：光盘\mingrisoft\03\096

实例说明

在网站开发时，经常需要限制输入字符串的长度。例如本实例所介绍的博客注册页面就要求注册用户名为10个字母或5个汉字。运行本实例，在“用户名”文本框中输入“lily的lover”后，单击“注册”按钮，将弹出如图3.13所示的对话框，原因是输入的用户名长度为11个字符了（每个汉字占两个字符）。



图3.13 限制输入字符串的长度

### 技术要点

在实现本实例时，需要应用 for 循环重新计算输入字符串中的每个字符的长度，并将计算的结果累加，得出该字符串的实际长度。计算单个字符的实际长度的方法如下。

首先应用string对象的charCodeAt()方法获取指定字符的Unicode编码，然后根据该编码判断字符的实际长度（如果该编码大于255则字符的长度为2，否则字符串的长度为1）。

string对象的charCodeAt()方法用于返回字符串中的第n个字符的Unicode编码，该方法的语法格式如下。

```
string.charCodeAt(n)
```

参数说明如下。

- n: 返回编码字符的下标。

返回值：0~65535之间的16位整数，也就是字符串中第n个字符的Unicode编码。如果n为负数或大于等于字符串的长度，则返回NaN。

### 实现过程

(1) 应用JavaScript编写判断输入的字符串是否大于指定长度的函数checkstr()，该函数有两个参数str（用于指定要判断的字符串）和digit（用于指定字符串的最大长度），返回值为true或false。代码如下。

```
<script language="javascript">
function checkstr(str,digit){
    var n=0;
    for(i=0;i<str.length;i++){
        var leg=str.charCodeAt(i);
        if(leg>255){
            n+=2;
        }else {
            n+=1;
        }
    }
    if (n>digit){
        return true;
    }else {
        return false;
    }
}
</script>
```

(2) 编写自定义的JavaScript函数check()，用于在提交表单前判断用户输入是否合法，在该函数中再调用checkstr()函数判断输入

的字符串是否大于指定长度，如果大于则给予提示，否则提交表单。  
关键代码如下。

```
<script language="javascript">
function check(myform) {
    varname=myform.username.value;
    if(name=="") {
        alert("用户名不能为空");
        myform.username.focus();
        return;
    }else{
        if(checkstr(name,10)) {
            alert("用户名长度不能超过10个字符，请重新输入！");
            myform.username.focus();
            return;
        }
    }
    myform.submit();
}
</script>
```

举一反三

根据本实例，读者可以：

实现在线论坛或留言簿中限制留言主题的字数；

实现新闻网中限制新闻标题的字数。

## 3.2 字符串处理

字符串是程序设计时不可缺少的组成部分，本节将结合实例对字符串处理的相关技术进行介绍。

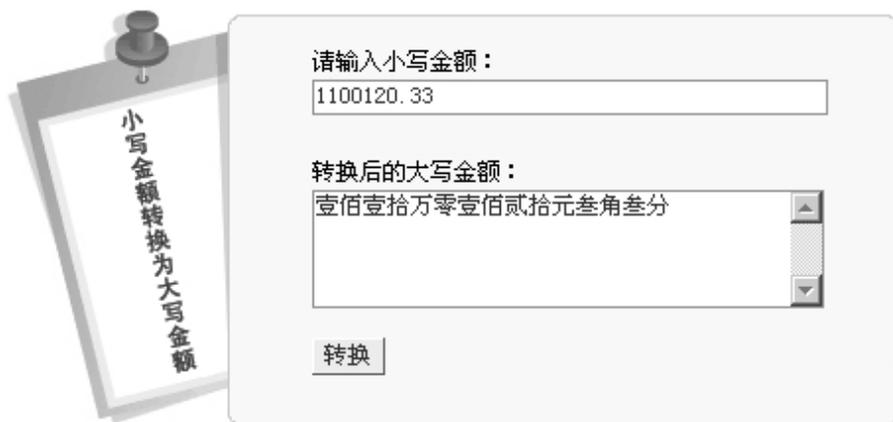
### 实例097 小写金额转换为大写金额

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\03\097

实例说明

在开发涉及财务或与金融相关的网络管理系统或网站时，经常会遇到的一个问题就是金额的大小写转换。在进行账目输入时，采用输入小写金额的方式可以方便用户，但是使用小写金额存在一个最大的问题就是它容易被篡改，这对于金融系统来说是不安全的，解决这种问题的方法就是直接将用户输入的小写金额转换为大写金额。运行本实例，在“请输入小写金额”的文本框中输入所要转换的小写金额后，单击“转换”按钮，在“转换后的大写金额”文本框中将显示转换后的结果，如图3.14所示。



### 图3.14 小写金额转换为大写金额

#### 技术要点

实现将小写金额转换为大写金额的自定义JavaScript函数的思路如下。

- (1) 将大写的数字和单位分别保存在两个字符串中。
- (2) 去除用户输入的小写金额中的小数点。
- (3) 根据用户输入的小写金额长度取出该字符串所用到的单位。
- (4) 取出小写字符对应的大写字符，并与单位组合成新的字符串，同时，还需要对小写金额中存在连续多个0的情况进行处理。

在本实例中主要应用了JavaScript的String对象的substr()方法和length属性。下面对substr()方法和length属性进行介绍。

(1) substr()方法用于返回一个从指定位路开始的指定长度的子字符串，其语法格式如下：

```
string.substr(start, length)
```

参数说明如下。

● start: 指定要获取的子字符串的起始位路。如果是一个负数，将从字符串的尾部开始计算起始位路。即当start的值为-1时，表示最后一个字符。

● length: 指定子字符串中的字符个数。如果省略该参数将返回从start开始到字符串结尾的子串。

(2) length属性用于获取字符串字符的个数，其语法格式如下。

```
string.length
```

注意: string对象的length属性是一个只读的整数，使用该属性时，一个汉字被认为是一个字符。而对于一个字符串，它最后一个字符的下标是其length属性值-1。

#### 实现过程

(1) 首先，对用户输入的小写金额的合法性进行判断，然后对合法数据的小数位数进行控制，使其保留两位小数，再调用自定义的 JavaScript 函数将小写金额转换为大写金额，最后将转换后的值赋给“转换后的大写金额”文本框。代码如下。

```
<script language="JavaScript">
    function convert(str) {
        if(str=="") {
            alert("请输入金额! ");form1.money.focus();return;
        }
        if(isNaN(str)) {
            alert("您输入的金额不合法!");return;
        }
        if(str>999999999999) {
            alert("您输入的金额不能大于999999999999!");return;
        }
        //小数点后保留两位小数
        if(str.indexOf(".")>0) {
            vardecimal=str.length-str.indexOf(".")+1;
            if(decimal>2) {
                str=str.substr(0, str.length+2-decimal);
            }
        }
        value=change(str); //调用自定义函数转换
        form1.moneyok.value=value; //将转换后的值赋给"转换后的大写金额"文本框
    }
</script>
```

(2) 将小写金额转换为大写金额的自定义函数change()的代码如下。

```
<script language="JavaScript">
function change(str) {
    je="零壹贰叁肆伍陆柒捌玖";
    cdw="万仟佰拾亿仟佰拾万仟佰拾元角分";
    varnewstring=(str*100).toString();
    newstringlog=newstring.length;
    newdw=cdw.substr(cdw.length-newstringlog);
    num0=0;    //记录零的个数
    wan=0;    //记录万位出现的次数
    dxje="";    //记录大写金额
    for(m=1;m<newstringlog+1;m++) {
        xzf=newstring.substr(m-1,1); //取出小写金额
        dzf=je.substr(xzf,1); //取出大写金额
        dw=newdw.substr(m-1,1); //取出单位
        if(dzf=="零") {
            dzf="";
            if(dw=="亿") {
            }else if(dw=="万") {
                dzf="";
                wan=1;
            }else if(dw=="元") {
            }else {
                dw=""; //记录单位
            }
        }
        num0=num0+1;
    }
}
```

```

    }else{
        if(num0-wan>0){
            if(dw!="角"){
                dzf="零"+dzf;
            }
        }
        num0=0;
    }
    dxje=dxje+dzf+dw;
}
if(newstring.length!=1){
    if(newstring.substr(newstring.length-2)=="00"){
        dxje=dxje+"整";
    }else{
        dxje=dxje;
    }
}
return dxje;
}
</script>

```

举一反三

根据本实例，读者可以：

实现销售单中销售金额的大小写转换；

实现汇款金额的大小写转换；

实现物资采购单中金额的大小写转换。

## 实例098 去除字符串中的空格

本实例是一个提高规范化的程序

实例位置：光盘\mingrisoft\03\098

实例说明

在开发网络程序时，经常需要对用户输入的字符串中的空格进行处理，例如，去除字符串中的尾部空格、全部空格或首尾空格。而在JavaScript中并没有提供解决该方法或函数，因此有必要自己编写一个这样的函数来解决实际工作中的问题。运行本实例，在“请输入原字符串”文本框中输入所要进行处理的字符串，然后单击“去除首尾空格”按钮，将去除原字符串中的首尾空格并显示在“转换后的字符串”文本框中；单击“去除全部空格”按钮，将去除原字符串中的全部空格并显示在“转换后的字符串”文本框中，如图3.15所示。

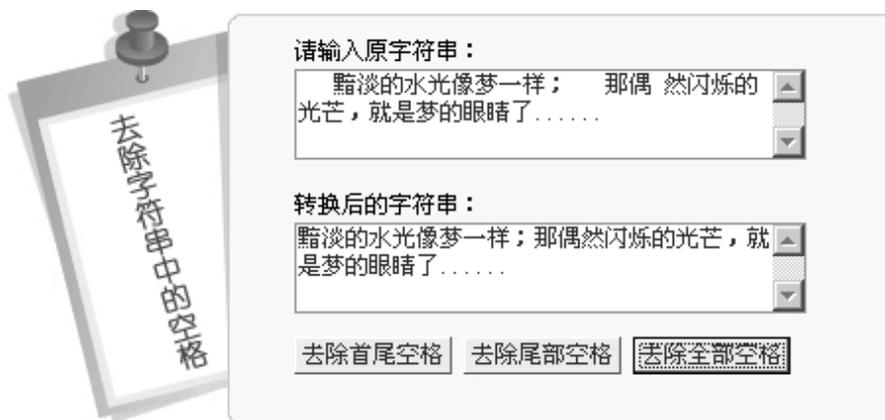


图3.15 去除字符串中的空格

技术要点

本实例主要应用string对象的replace()方法实现。string对象的replace()方法用于执行检索与替换操作。其语法格式如下。

```
string.replace(regExp, string);
```

参数说明如下。

- regExp: 是一个正则表达式。如果正则表达式中设路了标志g，那么该方法将用替换字符串替换检索到的所有与模式匹配的子串，否则只替换所检索到的第一个与模式匹配的子串。

● string: 是要进行替换的字符串。

下面将对本实例中用到的正则表达式的字符类、选择符及锚字符进行介绍。

(1) “\s”：正则表达式的字符类，代表任何Unicode的空白符。

(2) “|”：正则表达式的选择符，代表匹配的是该符号左边的子表达式或右边的子表达式。

(3) “^”：正则表达式的锚字符，代表匹配字符串的开头，在多行检索中，匹配一行的开头。

(4) “\$”：正则表达式的锚字符，代表匹配字符串的结尾，在多行检索中，匹配一行的结尾。

实现过程

(1) 编写去除首尾空格的自定义JavaScript函数trim()，具体代码如下。

```
<script language="javaScript">
    function trim(str) { //去除首尾空格
        var re=/(^\\s*)|(\\s*$)/g;
        str=str.replace(re,"");
        return str;
    }
</script>
```

(2) 编写去除尾部空格的自定义JavaScript函数rtrim()，具体代码如下。

```
<script language="javaScript">
    function rtrim(str) { //去除尾部空格
        var re=/(\\s*$)/g;
        str=str.replace(re,"");
    }
</script>
```

```
    return str;
}
%>
```

(3) 编写去除全部空格的自定义JavaScript函数alltrim(), 具体代码如下。

```
<script language="javaScript">
    function alltrim(str) { //去除全部空格
        var re=/\s/g;
        str=str.replace(re, "");
        return str;
    }
%>
```

(4) 通过相应的按钮, 调用自定义的 JavaScript 函数对字符串中的空格进行处理, 关键代码如下。

```
<input name="Button" type="button" class="btn_grey"
onClick="form1.result.value=trim(form1.source.value)" value="去除首尾空格"/>
<input name="Button22" type="button" class="btn_grey"
onClick="form1.result.value=rtrim(form1.source.value)" value="去除尾部空格"/>
<input name="Button2" type="button" class="btn_grey"
onClick="form1.result.value=alltrim(form1.source.value)" value="去除全部空格"/>
```

举一反三

根据本实例, 读者可以:

网上论坛中添加主题信息或回复信息页面;

应用在博客网站的添加文章页面中。

## 实例099 转换输入文本中的回车和空格

本实例是一个提高规范化的程序

实例位置：光盘\mingrisoft\03\099

实例说明

通常情况下，用户在多行文本框中输入的文字可能带有回车换行和空格符，但是，如果将输入的文字显示到页面上，结果并非和输入时一样，原本输入的回车和空格不见了。这是因为浏览器不能解释获取到的字符串中的回车换行和空格。因此需要编写一个自定义函数将字符串中的回车换行和空格转换为相应的 HTML 标识符或实体名称。本实例将介绍如何应用JavaScript实现转换输入文本中的回车和空格。运行本实例，在“留言”文本框中输入一段文字后，单击“转换后显示”按钮，在页面中将按照输入的格式显示，单击“直接显示”按钮，在页面中显示的结果和输入的格式不同，如图3.16所示。

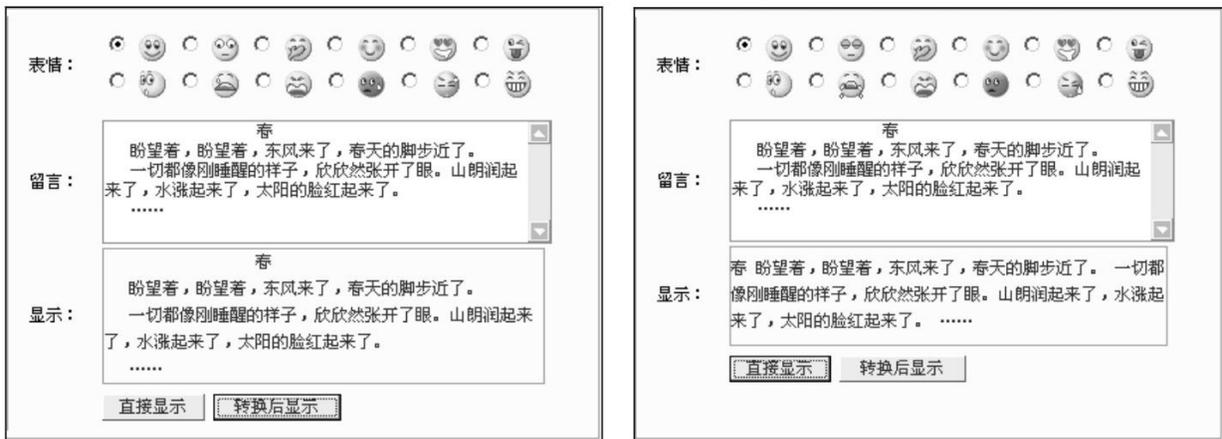


图3.16 转换输入文本中的回车和空格

技术要点

本实例主要应用string对象的replace()方法实现。string对象的replace()方法的详细介绍请参见实例098。

下面对本实例中用到的正则表达式的字符类、选择符及锚字符进行介绍。

- (1) “\s”：正则表达式的字符类，代表任何Unicode的空白符。
- (2) “\r”：正则表达式的字符类，代表回车符。
- (3) “g”：正则表达式的标志，代表执行一个全局匹配，即查找所有匹配的字符。

### 实现过程

(1) 编写转换输入文本中的回车和空格的 JavaScript 自定义函数 convert()，该函数只有一个参数str，用于传递需要转换的字符串，返回值为转换后的字符串。代码如下。

```
<script language="javascript">
function convert(str) {
if (str!="") {
    str=str.replace(/\r/g, "<br>");
    str=str.replace(/\s/g, "&nbsp;");
}
return str;
}
</script>
```

(2) 编写自定义JavaScript函数convertShow()和show()，分别用于将留言内容转换后显示在页面上和将留言内容直接显示在页面上。其中，在函数convertShow()中需要调用convert()函数转换输入文本中的回车和空格。代码如下。

```
<script language="javascript">
function convertShow() {
    if(form1.content.value=="") { //转换后显示
        alert("留言内容不允许为
空!");form1.content.focus();return;
    }
}
```

```

    }else{
        result.innerHTML=convert(form1.content.value);
    }
}
function show() { //直接显示
    if(form1.content.value=="") {
        alert("留言内容不允许为
空! ");form1.content.focus();return;
    }else{
        result.innerHTML=form1.content.value;
    }
}
</script>

```

(3) 在页面添加一个<div>标记，将其命名为“result”，用于显示留言内容。关键代码如下。

```
<div id="result" class="word_grey">&nbsp;  </div>
```

(4) 在页面的适当位路添加“直接显示”和“转换后显示”两个按钮，然后分别在两个按钮的onClick事件中调用convertShow()和show()函数，显示留言内容。关键代码如下。

```

<input name="Button" type="button" class="btn_grey"
value="直接显示" onClick="show()">
<input name="Submit2" type="button" class="btn_grey"
value="转换后显示"
onClick="convertShow()">

```

举一反三

根据本实例，读者可以：

在制作在线论坛时，显示主题信息和回复信息时都可以应用该函数；

在动态网站中通过后台进行添加信息时，也可以应用该函数。

## 实例100 将数字字符串格式化为指定长度

本实例是一个格式化字符串长度的程序

实例位置：光盘\mingrisoft\03\100

实例说明

在实现网站开发过程中，经常会用到将数字格式化为指定长度的字符串的情况，如果不足指定位数时，在该数字的前面用0填充，例如，要显示今天日期（假设今天为“2007-03-15”）时，使用 Date 对象的相应方法只能显示类似“2006-3-15”的字符串，所以需要将月份格式化为指定长度。为此笔者编写一个将数字字符串格式化为指定长度的函数，并将其制作为一个实例，运行本实例，首先在“请输入要格式化的数字”文本框中输入 7，然后在“请输入格式化后字符串的长度”文本框中输入 4，最后单击“转换”按钮后，在“格式化后的字符串”文本框中将显示“0007”，如图3.17所示。

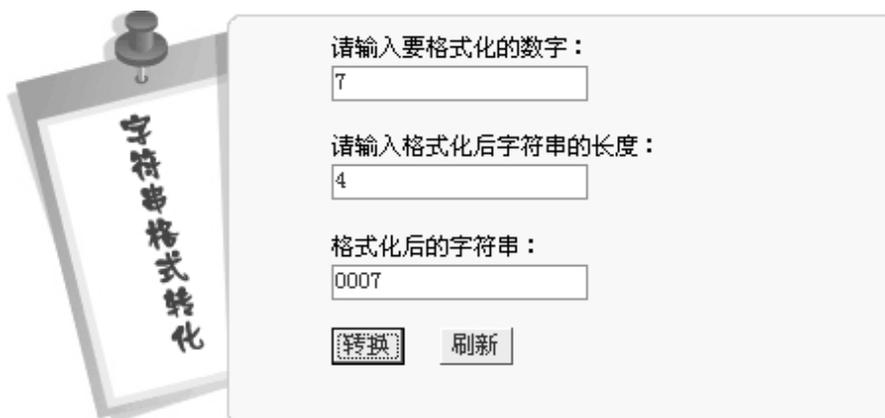


图3.17 将数字字符串格式化为指定长度

技术要点

要实现将数字字符串格式化为指定长度需要应用for循环语句和string对象的length属性实现。string对象的length属性的详细介绍请参见实例097。

### 实现过程

(1) 编写将数字字符串格式化为指定长度的 JavaScript 自定义函数 formatNO(), 该函数有两个参数分别是str (要格式化的数字)、len (格式化后字符串的长度), 返回值为格式化后的字符串。代码如下。

```
<script language="javascript">
function formatNO(str,len) {
    var strLen=str.length;
    for(i=0;i<len-strLen;i++) {
        str="0"+str;
    }
    return str;
}
</script>
```

(2) 编写JavaScript自定义函数deal(), 用于在验证用户输入信息后, 调用formatNO()函数将指定的数字格式化为指定长度。具体代码如下。

```
<script language="javascript">
functiondeal() {
    if(form1.str.value=="")
        {alert("请输入要格式化的数字!");form1.str.focus();return
false;}
    if(isNaN(form1.str.value)) {
```

```

        alert("您输入的数字不正确!");form1.str.focus();return
false;
    }
    if(form1.le.value=="")
    {alert("请输入格式化后字符串的长
度!");form1.le.focus();return false;}
    if(isNaN(form1.le.value)) {
        alert("您输入的格式化字符串的长度不正
确!");form1.le.focus();return false;
    }
    form1.lastStr.value=formatN0(form1.str.value,form1.le.val
ue);
}
</script>

```

(3) 在页面的合适位路添加“转换”按钮，在该按钮的onClick事件中调用deal()函数将指定的数字格式化为指定长度，代码如下。

```

<input name="Submit" type="button" class="btn_grey"
onClick="deal();" value="转换">

```

举一反三

根据本实例，读者可以：

在实现自动编号时需要使用该函数；

在网页中显示指定长度的日期字符串时也需要使用该函数。

## 实例101 把一个长数字分位显示

本实例是一个字符串处理程序

实例位置：光盘\mingrisoft\03\101

## 实例说明

实际网站开发过程中，很有可能遇到这样的情况：客户要求将一串长数字分位显示，例如将“13630016”显示为“13,630,016”。为此笔者编写了一个自定义函数，用来将输入的数字字符格式化为分位显示的字符串。运行本实例，在“请输入要转换的长数字”文本框中输入要转换的数字后，单击“提交”按钮，将会在转换结果中显示“13,630,016”，如图 3.18所示。



图3.18 把一个长数字分位显示

## 技术要点

实现把一个长数字分位显示的基本思路如下。

(1) 判断输入的字符串是否为数字，如果不是，则显示0，否则执行下面的操作。

(2) 判断字符串中是否包括小数，如果包括小数，则首先对字符串中的小数进行截取（包括小数点），然后再截取整数部分，如果不包括小数，则直接将该字符串作为需要格式化的整数。

(3) 通过for循环（设路步长为3）将整数部分的字符串每3位加一个逗号。

(4) 将转换后的字符串与前面截取的小数部分连接。

在判断字符串中是否包括小数时，需要应用string对象的indexOf()方法，该方法用于检索字符串，其语法格式如下：

```
string.indexOf(substring)
```

或

`string.indexOf(substring, start)`

参数说明如下。

- **substring**: 用于指定要在字符串 `string` 中检索的子串。
- **start**: 可选整数参数, 用于指定在字符串 `string` 中开始检索的位路。它的有效值为 0 到字符串 `string` 的长度-1。如果省略该参数, 默认将从字符串的第一个字符开始检索。

实现过程

(1) 编写把一个长数字分位显示的函数 `convert()`, 该函数只有一个参数 `num`, 用于传递需要转换的数字字符串, 返回值为转换后的字符串。代码如下。

```
<script language="javascript">
function convert(num) {
    var result=0;
    vardec="";
    if (isNaN(num)) {
        result=0;
    }else {
        if (num.length<4) {
            result=num;
        }else {
            pos=num.indexOf(".",1);
            if (pos>0) {
                dec=num.substr(pos); //小数部分的字符串, 包括小
                数点
                res=num.substr(0, pos);
            }else {
                res=num;
            }
        }
    }
}
```

```

    }
    var tempResult="";
    for(i=res.length;i>0;i-=3) {    //将整数部分分位显示
        if(i-3>0) {
            tempResult=","+res.substr(i-3,3)+tempResult;
        }else{
            tempResult=res.substr(0,i)+tempResult;
        }
    }
    result=tempResult+dec;
}
}
return result;
}
</script>

```

(2) 编写JavaScript自定义函数deal(), 用于将转换后的字符串输出到页面的指定位路, 代码如下。

```

<script language="javascript">
function deal() {
    result.innerHTML="&nbsp;转换结果: "+convert(form1.number.value);
}
</script>

```

(3) 在页面添加一个<div>标记, 将其命名为“result”, 用于显示转换后的字符串。关键代码如下。

```

<div id="result">转换结果: </div>

```

(4) 在页面的合适位路添加“转换”按钮，在该按钮的onClick事件中调用deal()函数将长数字分位显示，代码如下。

```
<input name="Submit" type="button" class="go-wenbenkuang" value="转换" onClick="deal()">
```

举一反三

根据本实例，读者可以：

在房地产信息网中的产品报价页可以应用该函数；

应用该方法也可以自行编写将字符串格式化为指定格式的字符函数。

## 实例102 将RGB格式的颜色值转换为十六进制格式

本实例是一个处理颜色值转换的程序

实例位置：光盘\mingrisoft\03\102

实例说明

在网站开发过程中，经常会用到将RGB格式的颜色值转换为十六进制格式的情况，例如，在实现文字动态变色时，就需要使用该方法将RGC值转换为十六进制格式。为此笔者编写了一个将RGB格式的颜色值转换为十六进制格式的函数，并将其制作为一个实例，运行本实例，首先在“R”文本框中输入180，在“G”文本框中输入7，在“B”文本框中输入177，然后单击“转换”按钮，在“格式化后的十六进制的颜色值”文本框中将显示转换后的颜色值“#b407b1”，如图3.19所示。

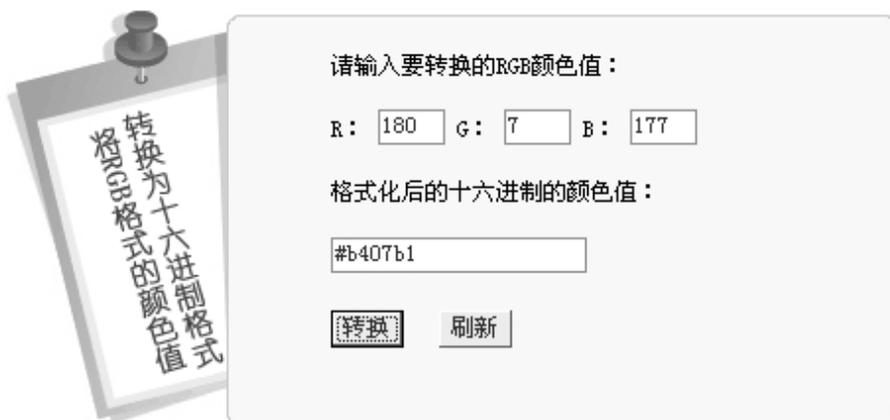


图3.19 将RGB格式的颜色值转换为十六进制格式

### 技术要点

本实例主要应用`parseInt()`方法和`Number`对象的`toString()`方法实现。下面将对`parseInt()`方法和`Number`对象的`toString()`方法进行详细介绍。

#### (1) `parseInt()`方法

`parseInt()`方法用于将非数字的原始值转换成整数，该方法有两种模式，即默认模式和基模式。下面分别介绍。

在默认模式下，`parseInt()`方法首先查看位路0处的字符，判断它是否为有效数字，如不是，将返回`NaN`，不再继续执行检查；否则将查看位路1，进行检查，依次类推，直到发现非有效数字为止，此时`parseInt()`方法将把第一个发现的非有效数字前面的有效字符转换成整数。例如：

```
parseInt("8007wgh"); //返回结果为8007
parseInt("8007wgh666"); //返回结果为8007
parseInt("wgh666"); //返回结果为NaN
parseInt("0xE"); //返回结果为14
parseInt("3.14"); //返回结果为3
```

采用基模式，可以把二进制、八进制、十六进制或其他任何进制的字符串转换成整数。基语法格式如下。

```
parseInt(str, basic)
```

参数说明如下。

- `str`: 表示要转换的原始字符串。
- `basic`: 表示基，用于指定要转换字符串采用的基数，二进制的基为2，八进制的基为8，十进制的基为10，十六进制的基为16。

例如：

```
parseInt("11", 2);    //返回结果为3
parseInt("11", 8);    //返回结果为9
parseInt("11", 10);   //返回结果为11
parseInt("11", 16);   //返回结果为17
```

## (2) Number对象的toString()方法

Number 对象的 `toString()` 方法用于将数值转换为字符串，该方法有两种模式，即默认模式和基模式。下面分别介绍。

采用默认模式，`toString()` 方法只是将指定的字符串转化为相应的十进制的数字值（无论是整数、浮点数还是科学计数法）。

说明：二进制的基为2，八进制的基为8，十进制的基为10，十六进制的基为16。

采用基模式，可以用不同的基数输出数字，例如：

```
var myNum=11;
myNum.toString(2);    //返回结果为1011
myNum.toString(8);    //返回结果为13
myNum.toString(10);   //返回结果为11
myNum.toString(16);   //返回结果为b
```

基模式对于 HTML 中处理数字时非常有用，在 HTML 中采用十六进制表示每种颜色，如 `#FF0000` 表示红色，本实例中应用的就是这种模式。

注意：在本实例中需要调用自定义函数formatNO()将不足两位的R或G或B的值格式化为2位，不足位置补0，formatNO()函数的具体代码请参见实例100。

### 实现过程

(1) 编写将RGB格式的颜色值转换为十六进制格式的JavaScript自定义函数convert()，该函数有3个参数，分别是r（RGB颜色中的红色值）、g（RGB颜色中的绿色值）和b（RGB颜色中的蓝色值），返回值为转换后的十六进制颜色值。代码如下。

```
<script language="javascript">
function convert(r, g, b) {
    if(isNaN(r) || 255-r<0 ) {
        r=0;
    }
    if(isNaN(g) || 255-g<0 ) {
        g=0;
    }
    if(isNaN(b) || 255-b<0 ) {
        b=0;
    }
    varhr=formatNO(parseInt(r).toString(16),2);
    varhg=formatNO(parseInt(g).toString(16),2);
    varhb=formatNO(parseInt(b).toString(16),2);
    var result="#" +hr+hg+hb;
    return result;
}
</script>
```

(2) 编写JavaScript自定义函数check(), 用于验证用户输入信息是否合法, 具体代码如下。

```
<script language="javascript">
function check(obj) {
    varExpression=/^[0-9]+(\d*$)/;
    varobjExp=newRegExp(Expression);
    if(objExp.test(obj.value)==false || 255-obj.value<0 ) {
        alert('要求输入0-255之间的整数!');
        obj.focus();
    }
}
</script>
```

(3) 编写JavaScript自定义函数deal(), 调用convert()函数将转换后的字符串显示在指定文本框中, 具体代码如下。

```
<script language="javascript">
function deal() {
    form1.lastStr.value=convert(form1.str1.value, form1.str2
.value, form1.str3.value);
}
</script>
```

(4) 在页面的合适位路添加“转换”按钮, 在该按钮的 onClick事件中调用 deal()函数将RGB格式的颜色值转换为十六进制格式, 代码如下。

```
<input name="Submit" type="button" class="btn_grey"
onClick="deal();" value="转换">
```

举一反三

根据本实例, 读者可以:

实现彩色渐变文字动画；  
实现表格或页面背景循环变色。

## 实例103 将IP地址转换为对应的数值

本实例是一个字符串处理程序

实例位置：光盘\mingrisoft\03\103

实例说明

在开发动态网站时，有时需要将IP地址转换为对应的数值。为此笔者编写了一个将IP地址转换为对应的数值的函数，并将其制作为一个实例。运行本实例，首先在“请输入IP地址”文本框中输入“192.168.1.66”，单击“转换”按钮，在“对应的数值”文本框中将显示转换后的对应数值“3194548521”，如图3.20所示。



图3.20 将IP地址转换为对应的数值

技术要点

实现本实例，首先需要验证输入的IP地址是否合法，如果合法，再将其转换为对应的数值。验证IP地址是否合法可以通过正则表达式实现。

IP地址使用4位地址定位设备，它采用点分十进制数的格式“xxx.xxx.xxx.xxx”，例如192.168.1.66。因此，验证IP地址是否合

法的正则表达式如下。

```
/(\d+).(\d+).(\d+).(\d+)/
```

将IP地址转换为对应的数值可以应用以下公式实现。

$$P1*255*255*255+P2*255*255+P3*255+P4*1$$

在上面的公式中，P1代表IP地址第1段的数值，P2代表IP地址第2段的数值，P3代表IP地址第3段的数值，P4代表IP地址第4段的数值。

实现过程

(1) 编写将IP地址转换为对应数值的JavaScript自定义函数convert()，该函数只有一个参数ip，用于指定IP地址，返回值为IP地址对应的数值。代码如下。

```
<script language="javascript">
function convert(ip) {
    ip=ip.split(".");
    vip=ip[0]*255*255*255+ip[1]*255*255+ip[2]*255+ip[3]*1;
    return vip;
}
</script>
```

(2) 编写JavaScript自定义函数deal()，用于验证用户输入IP地址是否合法，如果合法，再将其转换为对应的数值，并显示在相应的文本框中，具体代码如下。

```
<script language="javascript">
functiondeal(myform) {
    var ip=form1.ip.value;
    objExp=/(\d+).(\d+).(\d+).(\d+)/;
    if(ip=="") {
        alert("请输入IP地址!");form1.ip.focus();return;
    }
}
```

```

    if(objExp.test(ip)){
        myform.ipNum.value=convert(ip);
    }else{
        alert("你输入的IP地址不合法!");form1.ip.focus();return;
    }
}
</script>

```

(3) 在页面的合适位置添加“转换”按钮，在该按钮的onClick事件中调用deal()函数，将IP地址转换为相对应的数值，代码如下。

```

<input name="Button" type="button" class="btn_grey"
onClick="deal(form1)"
value="转换"/>

```

举一反三

根据本实例，读者可以：

用于限制指定范围内的IP地址投票页面中；

实现获取访问者IP所属地。

## [实例104 从指定URL中提取文件名](#)

本实例是一个对字符串进行操作的程序

实例位置：光盘\mingrisoft\03\104

实例说明

在开发网站时，可能会遇到从指定的URL中提取文件名的情况。为此笔者编写一个从指定URL中提取文件名的函数，并将其制作为一个实例。运行本实例，首先在“请输入URL地址”文本框中输入

“http://www.mingrisoft.com/board/index.asp”，单击“提取”按

钮，在“提取的文件名”文本框中将显示提取出的文件名“index”，如图3.21所示。



图3.21 从指定URL中提取文件名

#### 技术要点

本实例主要通过string对象的replace()方法实现，下面对replace()方法进行详细介绍。

string对象的replace()方法用于替换一个与正则表达式匹配的子串，其语法格式如下。

```
string.replace(regex, replacement)
```

参数说明如下。

- **regex**: 指明要替换的模式正则表达式对象。如果该参数是一个字符串，则将它作为要检索的直接量，而不是先将其转换为正则表达式对象。

- **replacement**: 一个字符串或函数，用于指明替换文本或生成替换文本的函数。该参数的值可以有以下3种情况。

- (1) 如果replacement是一个字符串，那么每个匹配都将由该字符串替换。

- (2) 如果replacement是一个包括\$字符的字符串，则具有特殊意义，具体意义参见表3.3。

表3.3 包括\$字符的字符串的具体意义

字 符	替 换 文 本
\$1、\$2、…、\$99	与 regexp 中的第 1 个到第 99 个子表达式相匹配的文本
\$&	与 regexp 相匹配的子串
\$`	位于匹配子串左侧的文本
`\${`	位于匹配子串右侧的文本
\$\$	直接量\$符号

(3) 如果replacement是一个函数，则每个匹配都调用该函数，并将返回字符串作为替换文本使用。

### 实现过程

(1) 编写从指定URL中提取文件名的JavaScript自定义函数convert()，该函数只有一个参数url，用于指定URL地址，返回值为提取出的文件名。代码如下。

```
<script language="javaScript">
function convert(url) {
    url=url.replace(/(.*/\/*)*([\^.]*)*/ig, "$2");
    return url;
}
</script>
```

(2) 编写JavaScript自定义函数deal()，用于验证用户输入URL地址是否合法。如果合法，再从该URL中提取文件名，并显示在相应的文本框中，具体代码如下。

```
<script language="javascript">
functiondeal(myform) {
    varurl=form1.url.value;
    objExp=/http(s)?:\//([\w-]+\.)+[\w-]+(\//[\w-. \//?
%&=]*)?(\//\w+)*.\w{3}$/;
    if(url=="") {
        alert("请输入URL地址!");form1.url.focus();return;
    }
}
```

```
    }  
    if(objExp.test(url)) {  
        myform.lastUrl.value=convert(url);  
    }else{  
        alert("您输入的URL地址不合法!");form1.url.focus();return;  
    }  
}  
</script>
```

(3) 在页面的合适位路添加“提取”按钮，在该按钮的onClick事件中调用deal()函数从指定URL中提取文件名，代码如下。

```
<input name="Button" type="button" class="btn_grey"  
onClick="deal(form1)"  
value="提取"/>
```

举一反三

根据本实例，读者可以：

用于限制指定范围内的IP地址投票页面中；

实现获取访问者IP所属地。

## 3.3 随机函数

在测试、模拟及游戏等程序中，经常要使用随机数。下面结合几个实例介绍一下随机函数的使用方法。

### 实例105 随机产生指定位数的验证码

本实例是一个应用随机函数的程序

实例位置：光盘\mingrisoft\03\105

实例说明

在Web应用程序中，为防止网站受到“注册机”之类的攻击，通常在网站的注册页面或登录页面中加入随机产生的验证码。为了方便读者，提高开发速度，笔者将实现该功能的代码封装在一个函数中，并提供了用户不修改该函数即可输入指定位数的验证码的功能。运行本实例，在“请输入要产生的随机数的位数”文本框中输入4后，单击“生成”按钮，即可在“产生的验证码：”后面显示产生的随机数，如图3.22所示。

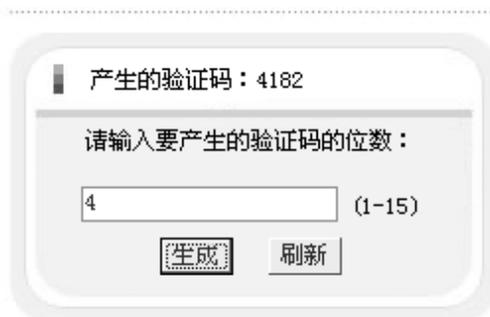


图3.22 随机产生指定位数的验证码

技术要点

在JavaScript中实现随机产生指定位数的验证码主要是通过Math对象的random()方法实现的。

Math对象的random()方法的语法如下。

```
Math.random()
```

功能：返回0.0~1.0之间的一个伪随机数。

为了生成一个0~9的随机整数的字符串，可使用以下公式。

```
(parseInt(Math.random()*10)).toString()
```

实现过程

(1) 编写随机产生指定位数的验证码的函数 checkCode ()，该函数只有一个参数 digit，用于指定生成的验证码的位数，返回值为指定位数的验证码，代码如下。

```
<script language="javascript">
function checkCode(digit) {
    //自动生成验证码
    var result="";
    for(i=0;i<parseInt(digit);i++) {
        result=result+
        (parseInt(Math.random()*10)).toString();
    }
    return result;
}
</script>
```

(2) 编写JavaScript自定义函数deal()，调用convert()函数将转换后的字符串显示在指定位路，具体代码如下。

```
<script language="javascript">
functiondeal() {
```

```
result.innerHTML+="&nbsp;&nbsp;&nbsp;产生的验证  
码："+checkCode(form1.digit.value);  
}  
</script>
```

(3) 在页面添加一个<div>标记，将其命名为“result”，用于显示生成的验证码。关键代码如下。

```
<div id="result">&nbsp;&nbsp;&nbsp;产生的验证码：</div>
```

(4) 在页面的合适位置添加“生成”按钮，在该按钮的onClick事件中调用deal()函数生成验证码，代码如下。

```
<input name="Submit" type="button" class="go-wenbenkuang"  
value="生成"  
onClick="deal()">
```

举一反三

根据本实例，读者可以：

在测试程序中应用随机函数；

在开发的游戏程序中应用随机函数。

## 实例106 生成随机字符串

本实例是一个应用随机函数的程序

实例位置：光盘\mingrisoft\03\106

实例说明

在开发网络应用程序时，经常会遇到由系统自动生成指定位数的随机字符串的情况，例如，生成随机密码或验证码等。为了方便读者，提高开发速度，笔者将实现该功能的代码封装在一个函数中，并提供了用户不修改该函数即可获取指定位数的随机字符串的功能。运行本实例，在“请输入要产生的随机字符的位数”文本框中输入4后，

单击“生成”按钮，即可在“产生的随机字符串”后面显示产生的随机字符串，如图3.23所示。

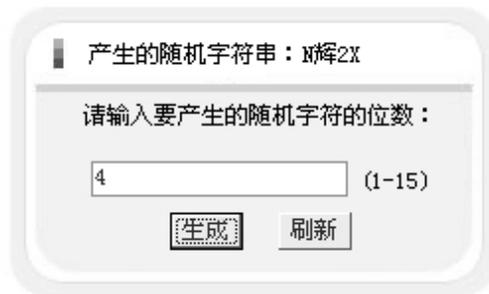


图3.23 生成随机字符串

### 技术要点

实现产生随机字符串首先需要定义一个由逗号分隔的基数字符串，并使用 `string` 对象的 `split()` 方法将其分解为数组，然后根据指定的位数循环生成指定位数的随机字符串。

`string`对象的 `split()` 方法用于将指定的字符串分割成字符串数组。其语法格式如下。

```
string.split(delimiter, limit)
```

参数说明如下。

- `delimiter`: 字符串或正则表达式，用于指定界符，即从什么位路分割字符串。在JavaScript 1.2 和 Jscript 3.0 中实现了将正则表达式作为定界符的用法，但是在JavaScript 1.1 中没有实现该用法。

- `limit`: 可选参数，用于指定返回数组的最大长度，如果没有设置该参数，整个字符串都会被分割，不考虑它的长度。在JavaScript 1.2 和 Jscript 3.0 中实现了该方法，但是在JavaScript 1.1 中没有实现。

例如：

```
"15:31:17".split(":")           //返回结果包括“15”、  
“31”和“17”3个元素的数组
```

```
"|w|g|h".split(":") //返回结果包括“w”、  
“g”和“h”3个元素的数组
```

```
"hello :wgh!".split(/\s*:\s*/) //返回结果为包括  
“hello”、“wgh!”两个元素的数组
```

### 实现过程

(1) 编写随机产生指定位数的验证码的函数checkCode(), 该函数只有一个参数digit, 用于指定生成的随机字符串的位数, 返回值为指定位数的随机字符串, 代码如下。

```
<script language="javascript">  
function checkCode(digit) {  
    var sourceStr="0, 好, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G,  
辉, I, J, K, L, 明, N, P, Q, 日,  
    宋, T, U, V, W, X, Y, Z"; //这里还可以继续添加  
    arrStr=sourceStr.split(",");  
    //自动生成验证码  
    var result="";  
    var suf=0;  
    for(i=0;i<parseInt(digit);i++) {  
        suf=parseInt(Math.random()*arrStr.length);  
        result=result+arrStr[suf];  
    }  
    return result;  
}  
</script>
```

(2) 编写JavaScript自定义函数deal(), 调用convert()函数将转换后的字符串显示在指定位路, 具体代码如下。

```
<script language="javascript">
```

```
function deal() {  
    result.innerHTML="&nbsp;&nbsp;&nbsp;产生的随机字符串：  
串："+checkCode(form1.digit.value);  
}  
</script>
```

(3) 在页面添加一个<div>标记，将其命名为“result”，用于显示生成的随机字符串。关键代码如下。

```
<div id="result">&nbsp;&nbsp;&nbsp;产生的随机字符串：</div>
```

(4) 在页面的合适位路添加“生成”按钮，在该按钮的onClick事件中调用deal()函数生成的随机字符串，代码如下。

```
<input name="Submit" type="button" class="go-wenbenkuang"  
value="生成"  
onClick="deal()">
```

举一反三

根据本实例，读者可以开发：

在测试程序中应用随机函数；

在开发的游戏程序中应用随机函数。

## 3.4 日期与时间

在应用网站中，经常需要动态地显示系统的当前日期及时间，以方便用户。下面通过几个实例详细介绍如何在网站中动态地显示日期与时间。

### 实例107 显示长日期格式的系统日期

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\03\107

实例说明

所谓长日期格式就是以“YYYY 年 MM 月 DD 日 星期 W”格式显示的日期。在网站的适当位路加入此格式的日期不仅可以美化界面，而且可以达到方便用户的目的。例如，在鸿博图书馆管理系统的导航条左侧就加入了以长日期格式显示的系统日期，运行本实例，在网页中将以长日期格式显示当前的系统日期，如图3.24所示。



图3.24 显示长日期格式的系统日期

技术要点

在JavaScript中获取系统日期可以使用Date对象。Date对象是一个有关日期和时间的对象。它具有动态性，即必须使用new运算符创建

一个实例。例如：

```
mydate=new Date();
```

Date对象没有提供直接访问的属性，只具有获取和设置日期和时间的方法。Date对象的方法如表3.4所示。

表3.4 Date对象的方法

方 法	说 明
getFullYear()	返回客户端的系统时间中的年数
getMonth()	返回客户端的系统时间中的月数，月的值是 0~11
getDate()	返回客户端的系统时间中的日数
getDay()	返回客户端的系统时间中的星期
getHours()	返回客户端的系统时间的小时数
getMinutes()	返回客户端的系统时间的分钟数
getSeconds()	返回客户端的系统时间的秒数
getTime()	返回客户端的系统时间的毫秒数
setYear()	设置年
setMonth()	设置月
setDate()	设置日
setHours()	设置小时
setMinutes()	设置分钟
setSeconds()	设置秒数
setTime()	设置毫秒数

### 实现过程

(1) 应用JavaScript编写将系统日期转换为长日期格式的函数clockon()，该函数只有一个参数bgclock，用于指定显示转换后的日期的<div>标记的名称，无返回值，并将其保存在onclock.JS文件中。代码如下。

```
<script language="javascript">  
function clockon(bgclock) {  
    varnow=newDate();  
    varyear=now.getFullYear();  
    varmonth=now.getMonth();
```

```

    vardate=now.getDate();
    varday=now.getDay();
    varweek;
    month=month+1;
    if(month<10)month="0"+month;
    if(date<10)date="0"+date;
    var arr_week=newArray("星期日","星期一","星期二","星期三",
"星期四","星期五","星期六");
    week=arr_week[day];
    var time="";
    time=year+"年"+month+"月"+date+"日"+week;
    if(document.all){
        bgclock.innerHTML+="&nbsp;&nbsp; "+time
    }
    var timer=setTimeout("clockon(bgclock)",200);
}
</script>

```

(2) 在要显示该日期页面的<body>标记的 onLoad事件中调用刚刚编写的 clockon()函数，并将该页面中要显示系统日期的表格单元格的id设路为bgclock，关键代码如下。

```

<body onload="clockon(bgclock)">
<td width="46%" class="word_white"
id="bgclock">&nbsp;&nbsp;&nbsp;</td>

```

举一反三

根据本实例，读者可以：

实现显示短日期格式的系统日期；

实现显示带有问候语的系统日期。

## 实例108 实时显示系统时间

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\03\108

实例说明

在网页中实时显示当前时间，不但可以给网页增色，还可以方便浏览者掌握当前时间，因此很多网页都加入了实时显示系统时间的功能。为了提高网站的开发速度，可以将实时显示系统时间的代码封装在一个单独的函数中，在需要时调用即可。例如，明日科技在线论坛网站中就使用了该函数，运行程序，在页面导航条的下方将实时显示系统时间，如图3.25所示。



图3.25 实时显示系统时间

技术要点

实现实时显示系统时间，大致可分为以下5个步骤。

(1) 创建Date()对象的实例，使用getYear()、getMonth()、getDate()、getDay()、getHours()、getMinutes()、getSeconds()方法获取当前系统时间中有关年、月、日、星期、时、分、秒的数据信息。

(2) 使用getMonth()方法获取的月信息是从0开始计数的，所以要将月份对应的数据自加1。

(3) 使用getDay()方法获取的星期信息是数值型数据，需要使用数组对象Array将其转换为对应的文字信息。

(4) 将所有数据组合并输出到浏览器上，需要使用<div>标记的 innerHTML 方法实现，该方法用于存取标记对象内的子标记和内容，不含标记对象本身。

(5) 使用 setTimeout() 方法使输出函数周期性运行。

### 实现过程

(1) 应用JavaScript编写实时显示系统时间的函数clockon()，该函数只有一个参数bgclock，用于指定显示转换后日期的<div>标记的名称，无返回值，代码如下。

```
<script language="javascript">
function clockon(bgclock) {
    var now=newDate();
    varyear=now.getYear();
    varmonth=now.getMonth();
    vardate=now.getDate();
    varday=ow.getDay();
    varhour=now.getHours();
    varminu=now.getMinutes();
    var sec=now.getSeconds();
    varweek;
    month=month+1;
    if(month<10)month="0"+month;
    if(date<10)date="0"+date;
    if(hour<10)hour="0"+hour;
    if(minu<10)minu="0"+minu;
    if(sec<10) sec="0"+sec;
    var arr_week=newArray("星期日","星期一","星期二","星期三",
"星期四","星期五","星期六");
```

```

week=arr_week[day];
var time="";
time=year+"年"+month+"月"+date+"日 "+week+"
"+hour+":"+minu+":"+sec;
if(document.all){
    bgclock.innerHTML="系统公告: ["+time+"]"
}
var timer=setTimeout("clockon(bgclock)",200);
}
</script>

```

(2) 在要实时显示系统时间页面的<body>标记的onLoad事件中,调用刚刚编写的clockon()函数,并在该页面中适当的位路加入<div>标记,命名为“bgclock”,关键代码如下。

```

<body onLoad="clockon(bgclock)">
<div id="bgclock" class="word_Green"></div>

```

举一反三

根据本实例,读者可以:

实现在适当的网页中显示系统时间。

## 第4章 日期和时间

日期时间显示

日期时间算法

日期时间特效

## 4.1 日期时间显示

在浏览网页时，常常会见到在网页的某一位置上显示着不同样式的系统日期时间，那么这些日期时间基本上都是使用JavaScript编写的。在本节中将使用JavaScript的Date对象来调用一些方法，实现在页面中显示不同类型的日期时间。

### 实例109 在标题栏中显示日期时间

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\109

实例说明

本实例用于实现在标题栏中显示日期时间。运行本实例，在浏览器的标题栏中就会显示当前的系统日期时间。程序运行结果如图4.1所示。

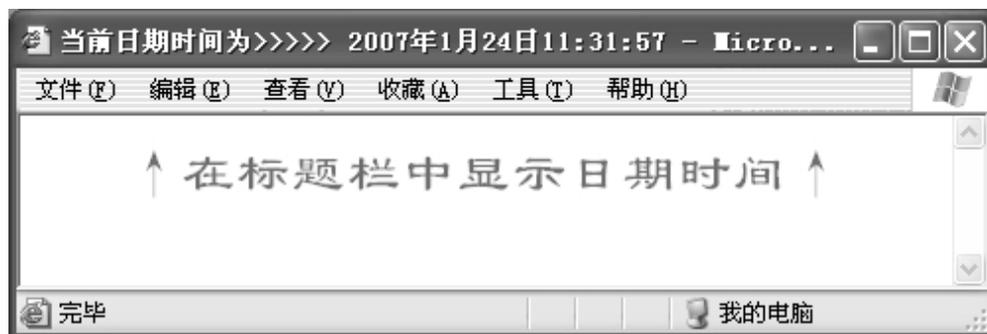


图4.1 在标题栏中显示日期时间

技术要点

本实例主要应用 Date 对象的 getFullYear( )、getMonth( )、getDate( )、getHours( )等方法获取一些时间的信息，然后将这些信息组成一段字符串，并使用document对象调用title属性在浏览器的标

题栏中显示所组成的字符串，最后以每隔1秒的时间间隔，在标题栏中输出当前系统日期时间。

下面介绍Date对象的一些常用方法，如表4.1所示。

表4.1 Date对象的常用方法

方 法	说 明
getFullYear()	返回日期的年份值，是 2 位或 4 位的整数
getFullYear()	返回日期的完整年份值
getMonth()	返回日期的月份值
getHours()	返回时间的小时值
getMinutes()	返回时间分钟值
getSeconds()	返回秒数值
getMilliseconds()	返回毫秒值
getDay()	返回一个处于 0~6 的整数，代表一周中的星期（即 0=星期天；1=星期一；2=星期二）
getTime()	返回从 1970 年 1 月 1 日 00:00:00 起的微秒数
toLocaleString	返回本地系统格式的日期字符串
setYear()	设置年份值

续表

方 法	说 明
setFullYear()	设置完整年份值
setMonth()	设置月份值
setHours()	设置小时值
setMinutes()	设置时间分钟值
setSeconds()	设置秒数值
setMilliseconds()	设置毫秒值
setTime()	设置从 1970 年 1 月 1 日 00:00:00 起的微秒数

在本例中还应用了Window对象的setTimeout()方法，使用此方法可以在指定的时间内重复执行某些操作。

### 实现过程

(1) 编写在标题栏中显示日期时间的JavaScript函数。代码如下。

```
function showtime()//显示当前的日期时间
{
```

```

var now=newDate();
years=now.getFullYear();
month=now.getMonth()+1;
dates=now.getDate();
hours=now.getHours();
Minutes=now.getMinutes();
Seconds=now.getSeconds();
if (hours<10)
    hours="0"+hours;
if(Minutes<10)
    Minutes="0"+Minutes;
if (Seconds<10)
    Seconds="0"+Seconds;
var titletext="当前日期时间为
>>>>> "+years+"年"+month+"月"+dates+"日"+hours+":"+Minutes
+":"+Seconds;
    setTimeout("showtime()",1000);
    ocument.title= titletext;//在页面的标题栏中显示当前日期
时间
}
</script>

```

(2) 添加页面设计代码，当页面载入后调用 showtime() 函数，在标题栏中显示当前系统日期时间。代码如下。

```

<body onload = "showtime();">
<form NAME="form1" METHOD="post" ACTION="">
<center>
    <font SIZE="6px" COLOR="#6666FF" FACE="隶书">

```

```
    ↑ 在标题栏中显示日期时间 ↑  
</font>  
</center>  
</form>  
</body>
```

举一反三

根据本实例，读者可以：

实现在页面中显示当前系统时间；

制作电子时钟。

## 实例110 使用toLocaleString()方法获取本地时间

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\04\110

实例说明

本实例利用Date对象的toLocaleString()方法获得本地时间。使用此方法可以直接获取本地系统日期时间，简化了实例109中调用Date对象的多个方法获取时间的繁琐步骤。程序运行结果如图4.2所示。



当前本地系统时间为：2007年2月9日 19:01:48

图4.2 使用toLocaleString()方法获取本地时间

技术要点

本实例主要应用 Date 对象的 toLocaleString() 方法来获取本地时间。Date 对象的 toLocaleString() 方法用于将Date转换为本地已格式化的字符串，其语法格式如下。

```
date.toLocaleString()
```

说明：toLocaleString()方法可以将日期转换成用本地时区表示的字符串，其日期和时间格式使用地方规则，所以在不同平台上以及不同国家之间，日期和时间的格式将有所不同。

### 实现过程

在<body>区域中编写JavaScript代码，使用Date对象调用toLocaleString()方法获取本地时间，然后利用document对象的write()方法在页面中输出本地时间。

```
<body>
<script language="javascript">
var now=newDate();
hours=now.getHours();
if (hours<25)//在页面中输出当前系统日期时间
{
    document.write("<font size=6pt face= 黑体 color=
ff9900>"+"当前本地系统时间为："+now.toLocaleString()+"
</font>");
}
</script>
</body>
```

### 举一反三

根据本实例，读者可以：

在用户添加注册信息时，通过toLocaleString()方法将本地系统时间加入到注册信息中；

显示用户登录网站的时间。

## [实例111 全中文显示日期](#)

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\111

实例说明

在网页中所看到的日期时间格式都是以数字形式显示的，而很少看到以全中文格式显示日期。本例将通过JavaScript实现此功能，如图4.3所示。



图4.3 全中文显示日期

技术要点

本例主要应用了substring()方法进行取位操作，此方法是string对象的方法。

substring()方法主要用于返回一个字符串的子串。

语法格式如下。

```
string.substring(from, to);
```

参数说明如下。

- from: 一个整数，声明了要抽取的子串的第一个字符在string中的位置。

- to: 一个可选的整数，比要抽取的子串的最后一个字符在string中的位置多1。如果省略了该参数，返回的子串直到字符串的结尾。

实现过程

在<body></body>标记之间编写全中文显示日期的JavaScript代码。

```
<body>
```

```
<script language="JavaScript">
```

```
functionnumberyear(index)//返回当前年份
```

```

{
    var i;
    var numberstring="零一二三四五六七八九";
    text= "";
    for (i=0; i<4; i++)
    {
        point=parseInt(index.toString().substring(i, i+1));
        text= text+numberstring.substring(point, point+1)
    }
    return "公元"+text;
}
function number(index1)//返回当前日期
{
    var numberstring="一二三四五六七八九十";
    if(index1==0)
        {document.write("十")}
    if(index1<10)
    {
        document.write(numberstring.substring(0+(index1-
1), index1));
    }
    else if(index1<20)
    {
        document.write("十"+numberstring.substring(0+(index1-
11), (index1-10)));
    }
    else if(index1<30)

```

```

    {
        document.write("二十"+numberstring.substring(0+(index1-
21), (index1-20)));
    }
    else
    {
        document.write("三十"+numberstring.substring(0+(index1-
31), (index1-30)))
    }
}

var today1=newDate();
varyear= today1.getYear();
varmonth= today1.getMonth()+1;
vardate= today1.getDate();
varday= today1.getDay();
document.write("<b><font size=6pt face= '华文彩
云' Color= #FF0000>" + numberyear(year) + "年");
    number(month);
    document.write("月");
    number(date);
    document.write("日"+"</font></b>");
</script>

```

举一反三

根据本实例，读者可以：

实现个性化页面日期显示样式。

## [实例112 在状态栏中显示日期时间](#)

本实例是一个提高工作效率的程序

实例位置：光盘\mingrisoft\04\112

实例说明

在一些网站中都会利用状态栏显示不同的信息，本例将实现在状态栏中显示日期时间。程序运行结果如图4.4所示。



图4.4 在状态栏中显示日期时间

技术要点

本实例主要应用了window对象的status（状态栏）属性，此属性主要用于在状态栏中显示一条信息。用户可以使用该属性在状态栏中显示一些帮助浏览者的提示信息或固定的自定义信息等。

实现过程

在<head></head>标记内创建在状态栏中显示日期时间的JavaScript函数，代码如下。

```
<script language="javascript">  
functionShowTime ()//获取当前日期时间  
{  
    var today=newDate ();
```

```

var hour= today.getHours();
var minu= today.getMinutes();
var seco= today.getSeconds();
if(hour<10)
hour="0"+hour;
if(minu<10)
minu="0"+minu;
if(seco<10)
seco="0"+seco;
window.status="-----"+today.getFullYear()+"年"+
(today.getMonth()+1)+"月"+today.getDate()+"日"+hour+"时"+mi
nu+"分"+seco+"秒"+"-----";//在状态栏中显示日期时间
window.setTimeout("ShowTime();",1000);
}
ShowTime();
</script>

```

举一反三

根据本实例，读者可以：

通过本例中的方法，在状态栏中显示各式各样的提示信息。

## [实例113 使用数组显示星期](#)

本实例是一个技巧性较强的程序

实例位置：光盘\mingrisoft\04\113

实例说明

为了使整个日期信息显示得更详细一些，一般都会会在日期后面显示一个星期。本实例将通过数组显示星期。

在本实例中首先通过数组对象创建了一个包含7个数组元素的数组，并为每个数组元素赋值，然后通过日期对象获取当前的星期值，再将获取的星期值加入到数组元素中，最后使用window对象的write()方法输出今天是星期几。程序运行结果如图4.5所示。

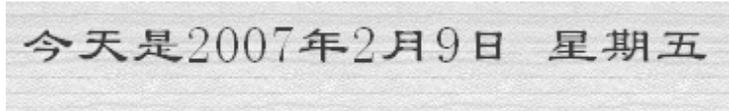


图4.5 使用数组显示星期

### 技术要点

本例中主要应用了数组和日期对象中的getDay()方法。getDay()方法用于返回一个星期值，其值是一个处于0~6之间的整数，代表一周中的某一天（即0是星期天；1是星期一；2是星期二；3是星期三；4是星期四；5是星期五；6是星期六）。

### 实现过程

在<body></body>标记中编写使用数组显示星期的JavaScript函数，代码如下。

```
<script language="javascript">
    var week, today, week_i;
    week=newArray(7);
    week[1]="星期一";
    week[2]="星期二";
    week[3]="星期三";
    week[4]="星期四";
    week[5]="星期五";
    week[6]="星期六";
    week[0]="星期日";
    today=newDate();
    week_i= today.getDay();
```

```
varyear= today.getFullYear();
varmonth= today.getMonth()+1;
vardate= today.getDate();
document.write("<font color=#3333FF size= '6pt' face=
隶书>"+"今天是"+year+"年"+month+"月"+date+"日 "+week
[week_i]+"</font>");
</script>
```

举一反三

根据本实例，读者可以：

实现在页面中显示中文格式的星期；

实现全中文格式显示日期时间。

## 实例114 在表格中显示时间

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\114

实例说明

本实例将介绍如何在表格中显示时间。运行本实例，在页面左边的表格中显示当前日期，右边的表格中显示时间，如图4.6所示。



图4.6 在表格中显示时间

技术要点

本实例首先需要创建两个函数，分别用于显示日期和时间，然后在页面中设计一个 1 行2列的表格，最后调用表格的innerHTML属性，在表格的第一列中显示日期，在第二列中显示时间。

实现过程

(1) 在页面的<body></body>标记之间编写在表格中显示时间的JavaScript代码。

```
<script>
functionShowDate (Elements)//显示日期
{
    var temp;
    vardatetime=newDate();
    var year=datetime.getYear();
    varmonth=datetime.getMonth()+1;
    vardate=datetime.getDate();
    varday=datetime.getDay();
    temp= year+"年"+month+"月"+date+"日 ";
    Elements.innerHTML= temp;
}
functionShowTime (Elements)//显示时间
{
    var temp= "";
    vardatetime=newDate();
    varhour=datetime.getHours();
    varminu=datetime.getMinutes();
    var seco=datetime.getSeconds();
    temp= temp+" "+hour+":"+minu+":"+seco+" ";
    Elements.innerHTML= temp; //这个Elements代表(表格)的id
    window.setTimeout("ShowTime("+Elements.id+")", 1000) //
这里Elements.name或Elements.id不是Elements
}
</script>
```

(2) 在<body></body>标记中添加表格，并在表格中调用JavaScript函数。

```
<table border="1" cellpadding="2" cellspacing="2" >
<tr>
<td id="Time1"><script>ShowDate(Time1);</script></td>
<td width="110" id="Time2"><script>ShowTime(Time2);
</script>
</td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

实现在表格中显示日期时间；

实现在页面的指定位置分别显示日期和时间。

## 实例115 退出页面时显示停留时间

本实例可以统计用户访问网页的时间

实例位置：光盘\mingrisoft\04\115

实例说明

本实例用于实现在退出页面时显示停留时间。当页面打开时程序将会开始计算用户在页面的停留时间，当用户关闭页面后，程序将会弹出信息框显示浏览者在本页中的停留时间，如图4.7所示。



## 图4.7 退出页面时显示停留时间

### 技术要点

本实例通过调用Date对象的一些方法，使变量minutes获得Date对象的当前分钟数，使变量seconds获得Date对象的当前秒数，最后使用alert()方法弹出信息框，显示用户在本页的停留时间。

### 实现过程

(1) 添加退出页面时显示停留时间的JavaScript函数。代码如下。

```
<script language="Javascript">
beginTime=newDate();
function outTime() {
    endTime=newDate();
    minutes= (endTime.getMinutes()-beginTime.getMinutes());
    seconds= (endTime.getSeconds()-beginTime.getSeconds());
    time= (seconds+(minutes *60));
    time= (time);
    alert(' 您在本页共停留了  '+time+"秒钟");
}
</script>
```

(2) 在页面中将body的onUnload事件设置为outTime()函数。代码如下。

```
<body onUnload="outTime();">
<center>

</center>
</body>
```

举一反三

根据本实例，读者可以：

如果用户想让浏览者看到在本站的停留时间，可以使用本实例；  
利用本实例制作一个计时器。

## 实例116 显示用户在页面的停留时间

本实例可以统计用户访问网站的时间

实例位置：光盘\mingrisoft\04\116

实例说明

本实例用于实现显示用户在页面的停留时间。本实例首先在页面中添加了一个 `textarea` 控件，用于显示浏览者在本站的停留时间。在 JavaScript 中声明3个变量，分别用于显示时、分、秒。接着声明一个显示时间的函数，并将用户在页面中的停留时间显示在 `textarea` 控件中。程序运行结果如图4.8所示。

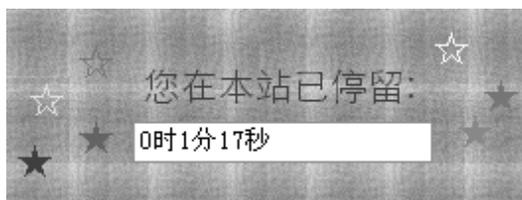


图4.8 显示用户在页面的停留时间

技术要点

在JavaScript所声明的 `interval()` 函数中，应用了 `if` 条件判断语句。首先判断当前秒数是否为60，如果为60则让分数加1，然后使秒数再次归0，接着判断分数是否为60，如果为60则让小时数加1。

实现过程

添加页面设计代码及编写JavaScript自定义函数，代码如下。

```
<body>
```

```
<form name="form1" method="post" action="">
```

```
<center>
<font size="6pt"color="#009933">您在本站已停留: </font>
<textareaname="textarea"></textarea>
</center>
<script language=javascript>
var second=0;
varminute=0;
var hour=0;
window.setTimeout("interval();",1000);
function interval()
{
    second++;
    if(second==60)
    {
        second=0;minute+=1;
    }
    if(minute==60)
    {
        minute=0;hour+=1;
    }
    document.form1.textarea.value=hour+"时"+minute+"分"+second+"秒";
    window.setTimeout("interval();",1000);
}
</script>
</form>
</body>
```

举一反三

根据本实例，读者可以：

在本站中加入一项本站会员在本站共访问的时间；

可以将站内会员的累计访问时间做为用户积分。

## 实例117 显示最后修改时间

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\117

实例说明

本实例将实现显示网页文件最后的修改时间。

本实例首先在页面中添加了一个文本框控件，然后使用JavaScript 创建了一个函数，在函数体中调用document对象的lastModified属性，然后在页面的文本框中显示文件上一次被修改的时间。通过这些方法可以查看网页的更新时间，如图4.9所示。

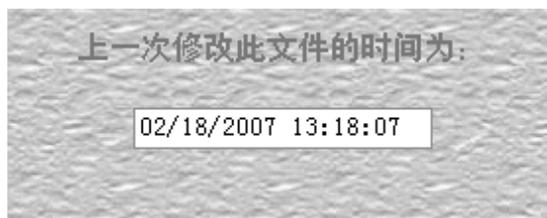


图4.9 显示最后修改时间

技术要点

本实例主要调用了document对象的lastModified属性，此属性主要以字符串的形式返回文档的修改日期时间。document对象还包含许多属性，例如alinkcolor、linkcolor、vlincolor、images、location、title、domain等。

实现过程

(1) 在<head></head>标记之间编写JavaScript函数，代码如下。

```
<head>
<script language="javascript">
    function alterdate()
    {
        document.form1.textfield.value =
document.lastModified;
    }
</script>
</head>
```

(2) 添加页面设计代码，在<body>的onLoad事件中调用alterdate()函数，代码如下。

```
<bodyonLoad="alterdate();">
<form name="form1"method="post"action="">
    <center>
        <p><font color="#FF0000" face="黑体" size="5pt">上一
次修改此文件的时间为： </font></p>
        <p>
            <input type="text"name="textfield">
        </p>
    </center>
</form>
</body>
```

举一反三

根据本实例，读者可以：

查看HTML文件是否被改动过。

## 4.2 日期时间算法

计算某一天是星期几、计算从出生到现在度过的时间、返回两个日期之间的间隔小时等这些都需要通过算法才能够实现，下面将通过一些算法来解决一些与日期时间相关的问题。

### 实例118 计算某一天是星期几

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\04\118

实例说明

在日常生活中，常常要知道某一天是星期几的问题。有时候，可能还需要知道历史上的某一天是星期几。这种情况通常需要查看日历，但是我们总不会随时随地带着日历，更不可能随时随身带着几千年的万年历，而如今已是信息化时代，可以随时在网上找到这样的工具进行查看。那么下面将通过 JavaScript 编写一个计算方法来计算某一天是星期几。程序运行结果如图4.10所示。

输入年：	<input type="text" value="2007"/>
输入月：	<input type="text" value="2"/>
输入日：	<input type="text" value="18"/>
星 期：	<input type="text" value="星期日"/>
<input type="button" value="计算"/>	

图4.10 计算某一天是星期几

技术要点

历史上的某一天是星期几？未来的某一天是星期几？关于这个问题，有很多计算公式（两个通用计算公式和一些分段计算公式），其中最著名的是蔡勒（Zeller）公式。

$$w=y+[y/4]+[c/4]-2c+[26(m+1)/10]+d-1$$

参数说明如下。

- w: 表示星期。
- c: 表示世纪。
- y: 表示后两位年份值（如果2049 则为49）。
- m: 表示月（m 大于等于 3，小于等于 14，即在蔡勒公式中，某年的 1、2 月要看作上一年的13、14月来计算，比如2003年1月1日要看作2002年的13月1日来计算）。
- d: 表示日。

说明：公式中[ ]代表取整，即只要整数部分。c是世纪数减一，y是年份后两位，m是月份，d是天数。

通过这个算法可以算一下自己出生日期那天是星期几。

实现过程

（1）如果用户在文本框中输入不符合条件的字符，则不计算星期。代码如下。

```
function checktext()  
{  
    if((form1.yeartext.value=="")&&  
(form1.monthtext.value=="")&& (form1.datetext.value==""))  
    {  
        alert("请输入相关信息!");  
        form1.yeartext.focus();return;  
    }  
}
```

```

        if((form1.yeartext.value.length !=4 )&&
(form1.monthtext.value.length !=1 )&&
(form1.datetext.value.length !=1 ))
    {
        alert("输入错误，只能输入4位数！");
        form1.yeartext.focus();return;
    }
}

```

(2) 编写取余函数。代码如下。

```

function mod(x, x_div)
{
    for (var i=x; i>=x_div; i-= x_div);
    return i;
}

```

(3) 编写计算星期函数。代码如下。

```

function getday()
{
    var currentyear=parseInt(form1.yeartext.value, 10);
    var currentmonth=parseInt(form1.monthtext.value, 10);
    var currentday=parseInt(form1.datetext.value, 10);
    var sig_val;
    varbegindate=newArray(0, 3, 3, 6, 1, 4, 6, 2, 5, 0, 3, 5);
    var rundate=newArray(-1, 2, 2, 5, 0, 3, 5, 1, 4, -1, 2, 4);
    varmontharray=newArray("星期日", "星期一", "星期二", "星期三", "星期四", "星期五", "星期六");
    sig_val=begindate[currentmonth-1];
}

```

```

        varval1=mod((currentyear+parseInt(currentyear/4)+current
tday+sig_val)-2,7);
        if ((currentyear%4==0&& currentyear%100!=0) ||
(currentyear%400==0))
        {
            sig_val=rundate[currentmonth-1];
            val1=mod((currentyear+parseInt(currentyear/4)+current
day+sig_val)-2,7);
            form1.resulttext.value=montharray[val1];
        }
        else
        {
            form1.resulttext.value=montharray[val1];
        }
    }

```

(4) 在<body></body>标记中添加页面设计代码，单击按钮调用getday()函数和checktext()函数。代码如下。

```

<body>
<center>
<form name="form1" method="post" action="">
    <table width="308" border="1" cellpadding="3"
cellspacing="1" bordercolor="#33CCFF" bgcolor="#CCFFFF">
        <tr bgcolor="#FFFFFF">
            <td align="center" class="style5">输入年: </td>
            <td width="170"><input name="yeartext" type="text"
id="yeartext" value="2008"></td>
        </tr>

```



</body>

举一反三

根据本实例，读者可以：

计算出出生日期是星期几。

## 实例119 判断指定年份是否为闰年

本实例是一个应用较为广泛的程序

实例位置：光盘\mingrisoft\04\119

实例说明

闰年就是能被4整除但不能被 100整除的数。本实例使用日期对象调用 `getFullYear()` 方法获取当前年份，然后通过三目运行符来返回用户输入的年份是否为闰年，如图 4.11所示。



请输入一个四位数的年份：	<input type="text" value="2007"/>	计算
2007年不是闰年		

图4.11 判断指定年份是否为闰年

技术要点

本例应用了三目运行符，它在JavaScript中是一种特殊的运行符，语法格式如下。

```
expression ? true_result:false_result;
```

如果expression 为真，则表达式的值为true\_result 子表达式的值，否则为false\_result 子表达式的值。

例如：

```
var x = 5;
```

```
var y = 10;
```

```
result = (x>y)?'x的值大':'y的值大';
```

```
document.write(result);
```

则返回结果为：y的值大。

实现过程

编写判断指定年份是否为闰年的JavaScript代码。

```
<script language="javascript">
function leapyear()
{
    var now=newDate();
    varyears=now.getFullYear(document.form1.textfield.value
);
    var str;
    varyears=document.form1.textfield.value;
    var result=(years%4==0&&years%100!=0) || (years%400==0)?
years+"是闰年":years+"不是闰年";
    document.form1.textfield2.value= result;
}
</script>
```

举一反三

根据本实例，读者可以：

通过闰月判断闰年。

## 实例120 计算从出生到现在度过的时间

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\120

实例说明

当页面显示时将弹出3次输入对话框，分别供用户输入出生年份、出生月份和出生日期，当输入完毕之后，将会在页面的文本框中显示

用户到当前为止已度过的时间。例如在输入对话框中输入1980、5、30，则计算结果为9751天7小时13分23秒，如图4.12、图4.13所示。



图4.12 在输入对话框中输入出生年月日

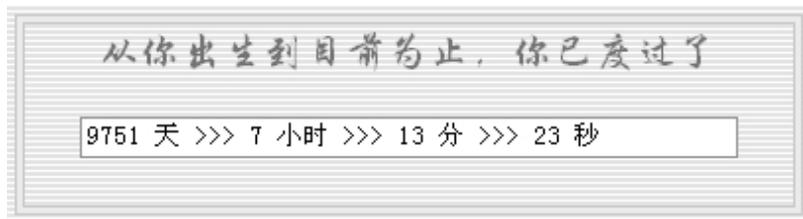


图4.13 计算从出生在现在度过的时间

#### 技术要点

本实例中应用了输入对话框，此对话框不但可以显示信息，而且还提供一个文本框供用户输入信息，用户单击此对话框中的“确定”按钮，则 `prompt()` 方法返回用户在文本框中输入的内容。`prompt()` 方法中包括两个参数，第一个参数是输入提示信息，第二个参数是显示文本框中的默认值，如果用户不想显示默认值，则将其值设置为空即可。

例如：

```
prompt('请输入你的性别：', '男');
```

#### 实现过程

(1) 编写计算从出生到现在度过时间的JavaScript代码。

```
<script language="javascript">  
    varbiryear=prompt('请输入你的出生年份', '1980');  
    varbirmonth=prompt('请输入你的出生月份', '5');  
    varbirday=prompt('请输入你的出生日', '30');
```

```

    varyear=newDate();
    vargetyear= year.getFullYear();
    TMonth=newArray('Jan','Feb','Mar','Apr','May','June','J
uly','Aug','Sep','Oct','Nov','Dec');
    nowMonth=birmonth-1;
    bmonth=TMonth[nowMonth];
    var timerid;
    var timerRunning= false;
    var today=newDate();
    var startday=newDate();
    var enday=newDate();
    varminPerDay=0;
    var secsLeft=0;
    var secsRound=0;
    var secsRemain=0;
    varminLeft=0;
    varminRound=0;
    varminRemain=0;
    var timeRemain=0;
    function showtime()//计算从出生到现在度过的时间
    {
        startday=newDate(""+bmonth+" "+birday+",
"+biryear+"00:00EDT");
        startday.setYear(""+biryear+"");
        today=newDate();
        secsPerDay=1000 ;
        minPerDay = 60 * 1000 ;
    }

```

```

hoursPerDay=60 *60 *1000;
PerDay=24 *60 *60 *1000;
secsLeft= (today.getTime()-
startday.getTime())/minPerDay;
secsRound=Math.round(secsLeft);
secsRemain= secsLeft-secsRound;
secsRemain= (secsRemain<0)?secsRemain=60-((secsRound-
secsLeft)*60) :secsRemain= (secsLeft-secsRound)*60;
secsRemain=Math.round(secsRemain);
minLeft= ((today.getTime()-
startday.getTime())/hoursPerDay);
minRound=Math.round(minLeft);
minRemain=minLeft-minRound;
minRemain= (minRemain<0)?minRemain=60-((minRound-
minLeft)*60) :minRemain= ((minLeft-minRound)*60);
minRemain=Math.round(minRemain-0.495);
hoursLeft= ((today.getTime()-
startday.getTime())/PerDay);
hoursRound=Math.round(hoursLeft);
hoursRemain=hoursLeft-hoursRound;
hoursRemain= (hoursRemain<0) ?hoursRemain=24-
((hoursRound-hoursLeft) *24) :hoursRemain= ((hoursLeft-
hoursRound) *24);
hoursRemain=Math.round(hoursRemain-0.5);
daysLeft= ((today.getTime()-
startday.getTime())/PerDay);
daysLeft= (daysLeft-0.5);

```

```

    daysRound=Math. round(daysLeft);
    daysRemain=daysRound;
    day_rem= "天 >>> "
    hour_rem= "小时 >>> "
    min_rem= "分 >>> "
    sec_rem= "秒"
    timeRemain=daysRemain+day_rem+hoursRemain+hour_rem+mi
nRemain+min_rem+secsRemain+sec_rem;
    document.up.face.value= timeRemain;
    timerid= setTimeout("showtime()",1000);
    timerRunning= true;
}
</script>

```

(2) 在<body></body>标记之间添加页面设计代码，并使<body>的onload事件调用showtime()函数。

```

<bodyonload=showtime()>
<tablewidth="400"height="50"border="1"align="center"cellp
adding="4"cellspacing="3"bordercolor="#CCCCCC"bgcolor="#CCFF0
0">
<tr>
<td align="center" valign="middle" bgcolor="#FFFFFF">
<FORMname=up onsubmit=1>
<font size="5pt"color="#FF6600" face="黑体">从你出
生到目前为止，你已度过了</font>
<p>
<INPUT size=46 value="计算出你的精彩人
生!"name=face>

```

```
</FORM>
</td>
</tr>
</table>
</body>
```

举一反三

根据本实例，读者可以：

计算从出生到现在一共过几个生日。

## 实例121 返回两个日期之间的间隔小时

本实例是一个计算小时数的程序

实例位置：光盘\mingrisoft\04\121

实例说明

本实例将实现返回两个日期之间的间隔小时。首先在页面中添加了3个文本框，分别用于输入开始日期和结束日期以及显示间隔小时，当用户单击“计算”按钮时，程序将会计算开始日期和结束日期之间的小时差，并将计算结果显示在间隔小时文本框中，如图4.14所示。

输入开始日期：	<input type="text" value="1/1/2006"/>
输入结束日期：	<input type="text" value="1/1/2007"/>
间隔小时：	<input type="text" value="8760"/>

图4.14 返回两个日期之间的间隔小时

技术要点

本实例通过Math对象的abs()方法和round()方法，将获得的数值进行舍入计算后使用abs求绝对值。

abs用于计算一个数的绝对值。

例如：Math.abs(-1.5)

返回值为：1.5。

round方法用于将一个数舍入到最接近的整数。

例如：Math.round(2.5)

则返回值为：3。

实现过程

(1) 在JavaScript所声明的函数中，使用Date对象获取开始时间和结束时间，然后将得到的差除以小时，最后将结果显示在文本框中。代码如下。

```
<script language="javascript">
function currentdate(begindate, enddate)//计算两个日期之间的间隔小时
{
    var begin=newDate(begindate);
    var end=newDate(enddate);
    var resultdate=begin.getTime()-end.getTime();
    varday=Math.abs(Math.round(resultdate/(1000 *60 *60)));//计算间隔小时
    document.form3.resulttext.value=day;
}
</script>
```

(2) 在<body></body>标记中添加页面设计代码，单击“计算”按钮时调用currentdate()函数。代码如下。

```
<body>
<center>
```

```

<tablewidth="285"border="1"cellpadding="4"cellspacing="
3"bordercolor="#9999FF">
  <tr>
    <tdwidth="84"align="left"valign="middle"nowrapborde
rcolor="#CC0000"><span class="style5">输入开始日期:
  </span></td>
    <formname="form1"method="post"action="">
  <tdwidth="170"align="center"valign="middle"
bordercolor="#CC0000">
    <inputname="begindatetext"
type="text"value="1/1/2006">
    </td></form>
  </tr>
  <tr>
    <td
align="left"valign="middle"nowrapbordercolor="#CC0000">
  <span class="style5">输入结束日期: </span></td>
    <formname="form2"method="post"action=""><td
align="center"valign="middle"bordercolor="#CC0000">
    <inputname="enddatetext"
type="text"value="1/1/2007">
    </td></form>
  </tr>
  <tr>
    <td
align="left"valign="middle"nowrapbordercolor="#CC0000">
  <span class="style5">间隔小时: </span></td>

```

```
<formname="form3"method="post"action=""><td
align="center"valign="middle"bordercolor="#CC0000">
    <input type="text"name="resulttext">
</td></form>
</tr>
</table>
</center>
```

举一反三

根据本实例，读者可以：

实现返回两个日期之间的相隔天数；

实现返回两个日期之间的相隔分钟。

## 实例122 倒计时

本实例是一个应用较为广泛的程序

实例位置：光盘\mingrisoft\04\122

实例说明

本实例中所应用的算法与实例 121 基本相同，本实例是求得两个日期之间的相隔天数，然后对相隔天数进行判断，并以不同的提示信息显示在页面中。本实例中将以2008年高考时间作为倒计时，程序每次运行时，都会在页面中显示距2008年高考还有多少天的提示信息，如图4.15所示。



距2008年高考时间还有446天！

图4.15 倒计时

技术要点

本实例中调用了 Math 对象的 floor() 方法，此方法将一个浮点值下舍入为最接近的整数。Math.floor() 执行的操作不同于 Math.round()，它总是进行下舍入，而不是上舍入或下舍入到最接近的整数。

例如：

```
Math.floor(1.66)//返回结果为：1.0
```

```
Math.floor(1.01)//返回结果为：1.0
```

实现过程

(1) 编写用于实现倒计时的JavaScript代码。

```
<script language="javascript">
function countdown(title,Intime)
{
var online=newDate(Intime);
var now=newDate();
var leave= online.getTime()-now.getTime();
var day=Math.floor(leave/(1000 *60*60 *24))+1;//获得相隔
天数
if (day>1)
{
if(document.all)
{
tools.innerHTML="距"+title+"还有"+day+"天！";
}
}
else{
if (day==1)//如果相隔天数只剩1天，则提示"明天就是2008年
高考啦，希望你能考个好成绩！"
```

```
{
  if(document.all)
  {
    tools.innerHTML="明天就是"+title+"啦，希望你能考个
好成绩！";
  }
}
else{
  if (day==0)//如果相隔天数等于指定日期，则提示"今天就是
2008年高考，不要太担心，只要发挥出自己的正常水平就可以，加
油^x^"
  {
    tools.innerHTML="今天就是2008年高考，不要太担心，只要
发挥出自己的正常水平就可以，加油^x^";
  }
  else{
    if(document.all)//如果当前日期超过指定日期，则提示"已
经过了，你考得怎么样呢！"
    {
      tools.innerHTML=title+"已经过了，你考得怎么样呢！";
    }
  }
}
}
}
</script>
```

(2) 在<body></body>标记之间调用JavaScript函数，代码如下。

```
<body>
<center>
  <font size="5pt"color="#FF0000" style="font-family:'黑
体'">
  <div id= "tools">
  <script language="javascript">
    countdown("2007年高考时间","6/8/2007");
  </script>
  </div>
  </font>
</center>
</body>
```

举一反三

根据本实例，读者可以：

在页面的指定位置显示一个节日倒计时；

生日提醒器。

## 实例123 访问时间限制

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\123

实例说明

当在某个网站下载资料或浏览信息时，因为不是该网站的会员有时会出现浏览页面限制的情况，这是因为该网站为了增加网站用户量

和防止非会员用户在该网站中获取信息而采用的一种手段。本实例将实现这一功能，如图4.16所示。

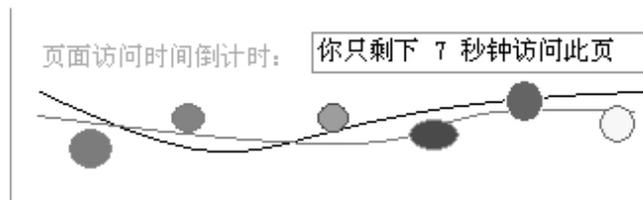


图4.16 访问时间限制

### 技术要点

本实例中使用了 String 对象的 split() 方法，它用于把字符串分割成一个字符串数组，在指定的分界字符处或正则表达式处执行分割。

### 实现过程

(1) 当用户打开页面时，在页面的文本框中显示页面倒计时访问时间，当倒计时为0时，页面将会跳转到一个新的页面中。代码如下。

```
<script language="JavaScript">
    var countdown= "0:10";
    varparselimit= countdown.split(":");
    parselimit=parselimit[0] *60+parselimit[1]*1;
    functionbegintimer()
    {
    if (parselimit==1)
        window.location= "www.mingrisoft.com";
    else
    {
        parselimit-=1;
        curmin=Math.floor(parselimit/60);
        cursec=parselimit;
        curtime= "你只剩下 "+cursec+"秒钟访问此页";
```

```
        setTimeout("begintimer()",1000);
        document.lefttime.left.value= curtime;
    }
}
</script>
```

(2) 添加页面设计代码，使<body>的onload事件调用 begintimer() 函数。代码如下。

```
<bodyonLoad="begintimer();">
<formNAME="lefttime"METHOD="post"ACTION="">
    <center>
        <span class="style2">页面访问时间倒计时: </span>
        <inputNAME="left"TYPE="TEXT" ID="left"SIZE="23">
    </center>
</form>
</body>
```

举一反三

根据本实例，读者可以：

当用户在没有登录的情况下访问本页时，可以使用本实例，提示用户在指定的时间内必须登录后才能够继续浏览本页；制作倒计时。

## 实例124 计步器

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\04\124

实例说明

本实例是一个计步器的示例，当用户单击“开始”按钮时，程序将会开始计数，同时两个文本框中将显示秒数和毫秒数。当用户单击“停止”按钮时，程序将会停止计数并将停止时的秒数和毫秒数显示在文本框中，如图4.17所示。



图4.17 计步器

### 技术要点

本例中应用了window对象的clearTimeout()方法，此方法用于终止setTimeout()执行的代码块。用户单击“停止”按钮时，程序将会调用clearTimeout()方法来停止setTimeout()所执行的操作。一般情况下setTimeout()方法和clearTimeout()方法并用。

### 实现过程

(1) 创建一个Date对象，通过调用其中的方法来获得秒数和毫秒数，代码如下。

```
function getdate()
{
    var now=newDate();
    var se=Math.floor(now.getTime()/1000)-
Math.floor(form1.hideform.value/1000);
    var mse=now.getTime()-form1.hideform.value;
    form1.se.value= se;//显示秒
    form1.mse.value=mse%1000;//显示毫秒
    time=window.setTimeout('getdate()',100);
}
```

(2) 使隐藏表单的值等于当前时间，代码如下。

```
function getdateq()
```

```

{
    varnow=newDate();
    form1.hideform.value=now.getTime();
    getdate();
}

```

(3) 当单击“停止”按钮时程序将会停止计数，代码如下。

```

function sstop()
{
    window.clearTimeout(time);
}

```

(4) 添加页面设计代码。单击“开始”按钮调用getdateq()函数，单击“停止”按钮调用sstop()函数，代码如下。

```

<body>
<formNAME="form1"METHOD="post"ACTION="">
<center>
    <spanCLASS="style1">秒</span>
    <inputNAME="se"TYPE="text" ID="se"SIZE="10">
    <spanCLASS="style1">毫秒</span>
    <inputNAME="mse"TYPE="text" ID="mse"SIZE="10">
    <inputTYPE="BUTTON"NAME="Submit"VALUE="开
始"onClick="getdateq()">&nbsp;
        <inputTYPE="BUTTON"NAME="Submit2"VALUE="停
止"onClick="sstop()">
    <inputNAME="hideform"TYPE="hidden" ID="hideform">
</center>
</form>
</body>

```

举一反三

根据本实例，读者可以：

制作秒表。

## 4.3 日期时间特效

在一些页面比较丰富的网站中，网站制作者为了突出页面的个性化，采用了一些特效技术来优化日期时间的显示风格，以增加页面的浏览效果。下面将使用 JavaScript制作一些具有个性化的日期时间显示特效，读者可以引用本节中的任意一个实例来美化网站中的日期时间。

### 实例125 分时间候

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\125

实例说明

本实例通过Date对象获得当前时间的小时，然后对小时进行判断，并在页面中输出指定的字符串，例如当前时间在8点到11点之间，则在页面中输出“上午好”。通过本实例用户可以在不同的时间内输出自定义的信息。程序运行结果如图4.18所示。



图4.18 分时间候

技术要点

本实例中主要的设计思路是通过Date对象的getHours()方法来获得当前时间的小时，然后判断当前的小时是否在指定的时间内，如果符合则在页面中显示指定的字符串。

## 实现过程

(1) 根据当前时间筛选出符合条件的记录，然后在页面中输出指定的字符串。代码如下。

```
<script language="javascript">
functionShowTime ()
{
    var temp= "";
    vardatetime=newDate ();
    varhour=datetime.getHours ();
    varminu=datetime.getMinutes ();
    var seco=datetime.getSeconds ();
    temp= temp+" "+hour+":"+minu+":"+seco+" ";
    if(hour<10)
    {
        hour= "0"+hour;
    }
    if(minu<10)
    {
        minu= "0"+minu;
    }
    if(seco<10)
    {
        seco= "0"+seco;
    }
    if(hour>=0&&hour<5)
    {
        temp= temp+"凌晨好";
    }
}
```

```

    }
    if(hour>=5 &&hour<8)
    {
        temp= temp+"早上好";
    }
    if(hour>=8 &&hour<11)
    {
        temp= temp+"上午好";
    }
    if(hour>=11 &&hour<13)
    {
        temp= temp+"中午好";
    }
    if(hour>=13 &&hour<17)
    {
        temp= temp+"下午好";
    }
    if(hour>=17 &&hour<24)
    {
        temp= temp+"晚上好";
    }
    window.setTimeout(temp, 1000);    //每1秒执行一次
    document.write("<font color=#CC00FF>" + temp + "</font>");
}
</script>

```

(2) 当页面加载时调用ShowTime()函数。

```
<body onLoad="ShowTime();">
```

</body>

举一反三

根据本实例，读者可以：

实现在日期时间中加入分时间候；

实现在不同的节日中显示不同的问候语。

## 实例126 节日提示

本实例可以显示节日提示

实例位置：光盘\mingrisoft\04\126

实例说明

在开发网络程序时，需要对特殊的日期进行提示，如一些法定节假日等，本实例将使用 JavaScript 实现此功能。运行结果如图4.19所示。



图4.19 节日提示

技术要点

实现特殊日期提示，大致可分为以下步骤。

(1) 创建 Date() 对象的实例，使用 getYear()、getMonth()，getDate()、getDay() 方法获取当前系统时间中有关年、月、日、星期的数据信息。

(2) 使用 getMonth() 方法获取月信息是从0开始计数的，所以要将月份对应的数据自动加1。

(3) 使用 getDay() 方法获取的星期信息是数值型数据，需要使用数据对象 Array 将其转换为对应的文字信息。

(4) 将所有的数据组合并输出到浏览器上，需要使用<div>标记的innerHTML方法实现。

## 实现过程

(1) 应用JavaScript编写特殊日期提示的函数countdown(), 该函数用于指定显示该特殊日期的<div>标记。特殊日期提示自定义函数的代码如下。

```
<SCRIPT language="javascript">
<!--
function countdown() {
calendar=newDate();
day= calendar.getDay();
month= calendar.getMonth();
date= calendar.getDate();
year= calendar.getYear();
if (year<100)year=1900+year;
cent=parseInt(year/100);
g= year%19;
k=parseInt((cent-17)/25);
i= (cent-parseInt(cent/4)-parseInt((cent-
k)/3)+19*g+15)%30;
i= i-parseInt(i/28)*(1-
parseInt(i/28)*parseInt(29/(i+1))*parseInt((21-g)/11));
j= (year+parseInt(year/4)+i+2-cent+parseInt(cent/4))%7;
l= i-j;
emonth=3+parseInt((1+40)/44);
edate= l+28-31*parseInt((emonth/4));
emonth--;
var dayname=newArray ("星期日", "星期一", "星期二", "星期三", "星期四", "星期五", "星期六");
```

```

    var monthname=
        newArray ("1月","2月","3月","4月","5月","6月","7月","8
月","9月","10月","11月","12月" );
    var time="<font style=font-
size:9pt>"+year+"年"+monthname[month]+date+"日"+dayname[day]+
" "+</span></font>";
    var holiday="";
    if ((month==0)&& (date==1))holiday="<font style=font-
size:9pt color=red>元旦";
    if ((month==4)&& (date==1))holiday="<font style=font-
size:9pt color=red>国际劳动节";
    if ((month==4)&& (date==4))holiday="<font style=font-
size:9pt color=red>青年节";
    if ((month==5)&& (date==1))holiday="<font style=font-
size:9pt color=red>国际儿童节";
    if ((month==9)&& (date==1))holiday="<font style=font-
size:9pt color=red>国庆节";
    if ((month==11)&& (date==25))holiday="<font style=font-
size:9pt color=red>圣诞节";
    time=time+holiday+"<br></span></font>";
    clock.innerHTML=time;
}
//-->
</SCRIPT>

```

(2) 在要实时显示特殊日期时间的页面的<body>标记的 onLoad 事件中, 调用刚刚编写的countdown() 函数, 并在该页面中适当的位置加入<div>标记, 代码如下。

```
<body onLoad="countdown()">
<td width="219" height="27" align="center"
background="images/1.JPG"><div id="clock">
</div></td>
```

举一反三

根据本实例，读者可以：

开发会议到时提醒的程序；

开发所有节日提示的功能。

## 实例127 倒影时钟

本实例是一个技巧性较强的程序

实例位置：光盘\mingrisoft\04\127

实例说明

在一些网站中，有时会看到显示着不同风格的日期时间，其中就包括倒影样式的日期时间，本实例将使用 JavaScript 来实现此功能。制作倒影时钟一般都是通过两个 div 层制作，第一层以正常的效果显示，而第二层则使用滤镜技术，将该层设置为水平翻转效果，这样就会形成倒影时钟的效果，如图4.20所示。



图4.20 倒影时钟

技术要点

本例中主要应用了 CSS 样式中的滤镜技术，滤镜是 CSS 样式中的一种扩充，使用滤镜可以在图片或文本容器中实现阴影、模糊、水平或垂直翻转、透明、波纹等特殊效果。

滤镜通过Filter设置滤镜参数，语法格式如下。

Filter: 滤镜名称 (参数)

其中的参数用于指定滤镜的显示效果。例如将图片设置为水平翻转效果显示,代码如下。

```

```

有关滤镜中的其他参数说明如表4.2所示。

表4.2 滤镜效果参数对照表

参 数	说 明
alpha	设置滤镜的透明度。语法格式如下。 <code>filter:alpha(opacity=0..100, finishOpacity=0..100, style=0..3, startX=0..100, startY=0..100, finishX=0..100, finishY=0..100)</code> 参数说明如下。 <ul style="list-style-type: none"><li>● opacity: 表示透明度</li><li>● finishOpacity: 设置渐变透明度效果的结束透明度</li><li>● style: 表示透明区域的形状。0表示统一形状;1表示线形;2表示放射状;3表示长方形</li><li>● startX 和 startY: 表示透明度的开始坐标</li><li>● finishX 和 finishY: 表示透明度的结束坐标</li></ul>
blendTrans	设置滤镜的淡入或淡出效果。语法格式如下。 <code>filter:blendTrans(duration=time)</code> 参数说明如下。 <ul style="list-style-type: none"><li>● duration: 表示效果持续的时间</li></ul>
blur	设置滤镜的模糊效果。语法格式如下。 <code>filter:blur(add=true false,direction=value,strength=value)</code> 参数说明如下。 <ul style="list-style-type: none"><li>● add: 确定图片是否为模糊效果</li><li>● direction: 设置模糊效果的方向,以度数为单位</li><li>● strength: 设置模糊效果的像素数</li></ul>
chroma	设置指定颜色为透明状态。语法格式如下。 <code>filter:chroma(color=value)</code> 参数说明如下。 <ul style="list-style-type: none"><li>● color: 指定颜色的颜色值</li></ul>
dropShadow	设置滤镜的阴影效果。语法格式如下。 <code>filter:dropShadow(color=value, offX=value, offY=value, positive=true false)</code> 参数说明如下。 <ul style="list-style-type: none"><li>● color: 表示阴影的颜色</li><li>● offX 和 offY: 表示阴影的偏移量,取值为像素数</li><li>● positive: 设置其值为 True 表示为透明的像素建立投影,设置为 False 表示不为透明的像素建立投影</li></ul>
flipH	设置滤镜为水平翻转效果。语法格式如下。 <code>filter:flipH</code>

续表

参 数	说 明
flipV	设置滤镜为垂直翻转效果。语法格式如下。 filter:flipV
glow	设置滤镜为发光效果。语法格式如下。 filter:glow(color=value, strength=value) 参数说明如下。 <ul style="list-style-type: none"> <li>● color: 用于设置滤镜发光的颜色</li> <li>● strength: 用于设置发光的亮度。取值为 0~255</li> </ul>
gray	可以去除可视对象的颜色信息, 使其变为灰度显示。语法格式如下。 filter:gray
invert	反转可视对象的色调和亮度, 创建底片效果。语法格式如下。 filter:invert
Light	通过使用滤镜的各种方法模拟光源在可视对象上的投影。语法格式如下。 filter:Light
Mask	设置透明膜效果。其语法格式如下。 filter:Mask(color=value) 参数说明如下。 <ul style="list-style-type: none"> <li>● color: 表示透明膜的颜色</li> </ul>
RevealTrans	设置滤镜转换效果, 显示或隐藏可视对象, 语法格式如下。 filter:RevealTrans(duration=time, transition=value) 参数说明如下。 <ul style="list-style-type: none"> <li>● duration: 表示转换效果持续的时间</li> <li>● transition: 表示转换的类型, 取值为 0~23</li> </ul>
shadow	设置滤镜为立体式阴影效果。语法格式如下。 filter:shadow(color=value, direction=value) 参数说明如下。 <ul style="list-style-type: none"> <li>● color: 表示阴影颜色</li> <li>● direction: 表示阴影的方向</li> </ul>
wave	设置滤镜的波形效果。语法格式如下。 filter:wave(add=true/false, freq=value, lightStrength=0~100, phase=0~100, strength=value) 参数说明如下。 <ul style="list-style-type: none"> <li>● add: 表示是否按正弦波形显示</li> <li>● freq: 设置波形的频率</li> <li>● lightStrength: 设置波形的光影效果</li> <li>● phase: 设置波形开始时的偏移量</li> <li>● strength: 设置波形的振幅</li> </ul>
xray	设置 X 光效果。语法格式如下。 filter:xray

## 实现过程

(1) 编写在层中显示时间的JavaScript代码。

```

<script language="javascript">
var timeout;
function settimes()
{
    varnow=newDate();
    hours=now.getHours();
    Minutes=now.getMinutes();
    Seconds=now.getSeconds();
    if (hours<10)
        hours="0"+hours;
    if(Minutes<10)
        Minutes="0"+Minutes;
    if (Seconds<10)
        Seconds="0"+Seconds;
    //使两个层全部显示时间
    divtime1.innerHTML=hours+":"+Minutes+":"+Seconds;
    divtime2.innerHTML=hours+":"+Minutes+":"+Seconds;
    timeout=setTimeout('settimes()',1000);
}
</script>

```

(2) 在页面中添加两个div层，并将divtime2层设置为滤镜效果。代码如下。

```

<bodyonLoad= "settimes();">
    <font color= "#FF0066" size= "8pt">
        <DIV class= time id=divtime1
            style= "HEIGHT:10px;POSITION:
                absolute;TOP:10px;WIDTH:10px"></DIV>

```

```
<DIV class= time id=divtime2
    style= "FILTER: flipv()
alpha(opacity=25);TOP:30px;FONT-STYLE: italic;POSITION:
absolute"></DIV>
</font>
</body>A
举一反三
根据本实例，读者可以：
制作倒影的图片；
制作倒影的文字。
```

## 实例128 带阴影的时钟

本实例使时钟呈现阴影样式

实例位置：光盘\mingrisoft\04\128

实例说明

本例主要应用了两个 div 层来实现带阴影的时钟，首先使两个层全部显示时间，然后设置第二个层的位置向右下偏移，当页面显示后，就会形成带阴影的时钟，如图4.21所示。



图4.21 带阴影的时钟

技术要点

本例通过调用Date对象的一些方法，然后将当前时间显示在两个层中。

实现过程

(1) 编写带阴影时钟的JavaScript代码。

创建一个Date对象，通过调用Date对象中的一些方法来获得秒数和毫秒数，然后将其显示在页面中。代码如下。

```
<script language=JavaScript>
function clockon()
{
    thistime=newDate()
    varhours=thistime.getHours()
    varminutes=thistime.getMinutes()
    var seconds=thistime.getSeconds()
    if (eval(hours)<10) {hours="0"+hours}
    if (eval(minutes)<10) {minutes="0"+minutes}
    if (seconds<10) {seconds="0"+seconds}
    thistime=hours+":"+minutes+":"+seconds
    if(document.all)
    {
        bgclocknoshade.innerHTML=thistime
        bgclockshade.innerHTML=thistime
    }
    if(document.layers)
    {
        document.bgclockshade.document.write("<div
id=bgclockshade>' +thistime+' </div>")
        document.bgclocknoshade.document.write(' <div
id="bgclocknoshade">' +thistime+' </div>')
        document.close()
    }
    var timer=setTimeout("clockon()", 1000)
```

```
}  
</script>
```

(2) 在页面中添加两个div层，并设置bgclocknoshade层的top为48px，left为343px。代码如下。

```
<body onLoad="clockon()">  
  <br>  
  <div id="bgclockshade"  
  style="position:absolute;visibility:visible;font-  
  family:Arial;color:0066CC;font-  
  size:50px;top:50px;left:345px"></div>  
  <div id="bgclocknoshade"  
  style="position:absolute;visibility:visible;font-  
  family:Arial;color:FFCC00;font-size:50px; top:48px;  
  left:343px"></div>  
</body>
```

举一反三

根据本实例，读者可以：

在层中加入滤镜技术，加强带阴影时钟的视觉效果；  
使页面中的字体或图片变为阴影效果。

## 实例129 动态石英钟

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\129

实例说明

上网时，有时可能会发现网站的某个位置上会显示一个石英钟，这样的石英钟一般都是使用Flash制作的，不过用JavaScript编写会比

较麻烦一些，本例将使用JavaScript编写一个动态石英钟，如图4.22所示。

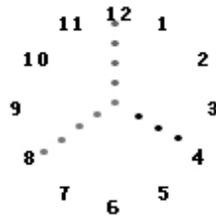


图4.22 动态石英钟

### 技术要点

本实例的技术要点是石英钟表盘以及指针的制作。

### 实现过程

(1) 创建一个Date对象，通过调用其方法来获得秒数和毫秒数，代码如下。

```
<script language=javascript>
pX=400;pY=200;
obs = new Array(13);
自定义函数ob()用于将表盘数字的位置存到一维数组obs中。
function ob()
{
  for (i=0; i<13; i++)
  {
    if (document.all)
      obs[i]=newArray (eval('ob'+i).style,-100,-100);
    else
      obs[i]=newArray (eval('document.ob'+i),-100,-100);
  }
}
```

自定义函数cl(a,b,c)用于在表盘显示秒针的位置。

```

function cl(a, b, c)
{
    if (document.all)
    {
        if (a!=0)
        b+=-1;
        eval('c'+a+'.style.pixelTop='+pY+(c));
        eval('c'+a+'.style.pixelLeft='+pX+(b));
    }
else
{
    if (a!=0)
    b+=10;
    eval('document.c'+a+'.top='+pY+(c));
    eval('document.c'+a+'.left='+pX+(b));
}
if (document.all)
    c0.style.pixelLeft=26
}
function runClock()
{
    for (i=0; i<13; i++)
    {
        obs[i][0].left=obs[i][1]+pX;
        obs[i][0].top=obs[i][2]+pY;
    }
}
}

```

```

var lastsec;
function timer()
{
    time=newDate ();
    sec= time.getSeconds();
    if (sec!=lastsec)
    {
        lastsec= sec;
        sec=Math.PI*sec/30;
        min=Math.PI*time.getMinutes()/30;
        hr=Math.PI*
((time.getHours()*60)+time.getMinutes())/360;
        for (i=1;i<6;i++)
        {
            obs[i][1]=Math.sin(sec) * (44-(i-1)*11)-16;
            if (document.layers)obs[i][1]+=10;
            obs[i][2]=-Math.cos(sec)* (44-(i-1)*11)-27;
        }
        for (i=6;i<10;i++)
        {
            obs[i][1]=Math.sin(min) * (40-(i-6)*10)-16;
            if (document.layers)obs[i][1]+=10;
            obs[i][2]=-Math.cos(min)* (40-(i-6)*10)-27;
        }
        for (i=10;i<13;i++)
        {
            obs[i][1]=Math.sin(hr)* (37-(i-10)*11)-16;

```

```

        if (document.layers)obs[i][1]+=10;
        obs[i][2]=-Math.cos(hr) * (37-(i-10)*11)-27;
    }
}
}

```

设置秒针、分针、时针在页面中的位置。

```

function setNum()
{
    cl (0,-65,-65);
    cl (1,10,-51);
    cl (2,30,-33);
    cl (3,35,-8);
    cl (4,28,17);
    cl (5,10,35);
    cl (6,-15,42);
    cl (7,-40,35);
    cl (8,-58,17);
    cl (9,-65,-8);
    cl (10,-58,-33);
    cl (11,-40,-51);
    cl (12,-16,-56);
}
</script>

```

(2) 在<body>中添加多个层，用于显示表盘及指针的位置，代码如下。

层c0~c12用于设置表盘。

```
<div id="c0" style="position:absolute;right:6;top:6; z-  
index:2;"></div>  
<div id="c1" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>1</b></div>  
<div id="c2" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>2</b></div>  
<div id="c3" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>3</b></div>  
<div id="c4" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>4</b></div>  
<div id="c5" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>5</b></div>  
<div id="c6" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>6</b></div>  
<div id="c7" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>7</b></div>  
<div id="c8" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>8</b></div>  
<div id="c9" style="position:absolute;left:20;top:-20; z-  
index:5;font-size:11px;"><b>9</b></div>  
<div id="c10" style="position:absolute;left:20;top:-20;  
z-index:5;font-size:11px;"><b>10</b></div>  
<div id="c11" style="position:absolute;left:20;top:-20;  
z-index:5;font-size:11px;"><b>11</b></div>  
<div id="c12" style="position:absolute;left:20;top:-20;  
z-index:5;font-size:11px;"><b>12</b></div>
```

层ob0~ob5用于设置秒针的位置。

```
<div id="ob0" style="position:absolute;left:-20;top:-20;z-index:1"></div>
```

```
<div id="ob1" style="position:absolute;left:-20;top:-20;z-index:8"> <font size="+3" color="#FF0000"><b>.</b></font></div>
```

```
<div id="ob2" style="position:absolute;left:-20;top:-20;z-index:8"> <font size="+3" color="#FF0000"><b>.</b></font></div>
```

```
<div id="ob3" style="position:absolute;left:-20;top:-20;z-index:8"> <font size="+3" color="#FF0000"><b>.</b></font></div>
```

```
<div id="ob4" style="position:absolute;left:-20;top:-20;z-index:8"> <font size="+3" color="#FF0000"><b>.</b></font></div>
```

```
<div id="ob5" style="position:absolute;left:-20;top:-20;z-index:8"> <font size="+3" color="#FF0000"><b>.</b></font></div>
```

层ob6~ob9用于设置分针的位置。

```
<div id="ob6" style="position:absolute;left:-20;top:-20;z-index:7"> <font size="+3" color="#0066FF"><b>.</b></font></div>
```

```
<div id="ob7" style="position:absolute;left:-20;top:-20;z-index:7"> <font size="+3" color="#0066FF"><b>.</b></font></div>
```

```
<div id="ob8" style="position:absolute;left:-20;top:-20;z-index:7"> <font size="+3" color="#0066FF"><b>.</b></font></div>
```

```
<div id="ob9"
style="position:absolute;left:-20;top:-20;z-index:7"> <font
size="+3" color="#0066FF"><b>.</b></font></div>
```

层ob10~ob12用于设置时针的位置。

```
<div id="ob10"
style="position:absolute;left:-20;top:-20;z-index:5"> <font
size="+3" color="#000000"><b>.</b></font></div>
```

```
<div id="ob11"
style="position:absolute;left:-20;top:-20;z-index:5"> <font
size="+3" color="#000000"><b>.</b></font></div>
```

```
<div id="ob12"
style="position:absolute;left:-20;top:-20;z-index:5"> <font
size="+3" color="#000000"><b>.</b></font></div>
```

(3) 应用<body>的onload事件调用JavaScript所定义的函数。

```
<body
onLoad="ob(),setNum(),setInterval('timer()',100);setInterval(
'runClock()',100)">
</body>
```

举一反三

根据本实例，读者可以：

将本实例中的指针改变为图片指针，使石英钟更加美观；

制作可以拖动的石英钟。

## 实例130 生日提醒器

本实例是一个适时显示提示信息的程序

实例位置：光盘\mingrisoft\04\130

## 实例说明

在我们见过的校友录中，可能会提供一个专门为同学过生日的提示栏，当某个同学快过生日时，提示栏就会显示某个同学将在多少天后过生日。本实例将使用JavaScript实现这一功能，如图4.23所示。



图4.23 生日提醒器

## 技术要点

本实例的技术点是如何判断月份之间的相隔天数，首先程序会得到当前日期，然后过滤数组中的数组元素，如果某个数组元素中的日期与当前日期相近或相等，则程序就会在页面中显示提示信息，如果不相近则提示“你的好朋友在这个月没有过生日的”。

## 实现过程

(1) 创建一个带参数的自定义函数

birthday(year, month, date, person)。代码如下。

```
<script language="JavaScript">
functionbirthday(year, month, date, person)
{
    this.year= year;
    this.month=month;
    this.date=date;
```

```
    this.person=person;
}
```

创建一个包含6个数组元素的数组对象，然后使每个数组元素获得 birthday() 函数赋给的值。

```
var friendlist = new Array(6);
friendlist[0] = new birthday(85, 2, 28, "小张");
friendlist[1] = new birthday(87, 1, 1, "小李");
friendlist[2] = new birthday(88, 7, 14, "小辉");
friendlist[3] = new birthday(88, 1, 8, "斌斌");
friendlist[4] = new birthday(87, 10, 12, "大黑");
friendlist[5] = new birthday(86, 12, 22, "君君");
```

使用Date对象获得当前的日期时间，然后与数组元素中的数据进行对比，如果当前日期与某个朋友的出生日期相近则提示“你的某个朋友将在多少天后过生日”。

```
var now=newDate();
today=newDate(now.getFullYear(),now.getMonth(),now.getDate());
;
var bool=false;
functionShowBirthday()
{
    for (i=0;i<6;++i)
    {
        thisDate=newDate(today.getFullYear(),friendlist[i].month-1,friendlist[i].date);
        varBmin=Math.round((thisDate.getTime()-today.getTime())/(24*60*60*1000));
        if (Bmin==0)
```

```

    {
        bool=true;
        note=i;
        break;
    }
    if (Bmin>0)
    {
        varnote=i;
        break;
    }
}
if (Bmin>=0)
{
    if ((note<5)&&(Bmin!=0))
    {
        for (i=note+1;i<6;++i)
        {
            thisDate=newDate(today.getYear(),friendlist[i].month-1,friendlist[i].date);
            varBmax=Math.round((thisDate.getTime()-today.getTime())/(24*60*60*1000));
            if (Bmin==0)
            {
                bool=true;
                note=i;
                break;
            }
        }
    }
}

```

```

        if (Bmax>0)
        {
            if (Bmin>Bmax)
            {
                Bmin=Bmax;
                note=i;
            }
        }
    }
}
if (bool==false)
{
    document.write("<Center>"+"提示"+"<br>"+"<br>"+"
</Center>");
    document.write("你的好朋友[
"+friendlist[note].person+" ]将在"+Bmin+"天后过生日!
<P>");
}
else
{
    document.write("<Center>"+"提示"+"<br>"+"<br>"+"
</Center>");
    document.write("你的好朋友[
"+friendlist[note].person+" ]在今天过生日! <P>");
}
}
else

```

```

    {
        document.write("<Center>"+"提示"+"<br>"+"<br>"+"
</Center>");
        document.write("你的好朋友在这个月没有过生日的!
<P>");
    }
}

```

在页面中输出当前日期。

```

function showdate()
{
    varnow=newDate();
    varyear=now.getFullYear();
    varmonth=now.getMonth()+1;
    varday=now.getDate();
    document.write("今天
是："+year+"年"+month+"月"+day+"日");
}
</script>

```

(2) 添加页面设计代码，在表格列中调用JavaScript自定义函数。

```

<body>
    <script language= "javascript">
        showdate();
    </script>
    <br>
    <br>
    <br>

```

```
<br>
<br>
<tablewidth="238"height="115"border="0"align="center"cellpadding="0"cellspacing="0">
<tr>
<tdheight="62"background="images/default_02.gif">&nbsp;
</td>
</tr>
<tr>
<tdwidth="238"height="176"align="center"valign="middle"background="images/default_01.jpg">
<tablewidth="200"height="100"border="0"cellpadding="0"cellspacing="0">
<tr>
<td align="left"valign="top">
<font color="#FF0000">
<script language="JavaScript">
ShowBirthday();
</script>
</font>
</td>
</tr>
</table>
</td>
</tr>
</table>
</body>
```

举一反三

根据本实例，读者可以：

制作节日提示；

在特殊日期的当天显示自定义信息。

## 实例131 数字时钟

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\04\131

实例说明

在网页中加入数字时钟既可以增加页面的活力，又可以方便浏览者。下面将通过一个具体实例介绍如何实现数字时钟。运行本实例，在页面中将显示一个走动的数字时钟，如图4.24所示。



图4.24 数字时钟

技术要点

制作本实例的基本思路如下：首先需要在页面中添加 7 个 image，主要用于显示两位数的时、分、秒以及am（上午）和pm（下午），然后在JavaScript所定义的函数中利用Date对象获得当前时间，最后随着时间的变化快速改变图片。

实现过程

（1）创建一个Date对象，通过调用其方法来获得当前时间，然后跟随时间变化图片。代码如下。

```
<script language="javascript">
function clock()
{
    date=newDate();
```

```
hour=date.getHours();
minu=date.getMinutes();
sec=date.getSeconds();
if(sec<10)
    {sec="0"+sec;}
if(minu<10)
    {minu="0"+minu;}
if(hour<10)
    {hour="0"+hour;}
time=hour+":"+minu+":"+sec;
document.img1.src="images/Clock/c"+time.charAt(0)+".gif";
document.img2.src="images/Clock/c"+time.charAt(1)+".gif";
document.img3.src="images/Clock/c"+time.charAt(3)+".gif";
document.img4.src="images/Clock/c"+time.charAt(4)+".gif";
document.img5.src="images/Clock/c"+time.charAt(6)+".gif";
document.img6.src="images/Clock/c"+time.charAt(7)+".gif";
if(hour>12)
    {document.img7.src="images/Clock/cpm.gif";}
else
    {document.img7.src="images/Clock/cam.gif";}
setTimeout('clock()',1000);
```

```
}
```

```
</script>
```

(2) 添加页面设计代码，当页面加载时调用clock()函数。

```
<body onLoad="clock();">
```

```
<img name="img1" width="16" height="21"><img name="img2"
width="16" height="21"><img name="img3" width="16"
height="21"><img name="img4" width="16" height="21"><img
name="img5" width="16" height="21" id="img5"><img name="img6"
width="16"height="21" id="img6"><img
name="img7" width="16" height="21" id="img7">
```

```
</body>
```

举一反三

根据本实例，读者可以：

以另一种图片风格显示类似于本例的数字时钟。

## 实例132 各地时间表

本实例是一个应用广泛的程序

实例位置：光盘\mingrisoft\04\132

实例说明

一般操作系统上使用的都是北京日期时间，如果要查看某一地区的日期时间，可以打开操作系统中的“日期和时间属性”对话框进行查看。本实例将简单地实现这一功能。本实例在页面中添加了6个按

钮，在按钮上分别显示着不同地区的名称，当用户单击某个按钮时，当地时间文本框中就会显示所选地区的日期时间，如图4.25所示。



图4.25 各地时间表

### 技术要点

本实例主要应用了Date 对象的getTimezoneOffset ()方法，此方法用于返回用分钟表示的本地计算机上的时间和全球标准时间（UTC）之间的差别。

getTimezoneOffset ()方法返回一个整数值，这个整数代表了当前计算机上的时间和 UTC之间相差的分钟数。这些值和执行脚本的计算机相关。如果这个方法被一个服务器脚本调用，那返回值和服务器相关，而如果这个方法被一个客户机脚本调用，那么返回值就根据客户机上的时间来确定。

### 实现过程

(1) 编写用于实现各地时间表的JavaScript代码。

```
<script language="JavaScript">
var timerRunning= false;
var zonetime= "格林威治标准时间: ";
var adjust=0;
function timeCheck(tzone,diff)
{
    if (timerRunning)
```

```

    {
        clearTimeout (updatetime);
        timerRunning= false;
    }
    gmtOffset= eval (diff+adjust);
    zonetime= tzone;
    checkDateTime ();
}
function checkDateTime ()
{
    varnow=newDate ();
    varyears=now. getYear ();
    varmonth=now. getMonth ()+1;
    vardate=now. getDate ();
    varday=now. getDay ();
    varhour=now. getHours ();
    varminute=now. getMinutes ();
    var second=now. getSeconds ();
    var lastSat=date-(day+1);
    while (lastSat<32) lastSat+=7;
    if (lastSat>31)
        lastSat+=-7;
    var firstSat=date-(day+1);
    while (firstSat>0) firstSat+=-7;
    if (firstSat<1)
        firstSat+=7;
}

```

```

    if (((month==4)&& (date>= firstSat)) || month>4)&&
(month<11 || ((month==10)&&day<= lastSat)))
    adjust+=60;
    yourOffset= (newDate()).getTimezoneOffset();
    yourOffset= yourOffset+adjust;
    varxx=navigator.appName
    varxy=navigator.appVersion;
    xy=xy.substring(0,1);
    if ((xy==4)&& (xx== "Netscape"))
        yourOffset=yourOffset+adjust;
    if (((month==4)&& (date> 20)) || month>4)&& (month<11
|| ((month==10)&&day<30))) adjust-=60;
    ourDifference= eval(gmtOffset-yourOffset);
    varhalf= eval(ourDifference%60);
    ourDifference=Math.round(ourDifference/60);
    hour= eval(hour-ourDifference);
    varm=newArray("", "1", "2", "3", "4", "5", "6", "7", "8", "9", "1
0", "11", "12");
    var leap= eval(years%4);
    if ((half== -30) || (half==30))
        minute+=30;
    if (minute>59)
        minute-=60, hour++;
    if (minute<0)
        minute+=60, hour--;
    if (hour>23)
        hour-=24, date+=1;

```

```

    if (((month==4) || (month==6) || (month==9) ||
(month==11))&& (date==31))
        date=1, month++;
    if (((month==2)&& (date>28))&& (leap != 0))
        date=1, month++;
    if ((month==2)&& (date>29))
        date=1, month++;
    if (hour<0)
        hour+=24, date--;
    if ((date==32)&& (month==12))
        month=m[1], date=1, years++;
    if (date==32)
        date=1, month++;
    if ((date<1)&& (month==1))
        month=m[12], date=31, years--;
    if (date<1)
        date=31, month--;
    if (((month==4) || (month==6) || (month==9) ||
(month==11))&& (date==31))
        date=30;
    if ((month==2)&& (date>28))
        date=29;
    if (((month==2)&& (date>28))&& (leap != 0))
        date=28;
    for (i=1; i<13; i++)
    {
        if (month== i)

```

```

    {
        month=m[i];break;
    }
}
vardateTime=hour;
dateTime= ((dateTime<10) ? "0:"+"")+dateTime;
dateTime+= ((minute<10) ? ":0" : ":")+minute;
dateTime+= ((second<10) ? ":0" : ":")+second;
//dateTime+= " "+year+"年"+month+"月"+date+"日" ;
zone.value= zonetime;
localtime.value= years+"/"+month+"/"+date+" "+dateTime;
updatetime=setTimeout("checkDateTime()", 1000);
timerRunning= true;
}
</script>

```

(2) 添加页面设计代码，当页面加载时调用timeCheck()函数，代码如下。

```

<bodyOnLoad= "timeCheck(zonetime,0)">
<p>&nbsp;  </p>
<tableWIDTH="359"HEIGHT="204"BORDER="1"ALIGN="CENTER"BO
RDERCOLOR="#006500">
<tr>
<tdCOLSPAN="3"ALIGN="CENTER"VALIGN="MIDDLE"BACKGROUND
="Images/map.GIF">
<tableWIDTH="265"BORDER="0"ALIGN="CENTER"BO
RDERCOLOR="#FF
FF00">
<trALIGN="RIGHT">

```

```

<tdCOLSPAN="2">
  <tableWIDTH="100%"BORDER="0">
    <tr>
      <tdCOLSPAN="4"ALIGN="RIGHT"CLASS="style1
style6"><strong>地 区: </strong></td>
      <tdWIDTH="165"ALIGN="LEFT"CLASS="style4">
<inputNAME="zone"TYPE="TEXT"VALUE=""SIZE="18">
</td>
    </tr>
  </table>
</td>
</tr>
<trALIGN="RIGHT">
  <tdWIDTH="88"HEIGHT="19"CLASS="style1"><strong>当
地时间: </strong></td>
  <tdWIDTH="167"ALIGN="LEFT">
<inputNAME="localtime"TYPE="TEXT"VALUE=""SIZE="18">
</td>
</tr>
<trALIGN="CENTER">
  <tdCOLSPAN="2">
<tableWIDTH="194"HEIGHT="75"BORDER="1"CELLPADDING="4"
CELLSPACING="3"BORDERCOLOR="#FFFF00">
  <tr>
    <tdWIDTH="55"ALIGN="CENTER">
<inputTYPE="BUTTON"NAME="bjbutton"VALUE="北
京"onClick="timeCheck(this.value,-500)"></td>

```

```
<tdWIDTH="55"ALIGN="CENTER">
<inputTYPE="BUTTON"NAME="djbutton"VALUE="东
京"onClick="timeCheck(this.value,-540)"></td>
<tdWIDTH="136"ALIGN="CENTER">
<inputTYPE="BUTTON"NAME="ldbutton"VALUE="伦
敦"onClick="timeCheck(this.value,+0)"></td>
</tr>
<tr>
<td>
<inputTYPE="BUTTON"NAME="mxgbutton"VALUE="墨西
哥"onClick="timeCheck(this.value,+360)"></td>
<td>
<inputTYPE="BUTTON"NAME="tpybutton"VALUE="太平
洋"onClick="timeCheck(this.value,+480)"></td>
<td>
<inputTYPE="BUTTON"NAME="xwybutton"VALUE="夏威
夷"onClick="timeCheck(this.value,+600)"></td>
</tr>
</table>
</td>
</tr>
</table>
</td>
</tr>
</table>
<p>&nbsp;</p>
<divALIGN="CENTER"></div>
```

</body>

举一反三

根据本实例，读者可以：

开发时区选择器；

查询某个时区的时间表。

## 实例133 位于页面顶层的时钟

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\04\133

实例说明

在浏览网站时，经常可以看到一些动态日期和时间的显示。本实例将时间以电子时钟的形式悬浮在页面上。程序运行结果如图4.26所示。



图4.26 位于页面顶层的时钟

技术要点

本实例主要应用Date函数获取当前系统的小时、分钟、秒。应用层的innerHTML属性将获取的系统时间动态添加到层中，再应用window对象的setTimeout( )方法将层中的时间输出到浏览器中。

### 实现过程

(1) 编写用于实现悬浮时钟的JavaScript代码，应用function函数创建自定义函数shotime()。代码如下。

```
<script language="javascript">
function shotime()
{
var now=newDate();
hour=now.getHours();
minute=now.getMinutes();
second=now.getSeconds();
if (minute<10) {minute='0'+minute}
if (second<10) {second='0'+second}
str=hour+":"+minute+":"+second;
str='<b><font style="font-size:50px ">' +str+' </font></b>';
div1.style.top="100px";
div1.style.fontFamily="verdana";
div1.style.color="#DDDDDD";
div1.style.left="30px";
div1.innerHTML=str;
div2.style.top="102px";
div2.style.color="#999999";
div2.style.fontFamily="verdana";
div2.style.left="32px";
div2.innerHTML=str;
```

```
window.setTimeout("shotime();", 1000);  
}
```

在页面加载时，调用自定义函数shotime()。

```
shotime();  
</script>
```

(2) 在<body></body>标记中添加两个层 (div)，分别设置id名称为“div2”、“div1”。代码如下。

```
<div id="div2" style="position:absolute "></div>  
<div id="div1" style="position:absolute "></div>
```

举一反三

根据本实例，读者可以：

在编写商务网站时，将时钟悬浮在网站首页；

在开发办公自动化管理系统时，将时钟悬浮在网站首页；

在开发宠物网站时，将时钟悬浮在网站首页。

## 第5章 文字特效

文字的颜色变换

文字显示效果

文字动态移动

文字动画效果

加密与解密

改变文字大小

指定文字位置

文字立体效果

文字特殊效果

## 5.1 文字的颜色变换

在网页制作时，可以通过对文字的颜色变换来实现网页的视觉动态效果，下面将通过几个典型实例进行详细讲解。

### 实例134 简单文字变色

本实例是一个美化界面的程序

实例位置：光盘\mingrisoft\05\134

实例说明

浏览页面时，经常会看到一些页面的标题颜色不断地变化。本实例是在指定的时间内对文字的颜色进行变化，使文字在网页中以不同的颜色显示。运行结果如图 5.1、图 5.2所示。



明日科技图书

图5.1 文字变色前的效果



明日科技图书

图5.2 文字变色后的效果

技术要点

本实例主要是通过窗体的onLoad事件调用JavaScript自定义函数来实现的，在该自定义函数中关键是修改当前页的fgColor属性来修改页面的前景颜色，并用window对象的setInterval()方法以指定的时间

调用 JavaScript 自定义函数 `changecolor()`。下面对 `setInterval()` 方法进行详细地介绍。

`setInterval()` 方法的语法格式如下。

```
setInterval (function, milliseconds, [arguments])
```

参数说明如下。

- **function:** 要重复调用的JavaScript 自定义函数名称。
- **milliseconds:** 设置超时时间（以ms 为单位）。

功能：经过超时时间后，重复调用函数。此值可以用 `clearInterval()` 函数清除。

实现过程

(1) 在<body>标记中，用<h1>标记在页面中输入最大号的标题，并将 `align` 属性设置为`center`，使文字在页面中居中，代码如下。

```
<h1 align="center">明日科技图书</h1> //在页面中显示标题文字
```

(2) 编写用于实现动态改变文字颜色的JavaScript代码。

```
<script language="JavaScript">
var
colors=newArray("red","orange","green","blue","browm","putple",
,"gray","white");
var ind=0;
function changecolor() {
    document.fgColor=colors[ind++];    //选择要修改的颜色
    if (ind==(colors.length)) {ind=0}
}
setInterval("changecolor()", 500);    //重复调用
changecolor() 函数
```

```
window.onload=changeColor;//在窗体载入事件中调用  
changeColor()函数
```

```
</script>
```

举一反三

根据本实例，读者可以：

开发在页面显示时，对标题进行颜色变换的程序；

开发当鼠标指向标题时，对标题进行颜色变换的程序。

## 实例135 文字变色

本实例是一个人性化实例

实例位置：光盘\mingrisoft\05\135

实例说明

本实例是在页面载入时以动态的形式在页面中显示文字，并以一定的时间间隔对文字的颜色进行变换。运行结果如图5.3、图5.4所示，其显示效果与实例134相同，只是上一实例在页面载入前就已经将文字显示在页面中。



在JavaScript中实现文字的颜色变换

图5.3 文字变色前的效果



在JavaScript中实现文字的颜色变换

图5.4 文字变色后的效果

技术要点

本实例主要是通过form组件的innerHTML属性将指定的文本动态添加到form组件中，并用window对象的setTimeout()方法以指定时间调用自定义函数Dong()。

setTimeout()方法的语法格式如下。

```
setTimeout (function, milliseconds, [arguments])
```

参数说明如下。

- function: 要调用的JavaScript 自定义函数名称。
- milliseconds: 设置超时时间 (以ms 为单位)。

功能: 经过超时时间后, 调用函数。此值可以用clearTimeout()函数清除。

注意: setTimeout()方法在超时时间后只调用一次函数, 而setInterval()方法是按一定时间重复调用指定的函数。

实现过程

(1) 在网页中添加一个form组件, 并设置id为object1, 代码如下。

```
<form id="object1" name="form1" method="post" action="">  
</form>
```

(2) 编写用于实现文字变色的JavaScript代码。

```
<script language="JavaScript">  
//定义一个数组, 用来记录要替换的颜色值  
var  
colors=newArray("F9F9F9", "F5F5F5", "F1F1F1", "99FF33", "99FF66",  
"99FF99", "CC3399",  
"CC3355", "CC3300", "CC00FF", "CC00AA", "CC0055", "66FFCC", "66  
FF88", "66FF11");  
var a=0, b=1;  
functionDong() {  
    color=colors[a]; //修改页面的前景颜色  
    aa="<font size='8'><font face='黑体'><font  
color="+color+"><p align='center'>
```

```
在JavaScript中实现文字的颜色变换</p></font></font>
</font>”;
```

```
    object1.innerHTML=aa;           //通过innerHTML属性动态修
改文字的颜色
    a+=b;
    if (a==15)b=-1;
    if (a==0)b=1;
    xx=setTimeout(“Dong()”, 300);    //以指定时间调用一次
Dong() 函数
    }
</script>
```

(3) 在<body>标记的窗体载入 (onLoad) 事件中调用自定义函数 Dong(), 代码如下。

```
<body onload=“Dong()”>
```

举一反三

根据本实例，读者可以：

开发以动态的形式在页面中显示不同的文字；

在页面中实现动态改变文字的大小。

## 实例136 变换的文字

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\05\136

实例说明

本实例是将指定的文字在页面上进行居中显示，并以一定的时间间隔动态显示不同的文字，同时修改其字体颜色。运行结果如图5.5、图5.6所示。

# 明日科技图书

图5.5 文字变换前的效果

明日科技JavaScript图书

图5.6 文字变换后的效果

## 技术要点

本实例主要是通过层的innerHTML属性将指定的文本动态添加到层中，并用window对象的setTimeout()方法以指定时间调用自定义函数Dong()来更换文本及其字体颜色。setTimeout()方法已在上一实例中进行了详细地说明，这里不再赘述。

Aletter.length表示获取Aletter数组中元素的数目，以实现循环显示文字的个数。

## 实现过程

(1) 在页面中添加一个层，代码如下。

```
<div id="cc" align="center" style="position: absolute;
left:-1; top: 5; width: 1003px;
height: 19"></div>
```

(2) 编写用于实现文字变换的JavaScript代码。

自定义保存颜色及文字的一维数组，其代码如下。

```
<script language="JavaScript">
varAletter=newArray(); //定义数组，用来保存要替换的文字
varAcolor=newArray("#3300CC","#00FFCC","#FF66CC","#CC0000",
"#00FF00","#9900FF",
"#FFCC33","#6633FF","#FFCCFF");
```

```

Aletter[0]="明日科技图书";
Aletter[1]="明日科技Delphi图书";
Aletter[2]="明日科技VB图书";
Aletter[3]="明日科技Java图书";
Aletter[4]="明日科技c图书";
Aletter[5]="明日科技.net图书";
Aletter[6]="明日科技JavaScript图书";
Aletter[7]="明日科技ASP图书";
Aletter[8]="明日科技JSP图书";
var i=0;
document.all.cc.style.width=document.body.clientWidth;
//将层的宽度设为页面工作区宽度

```

自定义函数letter()，用于改变层中的文字及字体颜色，其代码如下。

```

function letter() {
    if (i==Aletter.length) i=0;
    //通过innerHTML属性动态替换文字及颜色
    cc.innerHTML='<font color="" +Acolor[i]+' "
size="+5">' +Aletter[i]+' </font>';
    ++i;
    setTimeout("letter()",1000);    //每秒调用一次自定义函
数letter()
}
letter();    //在窗体载入时调用自定义函数
letter()
</script>

```

举一反三

根据本实例，读者可以：  
实现在页面的任意位置显示文字；  
在任意组件的上面改变文字的大小。

## 实例137 描边文字

本实例是一个美化界面程序

实例位置：光盘\mingrisoft\05\137

实例说明

在页面设计中，有许多文字都有描边效果。本实例是通过设置表格中文字的外边框颜色，使文字具有描边效果，如图5.7所示。

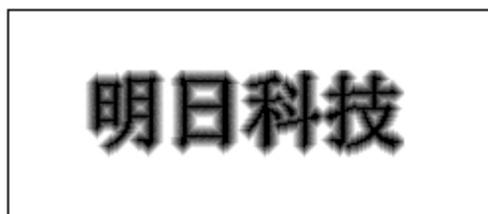


图5.7 描边文字

技术要点

本实例主要是通过<td>标记中的style样式的filter滤镜的glow属性来设置字体的外边框颜色及大小，并用document对象的write()方法将表格显示在浏览器中。write()方法的详细介绍如下。

write()方法的功能：该方法用一个字符串作为其参数，并在浏览器中显示。该字符串可以是普通文本或HTML。

语法格式如下。

```
document.write (str)
```

参数说明如下。

str：字符串，该字符串可以是普通文本或HTML。字符串可以用单引号或双引号括起来，这样可以引用那些包含引号的内容。

实现过程

(1) 编写用于实现描边文字的JavaScript代码。

```
<script language="JavaScript">
functionportrayal() {
    varpor='<table><tr><td style=';
    por=por+'font-family:'+'宋体'+'; font-size:40px;
filter:glow(color=0000ff,
    strenght=0)width:auto;height:auto;position:relative;'>';
    por=por+'明日科技</td></tr></table>';
    document.write(por);
}
</script>
```

(2) 在<body>标记中的onLoad事件中调用自定义函数portrayal(), 代码如下。

```
<body onLoad="portrayal();">
```

举一反三

根据本实例, 读者可以:

用write()方法在窗体中的指定位置显示图片、按钮等其他组件; 动态显示文字的外边框颜色。

## 实例138 霓虹灯文字

本实例是一个提高效率、人性化的程序

实例位置: 光盘\mingrisoft\05\138

实例说明

在页面设计中, 有许多文字都是以霓虹灯效果来显示。本实例是通过对文本中的每个文字的颜色进行修改, 使文本的颜色看上去在不停变换, 从而达到霓虹灯效果, 如图 5.8所示。



欢迎您购买明日科技图书

图5.8 霓虹灯文字

### 技术要点

本实例主要是通过 `substring()` 方法来获取指定文本中的单个文字，并将其设置成不同的颜色，再用 `innerHTML` 属性将文字动态显示在层中。用 `setTimeout()` 方法循环调用自定义函数 `Dcolor()`，使文字看上去具有霓虹灯效果。`substring()` 方法的详细说明如下。

`substring()` 方法的语法格式如下。

```
strVariable.substring (start, end)
```

参数说明如下。

- `strVariable`: 要截取的字符串，可以是变量形式，也可以是文本形式。

- `start`: 指明子字符串的起始位置，该索引从0 开始起算。

- `end`: 指明子字符串的结束位置，该索引从0 开始起算。

功能：`substring()` 方法将返回一个包含从 `start` 到最后（不包含 `end`）的子字符串的字符串。

### 实现过程

编写用于实现霓虹灯文字的JavaScript代码。

```
<script language="JavaScript">
varTname="欢迎您购买明日科技图书";
varTlen=Tname.length;    //获取文本的长度
//在页面中显示带有文字的层
document.write("<div id=' a' style=' font-
size:40px;'>" +Tname+"</div>");
```

```

//定义一个颜色数组
var
col=newArray("#FFCC00", "#3333FF", "#FFCC00", "#FF0000", "#FFCC00", "#CC33FF");
var ic=0;
functionDcolor() {
    varStrname="";
    for (i=0;i<Tlen;++i) {
        varStrname=Strname+"<font
color="+col[ic]+">" +Tname.substring(i, i+1)+"</font>";
        ic=ic+1;
        if (ic==col.length) ic=0;
    }
    a.innerHTML=Strname;    //用innerHTML属性动态修改文字的颜色
    setTimeout("Dcolor()", 200); //每隔一定时间调用自定义函数Dcolor()
}
Dcolor();
</script>

```

举一反三

根据本实例，读者可以：

实现动画图片效果；

使文本中的单个文字的大小进行变化。

## [实例139 追逐点亮的文字](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\139

实例说明

浏览网页时，经常会看到追逐点亮文字效果。本实例是通过对文本中的每个文字的颜色进行逐变，并将改变后的文字恢复到以前的颜色，使文本中的文字颜色看上去是逐个变色的。运行结果如图5.9所示。

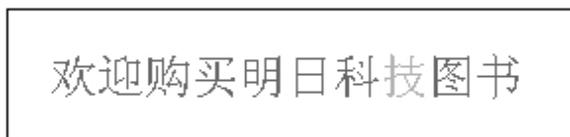


图5.9 追逐点亮的文字

技术要点

本实例主要是通过substring()方法逐个对文字的颜色进行改变，并用innerHTML属性将修改后的文字显示在浏览器中，再用setTimeout()方法循环调用自定义函数gogo()，使文字看上去具有逐个变色的效果。substring()方法的详细介绍请参见实例138。

实现过程

编写用于实现追逐点亮文字的JavaScript代码。

```
<script language="JavaScript">
Strname="欢迎购买明日科技图书";
StrLen=Strname.length;
document.write("<div id='a' style='color:#6666FF' >
</div>"); //在页面中创建一个层
var act=0;
function gogo() {
    ++act;
    act=(act>(StrLen-1))?0:act;
    act=(act<0)?(StrLen-1):act;
```

```
a.innerHTML=Strname.substring(0,act)+"<font  
color=' #00FF33' >"  
+Strname.substring(act,act+1)+"  
</font>" +Strname.substring(act+1,StrLen);  
}  
setInterval("gogo()",200); //重复调用自定义函数gogo()  
</script>
```

举一反三

根据本实例，读者可以：

逐个改变文字的大小；

制作一个红绿灯。

## 实例140 荧光文字

本实例是一个人性化实例

实例位置：光盘\mingrisoft\05\140

实例说明

在页面设计中，有些文字的外边框会以不同的颜色进行显示。本实例是通过对文字外边框的颜色进行变换，使文字看上去具有变换的荧光效果。运行结果如图5.10所示。

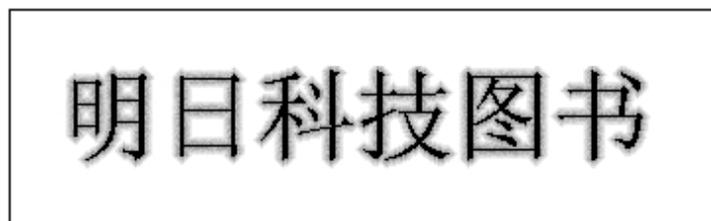


图5.10 荧光文字

技术要点

本实例主要是通过层的style样式的filter滤镜的glow属性来设置层中文字的外边框颜色及大小，并用window对象的setTimeout()方法

按一定时间调用自定义函数change(),使文字产生动态的荧光效果。

层中的style-filter-glow属性,可以用JavaScript脚本写成如下形式。

```
div1.filters.glow.color="#11CC55";//设置字体的外边框颜色
```

```
div1.filters.glow.strength=4; //设置字体外边框的高度
```

注意:div1为层的ID名称。div1.filters.glow.color语句只能在层中设置了glow属性后才可以进行修改。

### 实现过程

编写用于实现荧光文字的JavaScript代码。

(1) 自定义函数change(),将层中文字的外边框颜色进行动态改变。

```
<script>
varPname="明日科技图书";
varDivlay='<div id="div1" style="width:400px;
filter:glow(#CC0033,strength=2);
font-size:50px;FONT-FAMILY:CourierNew;">' +Pname+' </div>';
document.write(Divlay); //将层显示在浏览器中
function change() {
    if (div1.filters.glow.strength==2) {
        div1.filters.glow.color="#11CC55";//设置层中字体的外
        边框颜色
        div1.filters.glow.strength=4; //设置层中字体的外边框
        长度
    }
    else{
        div1.filters.glow.color="#CC0033";
        div1.filters.glow.strength=2;
    }
}
```

```
}  
if (col>5) col=1;  
    setTimeout("change()", 200);  
}
```

(2) 在窗体载入时，调用自定义函数change()。

```
change();
```

```
</script>
```

举一反三

根据本实例，读者可以：

对层中的文字进行缩放；

动态变换字体。

## 实例141 发光文字的闪烁效果

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\05\141

实例说明

在页面设计中，经常需要一些文字的外边框具有发光且闪烁的效果。本实例将通过对文字外边框颜色的设置，从而实现这一效果。运行结果如图5.11、图5.12所示。

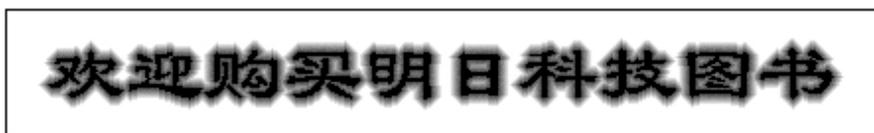


图5.11 发光文字闪烁前的效果

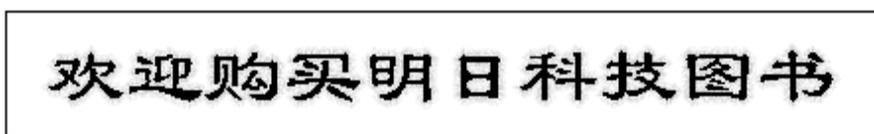


图5.12 发光文字闪烁后的效果

技术要点

本实例是用 `string` 对象中的 `fromCharCode()` 方法将指定的数转换成字符，并将这些字符组合成颜色值，通过这些值设置层的 `style-filter-glow` 属性中文字的外边框颜色，并用 `window` 对象的 `setTimeout()` 方法按一定时间调用自定义函数。

本实例主要通过以下代码对层的外边框进行设置。

```
//设置字体的外边框颜色及大小
```

```
document.all.Tdiv.style.filter="glow(color="+tcolor[color_x]+" , strength=4)";
```

注意：以上语句可以在任何时候对层的外边框进行修改，而无需事先在层中设置 `glow` 属性。

`string` 对象的 `fromCharCode()` 方法的详细说明如下。

`fromCharCode()` 方法的语法格式如下。

```
string.fromCharCode([code1[, code2[, ...[, codeN]]]])
```

参数说明如下。

- `string`：必选项，为 `string` 对象。

- `[code1[, code2[, ...[, codeN]]]]`：可选项，是要转换为字符串的 Unicode 字符值序列。如果没有给出参数，结果为空字符串。

功能：从一些 Unicode 字符值中返回一个字符串。

实现过程

(1) 在 `<body>` 标记中添加一个层，代码如下。

```
<div id="Tdiv" style="position:absolute;visibility:visible;width:600px;text-align:center; top:100px;left:70px;font-family:隶书;font-size:30pt"></div>
```

(2) 编写用于实现发光文字的闪烁效果的 JavaScript 代码。

自定义数组 `tcolor`，用于保存变化的颜色值。代码如下。

```

<SCRIPT language="JavaScript">
var tcolor=newArray();
for (i=2;i<=15;++i) {
    if (i>9) {
        varp=i-9;
        varTp=String.fromCharCode((96+p));    //将字符值转换成字符
        tcolor[i]=Tp+Tp+Tp+Tp+Tp+Tp;        //组合成颜色值
    }
    else{
        tcolor[i]=
(String(i)+String(i)+String(i)+String(i)+String(i)+String
(i));
    }
}

```

自定义函数fluorescence(), 用于变换文字的外边框颜色。代码如下。

```

varmessage="欢迎购买明日科技图书"
var color_x=0
var timer
function fluorescence() {
    if (color_x<=15) {
        Tdiv.innerText=message;
        document.all.Tdiv.style.filter="glow(color="+tcolor[c
olor_x]+"\", strength=4)"
        color_x++
        timer=setTimeout("fluorescence()",100)
    }
}

```

```

    }
    else{
        clearTimeout(timer)
        setTimeout("reversefluor()",1500)
    }
}

```

自定义函数reversefluor(), 用于反向显示文字外边框的颜色。代码如下。

```

function reversefluor() {
    if (color_x>=0) {
        Tdiv.innerText=message;
        document.all.Tdiv.style.filter="glow(color="+tcolor[co
olor_x]+",
        strength=4)";
        color_x--;
        timer=setTimeout("reversefluor()",100);
    }
    else{
        clearTimeout(timer);
        color_x=0;
        fluorescence();
    }
}
</SCRIPT>

```

(3) 在<body>标记中用窗体载入事件onLoad调用自定义函数fluorescence(), 代码如下。

```
<body bgcolor="#fef4d9" onLoad="fluorescence()">
```

举一反三

根据本实例，读者可以：

动态设置其变换颜色；

随机改变字体的颜色。

## 实例142 文字的舞台灯光效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\142

实例说明

在许多的网页特效中，在文字上面可以用灯光使文字具有动态照明的效果，并可以用鼠标来控制照明的强度和方向，如图5.13所示。



图5.13 文字的舞台灯光效果

技术要点

本实例主要是通过层的style-filter-light属性的addambient()、addcone()和moveLight()方法来设置灯光的大小及移动范围。addambient()是用来加入包围的光源；addcone()用来加入锥形光源；moveLight()用来设置光源的位置。

实现过程

(1) 在<body>标记中添加一个层，代码如下。

```
<DIV id=Tdiv style="COLOR: white; FILTER: light; HEIGHT: 300px; LEFT: 10px; POSITION: relative; TOP: 10px; WIDTH: 400px"><P align=center>明日科技图书</P></DIV>
```

(2) 编写用于实现文字的舞台灯光效果的JavaScript代码。

自定义函数illuminate(), 用于设置灯光的颜色及移动范围。其代码如下。

```
<SCRIPT language="JavaScript">  
function illuminate() {  
    document.all.Tdiv.filters.light.addambient(0, 0, 255, 30);  
    document.all.Tdiv.filters.light.addcone(400, 400, 120, 160  
, 100, 200, 124, 150, 120, 5);  
}
```

自定义函数Move(), 用鼠标来控制灯光的照明效果。其代码如下。

```
function Move() {  
    document.all.Tdiv.filters.light.moveLight(1, window.event.x-20, window.event.y, 0, 1)  
}  
</SCRIPT>
```

(3) 在<body>标记中用窗体载入事件 (onLoad) 调用自定义函数illuminate(), 在鼠标移动事件 (onmousemove) 中调用自定义函数Move(), 代码如下。

```
<body onload="illuminate();" onmousemove="Move()" >
```

举一反三

根据本实例, 读者可以:

实现对图片进行灯光效果;

实现对页面进行自动灯光循环效果。

## 实例143 制作彩色渐变文字动画

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\05\143

## 实例说明

在如今的互联网上，网站页面效果越来越养眼，如何吸引住众多网民的眼球，也就成为网页美工和网页设计师们追求的目标。在原本安静的静态页面中加入彩色渐变和文字动画的效果，既可以增加页面的动态效果，又可以吸引浏览者的眼球。本实例将介绍如何制作彩色渐变文字动画，当用户浏览网页时，页面中的文字“《Visual Basic 编程词典》软件的特点如下：”将不断变色，如图5.14所示。

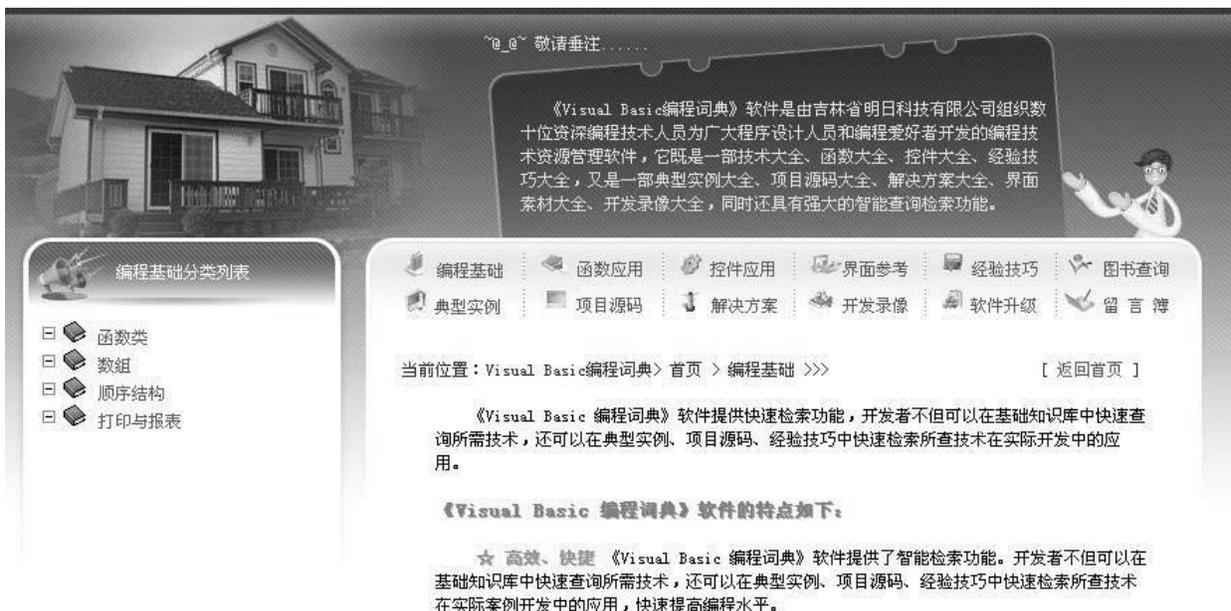


图5.14 制作彩色渐变文字动画

## 技术要点

本实例主要应用JavaScript脚本的Math对象的floor()方法、Number对象的toString()方法和 window 对象的 setInterval()方法来实现。Number 对象的 toString()方法和 window 对象的 setInterval()方法在前面已经介绍过，下面对Math对象的floor()方法进行详细介绍。

说明：Number对象的toString()方法的详细介绍请参见实例102；window对象的setInterval()方法的详细介绍请参见实例134。

Math对象的floor()方法用于将一个数字向下取整，返回小于或等于函数参数，并且与之最接近的整数。例如：

```
Math.floor(3.14); //返回的结果为3.0
```

```
Math.floor(3.99); //返回的结果为3.0
```

```
Math.floor(-3.99); //返回的结果为-4.0
```

```
Math.floor(-3.14); //返回的结果为-4.0
```

floor()方法的语法格式如下。

```
Math.floor(x)
```

参数说明如下。

x: 任意的数据或表达式。

实现过程

(1) 编写用于实现彩色渐变文字动画的JavaScript代码。

编写自定义函数makeColor()，用于生成十六进制的颜色值，代码如下。

```
<script language="javascript">
var rate=20;
var elmH=0;
var elmS=128;
var elmV=255;
functionmakeColor() {
if (elmS==0) {
    elmR= elmV; elmG= elmV; elmB= elmV;
}
else {
    t1= elmV;
```

```

t2= (255-elmS) *elmV/255;
t3= elmH%60;
t3= (t1-t2) * t3/60;
if (elmH<60) {
    elmR= t1; elmB= t2; elmG= t2+t3;
}else if (elmH<120) {
    elmG= t1; elmB= t2; elmR= t1-t3;
}else if (elmH<180) {
    elmG= t1; elmR= t2; elmB= t2+t3;
}else if (elmH<240) {
    elmB= t1; elmR= t2; elmG= t1-t3;
}else if (elmH<300) {
    elmB= t1; elmG= t2; elmR= t2+t3;
}else if (elmH<360) {
    elmR= t1; elmG= t2; elmB= t1-t3;
}else {
    elmR=0; elmG=0; elmB=0;
}
}

elmR=Math.floor(elmR);
elmG=Math.floor(elmG);
elmB=Math.floor(elmB);
clrRGB=
'#'+elmR.toString(16)+elmG.toString(16)+elmB.toString(16);
elmH= elmH+rate;
if (elmH>=360)
    elmH=0;

```

```
    return clrRGB;
}
```

编写自定义函数ChangeColor(), 将生成的十六进制的颜色应用在文字上, 代码如下。

```
function ChangeColor(tagname) {
    tagname.style.color = makeColor();
}
```

通过window对象的setInterval()方法每隔0.1秒调用一次ChangeColor()方法, 改变文字的颜色, 代码如下。

```
window.setInterval("ChangeColor(feature)", 100);
</script>
```

(2) 在页面中需要显示渐变文字动画的位置添加一个层, 命名为feature, 在该层中添加文字“《Visual Basic 编程词典》软件的特点如下:”, 并将该文字设置为粗体显示效果, 代码如下。

```
<div id="feature" class="shadow_3"><b>《Visual Basic 编程词典》软件的特点如下:</b></div>
```

举一反三

根据本实例, 读者可以:

为企业门户网站中的联系电话等信息设置动画效果;

将网站公告设置为动画效果。

## 5.2 改变文字大小

在网页开发过程中，通过对文字大小的改变，也可以使网页具有动态的效果。如果你想让网页中的标题具有动荡效果，那么需要对文字的大小进行改变。下面通过几个应用实例来介绍如何改变文字的大小。

### 实例144 文字伸缩

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\05\144

实例说明

在开发网页时，可以对一些文字进行伸缩效果，以突出其文字，从而使浏览者加以注意。运行结果如图 5.15、图 5.16所示，本实例是把“明日科技图书”字符串进行整体的放大和缩小。



明日科技图书

图5.15 文字缩小效果



明日科技图书

图5.16 文字放大效果

技术要点

本实例主要是用 innerHTML 属性在指定的层中添加一个用<span>标记和</span>标记设置的字符串，并用window对象的setTimeout()方法以一定时间对<span>标记和</span>标记中文字的字体大小进行修改。

<span>标记也是层，它与<div>标记之间的区别在于：不支持层的浏览器在<div>标签的前后放置额外的换行符；也就是说，<div>标签是块级别的元素，而<span>标签则是内联元素。

在<span>标记中的style-position属性十分重要，当将其设置成absolute时，可以改变层的位置，并且不会影响页面中其他的元素；如果没有设置style-position属性，当层的大小改变时，将改变层以下其他元素的位置。

#### 实现过程

(1) 在<body>标记上添加一个层，代码如下。

```
<span id="Tdiv" style='font-family:宋体;
position:absolute; font-size:1px;
color:#003399' ></span>
```

(2) 编写用于实现文字伸缩的JavaScript代码。

自定义函数magnify()，用于动态伸展文字。代码如下。

```
<script language="JavaScript">
var size=0;
var step=1;
varmaxsize=50;
var str="明日科技图书";
functionmagnify() {
    if (parseInt(Tdiv.style.fontSize)<0) {step=1; size=0}
    if (parseInt(Tdiv.style.fontSize)<maxsize) {
        Tdiv.innerHTML=str;
```

```

        step++;
        Tdiv.style.fontSize=parseInt(Tdiv.style.fontSize)+step;
    p;
    var timer=setTimeout("magnify()",50);
}
else {
    clearTimeout(timer);
    var intermezzo=setTimeout("reduce()",1000);
}
}

```

自定义函数reduce(), 用于动态缩小文字。代码如下。

```

function reduce() {
    if (parseInt(Tdiv.style.fontSize)>-0) {
        Tdiv.innerHTML=str;
        if (step>=2) {step=step-1} else {step=1}
        Tdiv.style.fontSize=parseInt(Tdiv.style.fontSize)-step;
        setTimeout("reduce()",50);
    }
    else setTimeout("magnify()",1000);
}
</script>

```

(3) 在<body>标记中用窗体载入事件 (onLoad) 调用自定义函数magnify(), 代码如下。

```
<body onLoad="magnify()">
```

举一反三

根据本实例, 读者可以:

实现图片自动缩放；  
实现文字在页面中走动。

## 实例145 动荡的文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\145

实例说明

在网页设计时经常需要标题文字进行动态显示效果。本实例将对标题的文字大小以正弦的形式进行改变，从而实现标题文字的动荡效果，运行结果如图5.17所示。



图5.17 动荡的文字

技术要点

本实例主要是用Math对象的sin()方法来计算出正弦值，并用Math对象的abs()方法将正弦值修改为绝对值后逐个改变文本中单个汉字的大小。Math对象的abs()方法的详细介绍请参见实例121。下面对sin()方法进行详细说明。

sin()方法的语法格式如下。

Math.sin(number)

参数说明如下。

● number：计算正弦值的数值表达式。

功能：返回数字的正弦值。

实现过程

(1) 在<body>标记上添加一个层，将align属性设置为center，使其居中。其代码如下。

```
<div align=center id=Tdiv></div>
```

(2) 编写用于实现文字动荡的JavaScript代码。

自定义函数wavy()，用于在字符串中按顺序取出单个汉字。其代码如下。

```
<SCRIPT language="JavaScript">
var theText= "明日科技图书";
functionwavy(n) {
    Cycle(theText,n);
    if (n> theText.length) {n=0}
    setTimeout("wavy("+n+1+")", 100);
}
```

自定义函数Cycle()，将文字按指定的大小及颜色进行显示。其代码如下。

```
functionCycle(text,dis) {
    str= "";
    for (i=0; i< text.length; i++){
        size=72*Math.abs(Math.sin((i+dis)/(text.length/3.14))
    ); //设置正弦值
        str+= "<font style='font-size:
"+size+"pt;color:RGB(100,0,"+(size+70)+")'>"
        +text.substring(i,i+1)+"</font>"; //设置字体的颜色及
大小
    }
    Tdiv.innerHTML= str; //通过innerHTML属性动态修改文字的颜色及大小
```

```
}  
</SCRIPT>
```

(3) 在<body>标记的载入事件中调用自定义函数wavy()。代码如下。

```
<body onload=wavy(0)>
```

举一反三

根据本实例，读者可以：

使文字具有霓虹灯效果；

随机改变文本中单个文字的大小。

## 实例146 文字伸展

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\146

实例说明

在网页设计时经常需要一些文本文字进行动态伸展的效果。本实例将扩大文本中的字间距，以实现文字的伸展效果，如图5.18、图5.19所示，文字正在伸展过程中。

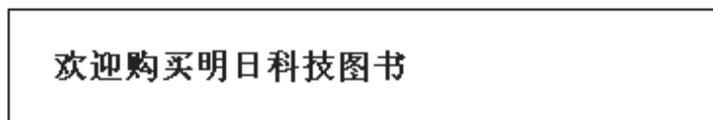


图5.18 文字显示效果



图5.19 文字伸展效果

技术要点

本实例主要是增加层（span）中style样式的letter-spacing属性的值，从而改变文本中文字的字间距，使文本具有动态的伸展效

果。

将伸展的文本放在<nobr>标记和</nobr>标记中，可以将超出层宽度的文本不换行显示。

### 实现过程

(1) 在<body>标记上添加一个层，将align属性设置为center，使其居中。其代码如下。

```
<div id="Tdiv" style="position:absolute; top:20px; left:20px;"></div>
```

(2) 编写用于实现文字伸展的JavaScript代码。

自定义函数extend\_min()，用于动态增加字间距。其代码如下。

```
<script language="JavaScript">
var
pcolor=newArray("000099","006699","0000CC","0066CC","0000FF",
"0033FF","00FFFF",
"3399FF","66CCFF");
var color_x=0;
var space=-8;
varTimer;
var textstr="欢迎购买明日科技图书";
function extend_min() {
    if (space<=1) {
        Tdiv.innerHTML="<span
style=' position:absolute;visibility:visible;
letter-spacing:"+space+"'><nobr><font size=5
face=Verdana color="+pcolor
[0]+"><b>&nbsp;"+textstr+"</b></font></nobr></span>";
        space=space+1;
```

```

        Timer=setTimeout("extend_min()",20);           //在指定
时间调用自定义函数extend_min()
    }
    else{
        clearTimeout(Timer);    //对setTimeout()方法进行清除
        showinfo();
    }
}

```

自定义函数extend\_min(), 用于清除setTimeout()方法, 并调用自定义函数extend\_max()。其代码如下。

```

function showinfo() {
    clearTimeout(Timer);
    Timer=setTimeout("extend_max()",2000);
}

```

自定义函数extend\_max(), 用于缩小字符串的字间距。其代码如下。

```

var step=1;
function extend_max() {
    if(space<=2000) {
        Tdiv.innerHTML="<span
style='position:absolute;visibility:visible;
letter-spacing:"+space+"'><nobr><font size=5
face=Verdana color="+pcolor
[color_x]+"><b>&nbsp;">+textstr+"</b></font></nobr>
</span>";    //改变字间距
        color_x++;
        if (color_x>=8) {color_x=8}
    }
}

```

```

    step++;
    space=space+step;
    Timer=setTimeout("extend_max()",30);
}
else{
    clearTimeout(Timer);
    color_x=0;
    space=-8;
    extend_min();
}
}

```

在窗体载入时调用自定义函数extend\_min()。其代码如下。

```
extend_min();
```

```
</script>
```

举一反三

根据本实例，读者可以：

使文字纵向伸展；

在文字缩放时不换行显示。

## 实例147 文字逐个放大

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\05\147

实例说明

为了让网页中的页面既有动态的效果又可以节省页面空间，可以将网页右上角或左上角标题以单个字符的形式进行放大显示。图

5.20、图5.21所示为正在放大的单个文字。



图5.20 文字没有放大前



图5.21 文字放大后

### 技术要点

本实例主要应用了 `eval()` 函数对 JavaScript 代码进行执行，下面将对 `eval()` 函数进行详细说明。

`eval()` 函数的语法格式如下。

`eval(codeString)`

参数说明如下。

● **codeString**: 必选项，`codeString` 参数是包含有效 JavaScript 代码的字符串值。这个字符串将由 JavaScript 分析器进行分析和执行。

功能：检查JavaScript代码并执行。

`eval`函数允许JavaScript源代码的动态执行。例如，下面的代码创建了一个包含Date对象的新变量mydate。

```
eval("var mydate = new Date();");
```

### 实现过程

(1) 在页面中添加一个层，将align属性设置为center，使层中的文字居中，代码如下。

```
<div align=center id=Tdiv style="POSITION: absolute; TOP: 10px"></div>
```

(2) 编写用于实现文字逐个放大的JavaScript代码。

自定义函数`zoom()`，用于逐个显示字符串中的文字，并动态改变字体的大小。其代码如下。

```

<SCRIPT>
text=new Array('明','日','科','技','图','书');
color=new
Array('#3300FF','#3366FF','#AAAAFF','#CDDDFC','#ffffff');
var maxSize=70;
var size=10;
var n=0;
function zoom(num,fn){
    if (size<maxSize){
        eval('document.all.Tdiv.innerText=text[num]');//执行
JavaScript的代码
        eval('document.all.Tdiv.style.fontSize=size');
        size+=5;
        setTimeout("zoom("+num+", '"+fn+"' )", 30);
    }else eval(fn);
}

```

自定义函数fadeIt(), 用于将最后一个文字隐藏。其代码如下。

```

function fadeIt(num){
    if (num<color.length){
        eval("document.all.Tdiv.style.color=color[num]");
        num+=1;
        setTimeout("fadeIt("+num+")", 200);
    }else document.all.Tdiv.style.visibility="hidden";
}

```

自定义函数init(), 设置Tdiv层的前景色, 并调用自定义函数change()。其代码如下。

```

function init(){

```

```
document.all.Tdiv.style.color=color[0];  
change(0);  
}
```

自定义函数change(), 用于调用自定义函数zoom()和 fadeIt()。其代码如下。

```
function change(num) {  
    n+=1;  
    size=10;  
    if(num<text.length) {zoom(num, 'change(' +n+')');}  
    else fadeIt(0);  
}  
</SCRIPT>
```

(3) 在<body>标记的载入事件 (onload) 中, 调用自定义函数init()。其代码如下。

```
<body onload=init()>
```

举一反三

根据本实例, 读者可以:

实现动态显示不同的图片;

实现对图片进行缩放。

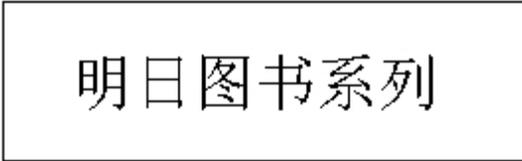
## 实例148 自动改变大小

这是一个实用的实例

实例位置: 光盘\mingrisoft\05\148

实例说明

在网页设计时, 可以将标题居中于页面, 并以动态的形式对标题进行缩放显示。运行结果如图5.22、图5.23所示。



明日图书系列

图5.22 文字缩小效果



明日图书系列

图5.23 文字放大效果

### 技术要点

本实例主要应用document对象的write()方法在页面的顶部动态显示一个层，并用<center>标记和</center>标记在层中显示文本，其作用是将文本居中显示。再通过window对象的setTimeout()方法对层的style样式的fontSize属性值进行递增或递减，以实现文本的自动缩放效果。

### 实现过程

编写用于实现文字动态放大的JavaScript代码。

(1) 自定义函数upwords()，对字体的大小进行增加。其代码如下。

```
<SCRIPT language="JavaScript">
var speed=20;
var cycledelay=1000;
var maxsize=48;
var x=y=0;
var themessage, size;
var esize= "";
var a=newArray();
a[0]="欢迎购买";
a[1]="明日图书系列";
```

```

a[2]="明日图书将是你成功的秘诀";
document.write(' <div id="div1"></div><br>');
functionupwords () {
    themessage= a[y];
    if (x<maxsize) {
        x++;
        setTimeout("upwords()", speed);
    }
    else setTimeout("downwords()", cycledelay);
    div1.innerHTML= "<center>" +themessage+ "</center>";
    div1.style.fontSize=x;
}

```

(2) 自定义函数downwords ()，对字体的大小进行减小。其代码如下。

```

functiondownwords () {
    if (x>1) {
        x--;
        setTimeout("downwords()", speed);
    }
    else {
        setTimeout("upwords()", cycledelay);
        y++;
        if (y> a.length-1) y=0;
    }
    div1.innerHTML= "<center>" +themessage+ "</center>";
    div1.style.fontSize=x;
}

```

(3) 在窗体载入时，调用自定义函数upwords()。其代码如下。

```
upwords();
```

```
</SCRIPT>
```

举一反三

根据本实例，读者可以：

实现图片动态居中显示；

实现文字的颜色渐变。

## 实例149 选择字体的大小

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\05\149

实例说明

在一些留言性的网站中，通常都可以在留言板中设置字体的大小，然后在留言板中进行留言。本实例将对文本框中文字的大小进行动态修改。运行结果如图5.24所示。



图5.24 选择字体的大小

技术要点

本实例主要是通过menu组件的options[index].value属性来获取当前所选对象的值，并对字体大小进行修改。menu组件的selectedIndex属性是选项列表中已选中的<option>的索引（位置），

并用options数组来获取选择框中的所有选项（<option>）的一个列表。下面对options数组的相关属性进行详细说明。

options数组的相关属性如表5.1所示。

表5.1 options数组的相关属性

属 性	描 述
defaultSelected	选项列表中的默认选项
index	选项列表中某选项的索引位置
length	选项列表中的选项数（<options>）
name	选项列表的名字（name 特性）
selected	表示选项列表中某选项<option>是否被选中的一个布尔类型值
selectedIndex	选项列表中已选中的<option>的索引(位置)
text	选项列表中<option>标记后的文本
value	选项列表中的 value=特性

### 实现过程

(1) 在页面中添加multi line 和menu 组件。代码如下。

```
<select id='select1' onchange='Gfont()'><option selected>  
选择字体大小  
</select>  
<p></p>
```

```
<textarea name="textarea" style="width:400px;  
height:100px " id='text1'>
```

这里的字体应该随选择变化

```
</textarea>
```

(2) 编写用于实现选择字体大小的JavaScript代码。

向menu组件中动态添加信息。其代码如下。

```
<script language="JavaScript">  
var Text=textarea.value;  
for (i=1;i<192;i++){
```

```
eval("select1.options[i]=new Option "+"(' "+(i+8)+"pt字  
体', ' "+(i+8)+"' )");  
}
```

自定义函数Gfont(), 用于改变multi line 组件的字体大小。其代码如下。

```
function Gfont() {  
    if((select1.selectedIndex!=-1)&&  
(select1.selectedIndex!=0)) {  
        text1.style.fontSize=select1.options[select1.selected  
Index].value;  
    }  
}  
</script>
```

举一反三

根据本实例，读者可以：

实现动态改变字体样式；

实现制作一个多级的级联菜单。

## 5.3 文字显示效果

文字的显示效果其实就是对文字进行一些特殊效果的显示，如文字的渐隐渐现、虚幻抖动等。下面将通过几个典型实例进行详细讲解。

### 实例150 文字渐隐渐现

本实例可以美化界面

实例位置：光盘\mingrisoft\05\150

实例说明

本实例是页面载入后，将文字以渐隐渐现的形式在页面中进行显示。运行结果如图5.25、图5.26所示。



图5.25 文字隐藏状态

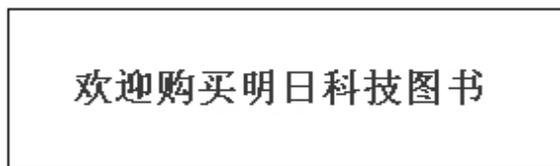


图5.26 文字显示状态

技术要点

本实例主要是通过修改表格的style样式的filter滤镜的alpha效果的opacity属性值来对单元格中的文字进行渐隐渐现的显示，透明度的取值范围为0~100。

当用如下代码修改透明度时，在创建表格时不必设置透明属性，代码如下。

```
document.all.aa.style.filter="alpha(opacity="+b+")";
```

也可以用如下代码对表格的透明度进行设置，但必须在创建表格时对其透明度设置默认值，代码如下。

```
<table id="aa" style="filter:alpha(opacity=10) "> //在  
创建表格时设置其透明度
```

```
document.all.aa.filters.alpha.opacity=b; //改变  
表格透明度的值
```

实现过程

(1) 在页面中添加一个一行一列的表格，并在单元格中添加文字，代码如下。

```
<table id="aa"width="30%"height="30"  
border="0"cellpadding="0"cellspacing="0">  
  <tr>  
    <th align="center" scope="col">欢迎购买明日科技图书  
  </th>  
</tr>  
</table>
```

(2) 编写用于实现文字的渐隐渐现的JavaScript代码如下。

```
<script language="JavaScript">  
var b = 1;  
var c= true;  
function latent() {  
  if(c== true)b++;  
  elseb--;  
  if(b==100) {
```

```
        b--;
        c= false;
    }
    if(b==10) {
        b++;
        c= true;
    }

    document.all.aa.style.filter="alpha(opacity="+b+)"; /
/修改表格的透明度
    setTimeout("latent()",50);
}
```

(3) 在窗体载入时，调用自定义函数latent()。代码如下。

```
latent();
</script>
```

举一反三

根据本实例，读者可以：

使图片渐隐渐现；

使两个图片叠加在一起。

## 实例151 文字虚幻变化

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\05\151

实例说明

在网页开发过程中，可以使文字以虚无的状态在页面中行走，本实例是将层以半透明的形式在页面中以指定的位置向上移动，当到顶端时，将层不透明显示。运行结果如图 5.27所示。



图5.27 文字虚幻变化

### 技术要点

主要应用到了 style样式的 filter滤镜的 alpha效果的 opacity属性。具体用法请参见实例150。

本实例的实现过程是在一个层中让另外两个层进行交互式的向上移动，当层移动到顶端时，将opacity属性设为100，当有任何一个层进行移动时，将里面的两个层进行半透明显示。

### 实现过程

编写用于实现文字虚幻变化的JavaScript代码。

(1) 将要显示的文字保存在一维数组 Sarry中，并在页面中添加3个层，其中一个层包含另外2个层。代码如下。

```
<script language="JavaScript">
var s_width=150;
var s_height=115;
var bgcolor='#E0EFD1';
var pause=3000;
varSarry=newArray()
Sarry[0]='欢迎光临 <ahref="http://www.mingrisoft.com/">明日科技主页</a>,您一定会
有许多收获!';
Sarry[1]='谢谢光临 <ahref="http://www.mingrisoft.com/">明日科技主页</a>,您一定会
有许多收获!';
```

```

    Sarry[2]='下次一定还要光临
<a href="http://www.mingrisoft.com/">明日科技主页</a>,
    您一定会有更多的收获!';
    document.write('<div
type="position:relative;width:'+s_width+';height:'+s_height+'
;
    overflow:hidden"><div id="div0"
style="position:absolute;background-color:'
    +bgcolor+';width:'+s_width+';height:'+s_height+';top:'+s_
height+';filter:alpha
    (opacity=20);-moz-opacity:0.2;"></div><div id="div1"
style="position:absolute;
    background-
color:'+bgcolor+';width:'+s_width+';height:'+s_height+';top:'
+s_height
    +';filter:alpha(opacity=20);-moz-opacity:0.2;"></div>
</div>');
    var curpos=s_height;
    var degree=10;
    var divis="div0";
    var curindex=0;
    var nextindex=1;
    (2) 自定义函数moveslide(), 使当前层向上移动。代码如下。
functionmoveslide() {
    if (curpos>0) {
        tempobj.style.top=curpos=Math.max(curpos-degree, 0);
    }
}

```

```

else{
    clearInterval(dropslide);
    if (crossobj.filters)
crossobj.filters.alpha.opacity=100;
    else if (crossobj.style.MozOpacity)
crossobj.style.MozOpacity=1;
    divis=(divis=="div0")? "div0" : "div1";
    tempobj=eval("document.all."+divis);
    tempobj.innerHTML=Sarry[curindex];
    nextindex=(nextindex<Sarry.length-1)?nextindex+1 :0;
    setTimeout("rotateslide()", pause);
}
}

```

(3) 自定义函数rotateslide(), 使当前层为半透明状态, 并获取一个字符串。代码如下。

```

function rotateslide() {
    resetit(divis); //调用自定义函数
    crossobj=tempobj=eval("document.all."+divis);
    crossobj.style.zIndex++;
    document.all.div0.filters.alpha.opacity=document.all.div1.filters.alpha.opacity=20;
    var temp='setInterval("moveslide()", 50)';
    dropslide=eval(temp);
    divis=(divis=="div0")? "div1" : "div0";
    curindex=(curindex<Sarry.length-1)? curindex+1 :0;
}

```

(4) 自定义函数resetit(), 用于显示层的上下位置。代码如下。

```
function resetit(what) {  
    curpos=s_height;  
    var crossobj=eval("document.all."+what);  
    crossobj.style.top=curpos=s_height;  
}
```

(5) 自定义函数startit(), 在页面的div0层中显示Sarry数组0元素中的字符串。代码如下。

```
function startit() {  
    crossobj=eval("document.all."+divis);  
    crossobj.innerHTML=Sarry[curindex];  
    rotateslide();  
}
```

(6) 在窗体载入时, 调用自定义函数startit()。代码如下。

```
window.onload=startit;  
</script>
```

举一反三

根据本实例, 读者可以:

实现将某企业的所有员工图片信息输出到浏览器中;

实现将个人写真集图片连接输出到浏览器。

## [实例152 文字虚幻抖动](#)

这是一个提高基本技能的实例

实例位置: 光盘\mingrisoft\05\152

实例说明

在浏览一些特殊的主页时，有些标题性文字的后面，会出现该标题的虚文字，并在标题文字的背面以一定的范围进行随机显示。运行结果如图5.28所示。



图5.28 文字虚幻抖动

### 技术要点

本实例主要是用 `Math` 对象的 `random()` 方法来获取一个随机数，并按指定的时间对下面层的 `style` 样式的 `top` 属性和 `left` 属性进行修改，以实现其抖动的效果。`random()` 方法的返回值在 `0~1` 之间，不包含 `1`。

在获取层的当前位置时，其值为 `string` 型，必须用 `parseInt()` 方法将其转换成 `int` 型，这样才可以对层的位置进行移动。下面对 `parseInt()` 方法进行详细说明。

`parseInt()` 方法的语法格式如下。

```
parseInt(numString, [radix])
```

参数说明如下。

- `numString`: 必选项。要转换为数字的字符串。
- `radix`: 可选项。在 `2` 和 `36` 之间的表示 `numString` 所保存数字的进制值。如果没有提供，则前缀为“`0x`”的字符串被当作十六进制，前缀为“`0`”的字符串被当作八进制，其他字符串都被当作是十进制。

功能：返回由字符串转换得到的整数。

`parseInt()` 方法的相关用法如下。

```
parseInt("JavaScript");//返回NaN
```

```
parseInt("12cc");//返回为整数值12
```

实现过程

(1) 设置页面的背景颜色为黑色。

(2) 在页面中添加2个相同的层，将字体颜色分别设置为gray和white，并且颜色为white的层在颜色为gray的层的上面，代码如下。

```
<DIV id=Title2 style="COLOR:gray;FONT:bold11pt
arial;POSITION: absolute;
visibility:hidden; top:300; left:300;"><P><FONT size=8>特
效JavaScript<br>明日
科技图书</FONT></P></DIV>
<DIV id=Title1 style="COLOR:white;FONT:bold 11pt
arial;POSITION: absolute;
visibility:hidden; top:300; left:300"><P><FONT size=8>特
效JavaScript<br>明日科
技图书</FONT></P></DIV>
```

(3) 编写用于实现文字虚幻抖动的JavaScript代码。

```
<script language="JavaScript">
varTle2="document.all.Title2.style";
varTle1="document.all.Title1.style";
//将层放在页面的中间部分
eval(Tle2).top=eval(Tle1).top=document.body.clientHeight/
2-20;
eval(Tle2).left=eval(Tle1).left=document.body.clientWidth
/2-150;
functionFade() {
    varBackX=Math.random()*10;    //获取层的移动位置
    varBackY=Math.random()*5;
```

```

    if(Math.random()<0.5){
        BackX=-BackX;
        BackY=-BackY;
    }
    eval(T1e1).visibility= eval(T1e2).visibility=
"visible";    //显示层
    eval(T1e2).left=parseInt(eval(T1e1).left)+BackX;//移动
下面的层
    eval(T1e2).top=parseInt(eval(T1e1).top) +BackY;
    varFadeID= setTimeout("Fade()",50);
}

```

(4) 在窗体载入时，调用自定义函数Fade()。代码如下。

```

Fade();
</script>

```

举一反三

根据本实例，读者可以：

实现文字的振动效果；

使文字在页面中走动。

## 5.4 指定文字位置

在设计页面时，有时会将文字固定在页面工作区的某一处，即使移动工作区的滚动条也不会改变其文字的位置。下面将通过几个典型实例进行详细讲解。

### 实例153 将文字置于工作区左上角

这是一个实用的实例

实例位置：光盘\mingrisoft\05\153

实例说明

在浏览网页时，经常会看到一些关键性的文字总是位于页面工作区的左上角，本实例将实现这一功能。运行结果如图5.29所示。

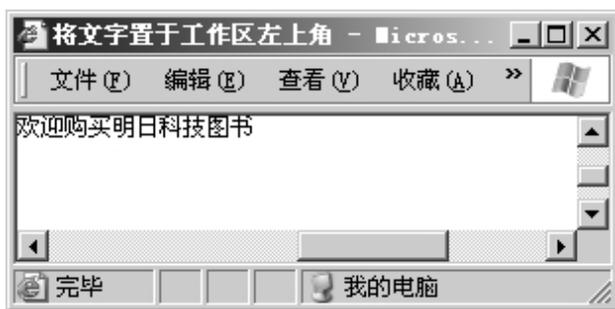


图5.29 将文字置于工作区左上角

技术要点

本实例主要运用了document对象中的<body>对象中的scrollTop、scrollLeft属性来获取页面中的纵向滚动条与横向滚动条被卷去的部分。

在使用scrollTop、scrollLeft属性时，应将页面中最上面的两行删除，删除代码如下。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
```

注意：如果不删除这两行代码，则无法获取滚动条被卷去部分的值。

实现过程

(1) 在页面中添加一个层，代码如下。

```
<DIV id=Tdiv style="HEIGHT: 45px; LEFT: 0px; POSITION:  
absolute; TOP: 0px; WIDTH: 463px;
```

```
Z-INDEX: 25"><FONT color=#000000 face="宋体">欢迎购买明日  
科技图书</FONT></DIV>
```

(2) 编写用于实现将文字置于工作区左上角的JavaScript代码。

```
<SCRIPT language=JavaScript>
```

```
functionpermute(tfloor) {
```

```
    RealTop=parseInt(document.body.scrollTop);
```

```
    document.all[tfloor].style.top=RealTop;
```

```
    RealLeft=parseInt(document.body.scrollLeft);
```

```
    document.all[tfloor].style.left=RealLeft;
```

```
}
```

```
setInterval('permute("Tdiv")', 50);
```

```
</SCRIPT>
```

举一反三

根据本实例，读者可以：

制作一个置于顶端的广告图片；

实现文字随滚动条的滚动而滚动。

## 实例154 右上角文字

本实例是一个应用性较强的实例

实例位置：光盘\mingrisoft\05\154

实例说明

本实例是将动态旋转的字符串文字总置于页面工作区的右上角，即使移动纵向或横向滚动条，旋转的文字仍然位于右上角。运行结果如图5.30所示。



图5.30 右上角文字

技术要点

本实例主要用`split()`方法将一个字符串分割为子字符串，然后再用`Math`对象的`sin()`和`cos()`方法将字符串中的文字在页面中画一个半圆，使其动态旋转。下面对`split()`方法进行详细说明。

`split()`方法的语法格式如下。

```
stringObj.split([separator[, limit]])
```

参数说明如下。

- `stringObj`：必选项，要被分解的`string`对象或文字。该对象不会被`split()`方法修改。

- `separator`：可选项，字符串或正则表达式对象。它标识了分隔字符串时使用的是一个还是多个字符。如果忽略该选项，返回包含整个字符串的单一元素数组。

- `limit`：可选项。该值用来限制返回数组中的元素个数。

功能：将一个字符串分割为子字符串，然后将结果作为字符串数组返回。

document对象中的body对象中的scrollTop、scrollLeft属性已经在实例153进行了说明。

### 实现过程

编写用于实现在工作区的右上角显示动态文字的过程。其主要程序如下。

(1) 对文字的字体、颜色进行设置，并将字符串中的文字添加到层中。其代码如下。

```
<SCRIPT language="JavaScript">
textstr="明日科技图书";
TColor="#AA92C7";
TFont="楷书";
textstr=textstr.split("");
n=textstr.length;
mass=360/n;
cy=cx=sb=0;
yb=50;    //设置半圆的高
xb=60;    //设置半圆的宽
sa=0.1;   //设置旋转的速度
pa=newArray();
pb=newArray();
for (i=0; i<n; i++)document.write(' <div id="logo"
style="position:absolute;
top:0;left:0;'+height:30;width:30;font-
family:' +TFont+' ;text-align:center;color:'
+TColor+' ">' +textstr[i]+' </div>');
```

(2) 自定义函数 sport(), 将字符串以半圆的形式显示在页面的右上角，并进行动态显示。其代码如下。

```

function sport() {
    //设置文本的显示位置
    cy=document.body.scrollTop+60;
    cx=document.body.scrollLeft+window.document.body.clientWidth-100;
    for (i=0; i<n; i++) {
        //设置单个文字的位置
        logo[i].style.top=cy+yb*Math.sin(sb+i*mass*Math.PI/180);
        logo[i].style.left=cx+xb*Math.cos(sb+i*mass*Math.PI/180);
        pb[i]=logo[i].style.pixelTop-cy;
        pa[i]=pb[i]-pb[i]*2;
        if (pa[i]<1) {
            pa[i]=0;
            logo[i].style.visibility='hidden';
        }
        else logo[i].style.visibility='visible';
        logo[i].style.fontSize=pa[i]/1.7;
    }
    sb-=sa;
    setTimeout('sport()', 100);
}

```

(3) 在窗体载入时，调用自定义函数sport()。代码如下。

```

sport();
</SCRIPT>

```

举一反三

根据本实例，读者可以：  
将文本在页面中以圆形进行旋转；  
将文字或图片置于工作区的右下角。

## 实例155 鼠标移动文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\155

实例说明

本实例是通过鼠标对页面中的文字进行控制，当鼠标移动到文字上方时，此时再拖动鼠标可以使工作区中的文字随鼠标移动。运行结果如图5.31所示。



图5.31 随鼠标移动的文字

技术要点

本实例主要是通过 document 对象的 OnMouseDown（鼠标按下）和 OnMouseMove（鼠标移动）事件来对文字进行移动控制，并用 event 对象的 clientX 和 clientY 属性来获取鼠标在当前工作区中的位置。在自定义函数 move() 中，event 对象的 button 属性是用来判断当前鼠标是否为左键按下状态。

event 对象的 button 属性的取值如表 5.2 所示。

表5.2 event对象的button属性的取值

可 选 值	说 明	可 选 值	说 明
0	没按键	4	按中间键
1	按左键	5	按左键和中间键
2	按右键	6	按右键和中间键
3	按左右键	7	按所有鼠标键

### 实现过程

编写用于实现鼠标移动文字的JavaScript代码。

(1) 自定义函数move(), 当在文字上单击鼠标左键时, 可以移动文字的位置。代码如下。

```
<script language="JavaScript">
var drag=false
var z, x, y
functionmove() {
    if (event.button==drag) {
        z.style.pixelLeft=temporarily1+event.clientX-x;
        z.style.pixelTop=temporarily2+event.clientY-y;
        return false;
    }
}
```

(2) 自定义函数grommet(), 当单击鼠标左键时, 获取文字的前位置。代码如下。

```
function grommet() {
    if (event.srcElement.className=="drag") {
        drag=true;
        z=event.srcElement;
        temporarily1=z.style.pixelLeft;
        temporarily2=z.style.pixelTop;
```

```
x=event.clientX;
y=event.clientY;
document.onmousemove=move;
}
}
```

(3) 在页面中单击鼠标左键时，调用自定义函数grommet()。其代码如下。

```
document.onmousedown=grommet;
</script>
```

举一反三

根据本实例，读者可以：

实现文字跟随鼠标的移动而移动；

实现随意改变图片的位置。

## 5.5 文字动态移动

在浏览网页时，经常会遇到在网页中自动滚动的文本，这样可以使浏览者根据滚动的速度进行阅读，而无需拖动滚动条。下面将通过几个典型实例进行详细讲解。

### 实例156 文字自动滚屏

本实例可提高工作效率

实例位置：光盘\mingrisoft\05\156

实例说明

在一些动态网页中，经常会看到一些文字自动进行滚屏，使浏览者在观看时不用做任何操作，本实例将实现这一过程。运行结果如图5.32所示。在打开网页后，文字将在文本框内自动循环向上移动。

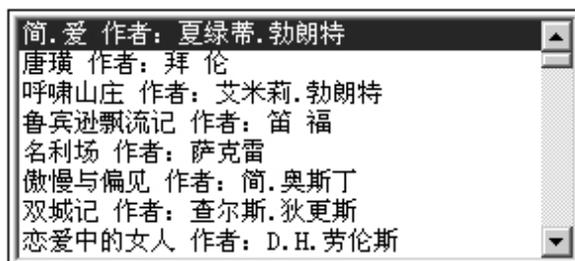


图5.32 文字自动滚屏

技术要点

本实例主要是将列表框中最上面的对象进行清空，使下一行移动到最上面，并利用列表框的selectedIndex属性将最上面的对象设为选中状态，并利用window对象的setTimeout()方法对自定义函数scroll()进行调用，以达到滚动的效果。

window对象的setTimeout()方法在实例002中进行了详细的说明。

## 实现过程

(1) 在页面中添加一个list组件，并输入相关的文本信息，代码如下。

```
<form name="form1">
<select name="Text1" size=8>
<option>
<option>世界名著及作者
.....
/*更多的文本信息*/
...
</select><option>浅滩迷船 作者：里縠
</select>
```

(2) 编写用于实现文字在列表框中自动向上滚屏的JavaScript代码。

```
<SCRIPT LANGUAGE="JavaScript">
function screenscroll() {
    var optlist = document.form1.Text1;
    optlist[0] = null;
    optlist.selectedIndex = 0;
    setTimeout("screenscroll()", 2000);
}
</script>
```

(3) 在<body>标记的载入事件（onLoad）中调用自定义函数screenscroll()。代码如下。

```
<body onLoad="screenscroll();">
```

举一反三

根据本实例，读者可以：

使浏览器背景自动滚屏；  
使文字在页面中循环滚动。

## 实例157 文字上下滚动

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\05\157

实例说明

在一些阅读性的网站中，为了使浏览者更便于阅读，页面中的文字将自动进行滚动。本实例在页面打开后，页面中的文字将在一定的范围内进行上下滚动，运行结果如图5.33所示。

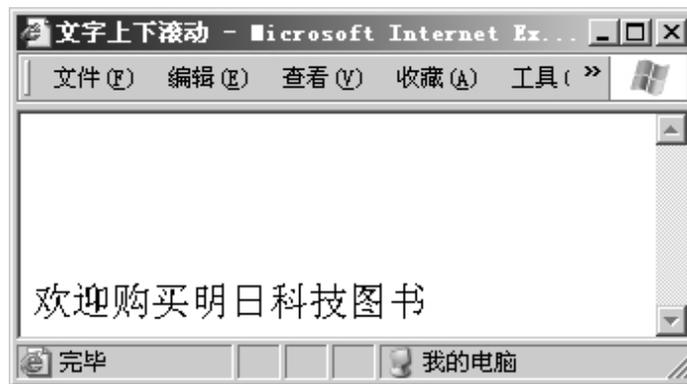


图5.33 文字上下滚动

技术要点

本实例主要是用document对象中的<body>对象的clientHeight属性来获取工作区的高，并设置文字移动的范围，并用window对象的setTimeout()方法使文字在指定的区域内进行上下移动。

注意：一定要将层（div）的style样式的position属性设为absolute，否则，层将不会移动。

实现过程

(1) 在<body>标记中添加一个层，代码如下。

```
<div id="TDiv" style="position: absolute;top: 50; color: #000000;font-size:20px;">
```

```
<p>欢迎购买明日科技图书</p></div>
```

(2) 编辑用于实现文字上下滚动的JavaScript代码。

自定义函数，用于设置文字所显示的位置，并调用自定义函数act()。代码如下。

```
<script language="JavaScript">
down=true;
function activity() {
    ob=document.all("TDiv");
    if (ob.style.posTop<=50) {
        varhei=document.body.clientHeight;
        act(50,hei-100,50)
    }
}
```

自定义函数act()，使文字在页面中进行上下滚动。代码如下。

```
function act(yp,yk,yx) {
    ob=document.all("TDiv");
    ob.style.posTop=yp;
    if (yp<=yx) down=true;
    if (yp>=yk) down=false;
    if (down) {step=4;}
    else {step=-4;}
    setTimeout('act('+yp+step+')','+yk+', '+yx+')', 35);
}
```

在窗体载入时，调用自定义函数activity()。代码如下。

```
activity();
```

</script>

举一反三

根据本实例，读者可以：

制作一个自动移动的广告；

制作一个左右移动的标题栏。

## 5.6 文字立体效果

在许多的网页中，文字被设为立体状态，并进行动态显示，这样可以使页面看上去更有空间感。下面将通过几个典型实例进行详细讲解。

### 实例158 文字的旋转

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\05\158

实例说明

本实例将一个字符串以单个字符的形式排成一个圆形，并以动态的形式进行旋转，在旋转的同时，由近到远改变字符的大小及颜色。运行结果如图5.34所示。



图5.34 文字的旋转

技术要点

本实例主要应用string对象的charAt()方法来获取字符串中的单个字符，并用Math对象的cos()方法和sin()方法来设置字符的大小及位置。下面对charAt()方法进行详细的说明。

charAt()方法的语法格式如下。

```
strObj.charAt(index)
```

参数说明如下。

- strObj: 必选项, 任意string 对象或文字。
- index: 必选项, 字符基于0 的索引值, 有效值是0 与字符串长度减1 之间的值。

功能: 返回指定索引位置处的字符。

实现过程

编写用于实现文字旋转的JavaScript代码。

(1) 用于将字符串的单个文字以层的形式显示, 并对文字的大小、间距及位置进行设置。代码如下。

```
<script language="JavaScript">
Phrase="欢迎购买明日科技图书";
Balises="";
Taille=40; //字体大小
Midx=150; //文字显示的纵向位置
Decal=0.5;//字间距
Nb=Phrase.length; //保存文字的个数
y=-10000;
for (x=0;x<Nb;x++) {
    Balises=Balises+' <DIV Id=L'+x+' STYLE="width:3;font-
family:CourierNew;
font-weight:bold;position:absolute;top:160;left:70;z-
index:0">' +Phrase.charAt(x)
+' </DIV>'
}
document.write(Balises);
Alpha=5;
I_Alpha=0.05;
```

(2) 自定义函数Alors(), 用于将文字以指定的位置、大小及颜色进行显示。代码如下。

```
functionAlors() {  
    Alpha=Alpha-I_Alpha;  
    for (x=0;x<Nb;x++) {  
        Alpha1=Alpha+Decal*x;  
        Cosine=Math.cos(Alpha1);  
        Ob=document.all("L"+x);  
        Ob.style.posLeft=Midx+100*Math.sin(Alpha1)+50;  
        Ob.style.fontSize=Taille+30*Cosine;  
        Ob.style.color="rgb("+ (27+Cosine*80+50) +", "+  
        (127+Cosine*80+50) +", 0)";  
    }  
}
```

(3) 在窗体载入时, 以指定的时间间隔, 重复调用自定义函数Alors(), 以实现旋转效果。

```
Time>window.setInterval("Alors()", 50);  
</script>
```

举一反三

根据本实例, 读者可以:

在网页中, 制作一个旋转的动态广告;

在网页中, 制作一个浮动的旋转文本。

## 实例159 立体旋转文字

这是一个可以启发思维的实例

实例位置: 光盘\mingrisoft\05\159

## 实例说明

目前，Internet上的很多网站都会用立体的动态文字作为标题或广告语，这样可以使页面更具有可观赏性，给浏览者以动态的空间感。运行结果如图5.35所示。

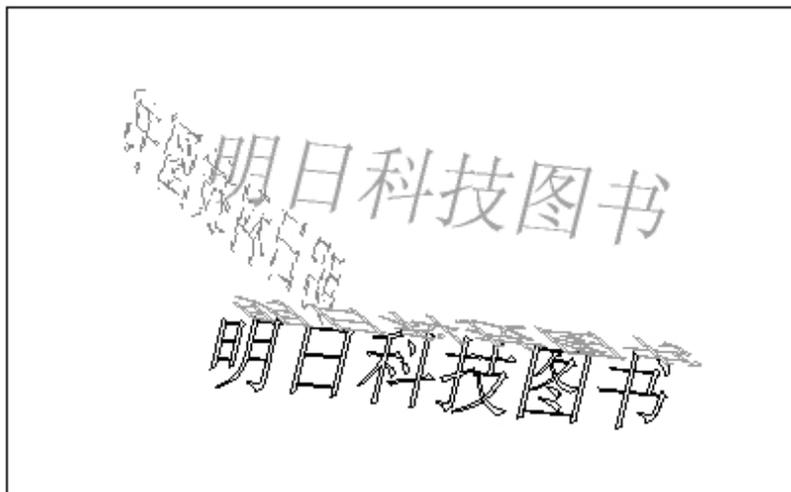


图5.35 立体旋转文字

## 技术要点

本实例主要应用到了<object>标记和<param>标记。<object>标记里的classid属性是告诉浏览器插件的类型，并用<object>标记的scale()和rotate()方法对其进行缩放和旋转。用<object>标记在页面中插入ActiveX控件或其他对象之后，有时候需要向该对象或者控件传递参数，这就要使用<param>标记，该标记没有相应的结束标志</param>，并且该标记仅在<object>标记中有效，并用<object>标记的scale()方法和rotate()方法对其进行旋转。下面将对<param>标记和<object>标记的scale()方法和rotate()方法进行详细说明。

<param>标记的一般格式如下。

```
<param name=* value=* valuetype=* type=*>
```

参数说明如下。

- name: 是参数的名字。
- value: 指定参数的值。

- **valuetype**: 指定表示参数的值。
- **type**: 指定媒体类型。

scale的语法格式如下。

scale(x-scale, y-scale, z-scale)

参数说明如下。

- **x-scale**: x 轴。
- **y-scale**: y 轴。
- **z-scale**: z 轴。

功能: 对ActiveX控件进行缩放。

rotate的语法格式:

rotate(x-rotation, y-rotation, z-rotation)

参数说明如下。

- **x-rotation**: x 轴。
- **y-rotation**: y 轴。
- **z-rotation**: z 轴。

功能: 对ActiveX控件进行旋转。

实现过程

(1) 在页面中按指定位置及大小添加4个Flash组件, 代码如下。

```
<OBJECT classid=CLSID:369303C2-D7AC-11D0-89D5-00A0C90833E6 height=18 id=SG4  
style="HEIGHT: 450px; LEFT: 25%; POSITION: absolute; TOP:  
64px; WIDTH: 450px; ZINDEX: 3" width=14><PARAMNAME="Line0001"  
VALUE="SetLineStyle(0)"><PARAM NAME="Line0002"  
VALUE="SetFillColor(164, 164, 255)"><PARAM NAME="Line0003"  
VALUE="SetFillStyle(1)"><PARAM NAME="Line0004"  
VALUE="SetFont('Arial', 100, 150, 0, 0, 0)"><PARAM
```

```
NAME="Line0005" VALUE="Text('明日科技图书',-95, 87)">
</OBJECT>
  <OBJECT classid=CLSID:369303C2-D7AC-11D0-89D5-
00A0C90833E6 height=18 id=SG1
  style="HEIGHT: 450px; LEFT: 25%; POSITION: absolute; TOP:
25%; WIDTH: 450px; ZINDEX: 0" width=14><PARAMNAME="Line0001"
VALUE="SetLineStyle(1)"><PARAM NAME="Line0002"
VALUE="SetLineColor(0, 0, 0)"><PARAM NAME="Line0003"
VALUE="SetFillStyle(0)"><PARAM NAME="Line0004"
VALUE="SetFont('Arial', 100, 150, 0, 0, 0)"><PARAM
NAME="Line0005" VALUE="Text('明日科技图书',-95, 87)">
</OBJECT>
  <OBJECT classid=CLSID:369303C2-D7AC-11D0-89D5-
00A0C90833E6 height=18 id=SG3
  style="HEIGHT: 350px; LEFT: 25%; POSITION: absolute; TOP:
25%; WIDTH: 450px; ZINDEX: 1" width=14><PARAMNAME="Line0001"
VALUE="SetLineStyle(2)"><PARAM NAME="Line0002"
VALUE="SetLineColor(66,158, 144)"><PARAM NAME="Line0003"
VALUE="SetFillStyle(0)"><PARAM NAME="Line0004"
VALUE="SetFont('Arial', 100, 150, 0, 0, 0)"><PARAM
NAME="Line0005" VALUE="Text('明日科技图书',-95, 87)">
</OBJECT>
  <OBJECT classid=CLSID:369303C2-D7AC-11D0-89D5-
00A0C90833E6 height=18 id=SG2
  style="HEIGHT: 450px; LEFT: 25%; POSITION: absolute; TOP:
25%; WIDTH: 450px; ZINDEX: 2" width=14><PARAMNAME="Line0001"
VALUE="SetLineStyle(3)"><PARAM NAME="Line0002"
```

```
VALUE="SetLineColor(255, 111, 210)"><PARAM NAME="Line0003"  
VALUE="SetFillStyle(0)"><PARAM NAME="Line0004"  
VALUE="SetFont('Arial', 100, 150, 0, 0, 0)"><PARAM  
NAME="Line0005" VALUE="Text('明日科技图书', -95, 0)"></OBJECT>  
</P>
```

(2) 编辑用于实现立体旋转文字的JavaScript代码。

自定义函数 Window\_OnLoad(), 用于设置 Flash组件在页面中的显示位置, 并调用自定义函数RotateAll()。代码如下。

```
<SCRIPT language="JavaScript">  
functionWindow_OnLoad() {  
    document.SG1.Scale(0.50, 0.50, 0.50);  
    document.SG2.Scale(0.50, 0.50, 0.50);  
    document.SG3.Scale(0.50, 0.50, 0.50);  
    document.SG4.Scale(0.50, 0.50, 0.50);  
    document.SG1.Rotate(0, 0, 0);  
    document.SG2.Rotate(90, 0, 0);  
    document.SG3.Rotate(0, 90, 0);  
    RotateAll();  
}
```

自定义函数RotateAll(), 用于在指定的时间动态旋转Flash组件。代码如下。

```
function RotateAll() {  
    document.SG1.Rotate(4, 6, 2);  
    document.SG2.Rotate(4, 6, 2);  
    document.SG3.Rotate(4, 6, 2);  
    document.SG4.Rotate(4, 6, 2);  
    window.setTimeout("RotateAll()", 20);  
}
```

```
}  
Window_OnLoad();  
</SCRIPT>
```

举一反三

根据本实例，读者可以：

在动态网页中，播入Flash动画；

在动态网页中，动态漂移立体文字。

## 实例160 文字3D效果

本实例可以美化界面

实例位置：光盘\mingrisoft\05\160

实例说明

在一些动画的网站中，经常会有一些 3D 效果的文字，这样可以使页面更有立体感。本实例将在页面中制作一个3D效果的文字，运行结果如图5.36所示。



图5.36 文字3D效果

技术要点

本实例主要是用 string 对象的 charAt() 方法将一组汉字拆分成单个汉字，再将每个汉字以层的形式重复显示 8 次，并将每个汉字的 8 个层从后往前叠加，同时向左下角移动，以形成 3D 效果。要想使 3D 效果看上去更加逼真，就应使用 RGB() 函数动态地显示每层的颜色。下面将详细介绍 RGB() 函数。

RGB()函数的语法格式如下。

RGB(r, g, b)

参数说明如下。

- r: 必要参数, 数值范围从0~255, 表示颜色的红色成分。
- g: 必要参数, 数值范围从0~255, 表示颜色的绿色成分。
- b: 必要参数, 数值范围从0~255, 表示颜色的蓝色成分。

功能: 将三原色 (RGB) 转换成Tcolor。

注意: 当系统设置为32色时, RGB 函数才能设置出更多的颜色。

charAt()方法已在实例158中进行了详细的讲解。

实现过程

编写用于实现文字3D效果的JavaScript代码。

(1) 将字符串中的每个文字用层来显示8次, 并按指定的间距进行排列。代码如下。

```
<script language=JavaScript>
Phrase="明日科技图书";
Balises="";
Nb=Phrase.length;
layer=8;
si=80;
for (x=0;x<Nb;x++) {
    for(y=0;y<layer;++y) {
        Balises=Balises+'<DIV id=L'+x+y+' style="FONT-WEIGHT:bold;FONT-SIZE:70px;LEFT: '+si+' px;WIDTH:5px;COLOR:rgb(195,195,0);FONT-FAMILY:CourierNew;POSITION: absolute;FONT-FAMILY:CourierNew;TOP:200px">' +Phrase.charAt(x)+'</DIV>';
```

```

    }
    si=si+80;
}
document.write(Balises);

```

(2) 自定义函数 max3D(), 将每个相同的文字从后往前叠加, 同时向左下角移动, 并对文字进行变色。代码如下。

```

function max3D() {
    for (x=0;x<Nb;++x) {
        for (y=0;y<layer;++y) {
            var Ob=document.all("L"+x+y);
            Ob.style.posLeft=Ob.style.posLeft-y-1;
            Ob.style.posTop=Ob.style.posTop+y*2+1;
            Ob.style.color="rgb("+ (30+y*35) +", "+
(y*20+30) +", 100)";
            Ob.style.fontSize=parseInt(Ob.style.fontSize)+15;
        }
    }
}
max3D();
</script>

```

举一反三

根据本实例, 读者可以:

制作一个动态的三维文字;

制作一个变色的三维文字。

## 5.7 文字动画效果

在网页设计时，经常会对一些文字进行移动、动态旋转，或将一组文字按一定的规律进行拆分并组合，这样可以使网页具有动态效果。本节将通过几个典型实例来介绍文字动画在网页中的应用。

### 实例161 反弹文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\161

实例说明

本实例是将一组文字以线形方式在页面中进行移动，当文字移动到工作区的边框时，文字将以随机方向进行反弹，如图5.37所示。



图5.37 反弹文字

技术要点

本实例主要是用 document 对象中的 write() 方法将数组中的单个汉字以层的形式添加在页面的左上角，用document对象中的body对象的clientHeight、clientWidth属性来获取工作区的高和宽。当首汉字移动到工作区的边框时，用Math对象中的random()和floor()方法获取一个方向值，使文字反弹后能够按一定的方向进行移动。下面对Math对象中的random()和floor()方法进行详细说明。

Math对象中的random()方法的语法格式如下。

Math.random()

功能：返回介于 0 和 1 之间的伪随机数。产生的伪随机数介于 0 和 1 之间（含 0，不含 1），也就是，返回值可能为0，但总是小于1。

Math对象中的floor()方法的语法格式如下。

Math.floor(number)

参数说明如下。

● number：必选项，数值表达式。

功能：返回小于等于其数值参数的最大整数。

实现过程

(1) 编写用于实现反弹文字的JavaScript代码。

将反弹的文字用层显示在工作区的左上角，并将每个层设为可移动状态。用一维数组xpos和ypos记录每个文字的位置。代码如下。

```
<SCRIPT language="javascript">  
var textsize=20;  
var tempo=40;  
var stepx=30;  
var stepy=20;  
var text=newArray("明","日","科","技","图","书");  
var x,y;
```

```

varmarginbottom;
var costL=costT=0;
varmarginright;
var timer;
var xpos=newArray();
var ypos=newArray();
var spanarray=newArray();
//定义两个数组，用于记录字符串的位置
for (i=0; i<text.length;i++) {
    xpos[i]=0
    ypos[i]=0
}
//将字符串的单个记录以层的方式显示在工作区中
for (i=0;i<text.length;i++) {
    spanarray[i]="<span style=' position:relative;font-
family:宋体;font-size:
    "+textsize+"pt;color:#0033FF' >" +text[i]+"</span>"
}
//将每个层设为可移动状态
for (i=0;i<text.length;i++) {
    document.write("<span id=' span"+i+"''
style=' position:absolute' ></span>");
    document.close();
}

```

自定义函数sport()，获取工作区的大小，并将层放在工作区的左上角，代码如下。

```
function sport() {
```

```

marginbottom=document.body.clientHeight-5;
marginright=document.body.clientWidth-5;
for (i=0;i<text.length;i++) {
    var thisspan= eval("document.all.span"+i);
    thisspan.innerHTML=spanarray[i];
    var thisspan= eval("document.all.span"+
(i)+".style");
    thisspan.posLeft=0;
    thisspan.postop=0;
}
toplace(); //使文字循环移动
}

```

自定义函数change(), 用于获取文字移动方向的随机数。代码如下。

```

function change(range) {
    rand=Math.floor(range*Math.random());
    return rand;
}

```

自定义函数toplace(), 用于调用自定义函数rim()和moveplace(), 使文字在工作区中循环移动。代码如下。

```

function toplace() {
    rim(); //用于判断字符串中首字符移动到工作区的四边时所
反弹的方向
    moveplace(); //使文字进行移动
    document.all.span0.style.posTop+=stepy;
    timer=setTimeout("toplace()", tempo);
}

```

自定义函数moveplace(), 用于在工作区中动态移动文字。代码如下。

```
function moveplace() {
    for (i=(text.length-1); i>=1; i--) {
        xpos[i]=xpos[i-1];
        ypos[i]=ypos[i-1];
    }
    xpos[0]=document.all.span0.style.posLeft+stepx;
    ypos[0]=document.all.span0.style.posTop+stepy;
    for (i=0; i<text.length; i++) {
        var thisspan=eval("document.all.span"+(i)+"style");
        thisspan.posLeft=xpos[i];
        thisspan.posTop=ypos[i];
    }
}
```

自定义函数rim(), 用于判断字符串中首字符移动到工作区边框时所反弹的方向。代码如下。

```
function rim() {
    if (document.all.span0.style.posLeft>marginright) { //工作区左边框
        stepx=(stepx+change(3))*-1;
        document.all.span0.style.posLeft-=1;
    }
    if (document.all.span0.style.posLeft<costL) { //工作区右边框
        stepx=(stepx+change(3))*-1;
        document.all.span0.style.posLeft+=1;
    }
}
```

```

    }
    if (document.all.span0.style.posTop>marginbottom)
{    //工作区下边框
    stepy=(stepy+change(3))*-1;
    document.all.span0.style.posTop-=1;
    }
    if (document.all.span0.style.posTop<costT) {    //工作区
上边框
    stepy=(stepy+change(3))*-1;
    document.all.span0.style.posTop+=1;
    }
}
</SCRIPT>

```

(2) 在<body>标记的载入事件 (onLoad) 中调用自定义函数 sport()。

```
<BODY onload=sport()>
```

举一反三

根据本实例，读者可以：

制作一个漂移的广告；

在页面中制作一个反弹的小球。

## 实例162 飞舞的文字

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\05\162

实例说明

本实例是将一组文字以曲线的形式动态地在页面中进行显示，当移动到页面工作区的最左边时，每个文字将不停地撞击工作区的上边框，然后再随机地往下降落。运行结果如图5.38、图5.39所示。



图5.38 以曲线向前移动的文字

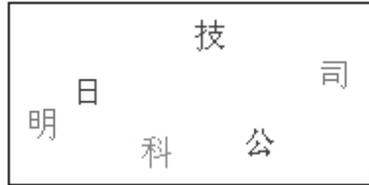


图5.39 随机下降的文字

### 技术要点

本实例主要是以一维数组（route）的形式设计了一个能让文字移动的轨迹，并设置了一个CSS 样式，使层的 position 属性为 absolute（使层可以移动），visibility 属性为 visible（让层在页面中可以显示）。通过Math对象的route()方法来获取一个随机数，可以让文字以随机的速度在工作区中往下降落，route()方法返回0~1（包含0，不包含1）之间的随机数。

### 实现过程

（1）在页面中添加一个层，并设置其id为a，代码如下。

```
<div id="a"></div>
```

（2）在页面中添加一个CSS样式，用于改变单个文字所在层的文字的颜色、大小、位置及将层设为可移动状态，代码如下。

```
<STYLE>
```

```
A {
```

```
    COLOR: #000000; FONT-FAMILY: Verdana; FONT-SIZE: 8pt;  
    FONT-WEIGHT: bold;  
    LEFT: 8px; POSITION: absolute; TOP: 250px
```

```

}
.dance {
    LEFT: 4000px; POSITION: absolute; VISIBILITY: visible
}
</STYLE>

```

(3) 编写用于实现飞舞文字的JavaScript代码。

用一维数组 `textstr` 来保存要飞舞的字符串，并用 `route` 数组来记录文字飞舞的轨迹。代码如下。

```

<script language="JavaScript">
    var textstr=newArray();    //定义一个一维数组，用于记录要
    动态显示的文字
    textstr[0]="明日科技公司";
    textstr[1]="欢迎购买明日科技图书";
    textstr[2]="明日科技的主页：www.mingrisoft.com";
    var
textcolor=newArray("FF00FF","0000FF","00FF00","FF0000");
    //设置文字的颜色
    varG=0;
    var routeL=newArray();
    var routeH=newArray();
    //定义一个一维数组，用于记录文字移动的轨迹
    var
route=newArray(554, 9, 552, 10, 550, 13, 542, 13, 538, 13, 532, 10, 528, 9
, 528, 9, 522,
    7, 519, 8, 512, 9, 510, 11, 507, 13, 505, 16, 500, 17, 495, 18, 492, 17, 4
88, 15, 482, 9, 477, 9, 473,

```

```

    9, 466, 12, 461, 17, 459, 19, 452, 20, 448, 21, 443, 18, 439, 15, 434, 9,
431, 9, 424, 8, 419, 9, 416,
    11, 412, 12, 410, 15, 408, 17, 406, 19, 398, 19, 391, 20, 383, 19, 379, 1
7, 372, 14, 369, 10, 364, 9,
    356, 7, 348, 8, 340, 9, 334, 11, 323, 13, 315, 17, 301, 18, 293, 17, 288,
12, 285, 10, 278, 7, 274, 7,
    269, 9, 268, 9, 265, 12, 260, 17, 250, 17, 241, 17, 229, 16, 220, 10, 216
, 9, 213, 9, 208, 9, 203, 9,
    194, 14, 188, 16, 177, 16, 166, 16, 157, 14, 149, 13, 146, 8, 135, 7, 124
, 6, 114, 9, 104, 14, 94, 16,
    69, 16, 61, 15, 41, 15, 34, 15, 22, 16, 12, 16, 13, 15);
var s="";
var text="";

```

自定义函数show()，将文字按一定的轨迹显示在页面中，并调用自定义函数sport()。代码如下。

```

function show() {
    text=textstr[G];
    s="";
    //循环显示文字，并通过CSS样式设置文字的样式
    for (i=0;i<text.length;i++) {
        s=s+"<font style=' font-size:16px;
color:#"+textcolor[Math.floor((
    textcolor.length)*Math.random())]+"'><span
id=' span"+i+"' class=' dance' 0>";
        s=s+text.substring(i, i+1);
        s=s+"</font></span>";
    }
}

```

```

a.innerHTML=s; //通过innerHTML属性动态加a层的值
varx=0;
vary=1;
var i=0;
for (i=0;i<route.length;++i) { //设置每个文字位置的初始
值
    routeL[i]=route[x];
    routeH[i]=route[y];
    x=x+2;
    y=y+2;
}
for (i=0;i<text.length;++i) { //在页面中将文字以
指定的位置进行显示
    varpp=eval("document.all.span"+(i)+".style");
    pp.posLeft=routeL[text.length-i]+(i*20);
    pp.posTop=routeH[text.length-i];
}
sport(text.length+1); //调用sport()函数,使文字能动态地向前移动
}

```

自定义函数sport(),使文字按一定的轨迹向前移动。代码如下。

```

var timer;
function sport(n) {
    if (n>=Math.floor(routeL.length/2)) { //判断文字是否到工作区的最左边
        window.clearTimeout(timer); //停止对文字的移动
        tingle(0,0); //调用tingle()函数,使文字不停的抖动
    }
}

```

```

}
else{
    for (i=0;i<text.length;++i) {          //使文字以指定的轨
迹向前移动
        var pp=eval("document.all.span"+(i)+".style");
        pp.posLeft=routeL[n-i]+(i*20);
        pp.posTop=routeH[n-i];
    }
    //循环调用sport()函数，使文字以动画的形式向前移动
    timer=setTimeout("sport"+"+(n+1)+"",30);
}
}

```

自定义函数tingle()，使文字按随机的高度进行跳动。代码如下。

```

function tingle(p,e) {
    for (i=0;i<text.length;++i) {          //用来修改文字与工
作区上边框的高
        varpp=eval("document.all.span"+(i)+".style");
        pp.posTop=routeH[p+i];
    }
    p=p+2;
    e=e+1;
    if (e<30)
    {timer=setTimeout("tingle"+"p+", "e+", 15);} //使文字出现
抖动效果
    else {
        window.clearTimeout(timer);      //停止文字的抖动
    }
}

```

```

        whereat(); //调用whereat()函数，使文字随机向下降落
    }
}

```

自定义函数whereat()，使文字按随机的高度下降。代码如下。

```

function whereat() {
    for (i=0; i<text.length; ++i) {
        var pp=eval("document.all.span"+(i)+".style");
        pp.posTop=pp.posTop+Math.random()*200; //随机修改文字
        与工作区上边框的高
        var just=true;
        for (j=0; j<text.length; ++j) {
            var pp=eval("document.all.span"+(j)+".style");
            //判断文字是否降落在工作区以外
            if (pp.posTop<document.body.clientHeight) {
                just=false;
                break;
            }
        }
        if (just==true) {break;} //如果没有文字在当前工作区
        内，退出本次循环
    }
    if (just==true) {
        window.clearTimeout(timer); //停止对文字的下降效果
        G=G+1; //指定下一个数组文字
        if (G>2) {G=0;}
        show(); //让下一组文字重新显示其动画效果
    }
}

```

```
else {timer=setTimeout("whereat()",20);} //让文字继续下降  
}
```

在窗体载入时，调用自定义函数show()。代码如下。

```
show();  
</script>
```

举一反三

根据本实例，读者可以：

让文字在页面中按一定的轨迹进行移动；

在页面中制作一个文字跳动的效果。

## 实例163 飞翔的文字

本实例可以美化界面

实例位置：光盘\mingrisoft\05\163

实例说明

本实例主要是将一组文字随机显示在页面工作区的某一点上，然后将这些层向左下角进行扩散，当扩散到一定距离后，将所有的文字向页面工作区的右上角进行扩散。运行结果如图5.40所示。



图5.40 飞翔的文字

技术要点

本实例主要是用document对象中的write()方法在浏览器中创建一个CSS样式，来定义层的字体颜色、大小、初始位置及将每一个层设为可移动状态。

在对层进行移动前，用eval()方法将指定层中的style样式保存在一个变量中，在修改层的位置时，可以直接用该变量调用style样式下的所有属性对其值进行修改，在对层的位置进行修改时用Math对象的round()方法获取值的最大整数。下面对eval()方法和round()方法进行详细说明。

eval()方法的语法格式如下。

```
eval(codeString)
```

参数说明如下。

● **codeString**: 必选项参数，是包含有效 JavaScript 代码的字符串值。这个字符串将由JavaScript分析器进行分析和执行。

功能：检查JavaScript代码并执行。

round()方法的语法格式如下。

```
Math.round(number)
```

参数说明如下。

● **number**: 必选项参数，是要舍入到最接近整数的值。如果number的小数部分大于或等于0.5，那么返回值是大于number的最小整数。否则，round返回小于或等于number的最大整数。

功能：返回与给出的数值表达式最接近的整数。

实现过程

编写用于实现飞翔文字的JavaScript代码。

(1) 将字符串中的每个文字用层来进行显示，并用CSS样式来设置层的位置及前景色。其代码如下。

```
<SCRIPT language="JavaScript">  
var textstr="欢迎购买明日科技图书";
```

```

    varCarray=newArray("lime","fuchsia","blue","orange","red"
, "yellow",
    "pink","COA5AC","aqua","9999FF");    //定义文字的随机颜色
//在窗体中添加CSS样式
document.write("<STYLE>");
for (p=0;p<textstr.length;++p) {
    document.write("#sDiv"+p+" {COLOR: "+Carray[p]+";FONT-
FAMILY: arialblack;
    FONT-SIZE:15px;HEIGHT:1px;POSITION:
absolute;WIDTH:1px}");
}
document.write("</STYLE>");
for (p=0;p<textstr.length;++p) {    //将每个文字以层的形式
在页面中进行显示
    document.write("<DIV
id=sDiv"+p+">"+textstr.substring(p,p+1)+"</DIV>");
}
var time=1000;
var speed=100;
var power=3;
var ifshift= false;
var sArray=newArray(10);
var early=0;

```

(2) 自定义函数initAll(), 用于循环调用firework()函数, 并用数组获取每个层style样式下的属性。其代码如下。

```

function initAll() {    //循环调用firework()函数, 并用数组
获取每个层style样式下的属性

```

```

    setInterval("firework()", time)
    for(j=0; j<textstr.length;++j)
sArray[j]=eval('document.all.sDiv'+j+'.style');
}

```

(3) 自定义函数firework(), 用于设置每个层的初始化位置。其代码如下。

```

function firework() { //设置层移动位置
    vardWidth=document.body.clientWidth; //
//获取当前工作区的宽
    vardHeight=document.body.clientHeight;
//获取当前工作区的高
    var
cleft=document.body.scrollLeft; //获取
横向滚动条卷去的部分
    var
ctop=document.body.scrollTop; //获取纵
向滚动条卷去的部分
    var
randomx=cleft+Math.floor(Math.random()*dWidth); //设置
层的随机位置
    var randomy=ctop+Math.floor(Math.random()*dHeight);
    if(!ifshift) {
        ifshift=true;
        early=0;
        for(var spark=0;spark<textstr.length;spark++)
    { //初始化每个层的随机位置
        dx=Math.round(Math.random()*50);

```

```

        dy=Math.round(Math.random()*50);
        moveTo(spark, randomx, randomy, dx, dy);           //
调用自定义函数moveTo()
    }
}
}

```

(4) 自定义函数moveTo(), 用于判断文字移动的方向, 并进行动画移动效果。其代码如下。

```

functionmoveTo(i, rx, ry, dx, dy) { //判断文字移动的方向, 并
进行动画移动效果
    if(ry+80>
(document.body.offsetHeight+document.body.scrollTop))
        ry=document.body.offsetHeight+document.body.scrollTop
-80;
    if(rx+80>
(document.body.offsetWidth+document.body.scrollLeft))
        rx=document.body.offsetWidth+document.body.scrollLeft
-80;
    if(rx>-50&&ry>-50) {
        rx+=dx;
        ry+=dy;
        sArray[i].left=rx;
        sArray[i].top=ry;
        dx-=power;dy-=power;
        setTimeout("moveTo("+i+", "+rx+", "+ry+", "+dx+", "+dy+")
", speed)
    }
}

```

```
else
    ++early;
if(early==10) {
    ifshift=false;
    early=0;
}
}
```

(5) 在窗体载入时，调用自定义函数initAll()。其代码如下。

```
window.onload=initAll;
</SCRIPT>
```

举一反三

根据本实例，读者可以：

在页面中制作一个绽放的礼花；

使文字在页面中随机地游动。

## 实例164 平面旋转的文字

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\05\164

实例说明

本实例主要是将一组文字分成不同的颜色块，并以指定的半径在页面中组成一个圆，并以动态的形式进行旋转，如图5.41所示。

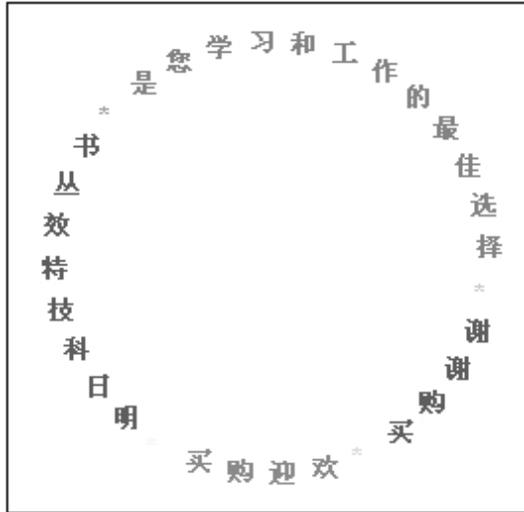


图5.41 平面旋转的文字

### 技术要点

本实例主要是用String对象的charAt()方法获取字符串中的每一个汉字，并以\*为节点，将文字设成不同的颜色块，用公式 $s=2\pi/n$ 可以计算出n个汉字组成一个圆的字间距，在存储字间距时，一定要存储成单精度，再用公式： $r*\sin(s+\pi/180)$ 和 $r*\cos(s+\pi/180)$ 来获取每个文字的位置，这样就能在页面中通过r（半径）值的变化来显示指定大小的圆。

### 实现过程

编写用于实现平面旋转文字的JavaScript代码。

(1) 将字符串中的文字以层进行显示，并以\*号为节点，将每个层的前景色设置成不同的颜色。用Radius变量保存圆的半径。其代码如下。

```
<script language="JavaScript">
var str="欢迎购买*明日科技特效丛书*是您学习和工作的最佳选择*谢谢购买*";
var s="";
var
a=newArray("#6699CC", "#FFFF00", "#CC0000", "#FF80C0", "#8080FF",
```

```

"#FFCC00", "#CC0000", "#66FF00");
var j=0;
for (var i=0;i<str.length;i++) {//以*为节点，将字符串以不同的颜色进行显示
    if (str.charAt(i)=="*") {j=j+1}
    s=s+'<DIV id=Circle'+i+' style="COLOR: '+a[j]+';FONT:bold 11pt arial;POSITION: absolute;visibility:hidden;"><P><FONT size=4>'+str.charAt(i)+' </FONT></P></DIV>'
    if (str.charAt(i)=="*") {j=j+1}
}
document.write(s);
var Pos=new Array();
var Radius = 160;//半径
var Pi = Math.PI;
var Inc = Pi/180;

```

(2) 自定义函数showfont(), 用数组Pos记录文字显示成圆的间距。其代码如下。

```

function showfont() {
    Pos[0] = 0;
    for(i=1;i<str.length;i++) Pos[i] =
eval(parseFloat(Pos[i-1]+((2*Pi)/
str.length)));
    cycle();
}

```

(3) 自定义函数cycle(), 在页面中用文字显示圆，并进行动态的旋转。其代码如下。

```

function cycle() {
    for(i=0;i<Pos.length;i++) {
        Pos[i]=Pos[i]+Inc;
        varpp="document.all."+"Circle"+i+".style";
        eval(pp).left=Radius*Math.cos(Pos[i])+document.body.offsetLeft/2-10;
        eval(pp).top
=Radius*Math.sin(Pos[i])+document.body.offsetTop/2-10;
        eval(pp).visibility="visible";
    }
    setTimeout("cycle()",10)
}

```

(4) 在窗体载入时，调用自定义函数showfont()。其代码如下。  
showfont() ;

</script>

举一反三

根据本实例，读者可以：

使一个圆在自转的时候进行旋转；

制作一个扩散的圆环。

## 实例165 输出文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\165

实例说明

本实例是将一组带阴影的文字在页面中一个一个地显示出来，并将该组文字以不同的语句在窗体上循环显示。运行结果如图5.42所

示。



图5.42 输出文字

### 技术要点

本实例主要是用<span >标记中的style 样式的clip 属性的rect()方法来实现字符串的输出效果，下面对rect()方法进行详细说明。

rect()方法的语法格式如下。

```
clip : auto | rect( number number number number )
```

参数说明如下。

- auto: 对象无剪切。
- rect (number number number number) : 依据上-右-下-左的顺序提供指定对象左上角为(0,0)坐标的4个偏移数值，其中任一数值都可用auto替换，即此边不剪切。

功能：检索或设置对象的可视区域。区域外的部分是透明的，必须将 position 的值设为absolute，此属性方可使用。

### 实现过程

(1) 在页面中添加两个层，shadow 层用于显示实体的文字，showstr 层用于显示文字的阴影，代码如下。

```
<div id="shadow"  
style="position:absolute;visibility:visible;top:12px;  
left:12px">  
</div>  
<div id="showstr"  
style="position:absolute;visibility:visible;top:10px;  
left:10px">
```

```
</div>
```

(2) 编辑用于实现输出文字的JavaScript代码。

用一维数组来保存要进行输出的字符串。其代码如下。

```
<script language="JavaScript">
```

```
var space=0
```

```
var word=0
```

```
var Tstr = new Array()
```

```
Tstr[0]="明日科技"
```

```
Tstr[1]="欢迎购买明日科技图书"
```

```
Tstr[2]="明日科技网址: www.mingrisoft.com"
```

自定义函数exportstr(), 以输出的方式循环显示Tstr数组中的字符串。其代码如下。

```
function exportstr() {
```

```
if (space<=document.body.clientWidth) {
```

```
    showstr.innerHTML="<span style=' position:absolute;font-family:宋体;
```

```
    font-size:30pt;color:3399FF; clip:rect(2px "+space+"px  
200px0px)'>" +
```

```
Tstr[word]+"</span>"
```

```
    shadow.innerHTML="<span style=' position:absolute;font-family:宋体;
```

```
    font-size:30pt;color:888888; clip:rect(2px "+space+"px  
200px0px)'>" +
```

```
Tstr[word]+"</span>"
```

```
    space=space+30
```

```
    var timer=setTimeout("exportstr()", 30)
```

```
}
```

```
else {
    document.close()
    clearTimeout(timer)
    word++
    space=0
    if (word>=Tstr.length) {word=0}
    var intermezzo=setTimeout("exportstr()",100)
}
}
exportstr();
</script>
```

举一反三

根据本实例，读者可以：  
制作一个图片横向展示效果；  
为图片添加阴影。

## 实例166 文字打字效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\166

实例说明

本实例是将一组文字以单个文字不断累加的形式在页面中进行显示，并改变每次累加的最后一个文字的颜色。运行结果如图5.43所示。

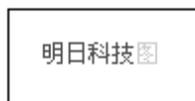


图5.43 文字打字效果

## 技术要点

本实例主要是用string对象的substring()方法来获取上一次所显示的文字，再将通过String对象的charAt()方法所获取的下一个文字添加到上一次所显示文字的后面，并将其改变颜色。下面对substring()和charAt()方法进行详细说明。

substring()方法的语法格式如下。

```
strVariable.substring(start, end)
```

```
"String Literal".substring(start, end)
```

参数说明如下。

- strVariable: 必选项，字符串变量。
- "String Literal": 必选项，字符串。
- start: 必选项，指明子字符串的起始位置，该索引从0 开始起算。
- end: 必选项，指明子字符串的结束位置，该索引从0 开始起算。

功能:

返回位于 string 对象中指定位置的子字符串。

charAt()方法的语法格式如下。

```
strObj.charAt(index)
```

参数说明如下。

- strObj: 必选项，任意string 对象或文字。
- index: 必选项，字符基于0 的索引值，有效值是0 与字符串长度减1 之间的值。

功能: 返回指定索引位置处的字符。

实现过程

(1) 在页面中添加一个层，并设置id为TDiv，代码如下。

```
<DIV class=Cla style1 id="TDiv"></DIV>
```

(2) 编辑用于实现文字打字效果的JavaScript代码。

自定义函数ingo(), 调用自定义函数typed(), 并设置初始值。其代码如下。

```
<script language="JavaScript">
var dispStr=newArray("明日科技图书");
function ingo() {
    typed(dispStr[0], 0, document.all.TDiv,
'Cla', ' #3333FF', ' #FFCC00', 600, 0);
}
```

自定义函数typed(), 用于实现文字的打印效果。其代码如下。

```
function typed(Astr, x, id, clas, col1, col2, wit, tim) {
    if (document.all) {
        var str0=str1="";
        str0=Astr.substring(0, tim); //获取上一次显示的文字
        str1=Astr.charAt(tim++); //获取上一次显示文字的下一个文字
        var str="<span class="+clas+"><font
color=' "+col1+" '>"+str0+"</font><font color='
"+col2+" '>"+str1+"</font></span>";
        var Stim=setTimeout("typed(' "+Astr+"', "+x+" , ' "+id+"',
' "+clas+"', ' "+col1+"',
' "+col2+"', "+wit+" , "+tim+" )", 600);
        document.all.TDiv.innerHTML=str; //用
innerHTML属性改变层的值
        if (tim>Astr.length) //判断是否
已显示所有的文字
            window.clearTimeout(Stim);
```

```
}  
}
```

在窗体载入时，调用自定义函数ingo()。其代码如下。

```
ingo();  
</script>
```

举一反三

根据本实例，读者可以：

制作弹簧式的文字伸缩效果；

制作一个追逐的变色效果。

## 实例167 文字抖动

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\05\167

实例说明

本实例是将一组文字按从左到右的顺序，将一定范围内的文字以凸起的状态向后移动，使文字出现动态的曲线效果，结果如图5.44所示。

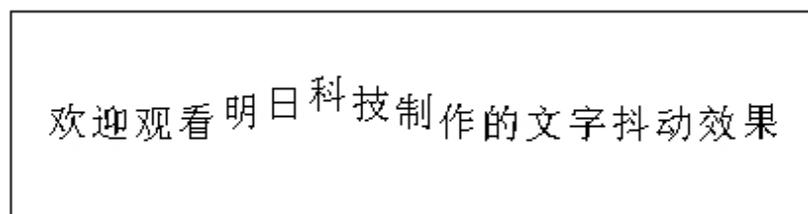


图5.44 文字抖动

技术要点

本实例主要是将字符串中的每个汉字用层来进行显示，并将层的id用数字从小到大进行排列，并自定义一个数组（sumstr），用于改变字符串指定个数的高度。用 setTimeout()方法调用自定义函数dmove()时，将字符串的位置向后移动一个汉字，当移动到最后一个文

字时，将位置移动到第一个汉字，使其出现动态的抖动效果。其关键主要是通过 `document.all("aa"+(n+i2))` 来获取要移动的层，`n` 表示移动层的起始位置，`i2` 表示要移动层的个数的一个动态变量，并通过层的 `style` 样式的 `top` 属性来设置层在 `x` 轴上的位置。

### 实现过程

编写用于实现文字抖动的 JavaScript 代码。

(1) 用层来显示字符串中的每个文字。其代码如下。

```
<script language="JavaScript">
var strname="欢迎观看明日科技制作的文字抖动效果";
var strlen=strname.length;
var dstr="";
var ssize=50;
function fdiv() { //将字符串中的每个文字用层来显示
    for (i=0;i<strlen;++i) {
        dstr=dstr+'<div id=aa'+i+' style="font-size:20px;
top:200px; left:'+ssize+' px;
position:absolute;">'+strname.charAt(i)+'</div>';
        ssize=ssize+22;
    }
    document.write(dstr);
}
```

(2) 自定义函数 `dmove()`，该函数用于调用自定义函数 `Parray()` 和 `Larray()`，对字符串进行抖动效果的设置。其代码如下。

```
var sumstr=newArray(-1,-6,-10,-15,-10,-6,-1); //用数组设置文字抖动的高度
var sublen=sumstr.length; //获取字符串的长度
var i=1;
```

```
function dmove() { //调用其他的函数，对字符串进行抖动效果
```

```
    var s="";  
    if (i!=1) {Parray(i);}   
    Larray(i);  
    ++i;  
    if (i==strlen) i=0;  
    setTimeout("dmove()",100);  
}
```

(3) 自定义函数Parray(), 在文字抖动时, 将前面的文字置于水平状态。其代码如下。

```
function Parray(n) {在文字抖动时，将前面的文字置于水平状态  
    for (i1=1;i1<n;++i1) {  
        ob=document.all("aa"+(i1));  
        ob.style.top=200+"px";  
    }  
}
```

(4) 自定义函数Larray(), 通过sumstr数组来设置文字的高, 出现抖动效果。其代码如下。

```
function Larray(n) { //通过sumstr数组来设置文字的高，出现抖动效果
```

```
    for (i2=0;i2<sublen;++i2) {  
        if ((n+i2)<strlen) {  
            ob=document.all("aa"+(n+i2));  
            ob.style.top=((200+sumstr[i2])+"px");  
        }  
    }  
}
```

```
}
```

(5) 在窗体载入时调用自定义函数fdiv()和dmove()。其代码如下。

```
fdiv();  
dmove();  
</script>
```

举一反三

根据本实例，读者可以：

制作一个S形的文字动画走动效果；

制作一个文字下降效果。

## 实例168 指向文字时飞出星形标记

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\05\168

实例说明

本实例是将鼠标移动到指定文字的上面时，动态显示一个星形标记，并不断地变大，鼠标移出该段文字时，飞出的星形标记将消失。结果如图5.45所示。

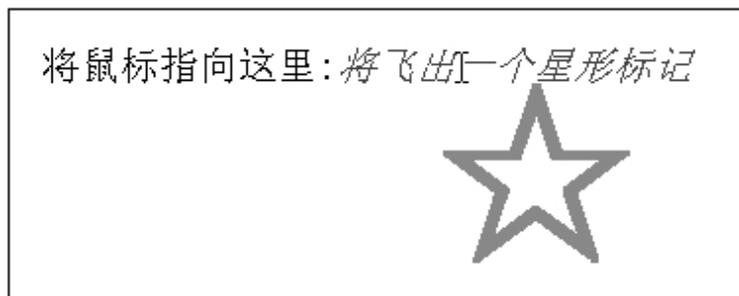


图5.45 指向文字时飞出星形标记

技术要点

本实例主要是通过层的style样式中visibility属性的visible和hidden值来动态设置层的显示和隐藏状态，并通过window对象中的event对象的x和y属性来获取鼠标的当前位置。如果要使层在指定的位置显示，必须将层的style样式的position属性设为absolute，使层为可移动状态，并通过对层的style样式的fontSize属性值进行递增修改，使层不断地变大。

本实例是通过setTimeout()方法按指定的时间调用divsize()函数来改变层的大小，当动画结束时应用window对象的clearTimeout()方法将setTimeout()方法消除。

### 实现过程

(1) 在页面中添加一串文字及一个层，当鼠标指向文字的后半部时，将通过层动态显示飞出来的星形标记，代码如下。

```
<font style="font-size:20px ">将鼠标指向这里:</font><a  
style="color:#3300FF; font-size:  
20px; font-style:italic " onMouseMove="move()"  
onMouseOut="return out();">将飞出一个星形标记</a>  
<div id="div1" style="width:60px; height:30px; font-  
size:40px; color:#FF00FF; font-weight:  
bold;">☆</div>
```

(2) 编辑用于实现当鼠标指向文字时飞出星形标记的JavaScript代码。

自定义函数out()，当鼠标移出指定的文字时，隐藏星形标记。其代码如下。

```
<script language="JavaScript">  
div1.style.position="absolute";           //将层设置为可移动  
的状态  
div1.style.visibility="hidden";           //将层隐藏
```

```

varPT;
var bool=false;
var size=40;
function out() {                                //当鼠标移出指定的文字
时，调用自定义函数out()
    div1.style.visibility="hidden";            //隐藏层
    div1.style.fontSize="40px";将层的大小设置为初始状态
    size=40;
    bool=false;
    //在这里再调用一次divsize()函数，是为了关闭setTimeout()
的调用，以减少内存的使用
    divsize();
}

```

自定义函数move()，当鼠标移动到指定的语句时，将层显示在指定位置，并进行动态放大。其代码如下。

```

functionmove() {//当鼠标移动到指定的语句时，将层显示在指定
位置，将调用divsize()函数
    //获取鼠标的当前位置
    varx=window.event.x+document.body.clientLeft;
    vary=window.event.y+document.body.clientTop;
    div1.style.left=x;                            //设置层的位置
    div1.style.top=y;
    div1.style.visibility="visible";            //使层为显示状态
    bool=true;
    divsize();                                    //调用自定义函数divsize()
}

```

自定义函数divsize()，用于动态改变层的大小。其代码如下。

```

functiondivsize() { //用于动态改
变层的大小
    if (bool==true) {
        if (size>=200) {out();} //当层的大小
为最大时，调用自定义函数out()
        else{
            size=size+2;
            div1.style.fontSize=String(size)+"px"; //增加层中
字体的大小
            PT=setTimeout("divsize()",200); //重复调用
divsize()函数，使层逐渐变大
        }
    }elsewindow.clearTimeout(PT); //清除对层的调
用
}
</script>

```

### 举一反三

根据本实例，读者可以：

制作一个动态伸缩的文字效果；

制作一个随意拖动的文字。

## 实例169 文字的抛出效果

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\05\169

实例说明

本实例是将一组文字以正弦曲线的值来改变字符串中每个文字的字体大小，并以动态的形式进行显示，从而出现文字抛出的效果，结果如图5.46、图5.47所示。



图5.46 文字抛出前的效果



图5.47 文字抛出后的效果

### 技术要点

本实例主要是用公式： $\text{Math.sin}(i/(\text{textLength}/3.14))$ 来计算正弦曲线各点的值，并用这些值来修改字符串中每个文字的字体大小， $i$ 表示字符串中第几个文字， $\text{textLength}$ 表示字符串文字的个数。用层的`innerHTML`属性将修改后的文字动态添加到层中进行显示。

### 实现过程

(1) 在页面中添加一个一行一列的表格及一个层，并将层放到单元格中，代码如下。

```
<tableborder="0" cellpadding="0" style="FONT-  
SIZE:9pt"width="100%">  
  <TR>  
    <TD align=middle style="COLOR:  
rgb(255,255,0)"width="100%">  
      <DIV align=center id="div1"></DIV>  
    </TD>  
  </TR>  
</table>
```

(2) 编辑用于实现当鼠标指向文字时飞出星形标记的JavaScript代码。

自定义函数nextSize ()，用于计算相邻文字的缩放比例。其代码如下。

```
<script language="JavaScript">
function nextSize(i, textLength) {
    return (32*Math.abs( Math.sin(i/(textLength/3.14))));
}
```

自定义函数Ssize()，动态修改字符串中每个文字的字体大小。其代码如下。

```
function Ssize(text, few) {
    s = "";
    for (i = 0; i < text.length; i++) {
        size = parseInt(nextSize(i+few, text.length));
        s+= "<font style='font-size:
"+size+"pt'>" + text.substring(i, i+1) + "</font>";
    }
    div1.innerHTML = s;
}
```

自定义函数 start()，用于循环调用自定义函数 Ssize()，使文字出现动态的抛出效果。其代码如下。

```
function start(n) {
    str= "※※※※※明日科技※※※※※";
    Ssize(str, n);
    if (n> str.length) {n=0}
    setTimeout("start('+(n+1)+')", 50);
}
```

```
</script>
```

(3) 在<body>标记中设置页面的背景颜色，并将页面中的左边和上边的页面空白设为 0。在窗体载入时调用自定义函数start()。代码如下。

```
<body bgColor="#0033CC" leftMargin="0" onload=start(0)  
topMargin="0">
```

举一反三

根据本实例，读者可以：

制作一个文字爆炸的效果；

拖动文字时，文字动画显示。

## 5.8 文字特殊效果

随着网页制作的不断进步，文字的特殊效果被广泛地应用，如文字的阴影效果、波浪效果等。下面我们通过几个实例来进行介绍。

### 实例170 波浪文字

这是一个实用的实例

实例位置：光盘\mingrisoft\05\170

实例说明

在浏览网页时，经常会看到一些文字或图片以水波的形式显示。本实例将实现文字的动态波浪效果。运行结果如图5.48所示。



图5.48 波浪文字

技术要点

本实例主要是通过 window 对象的 write() 方法在页面中动态添加一个层，并在层中用<font>标记显示一段文字，并通过 setTimeout() 方法来循环修改层中style样式的filter滤镜中wave属性的phase值。下面对层中style样式的filter滤镜中的wave属性进行详细的介绍。

wave的语法格式如下。

filter: Wave(Add=add, Freq=freq, LightStrength=strength, Phase=phase, Strength=strength)

参数说明如下。

- Add: 一般为1 或0。
- Freq: 变形值。
- LightStrength: 变形百分比。
- Phase: 角度变形百分比。
- Strength: 变形强度。

功能: 波浪效果。

实现过程

(1) 在<body>标记中的窗体载入事件中调用自定义函数 shadow(), 代码如下。

```
<BODY onload="if (document.all) test()">
```

(2) 编写用于实现波浪文字的JavaScript代码。将字符串以层的形式显示在页面中。其代码如下。

```
<SCRIPT language="JavaScript">
```

```
    if (document.all) {  
        var step=2;  
        var cstep=0;  
        var msg='明日科技';  
        document.write("<center><div id='water'  
style='position:relative;width:650px;  
height:150px;font-family:黑体;font-  
size:100px;color:#00AAFF' ">"+msg+"</div></center>");
```

自定义函数test(), 将层以波浪的效果进行动态显示。其代码如下。

```
function test() {
```

```
document.all.water.style.filter='wave(freq=5,
strength=5, phase='+cstep+',
lightstrength=45, add=0, enabled=1)';
cstep+=step;
TIMER=setTimeout('test()',10);
}
}
</SCRIPT>
```

举一反三

根据本实例，读者可以：

制作一个慢慢消失的波浪效果；

制作一个扭曲的图片。

## 实例171 梦幻文字效果

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\05\171

实例说明

本实例是通过对层的滤镜的设置，使文字以点状的不断增加来进行显示，给人一种虚幻的梦幻效果。运行结果如图5.49、图5.50所示。

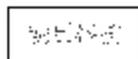


图5.49 文字梦幻消失效果

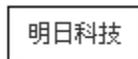


图5.50 文字显示效果

技术要点

本实例主要通过innerHTML属性来动态改变Tdiv层中的文字及其相关设置，并通过style样式的filter滤镜的revealTrans属性使文字显示时具有梦幻效果。下面对revealTrans属性进行详细说明。

revealTrans语法格式如下。

revealTrans(Transition, Duration)

参数说明如下。

- Duration: 是切换时间，以s 为单位。
- Transition: 是切换方式，它有24 种方式。

revealTrans属性的Transition参数取值如表5.3所示。

表5.3 revealTrans属性的Transition参数取值

Transition 参数值	切换效果	Transition 参数值	切换效果
0	矩形从大至小	12	随机溶解
1	矩形从小至大	13	从上下向中间展开
2	圆形从大至小	14	从中间向上下展开
3	圆形从小至大	15	从两边向中间展开
4	向上推开	16	从中间向两边展开
5	向下推开	17	从右上向左下展开
6	向右推开	18	从右下向左上展开
7	向左推开	19	从左上向右下展开
8	垂直形百叶窗	20	从左下向右上展开
9	水平形百叶窗	21	随机水平细纹
10	水平棋盘	22	随机垂直细纹
11	垂直棋盘	23	随机选取一种特效

功能：获得不同的网页切换效果。

实现过程

(1) 在<body>标记中的窗体载入事件中调用自定义函数shadow()，代码如下。

```
<body onload=dream()>
```

(2) 在页面中添加一个层，用于动态显示文字，代码如下。

```
<DIV id=Tdiv style="FILTER: revealTrans(Transition=12,
Duration=2); HEIGHT: 30px; LEFT:
131px; POSITION: absolute; TEXT-ALIGN: center; TOP:
155px; WIDTH: 500px"></DIV>
```

(3) 编写用于实现梦幻文字效果的JavaScript代码。

用自定义数组textstr和textcolor来保存字符串及变换的颜色。其代码如下。

```
<SCRIPT Language="JavaScript">
textstr = new Array()
textstr[0] = "明日科技"
textstr[1] = "欢迎购买明日科技图书"
textstr[2] = "明日科技图书是您学习的最佳材料"
textstr[3] = "如有意见请到www.mingrisoft.com网站中"
textcolor = new Array()
textcolor[0] = "003399"
textcolor[1] = "66CC00"
textcolor[2] = "FF3300"
textcolor[3] = "0000FF"
var i = 0
```

自定义函数dream(), 以增加点的形式来显示文字。其代码如下。

```
functiondream() {
    Tdiv.filters[0].apply();
    Tdiv.innerHTML= "<span
style=' color:"+textcolor[i]+"'font-size: '20px';filter:
    revealTrans(Transition=10,Duration=8)'>"+textstr[i]+
</span>";
    Tdiv.filters[0].play();
```

```
if (i >= textstr.length-1) {i = 0}
else {i++}
setTimeout("dream()", 5000);
}
</SCRIPT>
```

举一反三

根据本实例，读者可以：

制作一个梦幻图片的显示效果；

用梦幻效果来打开一个用层制作的窗体。

## 实例172 屏风文字

本实例可以美化界面

实例位置：光盘\mingrisoft\05\172

实例说明

在浏览一些网站时，经常会看到一些屏风文字，本实例将屏风文字以动态的方式进行显示，运行结果如图5.51所示。

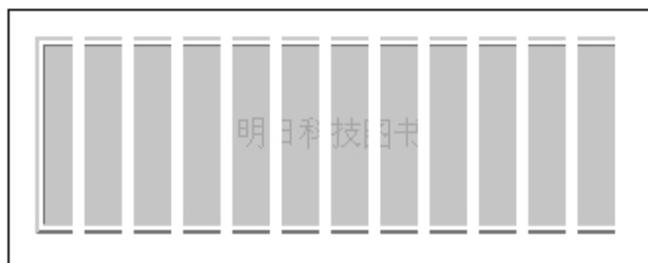


图5.51 屏风文字

技术要点

本实例的设计思想是在页面中添加 12 个层，并在每个层中添加相同的一行一列的表格，然后将每个层重复叠加在一起，再用style样式中clip的rect属性在每个层指定的局部区域中使宽度不断地放大，这样将显示这个区域中的表格内容。下面对rect属性进行详细说明。

rect属性的语法格式如下。

clip : auto | rect( number number number number )

参数说明如下。

- auto: 对象无剪切。

- rect (number number number number) : 依据上-右-下-左的顺序提供自对象左上角为 (0,0) 坐标计算的4个偏移数值, 其中任一数值都可用auto替换, 即此边不剪切。

功能: 检索或设置对象的可视区域。区域外的部分是透明的, 必须将 position 属性的值设为absolute, 此属性方可使用。

实现过程

(1) 编写用于实现屏风文字的JavaScript代码。

定义一些全局变量, 以及在页面中添加12个层。其代码如下。

```
<script language="JavaScript">
var textwidth=300;    //设置表格的宽度
var textheight=100;  //设置表格的高度
var content="";
var font_face="宋体";
var font_color="#87BC78";
var font_size=5;
var bg_ticker="#A7D2E4";
var x_finalpos=10;   //设置层的位置
var y_finalpos=10;
var x_slices=12;     //设置层的个数
var i_message=1;
var cli=1;
var cliptop=0;
var clipbottom=textheight;
```

```

varwidth_slice=Math.ceil(textwidth/x_slices);
varmessage="明日科技图书";
//在页面中添加12个层
for (i=0;i<=x_slices;i++)document.write("<span
id='s"+i+"' style='POSITION: absolute;
'></span>");

```

自定义函数initiate(), 用于在每个层中添加相同的表格, 并将层放在同一个位置。代码如下。

```

function initiate() { //在每个层添加相同的表格, 并将层
放在同一个位置

```

```

    content+="

```

```

    for (i=0;i<=x_slices;i++) {
        varTdiv=eval("s"+i);

```

```

        Tdiv.innerHTML=content; //利用innerHTML属性,

```

动态向层中添加表格

```

        var sty=eval("document.all.s"+i+".style");
        sty.posLeft=x_finalpos; //设置层的位置
        sty.posTop=y_finalpos;

```

```

    }

```

```
    i_message++;
    openlamellar();
}
```

自定义函数openlamellar(), 将多个层按指定位置进行局部截取, 并使它们的宽度逐渐变大。其代码如下。

function openlamellar() { //局部截取层, 并使它的宽度逐渐变大

```
    clipleft=-width_slice;
    clipright=0;
    if (cli<=width_slice) {
        for (i=0;i<=x_slices;i++) {
            var sty=eval("document.all.s"+i+".style");
            sty.clip="rect("+cliptop+" "+clipright+"
"+clipbottom+" "+clipleft+)";
            clipleft+=width_slice;
            clipright=clipleft+cli;
        }
        cli++;
        var timer=setTimeout("openlamellar()",20);
    }
    else clearTimeout(timer);
}
</script>
```

(2) 在<body>标记的窗体载入事件中调用自定义函数initiate(), 代码如下。

```
<body onload="initiate()">
```

举一反三

根据本实例，读者可以：  
制作一个文字的打印效果；  
制作一个闪烁的星空。

## 实例173 文字阴影

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\05173

实例说明

在一些网站中，为了使文字具有立体效果，一般在文字的后面加上阴影效果，使文字更加美观。本实例将实现这一效果。运行结果如图5.52所示。



图5.52 文字阴影

技术要点

本实例主要是通过层的style样式的filter滤镜中的shadow属性来设置层的阴影效果。下面对shadow属性进行详细说明。

shadow语法格式如下。

filter:Shadow(Color=color, Direction=direction)

参数说明如下。

- Color: #rrggbb 格式。
- Direction: 角度， $0^{\circ}$  ~  $315^{\circ}$ ，步长为 $45^{\circ}$ 。

功能：创建偏移固定影子。

实现过程

(1) 编写用于实现文字阴影的JavaScript代码。

```
<script language="javascript">
```

```
function shadow() {  
    var sha='<div style="font-size:40px;  
color:#CC00FF;FILTER:glow(color=#9900FF,  
    strength=0)  
shadow(color=#6699FF,direction=120);WIDTH:100%">明日科技  
</div>';  
    document.write(sha);  
}  
</script>
```

(2) 在<body>标记中的窗体载入事件中调用自定义函数 shadow(), 代码如下。

```
<body onLoad="shadow();">
```

举一反三

根据本实例，读者可以：

制作一个动态的文字阴影效果；

制作一个图片的阴影效果。

## 5.9 其他

在制作网页时，对文字的处理还有许多方式，因篇幅有限，下面我们通过具体的几个应用实例来进行介绍。

### 实例174 随机显示文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\05\174

实例说明

在浏览网页时，我们经常会看到一些新闻随机性地显示其内容，本实例将通过一个简单的方法来实现其效果，如图5.53、图5.54所示。

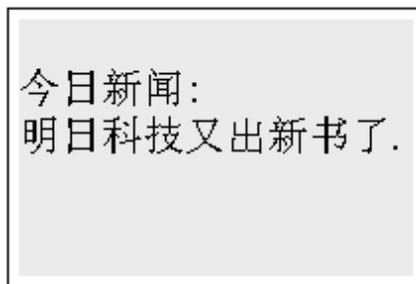


图5.53 随机显示文字

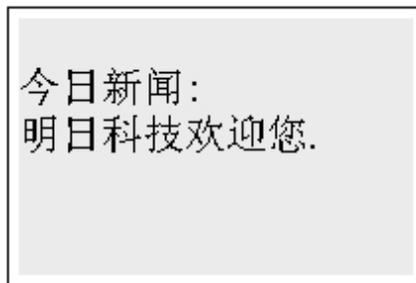


图5.54 显示下一个随机文字

技术要点

本实例主要是用 Math 对象中的 random() 方法来随机获取数组中的元素，并用层的innerHTML属性将数组中的信息以动态形式添加到层中。window对象中的setInterval()方法用于在一定间隔时间重复显示随机获取的数组信息。<BR>为换行标记。

### 实现过程

(1) 在页面中添加一个层，并设置id名为div1，代码如下。

```
<div id="div1"></div>
```

(2) 编写用于实现随机显示文字的JavaScript代码。

自定义一个数组，用于保存随机显示的文字，并对层进行初始化设置。其代码如下。

```
<script language="JavaScript">
var arr=newArray("今日新闻:<BR>明日科技又出新书了。","今日新闻:<BR>明日科技网站为:www.mingrisoft.com","今日新闻:<BR>明日科技欢迎您。","今日新闻:<BR>明日科技图书是您学习的好帮手。");
div1.style.backgroundColor="#CCFF00";
div1.style.width="200px";
div1.style.height="130px";
div1.style.visibility="hidden";
div1.style.position="absolute"
```

自定义函数text()，用于在层中随机显示数组中的文字。其代码如下。

```
function text() {
var n=Math.floor(Math.random()*3);
div1.style.visibility="visible";
div1.innerHTML="<BR>"+arr[n];
```

```
div1.style.width="200px";
div1.style.height="130px";
}
setInterval("text()",2000);
</script>
```

举一反三

根据本实例，读者可以：

随机显示广告；

制作一个猜数的小游戏。

## 实例175 文字加密及解密

本实例可以提高网络安全性

实例位置：光盘\mingrisoft\05\175

实例说明

在浏览一些网页时，我们经常会看到一些关键性的文字或代码被加密了，一般的浏览者无法获得其信息。本实例将对指定的文本进行加密及解密。运行结果如图5.55、图5.56所示。

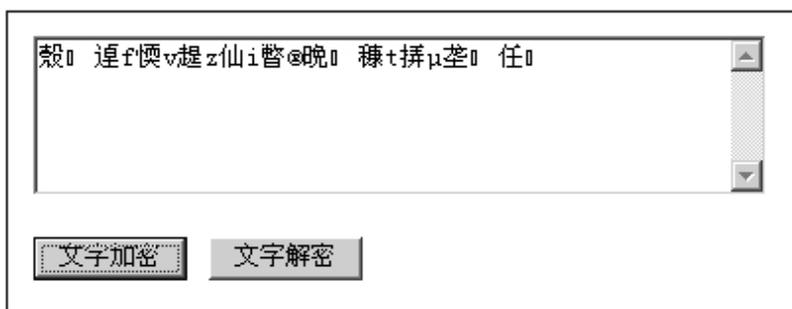


图5.55 文字加密



图5.56 文字解密

### 技术要点

本实例主要是用string对象中的charCodeAt()方法和string对象中的fromCharCode()方法对字符的ASCII编码进行获取和修改。下面对这两个方法进行详细说明。

charCodeAt的语法格式如下。

stringObj.charCodeAt(index)

参数说明如下。

- stringObj: 必选项，一个string对象或文字。
- index: 必选项，指定字符基于0的索引。如果在指定index处没有字符，则返回NaN。

功能：返回指定字符的Unicode编码。

fromCharCode的语法格式如下。

String.fromCharCode(number1, number2, ...)

number1, number2, ... ASCII 编码。

功能：用于将ASCII编码变为字元。

### 实现过程

(1) 在页面中添加一个层，并设置id名为div1，代码如下。

```
<p><textarea id="text1" name="textfield" cols="50"
rows="5">欢迎您购买明日科技图书
</textarea></p>
```

```
<input type="button" name="Button1" value="文字加密"
onClick="Encrypt(text1.value)">
```

```
<input type="button" name="Button2" value="文字解密"
onClick="unEncrypt(text1.value)">
```

(2) 编写用于实现文字加密及解密的JavaScript代码。

自定义函数Encrypt(), 将文本框中的文字进行加密。其代码如下。

```
<script language="JavaScript">
var p=0;
functionEncrypt(Text) {
    p=p+1;
    output=newString;
    alterText=newArray();
    varCost=newArray();
    TextSize=Text.length;
    for (i=0; i<TextSize; i++) {
        idea=Math.round(Math.random() *111)+77;
        alterText[i]=Text.charCodeAt(i)+idea;
        varCost[i] = idea;
    }
    for (i = 0; i < TextSize; i++) {
        output+= String.fromCharCode(alterText[i],
varCost[i]);
    }
    text1.value=output;
}
```

自定义函数unEncrypt()，将文本框中的文字进行解密。其代码如下。

```
function unEncrypt(Text) {  
    if (p>0) {  
        p=p-1;  
        output = new String;  
        alterText1 = new Array();  
        varCost1 = new Array();  
        TextSize = Text.length;  
        for (i = 0; i < TextSize; i++) {  
            alterText[i] = Text.charCodeAt(i);  
            varCost[i] = Text.charCodeAt(i+1);  
        }  
        for (i = 0; i < TextSize; i = i+2) {  
            output+= String.fromCharCode(alterText[i]-  
varCost[i]);  
        }  
        text1.value=output;  
    }  
}  
</script>
```

举一反三

根据本实例，读者可以：

实现对文本文件进行加密；

实现当进入网页浏览的不是注册用户时，将网页中的部分文字进行加密。

## 第6章 超级链接特效

超级链接样式

超级链接控制

超级链接特效

## 6.1 超级链接样式

超级链接是互联网的基础，是Web网站的根本，也是我们上网时将当前页跳转到另一个页面的主要手段。在一些网站中或多或少都会有一些超级链接。本节将通过 JavaScript 脚本来修改超级链接的字体样式以改变超级链接的一些显示效果。

### 实例176 改变超级链接字体样式

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\06\176

实例说明

一般网站中都有很多超级链接，有时当我们将鼠标移动到某一超级链接上时，此超级链接就会以不同的字体样式显示，例如超级链接的字体样式显示为斜体、粗体、下划线或是删除线等。本例将通过 JavaScript 改变超级链接字体样式，如图6.1所示。



图6.1 改变超级链接字体样式

### 技术要点

本实例中应用了字体样式中的fontWeight、fontStyle 以及 textDecoration 属性，通过设置其属性值，改变超级链接的字体样式。

### 实现过程

(1) 当鼠标移动到超级链接时改变指定链接的字体样式，代码如下。

```
<script language="javascript">
function onmovein(v)
{
    if (v=="a")
    {
        a.style.fontWeight= "Bold";           //粗体
    }
    if (v=="b")
```

```

{
  b.style.fontStyle= "Italic";           //斜体
}
if (v=="c")
{
  c.style.textDecoration= "underline";    //下划线
}
if (v=="d")
{
  d.style.textDecoration= "line-through"; //删除线
}
if (v=="e")
{
  e.style.fontWeight= "Bold";           //粗体
  e.style.fontStyle= "Italic";         //斜体
}
}
</script>

```

(2) 当鼠标移出超级链接时，恢复超级链接的字体样式，代码如下。

```

function onmouseout()
{
  a.style.fontWeight = "normal";
  b.style.fontStyle = "normal";
  c.style.textDecoration = "none";
  d.style.textDecoration = "none";
  e.style.fontStyle = "normal";
}

```

```
e.style.fontWeight = "normal";  
}
```

(3) 在超级链接的onMouseOver事件和onMouseOut事件中调用自定义的JavaScript函数onmovein()和onmoveout(),代码如下。

```
<body vlink="#FF3300" bgcolor="#CCFFFF">  
  <a href="#" name="a" onMouseMove="onmovein('a');" onmouseover="onmoveout();">用户注册</a> |  
  <a href="#" name="b" onMouseMove="onmovein('b');" onmouseover="onmoveout();">用户登录</a> |  
  <a href="#" name="c" onMouseMove="onmovein('c');" onmouseover="onmoveout();">查看留言</a> |  
  <a href="#" name="d" onMouseMove="onmovein('d');" onmouseover="onmoveout();" onclick="window.location.reload();">刷新页面</a> |  
  <a href="#" name="e" onMouseMove="onmovein('e');" onmouseover="onmoveout();">版主登录</a>  
</body>
```

举一反三

根据本实例,读者可以:

在设计页面中的超级链接时,如果认为某个超级链接有特别意义,可以使用本实例介绍的方法来改变链接的属性。

## 实例177 当鼠标移动到超级链接时改变超级链接颜色

本实例是一个人性化的程序

实例位置: 光盘\mingrisoft\06\177

实例说明

本实例通过修改超级链接的颜色样式来修改超级链接颜色，当用户将鼠标移动到超级链接上时，超级链接颜色将变为红色，如图6.2所示。



图6.2 当鼠标移动到超级链接时改变超级链接颜色

### 技术要点

本实例主要应用超级链接样式中的color属性修改超级链接的前景颜色。

### 实现过程

(1) 当鼠标移动到超级链接时，将超级链接文字颜色设置为红色，代码如下。

```
<script language="javascript">
function color1()
{
    a.style.color="#FF0000";
}
</script>
```

(2) 当鼠标移出超级链接时，将超级链接文字颜色设置为黑色，代码如下。

```
<script language="javascript">
function color2()
{
    a.style.color = "#000000";
}
</script>
```

(3) 在页面中添加一个超级链接，并将onMouseOver事件和onMouseOut事件设置为自定义的JavaScript函数color1()和color2()，代码如下。

```
<body vlink="#000000">
<a name="a" href="http://www.mingrisoft.com"
onMouseMove="color1();" onMouseOut="color2();" style="text-
decoration:none;">吉林省明日科技有限公司</a><p>
</body>
```

举一反三

根据本实例，读者可以：

实现当鼠标移动到超级链接时，使超级链接链接的字体变粗。

实现当超级链接获取焦点时，使超级链接快速闪动。

## [实例178 改变超级链接背景色](#)

本实例是一个美化界面的程序

实例位置：光盘\mingrisoft\06\178

实例说明

在一些网页中为了突出超级链接被选中时的状态，需要设置超级链接的背景颜色，以使超级链接更具有焦点性。本实例实现了当鼠标移动到超级链接时，设置超级链接的背景颜色为红色，如图6.3所示。



图6.3 改变超级链接背景色

### 技术要点

本实例中主要应用了超级链接样式中的backgroundcolor 属性修改超级链接的背景色。

### 实现过程

(1) 当鼠标移动到超级链接时，显示超级链接的背景色，代码如下。

```
<script language="javascript">
function setBackColor1()
{
    aa.style.backgroundColor = "#FF0000";
}
</script>
```

(2) 当鼠标移出超级链接时，将超级链接背景色设置为白色，代码如下。

```
<script language="javascript">
function setbackcolor2()
{
    aa.style.backgroundColor = "#FFFFFF";
}
</script>
```

(3) 在页面中添加一个超级链接，并将onMouseMove和onMouseOut设置为JavaScript所定义的setbackcolor1()函数和setbackcolor2()函数，代码如下。

```
<body vlink="#0066FF">
<a name="aa" href="http://www.mingrisoft.com"
onMouseMove="setbackcolor1();" onMouseOut="setbackcolor2();"
style="text-decoration:none; font-size: 18px; color:
#FFFF00;"> 吉林省明日科技有限公司</a>
<p>
</body>
```

举一反三

根据本实例，读者可以：

在制作单个超级链接时改变链接背景色。

## 6.2 超级链接控制

在实际网站开发过程中，经常会遇到对超级链接进行控制的情况，例如建立E-mail超级链接，返回默认主页，将网站设为首页，获取页面中的全部超级链接等。下面将通过具体实例介绍如何通过JavaScript控制超级链接。

### 实例179 建立E-mail超级链接

这是一个实用的程序

实例位置：光盘\mingrisoft\06\179

实例说明

在网站的适当位置添加建立E-mail的超级链接，不但可以帮助用户快速启动电子邮件软件（例如 Outlook），发送邮件，而且还可以避免因 E-mail地址输入错误而造成邮件不能正确到达的情况。例如，在明日科技的软件网站中，就应用了该功能。运行本实例，单击“联系我们”超级链接将启动 Outlook 软件，并创建新邮件，在新创建邮件的“收件人”文本框中将显示收件人的E-mail地址，在“抄送”文本框中将显示抄送人的E-mail地址，在“主题”文本框中显示主题内容，如图6.4所示。



图6.4 建立E-mail超级链接

### 技术要点

本实例主要应用JavaScript建立E-mail超级链接，当浏览者单击E-mail超级链接后，系统会启动默认的电子邮件软件进行E-mail的发送。

通过JavaScript建立E-mail超级链接的语法格式如下。

```
window.location.href="mailto:mail?  
subject=strSubject&CC=strCc&BCC=strBcc"
```

参数说明如下。

- mail: 代表收件人邮件地址，例如  
mingrisoft@mingrisoft.com。

- strSubject: 代表电子邮件主题，例如，“用户反馈！”。

- strCc: 代表抄送收件人。

- strBcc: 代表暗送收件人。

注意：上面的参数均可以省略，如果省略，将只启动电子邮件软件，并不设置收件人地址等信息。

### 实现过程

(1) 编写自定义函数mailto(), 用于启动Outlook, 并新建一封邮件, 该函数包括3个参数, 分别是mail (用于指定接收邮件的邮箱地址)、subject (用于指定主题内容) 和cc (用于指定抄送接收人), 无返回值, 代码如下。

```
<script language="javascript">  
function mailto(mail,subject,cc) {  
    window.location.href="mailto:"+mail+"?  
subject="+subject+"!&CC="+cc;  
}  
</script>
```

(2) 在页面中添加一个“联系我们”超级链接, 并将该超级链接的href属性设置为“#”, 在onClick事件中调用mailto()函数, 代码如下。

```
<a href="#"  
onClick="mailto('mingrisoft@mingrisoft.com','用户反  
馈!','mingrisoft@mingrisoft.com')">联系我们</a>
```

举一反三

根据本实例, 读者可以:

为指定的用户发送电子邮件;

实现在线投放稿件或简历功能。

## [实例180 获取页面中的全部超级链接](#)

本实例可以方便操作、提高效率

实例位置: 光盘\mingrisoft\06\180

实例说明

在设计网站时，有时为了页面的整体效果，会将导航菜单设置为通过 JavaScript 实现的下拉菜单，但这种菜单并不是所有浏览器都支持。为了使不支持该功能的浏览者也能正常访问网站的子链接，可以在页面中加入获取页面中全部超级链接的功能。运行本实例，单击“获取全部超级链接”超级链接将打开“获取全部超级链接”页面，在该页面中将显示原页面中的全部超级链接，如图6.5所示。



图6.5 获取页面中的全部超级链接

### 技术要点

本实例首先通过 DOM提供的 `getElementsByTagName()` 方法获取页面中的全部超级链接，然后应用 `window` 对象的 `open()` 方法打开一个新的窗口，最后通过 `for` 循环和超级链接对象的 `outerHTML` 属性将获取的超级链接显示在新打开的窗口中。下面将对DOM提供的 `getElementsByTagName()` 方法和超级链接对象的 `outerHTML` 属性进行详细介绍。

#### (1) `getElementsByTagName()` 方法。

DOM（文档对象模型）是表示文档和访问、操作构成文档的各种元素的应用程序的接口。DOM树的根节点是Document对象，通过Document对象可以访问`getElementsByTagName()`方法。

`getElementsByTagName()`方法用来返回一个NodeList对象（可以作为只读数组处理），该对象存放文档中具有指定标记名的Element节

点，它的存放顺序与源文档中出现的顺序相同。

getElementsByTagName() 方法的语法格式如下：

```
document.getElementsByTagName(String tagName)
```

参数说明如下。

● **TagName**：HTML 标记名，不包括定界符“<”和“>”，对于 HTML 文档，标记名不区分大小写，例如要获取 HTML 页面中的全部超级链接，可以使用 A 或 a 标记名。如果要返回页面中的所有元素，可以使用一个星号（如 document.getElementsByTagName("\*") 可以返回页面中的所有元素）。

(2) outerHTML 属性。

为方便对文档进行操作，DOM 还为所有的元素提供了 outerHTML 属性，该属性将返回指定元素完整的 HTML 代码，包括元素本身。

实现过程

(1) 编写自定义函数 getUrl()，用于获取页面中的全部超级链接，该函数无参数和返回值，代码如下。

```
<script language="javascript">
function getUrl() {
    var allLink=document.getElementsByTagName("A");
    var linktotal=allLink.length;
    varnewWin=window.open("")
    newWin.document.write("<title>获取的全部超级链接
</title>");
    newWin.document.write("<style>a:hover {font-
size:9pt;color:#FF6600;}");
    newWin.document.write("a {font-size:9pt;text-
decoration:none;color:#676767;}</style>");
    for (i=0;i<linktotal;i++){
```

```
        newWin.document.write(allLink[i].outerHTML+"<br>")
    }
}
</script>
```

(2) 在页面中适当的位置添加一个“获取全部超级链接”的超级链接，并将该超级链接href属性设置为“#”，在onClick事件中调用getUrl()函数，代码如下。

```
<a href="#" onClick="getUrl()" class="word_white">[获取全部超级链接]</a>
```

举一反三

根据本实例，读者可以：

获取页面中的全部图片；

获取页面中的表格。

## 实例181 将网站设为首页

本实例是一个实用的程序

实例位置：光盘\mingrisoft\06\181

实例说明

在网站中加入“设为首页”超级链接，可以使用户很方便地将该网站设置为IE的默认主页，这样用户在下次启动IE时将自动打开该网站。例如，在明日科技的软件网站中，就应用了该功能。运行本实例，单击“设为首页”超级链接将弹出“是否将

“http://www.mingrisoft.com”设为主页？”的对话框，如图6.6所示，单击“是”按钮，即可将http://www.mingrisoft.com设置为IE的默认主页。



图6.6 将网站设为首页

## 技术要点

实现本实例首先应用超级链接对象的style属性将CSS样式behavior设置为主页的样式，然后通过超级链接对象的sethomepage()方法指定CSS样式homepage的值为指定的URL地址，该地址将被设置为IE的默认主页地址。

## 实现过程

(1) 编写自定义函数defaultPage(), 用于将指定网站设为IE的默认首页，该函数只包括一个参数url，用于指定网站的URL地址，该函数无返回值，代码如下。

```
<script language="javascript">
functiondefaultPage(url){
    defaultHomePage.style.behavior="url(#default#homepage)"
;
    defaultHomePage.sethomepage(url);
}
</script>
```

(2) 在页面中添加一个“设为首页”超级链接，并将该超级链接 href 属性设置为“#”，在 onClick 事件中调用 defaultPage() 函数，将指定的网站设为首页，代码如下。

```
<a href="#" name="defaultHomePage" class="word_white"
onClick="defaultPage('http://www.mingrisoft.com')">设为首页</a>
```

举一反三

根据本实例，读者可以：

在网上论坛中加入设为首页功能；

在企业门户网站中加入设为首页功能。

## 实例182 单击鼠标右键自动链接到指定网站

这是一个自娱自乐的实例

实例位置：光盘\mingrisoft\06\182

实例说明

当用户浏览网站时，经常利用鼠标右键进行快捷键操作。为了提高网站的安全性和有效利用资源，可以在页面中添加单击鼠标右键自动链接到指定网站的功能。运行本实例，在页面的任意位置单击鼠标右键都将弹出“欢迎访问明日科技网站！”的对话框，单击“确定”按钮，即可打开新的窗口进入到明日科技网站的首页，如图6.7所示。



图6.7 单击鼠标右键自动链接到指定网站

### 技术要点

本实例主要通过event对象的button属性实现。button属性用于指示哪一个鼠标键被按下，0表示无键被按下；1表示左键被按下；2表示右键被按下；3表示在按下左键的同时按下右键；4表示中键被按下。

### 实现过程

(1) 编写自定义函数goSite()，用于单击鼠标右键时，弹出提示对话框并打开新窗口访问指定网站，代码如下。

```
<script language="javascript">
function goSite() {
    if(event.button==2) {
        alert("欢迎访问明日科技网站!");
        window.open("http://www.mingrisoft.com");
    }
}
```

```
        return false;
    }
}
</script>
```

(2) 当按下鼠标按键时触发goSite方法，代码如下。

```
<script language="javascript">
document.onmousedown=goSite;
</script>
```

举一反三

根据本实例，读者可以：

实现在指定的元素上按下鼠标右键链接到指定网站；

实现在不同元素上按下鼠标右键链接到不同网站。

## 实例183 单击超级链接将本页加入收藏夹

本实例是一个实用的程序

实例位置：光盘\mingrisoft\06\183

实例说明

为了方便用户能够迅速地浏览网站，在一些网站的链接中都会提供一个“加入收藏夹”的链接，当用户单击此超级链接时，则会弹出一个对话框，用户可以通过对话框中的一些设置，将网站加入到收藏夹中，当用户再次访问该网站时，只需在收藏菜单中选择被收藏的网站即可浏览该网站，如图6.8、图6.9所示。

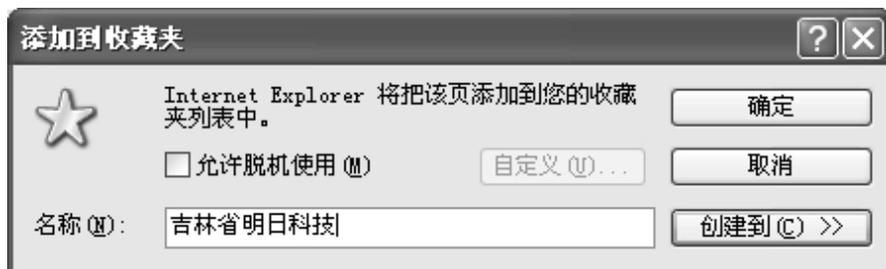


图6.8 添加到收藏夹对话框



图6.9 在收藏菜单中显示被收藏的网站

### 技术要点

本实例主要应用了window.external.AddFavorite( )方法，使用此方法可以提供对话框让用户将指定的URL加入到收藏夹。语法格式如下。

window.external.AddFavorite(sURL, sTitle)

参数说明如下。

- sURL：指定要加入收藏夹的URL 地址。
- sTitle：指定加入收藏夹后显示的标题名称。

例如：

window.external.AddFavorite(http://www.mingrisoft.com, 吉林省明日科技)

### 实现过程

(1) 使用JavaScript代码编写加入收藏夹代码。

```
<script language="javascript">  
function addfavorite()  
{
```

```
links.href="javascript:window.external.AddFavorite('http://www.mingrisoft.com', '吉林省明日科技');  
}  
</script>
```

(2) 在<body>区域中制作一个超级链接，并设置其单击事件为addfavorite()函数，代码如下。

```
<body>  
<a href="#" name="links" onClick="addfavorite();" > 单击链接将本站加入收藏夹</a>  
</body>
```

举一反三

根据本实例，读者可以：

在个人网站中提供单击超级链接将本页加入收藏夹的功能。

## 实例184 访问指定的链接地址

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\06\184

实例说明

当用户在浏览一个网站时，有时可能会访问其他网站，但是又不想关闭当前正在浏览的网页，此时，可以在本站中提供一个访问新网页的功能，这样可以避免用户再次打开一个IE浏览器。运行实例，在“请输入您要访问的URL地址”文本框中输入网站地址，单击“链接”按钮，即可链接到该网站，如图6.10所示。



图6.10 访问指定的链接地址

### 技术要点

本实例中主要应用了window对象的open()方法实现打开一个浏览器窗口，语法格式如下。

```
window.open(url,windowname[,location]);
```

参数说明如下。

- url: 目标窗口的URL。如果URL 是一个空字符串，则浏览器将打开一个空白窗口，允许用write()方法创建动态HTML。
- windowname: window对象的名称。
- location: 对窗口属性进行设置，其可选值如表6.1 所示。

表6.1 对窗口属性进行设置的可选值

可 选 值	说 明
width	窗口的宽度
height	窗口的高度
scrollbars	是否显示滚动条
resizable	设定窗口大小是否固定
toolbar	浏览器工具条，包括后退及前进按钮等
menubar	菜单条，一般包括有文件、编辑及其他一些条目
location	定位区，也叫地址栏，可以输入 URL 的浏览器文本区
direction	更新信息的按钮

### 实现过程

(1) 编写用户单击“链接”按钮时执行的JavaScript自定义函数。

主要用于判断用户在地址栏中是否输入内容，如果输入则打开一个新窗口以显示用户所要访问的网站，代码如下。

```
<script language="javascript">
function openform()
{
if (document.form1.textfield.value== "")
{
    alert("请输入URL地址！");
}
else
{
    window.open(document.form1.textfield.value,'newwindow',
'width=800,height=600,
top=10,left=10,toolbar=no,menubar=no,resizable=no,location=
no ,statusbar=yes');
    document.form1.textfield.clear;
}
}
</script>
```

(2) 编写页面设计代码，当用户单击“链接”按钮时调用 openform() 函数。

```
<body>
<form name="form1"action="">
    <font color="#0033FF" size="3pt">请输入您要访问的URL地
址： &nbsp; </font>
    <inputname="textfield"
type="text"value="www.mingrisoft.com">
```

```
<input type="submit" name="Submit" value="链接" onClick="openform();">
</form>
<a href="#" style="text-decoration:none ">首页 |</a>
<a href="#" style="text-decoration:none ">图书介绍 |
</a>
<a href="#" style="text-decoration:none ">新书预告 |
</a>
<a href="#" style="text-decoration:none ">勘误发布 |
</a>
<a href="#" style="text-decoration:none ">技术支持 |
</a>
<a href="#" style="text-decoration:none ">明日论坛</a>
</body>
```

举一反三

根据本实例，读者可以：

在制作网页导航时，可以增设此功能为用户提供方便；

将此实例进一步改进，将其设置为个人网站内部的搜索工具。

## [实例185 返回默认主页](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\06\185

实例说明

本实例使用 JavaScript 制作一个快速回到默认主页的超级链接，本例的功能类似于用户在IE浏览器中单击工具栏中的 按钮。运行本实例前，首先用户要在 IE的属性对话框中设置一个默认的URL地

址，当用户单击返回默认主页的超级链接后，则IE浏览器会跳转到指定的URL页中，如图6.11、图6.12所示。



图6.11 单击返回默认主页



图6.12 将页面跳转到默认主页

## 技术要点

本实例主要应用了 `window.location.href` 方法，使用该方法可以使当前页面跳转到指定页面中。

语法如下。

```
window.location.href = 'url';
```

参数说明如下。

- `url`: 指定网站的URL 地址。

实现过程

(1) 编写用于实现返回默认主页的JavaScript自定义函数，代码如下。

```
<script language="javascript">  
function gohome ()
```

```
{
  if (document.all)
  {
    window.location.href="about:home"
  }
}
</script>
```

(2) 在页面中添加一个超级链接，并将超级链接的单击事件设置为gohome()函数，当用户单击超级链接时，使当前页面跳转到默认页。

```
<body>
<a href="#" onClick="gohome();">返回默认主页</a>
</body>
```

举一反三

根据本实例，读者可以：

在链接中添加一项“返回默认首页”的链接项，以方便网站浏览者使用。

## [实例186 导航链接](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\06\186

实例说明

在一些网站中，为了方便用户快速地浏览其他网站，该网站中会在指定位置提供一个导航栏，这些导航栏一般都是以列表形式或菜单形式显示。本实例将使用列表框作为导航栏，当用户选择导航栏中的

某一超级链接后单击“去看看”按钮时，程序会打开用户在列表中选择网站，如图6.13所示。

### 技术要点

本实例主要使用了switch case 语句来实现链接导航。switch case 语句是最典型的多路分支结构，它会根据一个表达式的值，来执行不同的分支语句。

在本实例中将以用户在列表框中所选择项目的索引值作为switch 的表达式，然后根据表达式的值来访问指定的链接地址。

### 实现过程

(1) 编写实现导航链接的自定义 JavaScript 函数，主要根据用户在列表中所选择的项目来浏览指定的网页，代码如下。

```
<script language="javascript">
function text(index)
{
    index = document.form1.select1.selectedIndex;
    switch (index)
    {
        case 0 :
            window.location = "http://www.mingrisoft.com";
            break;
        case 1 :
            window.location = "http://www.dearbook.com";
            break;
        case 2 :
            window.location = "http://www.dangdang.com";
            break;
        case 3 :
```

```

window.location = "http://www.huachu.com.cn";
break;
default :
window.location = "#";
}
}
</script>

```



图6.13 导航链接

(2) 在下面的代码中添加了一个列表框控件，并在其中添加了4个选项，同时设置第1个选项为被选中状态，最后将按钮的单击事件设置为text(0)。代码如下。

```
<body>
```



## 6.3 超级链接特效

在制作网站时，有时为了使超级链接更加突出，或是使浏览者快速了解超级链接的作用，需要为超级链接加入特殊的效果，例如快速闪动页面中的超级链接，显示超级链接提示信息等。下面将通过具体实例介绍网站中常用的超级链接特效。

### 实例187 快速闪动页面中的超级链接

本实例是一个美化界面程序

实例位置：光盘\mingrisoft\06\187

实例说明

在一些网站中为了使某条超级链接信息或广告更具有视觉效果，特意将其设置为快速闪动效果，这样不但会吸引许多浏览者的注意力，同时也扩大了对此网页的单击率。运行本实例，页面中的“吉林省明日科技”将快速闪动。实例运行结果如图6.14所示。



图6.14 快速闪动页面中的超级链接

### 技术要点

本例主要通过循环数组中的颜色值来改变超级链接的前景色。本例中应用了数组arguments[], 通过arguments[]的RGBArray()函数可以处理可变数量的参数。arguments对象的length属性包含了传递给函数的参数的数目。对于arguments对象所包含的单个参数, 其访问方法与数组中所包含的参数的访问方法相同。

### 实现过程

(1) 在<body>区域中编写快速闪动的超级链接文字的JavaScript自定义函数, 代码如下。

```
<SCRIPTLANGUAGE="JavaScript">
functionRGBArray()
{
    for (var i=0;i<RGBArray.arguments.length;i++)
    {
        this[i]=RGBArray.arguments[i];
    }
    this.length=RGBArray.arguments.length;
}
var
color=newRGBArray("red","green","blue","purple","tan","yellow",
"lime","silver","gold");
var vlink=0;
function twinkling() {
vlink= (vlink+1)%color.length;
a1.style.color= color[vlink];
setTimeout("twinkling()",500);
```

```
}  
twinkling();  
</script>
```

(2) 在<body>区域中编写设计页面代码，并将超级链接“吉林省明日科技”的ID设置为“a1”，主要用于在JavaScript中调用该超级链接，代码如下。

```
<table id="table1" width="479" border="0">  
  <tr>  
    <td id="td1" width="284" height="72">  
      <font size="10pt" face="隶书">  
        <a id="a1" href="www.mingrisoft.com">吉林省明日科技  
      </a>  
    </font>  
  </td>  
  <td width="185" valign="middle"><font size="10pt"  
face="隶书"></font></td>  
</tr>  
</table>  
<p>  
<a href="#" style="text-decoration:none ">首页&nbsp;|</a>  
<a href="#" style="text-decoration:none ">图书介绍&nbsp;|  
</a>  
<a href="#" style="text-decoration:none ">新书预告&nbsp;|  
</a>  
<a href="#" style="text-decoration:none ">勘误发布&nbsp;|  
</a>
```

```
<a href="#" style="text-decoration:none ">技术支持 |  
</a>
```

```
<a href="#" style="text-decoration:none ">明日论坛</a>
```

举一反三

根据本实例，读者可以：

将网站中特殊的超级链接设置为快速闪动。

## 实例188 滚动的超级链接提示信息

本实例可以美化界面、简化操作

实例位置：光盘\mingrisoft\06\188

实例说明

本实例主要实现链接滚动特效，当用户将鼠标移动到超级链接上时，超级链接文字会变成滚动显示的字符串“明日科技最新推出《JavaScript 范例宝典》”，当鼠标移出超级链接时则会停止滚动，同时超级链接文字将显示为“吉林省明日科技有限公司”，如图6.15、图6.16所示。



图6.15 滚动链接1



图6.16 滚动链接2

技术要点

本实例主要技术要点在于函数之间的嵌套调用。首先在程序中定义一个自定义函数scrolltime()以实现字幕滚动。mousein()函数和mouseout()函数用于实现当鼠标移动到链接时，链接开始滚动指定的字符串；当鼠标移出链接时则停止滚动，并且将链接显示为“吉林省明日科技有限公司”。

### 实现过程

(1) 在</head>和<body>区域之间编写滚动链接的JavaScript自定义函数，代码如下。

```
<script language="javascript">
var scrolltext,go,word;
function scrolltime()
{
    clearTimeout(go);
    scrolltext=scrolltext.substring(1,30)+scrolltext.charAt
(0);
    Scroll.innerText=scrolltext;
    go=setTimeout("scrolltime()",150);
}
functionmousein(word)
{
    scrolltext="明日科技最新推出《JavaScript程序开发范例宝
典》";
    word.style.color="#FFFF00";
    scrolltime();
}
function mouseout(word)
{
```

```

clearTimeout(go);
Scroll.innerText="明日科技新书预告";
word.style.color="#FFFFFF";
}
</script>

```

(2) 在<body>区域中添加一个超级链接，并将onMouseOver事件设置为mousein(this)、将onMouseOut事件设置为mouseout(this)，代码如下。

```

<body vlink="#000000">
  <table width="553" height="370" border="0"
align="center">
    <tr>
      <td width="547" height="366"
background="Images/backgroup.jpg"><table width="360"
height="297" border="0" align="right">
        <tr>
          <td width="354" height="293" align="right"
valign="top"><a href="http://www.mingrisoft.com"
class="style1" onMouseOver=" mousein(this)"
onMouseOut="mouseout(this)";><span id="Scroll">明日
科技新书预告</span> </a></td>
        </tr>
      </table></td>
    </tr>
  </table>
</body>

```

举一反三

根据本实例，读者可以：  
增加超级链接的特殊效果；  
突出网站中的个性化超级链接；  
在链接上显示一些链接页中的主要内容。

## 实例189 公告栏中显示超级链接

本实例是一个实用的程序

实例位置：光盘\mingrisoft\06\189

实例说明

公告栏对于一些门户网站来说是必不可少的工具，通过公告栏可以发布一些最新的新闻资讯及公告事项。

公告栏不但可以美化页面，同时也为网站浏览者提供了许多方便。本实例中使用了 `marquee` 标记来实现公告栏中的字幕滚动效果，当用户单击公告栏中的链接时，程序会通过数组中的链接地址，将当前页面跳转到指定页面中，如图6.17所示。



图6.17 公告栏中显示超级链接

技术要点

在制作本实例时，首先将超级链接的URL地址保存到数组 `link[]` 中，超级链接文字保存到数组 `affiche[]` 中，然后通过 `document` 对象的

write()方法输出作为公告栏的表格及超级链接。在公告栏中，通过<marquee>标记实现超级链接文字的滚动。

### 实现过程

在</head>和<body>区域之间编写公告栏中滚动链接的代码。

```
<script language="javascript">
var affiche_index = 6;
link = new Array(5);
affiche = new Array(5);
link[0]= 'http://www.mingrisoft.com/bookJS.asp#';
link[1]=
' http://www.mingrisoft.com/bookdelphitechnic.asp#';
link[2]=
' http://www.mingrisoft.com/bookDelphifunction.asp#';
link[3]= 'http://www.mingrisoft.com/bookVBCompoent.asp#';
link[4]= 'http://www.mingrisoft.com/bookASPNET.asp#';
link[5]= 'http://www.mingrisoft.com/bookVC.asp#';
affiche[0] = 'JavaScript范例宝典';
affiche[1] = 'Delphi技术大全';
affiche[2] = 'Delphi函数大全';
affiche[3] = 'VB组件大全';
affiche[4] = 'ASP.NET完全手册';
affiche[5] = 'VC案例精选';
document.write("<table><tr><td background='公告栏.jpg' ;
width=154px; height=180px>");
document.write("&nbsp;&nbsp;&nbsp;<marquee scrollamount = '1'
direction = 'up' width = '130' height = '80' scrolldelay = '80'
```

```
id = table onMouseover = table.stop() onMouseout =
table.start(>");
    for (i = 0;i<affiche_index;i++)
    {
        document.write("<a href = '"+link[i]+"target='_blank'>
<font color = '#003399'>");
        document.write(affiche[i]+"</font></A><br>");
    }
document.write("</marquee>");
document.write("</td></tr></table>");
</script>
```

举一反三

根据本实例，读者可以：

在网站中以超级链接的形式显示一些最新的新闻资讯、发布信息。

## [实例190 显示超级链接站点相关信息](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\06\190

实例说明

目前，大多网站都设置了超级链接站点、友情链接之类的信息资源，随着个性化网站的增强，一些站点资源可以通过超级链接的方式显示其站点的详细资源信息。这样设计的好处不仅考虑到网站的整洁、美观程度等因素，还可以简化页面。本实例便实现了这样一个个性化的功能。首先将要显示的连接信息存储在层中，当单击链接站点

时，通过调用Javascript脚本调用该层显示其信息资源，如图6.18所示。

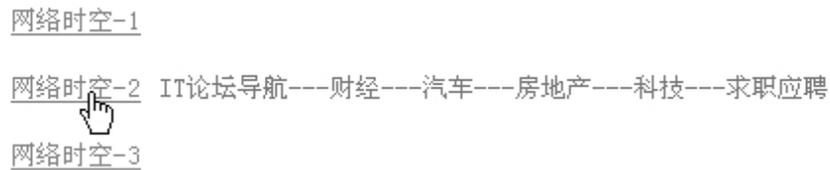


图6.18 显示超级链接站点的相关信息

### 技术要点

本实例中主要使用innerHTML方法实现当鼠标移动到超级链接时，在页面中输入指定的信息。

### 实现过程

(1) 在</head>和<body>区域之间编写显示链接站点相关信息的JavaScript自定义函数，代码如下。

```
<script language="javascript">  
var hint=newArray(3)  
hint[0]= "<font color= red>新闻---IT---游戏---影视---音  
乐---娱乐---旅游---体育</font>";  
hint[1]= "<font color= red>IT论坛导航---财经---汽车---房  
地产---科技---求职应聘</font>";  
hint[2]= "<font color= red>精美壁纸---铃声下载---游戏专  
区---奇闻奇事---时尚---邮件</font>";  
</script>
```

(2) 在<body>区域中设计一个一行两列的表格，并在表格的第一列中添加3个超级链接，代码如下。

```
<table border="0">  
<tr>  
<td valign=top width="100">
```

```
<a href="#" onMouseOver="Hints.innerHTML = hint[0];">网络  
时空-1</a>  
<p>  
<a href="#" onMouseOver="Hints.innerHTML = hint[1];">网络  
时空-2</a>  
<p>  
<a href="#" onMouseOver="Hints.innerHTML = hint[2];">网络  
时空-3</a>  
</td>  
<td width="600">  
<a id="Hints">请将鼠标指向链接</a>  
</td>  
</tr>  
</table>
```

举一反三

根据本实例，读者可以：

显示一些链接页中的主要内容；

以中文形式提示用户此链接中的相关内容。

## [实例191 显示超级链接的提示信息](#)

本实例可以美化界面、简化操作

实例位置：光盘\mingrisoft\06\191

实例说明

为了明确超级链接所表达的具体含义，可以通过Javascript脚本为超级链接文字或超级链接图片添加提示信息，当鼠标指向该超级链接文字或图片时，即可弹出相应的信息提示，并可设置提示信息的文

字属性，从而做到所见即所得，快速加深用户对超级链接文字的理解。运行效果如图6.19所示。

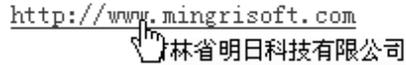


图6.19 显示超级链接的提示信息

### 技术要点

本实例中主要通过 div 层改变层的样式，来实现改变层的背景颜色、宽度、字体大小以及x和y坐标位置等。

### 实现过程

(1) 编写改变层样式的JavaScript自定义函数，代码如下。

```
<script language="javascript">
function over(hint)
{
    document.all.Tdiv.innerHTML = hint;
    document.all.Tdiv.style.position="absolute";
    document.all.Tdiv.style.backgroundColor="#FFFFCC";
    document.all.Tdiv.style.width="140px";
    document.all.Tdiv.style.fontSize="9pt";
    document.all.Tdiv.style.posLeft=window.event.x+document
.body.clientLeft+4;
    document.all.Tdiv.style.posTop
=window.event.y+document.body.clientTop+2;
    document.all.Tdiv.style.visibility = "visible";
}
</script>
```

(2) 编写鼠标移出层时隐藏层的自定义JavaScript函数，代码如下。

```
<script language="javascript">
function out()
{
    document.all.Tdiv.style.visibility = "hidden";
}
</script>
```

(3) 在<body>区域中设计一个层和一个超级链接，设置超级链接的onMouseOut事件为out()函数，onMouseOver事件为over()。同时将body的onLoad事件设置为out()函数，使页面显示时不显示层的提示信息。

```
<body onLoad="out()">
    <a style="width:100px " href="http://www.mingrisoft.com"
onMouseOut="out()" onMouseOver="over('吉林省明日科技有限公司')">
        <p>http://www.mingrisoft.com</p>
    </a>
<div id="Tdiv"></div>
</body>
```

举一反三

根据本实例，读者可以：

在链接文件或是链接图片上加入提示信息；

将页面中的所有链接统一显示为指定的提示信息。

## [实例192 半透明背景的超级链接提示](#)

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\06\192

实例说明

为网页的超级链接添加半透明背景的提示信息，可以在方便读者快速了解各超级链接作用的同时，美化了页面。运行本实例，将鼠标移动到各超级链接上时，将显示半透明背景的提示信息，例如，将鼠标移动到“图书网站”超级链接上时，将显示“单击进入明日科技图书网站！”的提示信息，如图6.20所示。



图6.20 半透明背景的超级链接提示

技术要点

在实现本实例时，首先需要在页面中加入div层，并且通过JavaScript控制层的显示或隐藏及跟随鼠标移动，然后再通过CSS样式控制层的背景为半透明效果。在页面中加入div层的语法及其属性说明，具体请读者参见实例042。

实现过程

(1) 编写用于实现显示半透明背景的超级链接提示信息的JavaScript代码。

编写自定义函数toolTip(), 用于控制层toolTipLayer (用于显示超级链接提示信息) 的显示或隐藏，并指定该层的显示内容，代码如下。

```
<script language="JavaScript">
```

```

var offsetX=-30;
var offsetY=20;
var toolTipSTYLE="";
function toolTip(title) {
    if(toolTip.arguments.length<1) { //隐藏
        toolTipSTYLE.display= "none";
    } else { //显示
        var content= "<table align=' left' ><tr><tdnowrap
class=' ToolTip' style=' padding:5px' >" +title+"</td></tr>
</table>";
        document.all("toolTipLayer").innerHTML=content;
        toolTipSTYLE.display=' block'
    }
}

```

编写自定义函数moveToMouse(), 用于控制显示超级链接提示信息层的位置, 代码如下。

```

function moveToMouse() {
    x = event.x+document.body.scrollLeft;
    y = event.y+document.body.scrollTop;
    toolTipSTYLE.left = x+offsetX;
    toolTipSTYLE.top = y+offsetY;
    return true;
}

```

编写自定义函数 initToolTips(), 用于控制显示超级链接提示信息的层跟随鼠标移动, 代码如下。

```

function initToolTips() {
    toolTipSTYLE = document.all.toolTipLayer.style;

```

```
document.onmousemove = moveToMouse;
}
</script>
```

(2) 在页面的<body>和</body>标记中间首先添加一个用于显示超级链接提示信息的层，命名为toolTipLayer，然后再调用自定义的JavaScript函数initToolTips()，代码如下。

```
<DIV id="toolTipLayer" style="POSITION: absolute"></DIV>
<SCRIPT language=JavaScript>initToolTips(); </SCRIPT>
```

(3) 在页面的每个超级链接的onMouseOver事件中调用toolTip()函数，并将提示信息作为参数传递；在onMouseOut事件中调用toolTip()函数，此时不传递参数，关键代码如下。

```
<a href="#" onMouseOver="toolTip('单击进入明日科技软件网站!');" onMouseout="toolTip();"
target="_blank">软件网站</a>
<a href="#" onMouseOver="toolTip('单击进入明日科技图书网站!');" onMouseout="toolTip();"
target="_blank">图书网站</a>
```

(4) 编写控制显示超级链接提示信息的层为半透明背景的CSS样式，代码如下。

```
<style>
.ToolTip {
    filter:Alpha(Opacity=50, style=1, Finishopacity=80);BORDER: #ff8c00 1px solid;
    BACKGROUND-COLOR: #FFFFFF;
}
</style>
```

举一反三

根据本实例，读者可以：

应用该方法在在线论坛的浏览主题信息页面中显示会员基本信息；

应用该方法显示网站公告详细信息。

## 第7章 操作表格

表格的颜色效果

对单元格的焦点进行控制

对表格的行、列进行修改

单元格的相关操作

表格的特殊效果

## 7.1 表格的颜色效果

在网络设计时，为了使表格中的数据更加突出，经常需要表格的边框和单元格变色，或者在选择某一行时，使行变色。下面将通过几个典型实例进行详细讲解。

### 实例193 闪烁的表格边框

这是一个实用的实例

实例位置：光盘\mingrisoft\07\193

实例说明

本实例是将表格的边框设置为指定的宽度，然后在窗体载入时按一定的时间间隔改变表格的外边框颜色。运行结果如图7.1、图7.2所示。

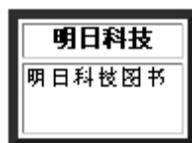


图7.1 表格边框闪烁前的效果

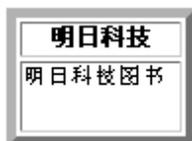


图7.2 表格边框闪烁后的效果

技术要点

本实例主要是通过setTimeout()方法按一定的时间修改表格的外边框颜色，修改表格的外边框颜色是用表格的style对象的

borderColor属性来进行修改的。下面对setTimeout()方法进行详细说明。

setTimeout()方法的语法格式如下。

```
setTimeout (function, milliseconds)
```

参数说明如下。

● function: 要调用的JavaScript 自定义函数名称。

● milliseconds: 设置超时时间 (以ms 为单位)。

功能: 超过超时时间后, 调用函数。此值可以用clearTimeout()函数清除。

注意:

①setTimeout()方法在超时时间后只调用一次函数, 而setInterval()方法是按一定时间重复调用指定的函数。

②在HTML语言中一些对象的属性中间是带“-”符号的, 例如border-color, 要想在脚本中进行应用, 应将“-”符号删除, 并将后面的首字母大写(borderColor)。

实现过程

(1) 在页面中添加一个表格, 并设置表格外边框的宽度为 5, 这样可以更直观地看到表格的变色效果, 代码如下。

```
<table id="table1" border=5>
<tr><td align=center><strong>明日科技</strong></td>
</tr>
<tr><td align=center>
<pre>明日科技图书</pre>
</td>
</tr>
</table>
```

(2) 编写用于实现表格边框变色的JavaScript代码。

```
<script language="JavaScript">
var i=0;
```

```
varColor=newArray("#0000FF","#99FF00","#660033","#CC66CC",
"#FFFF33");
functionpp(){
    if (i>Color.length-1){i=0}
    table1.style.borderColor=Color[i]; //改变表格外边框的
颜色
    i=i+1;
    setTimeout("pp()",100); //按间隔时间调用当前自定义函数
}
pp();
</script>
```

举一反三

根据本实例，读者可以：

实现动态改变表格的外边框大小；

制作一个可以动态取消的表格外边框。

## 实例194 单元格边框变色

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\07\194

实例说明

本实例是将表格中每个单元格外边框的颜色按一定的时间间隔进行改变。运行结果如图7.3、图7.4所示。

	箱数	单价
苹果	50	30
香蕉	60	35

图7.3 单元格边框变色前

	箱数	单价
苹果	50	30
香蕉	60	35

图7.4 单元格边框变色后

### 技术要点

本实例主要是通过setTimeout()方法按一定的时间改变表格中单元格的外边框颜色，修改表格单元格颜色是用单元格的style对象的borderColor属性来进行修改的，通过一个布尔型的变量来选择变化的两种颜色。

setTimeout()方法已在实例193中进行了详细的说明。

### 实现过程

(1) 在页面中添加一个宽为200的表格，并对表格中的每个单元格设置id名称，代码如下。

```
<table id="table1" width="200">  
  <tr>  
    <td width="60" id="Ttd0">&nbsp;</td>  
    <td width="58" id="Ttd1">箱数</td>  
    <td width="66" id="Ttd2">单价</td>  
  </tr>  
  <tr>  
    <td id="Ttd3">苹果</td>  
    <td id="Ttd4">50</td>  
    <td id="Ttd5">30</td>  
  </tr>  
  <tr>  
    <td id="Ttd6">香蕉</td>  
    <td id="Ttd7">60</td>  
    <td id="Ttd8">35</td>
```

```
</tr>
</table>
```

(2) 编写用于实现单元格边框变色的JavaScript代码。

```
<script language="JavaScript">
var tt=true;
table1.cellSpacing="0";//设置单元格的间隔
table1.border="2";    //设置单元格的边框宽度
function changcolor() {
    for (var i=0;i<9;i++) {    //修改所有单元格的外边框颜色
        if (tt==true) {eval("Ttd"+i).borderColor="#00FF00";}
        if (tt==false) {eval("Ttd"+i).borderColor="#6666FF";}
    }
    tt=!tt; //对布尔型的值进行改变
    setTimeout("changcolor()",200);
}
changcolor();
</script>
```

举一反三

根据本实例，读者可以：

实现动态取消表格的单元格边框；

实现鼠标指向表格时，改变单元格外边框的颜色。

## 实例195 选中的行变色

本实例可提高工作效率

实例位置：光盘\mingrisoft\07\195

实例说明

本实例是在鼠标指向表格中的任意一个单元格时，将该单元格所在行的背景颜色及字体颜色进行改变。运行结果如图7.5所示。



	姓名	性别
一班	张同学	男
二班	李同学	女

图7.5 选中的行变色

### 技术要点

本实例的主要思想是当鼠标指向表中的单元格时通过onMouseOver 事件调用自定义函数over()，从而改变单元格所在行的前景色和背景色；当鼠标纵向离开单元格时，通过onMouseOut事件将所选行的前景色和背景色改变为初始状态。

主要是通过<tr>标记的style方法的backgroundColor和color属性来改变行的背景色和前景色。

### 实现过程

(1) 在页面中添加一个宽为200的表格，并对表格中的行设置id名称，代码如下。

```
<tablewidth="200"border="1">  
  <tr  
  id="tr1"onMouseOver="over(this.id)"onMouseOut="out(this.id)"  
  >  
    <td>&nbsp;   </td>  
    <td>姓名</td>  
    <td>性别</td>  
  </tr>  
  <tr  
  id="tr2"onMouseOver="over(this.id)"onMouseOut="out(this.id)"  
  >
```

```

        <td>一班</td>
        <td>张同学</td>
        <td>男</td>
    </tr>
    <tr
id="tr3"onMouseOver="over(this.id)"onMouseOut="out(this.id)
">
        <td>二班</td>
        <td>李同学</td>
        <td>女</td>
    </tr>
</table>

```

(2) 编写用于实现选中行变色的JavaScript代码。

```

<script language="JavaScript">
function over(trname) {
    eval(trname).style.backgroundColor="0000FF";
    eval(trname).style.color="FFFFFF";
}
function out(trname) {
    eval(trname).style.backgroundColor="FFFFFF";
    eval(trname).style.color="000000";
}
</script>

```

举一反三

根据本实例，读者可以：

实现在数据比较多的表格中使用行变色；

实现当选中单元格时，使单元格所在的列改变颜色。

## 7.2 对单元格的焦点进行控制

在页面中对表格进行操作时，经常要上下左右移动表格中单元格的焦点，以便对单元格进行相应的操作。下面将通过几个典型实例进行详细讲解。

### 实例196 选定表格中的单元格

本实例可以方便操作、提高工作效率

实例位置：光盘\mingrisoft\07\196

实例说明

在页面上浏览表格中的数据时，有时会忘记查看到了哪条数据，这时就应该用鼠标选中已读取到的数据并改变表格中单元格的背景颜色，这样就不会忘记读取数据的位置，以便以后继续读取，如图7.6所示。

	产品名称	进出口
2000年	葡萄	出口
2002年	山查	出口

图7.6 选定表格中的单元格

技术要点

本实例主要应用了表格的单击事件（onClick）中的event对象中的srcElement属性来获取发生事件的文档元素，并将其保存在变量e中，在用e.tagName（用于获得当前位置的标志名称）来判断发生事件的文档元素是否在表格的单元格上，并用变量e中的runtimeStyle样式中的color和backgroundColor属性来改变当前单元格的前景色

和背景色。window 对象的lastSelection属性用于获取最后一次选中的单元格焦点。

注意：在选中单元格后，一定要将上一次选中的单元格的前景色和背景色设置为空，使当前单元格为唯一选中状态。

### 实现过程

(1) 在页面中添加一个表格。代码如下。

```
<table id="table1" width="200" border="0" cellpadding="1"
cellspacing="1"
bordercolor="#000000" bgcolor="#000000">
  <tr bgcolor="#FFFFFF">
    <td>&nbsp;</td>
    <td>产品名称</td>
    <td>进出口</td>
  </tr>
  <tr bgcolor="#FFFFFF">
    <td>2000年</td>
    <td>葡萄</td>
    <td>出口</td>
  </tr>
  <tr bgcolor="#FFFFFF">
    <td>2002年</td>
    <td>山楂</td>
    <td>出口</td>
  </tr>
</table>
```

(2) 编写用于实现选中表格中单元格的JavaScript代码。

自定义函数select()用来保存发生事件的文档元素信息，其代码如下。

```
<script language="JavaScript">
var lastSelection=null;
function select(element) {
    var e, r, c;
    if (element==null) {e= event.srcElement;} //获取发生
事件的文档元素
    else { e= element; }
    if (e.tagName== "TD") {
        c= findCell(e);
        if (c != null) {
            if (lastSelection != null)
                {deselectRowOrCell(window.lastSelection);}
            selectRowOrCell(c);
            lastSelection= c;
        }
    }
    window.event.cancelBubble= true; //取消冒泡语句,用于防
止向下一个外层对象冒泡
}
```

Tdiv.onclick=select;

自定义函数findCell(), 用于判断选中的位置是否是单元格，其代码如下。

```
function findCell(e) {
    if (e.tagName == "TD") {return e;}
    else if (e.tagName == "BODY") {return null;}
}
```

```
        else{return findCell(e.parentElement);}  
    }  
}
```

自定义函数selectRowOrCell(), 用于改变单元格的前景色和背景色, 其代码如下。

```
function selectRowOrCell(r) {  
    r.runtimeStyle.backgroundColor="darkblue";  
    r.runtimeStyle.color="white";  
}
```

自定义函数deselectRowOrCell(), 使前景色和背景色恢复正常, 其代码如下。

```
function deselectRowOrCell(r) {  
    r.runtimeStyle.backgroundColor = "";  
    r.runtimeStyle.color = "";  
}
```

```
</script>
```

举一反三

根据本实例, 读者可以:

使选中的单元格的前景色不停变化;

实现当单击单元格时文字变大, 松开时, 文字恢复正常。

## [实例197 左右移动单元格的信息](#)

这是一个可以提高基础技能的实例

实例位置: 光盘\mingrisoft\07\197

实例说明

本实例是在单元格获得焦点后, 通过页面中的两个按钮来使单元格的信息左右移动, 当信息移到最左边或最右边时, 将无法移出。运

行结果如图7.7、图7.8所示。

书名	2006年销量	2007年销量
Delphi XX参考大全	40000册	50000册
Delphi XX完全手册	45000册	55000册

单元格左移      单元格右移

图7.7 单元格移动前的效果

书名	2006年销量	2007年销量
Delphi XX参考大全	50000册	40000册
Delphi XX完全手册	45000册	55000册

单元格左移      单元格右移

图7.8 单元格移动后的效果

### 技术要点

在实例196中已经对在表格中单击鼠标时获得焦点进行了详细的说明，本实例将详细介绍单元格焦点如何进行左右移动的效果。

在本实例中虽然利用 event对象的 srcElement属性获取发生事件的文档元素，并将其数据保存在变量c 中。在移动单元格信息之前，首先要利用c.tag Name 来判断当前焦点的标签名是否为 TD（单元格标记），当为 TD 时，表示表格中的单元格获得了焦点，用 c.previousSibling或 c.nextSibling来获取当前单元格同级（同一行）的前一个或下一个单元格的位置，并将其保存在 ls 变量中，再用 ls.parentElement 来返回当前位置的上一级（TR），最后用行标记中的insertBefore()方法使单元格的内容向左或向右插入。

### 实现过程

（1）在页面中添加一个表格和两个按钮，代码如下。

```
<table id="table1"width="273"border="0" cellpadding="1"
cellspacing="1"
bordercolor="#000000"bgcolor="#000000">
<trbgcolor="#FFFFFF">
```

```

    <tdwidth="120">书名</td>
    <tdwidth="72">2006年销量</td>
    <tdwidth="71">2007年销量</td>
</tr>
<trbgcolor="#FFFFFF">
    <td>DelphiXX参考大全</td>
    <td>40000册</td>
    <td>50000册</td>
</tr>
<trbgcolor="#FFFFFF">
    <td>DelphiXX完全手册</td>
    <td>45000册</td>
    <td>55000册</td>
</tr>
</table>

```

(2) 编写用于实现将单元格信息左右移动的JavaScript代码。自定义函数select(), 用来保存发生事件的文档元素信息, 其代码如下。

```

<script language="JavaScript">
var lastSelection=null;
function select(element) {
    var e, r, c;
    if (element==null) {e= event.srcElement;} //获取发生
事件的文档元素
    else { e= element; }
    if (e.tagName=="TD") {
        c= findCell(e);

```

```

    if (c != null) {
        if (lastSelection != null)
            {deselectRowOrCell(window.lastSelection);}
        selectRowOrCell(c);
        lastSelection= c;
    }
}

window.event.cancelBubble= true;           //取消冒泡语
句,用于防止向下一个外层对象冒泡
}

```

Tdiv.onclick=select;

自定义函数findCell(), 判断选中的位置是否是单元格, 其代码如下。

```

function findCell(e) {
    if (e.tagName == "TD") {return e;}
    else if (e.tagName == "BODY") {return null;}
    else{return findCell(e.parentElement);}
}

```

自定义函数selectRowOrCell(), 改变单元格的前景色和背景色, 其代码如下。

```

function selectRowOrCell(r) {
    r.runtimeStyle.backgroundColor="darkblue";
    r.runtimeStyle.color="white";
}

```

自定义函数deselectRowOrCell(), 使前景色和背景色恢复正常, 其代码如下。

```

function deselectRowOrCell(r) {

```

```
    r.runtimeStyle.backgroundColor = "";  
    r.runtimeStyle.color = "";  
}
```

自定义函数moveLeft(), 用于将当前单元格的信息向左移动, 其代码如下。

```
functionmoveLeft() {  
    var c, p, ls;  
    if (lastSelection==null) return false;    //是否对页面  
进行了单击  
    c= lastSelection;  
    if (c.tagName != "TD") { returnnull; } //是否在单元  
格上单击  
    ls= c.previousSibling;                    //使当前位置移动  
到单元格同级的前一个  
    if (ls==null) returnnull;                //判断当前单元格  
的前一个是否有单元格  
    p= ls.parentElement;                    //获取当前单元格的  
上一级  
    p.insertBefore(c, ls);                  //将信息移到当  
前单元格的前一个位置上  
    return c;  
}
```

自定义函数moveRight(), 用于将当前单元格的信息向右移动, 其代码如下。

```
functionmoveRight() {  
    var c, p, ls;  
    if (lastSelection==null) return false;
```

```

c= lastSelection;
if (c.tagName != "TD") {returnnull;}
ls= c.nextSibling;
if (ls==null) returnnull;
p= ls.parentElement;
ls= ls.nextSibling;           //使当前位置移动到
单元格同级的下一个
p.insertBefore(c, ls);
return c;
}
</script>

```

举一反三

根据本实例，读者可以：

使单元格信息上下移动；

使单元格信息移动到任意位置。

## 实例198 通过键盘使单元格焦点任意移动

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\07\198

实例说明

本实例是在单元格获得焦点后，通过键盘上的方向键，可以使单元格的焦点进行上下左右的移动，当持续按方向键时，单元格焦点将在表格中进行循环移动。运行结果如图7.9所示。

单位名称	成立年月	职工人数
天明有限公司	2005年6月	300人
四海股份有限公司	2006年1月	1500人

图7.9 通过键盘使单元格焦点任意移动

技术要点

本实例主要是在JavaScript脚本中创建了一个ObjTable对象，将这个对象来对表格中的焦点移动进行控制。具体思路如下。

(1) 通过当前表格的rows的length属性来获取表格的总行数，通过某一行的单元格个数来获取表格的总列数

(sTable.rows[0].cells.length)。

(2) 创建一个全局 ObjTable 对象，并将表格的行数和列数分别赋给 ObjTable 对象的rowCount 和colCount 属性。

(3) 通过ObjTable的单击事件 (onClick) 来获得表格中的焦点。

(4) 通过ObjTable的键盘按下事件 (onkeydown)，对键盘的操作进行监控。其中“→”键的键值为 3，“←”键的键值为 2，

“→”键的键值为 1，“↑”键的键值为 0，回车键的键值为13。通过键盘上的移动键对表格中单元格的焦点进行移动，用回车键来实现换行的操作。

(5) 通过对键盘的操作，对当前单元格坐标进行修改，并用 ObjTable 对象的 getNode() 方法重新获取单元格的焦点。

#### 实现过程

(1) 在页面中添加一个表格，并将id设置为table1，代码如下。

```
<table id="table1" width="273" border="0" cellpadding="1" cellspacing="1" bordercolor="#000000" bgcolor="#000000">
  <tr bgcolor="#FFFFFF">
    <td width="120">单位名称</td>
    <td width="72">成立年月</td>
    <td width="71">职工人数</td>
  </tr>
  <tr bgcolor="#FFFFFF">
```

```

        <td>天明有限公司</td>
        <td>2005年6月</td>
        <td>300人</td>
    </tr>
    <tr bgcolor="#FFFFFF">
        <td>四海股份有限公司</td>
        <td>2006年1月</td>
        <td>1500人</td>
    </tr>
</table>

```

(2) 编写用于实现通过键盘方向键使单元格焦点进行移动的JavaScript代码。

自定义一个ObjTable对象，用于存放页面中表格的相关信息，其代码如下。

```

<script language="JavaScript">
varObjTable=newObject();
var prevbool= true;
ObjTable.colCount=0;      //列数
ObjTable.rowCount=0;     //行数
ObjTable.Map=null;       //TableMap
ObjTable.prevRow=-1;     //前一单元格的行号，默认为-1
ObjTable.prevCol=-1;     //前一单元格的列号，默认为-1
ObjTable.curRow=-1;      //当前行号，默认为-1
ObjTable.curCol=-1;      //当前列号，默认为-1
ObjTable.Element="TD";   //待操作元素的TagName

```

在窗体载入时，将表格的相关信息保存在ObjTable对象中，其代码如下。

```

ObjTable.Load= function(a_sID, a_sTagName) { //载入table
    var iRowCount=0, iColCount=0, i, j, m, n, iIndex=0,
iCount;
    var sTable=document.getElementById(a_sID);
    if (sTable!=null) { //设置sTable属性
        var tableMap= [];
        iColCount=sTable.rows[0].cells.length;
        iRowCount= sTable.rows.length; //获取总行数
        //监听onClick事件
        sTable.getElementsByTagName('tbody')
[0].onclick=ObjTable_setFocus;
        //监听onkeydown事件
        document.getElementsByTagName('body')
[0].onkeydown=ObjTable_moveFocus;
        ObjTable.colCount= iColCount; //列数
        ObjTable.rowCount= iRowCount; //行数
        var aCols=null, iCell;
        for (i=0; i<iRowCount;++i) {
            aCols=newArray(iColCount);
            tableMap.push(aCols);
        }
        for (i=0; i<iRowCount;++i) {
            iIndex=0;
            for (j=0; j<iColCount; j+=iCell.colSpan) {
                if (tableMap[i][j]==null) {
                    iCell= sTable.rows[i].cells[iIndex++];
                    for(m=i;m<i+iCell.rowSpan;++m) {

```

```

        for (n=j;n<j+iCell.colSpan;++n)
            {tableMap[m][n]= i+' , '+j;}
    }
    tableMap[i][j]= iCell;
}
}
}
ObjTable.Map= tableMap;//tablemap结束
}
};

```

自定义函数 ObjTable\_setFocus(), 当鼠标单击单元格时, 设置单元格的前景色和背景色, 其代码如下。

```

var iCurRow,iCurCol;
functionObjTable_setFocus(event) {
    var e= event ||window.event;
    var obj= e.target || e.srcElement,
oParent=obj.parentNode;
    var iCurRow=ObjTable.curRow, iCurCol=ObjTable.curCol;
    if (prevbool==true){var iCurRow=ObjTable.curRow,
iCurCol=ObjTable.curCol;}
    else{ObjTable.prevRow= iCurRow;ObjTable.prevCol=
iCurCol;}
    RecallFocus(ObjTable);
    var
oPrevNode=ObjTable.getNode(ObjTable.prevRow,ObjTable.prevCo
l);
    var iColCount=ObjTable.colCount;

```

```

var aMaps=ObjTable.Map;
if (oParent.tagName.toUpperCase() != "TR") {return;}
iCurRow= oParent.rowIndex;
for (var i=0; i<iColCount; i++){
    if (aMaps[iCurRow][i]==obj) {
        iCurCol= i;
        break;
    }
}
//设置节点的位置
ObjTable.curRow= iCurRow;
ObjTable.curCol= iCurCol;
//记录前一个节点的位置
ObjTable.prevRow= iCurRow;
ObjTable.prevCol= iCurCol;
obj.focus();
obj.backgroundColor="#336699";
obj.style.color="#FFFFFF";
prevbool=true;
}

```

自定义函数 ObjTable\_moveFocus(), 当用键盘的方向键移动单元格焦点时, 取消上一次的单元格焦点颜色, 使当前单元格获得焦点, 其代码如下。

```

functionObjTable_moveFocus(event) {
    //获取当前节点
    var e= event ||window.event;
    varoNode= e.target || e.srcElement;
}

```

```

varoNext=null;
switch (e.keyCode) {
    case37://左键 2
        oNext=CTable_getNextNode(2);
        break;
    case38://上键 0
        oNext=CTable_getNextNode(0);
        break;
    case13://回车键
    case39://右键 3
        oNext=CTable_getNextNode(3);
        break;
    case40://下键 1
        oNext=CTable_getNextNode(1);
        break;
}
//改变背景色
if (oNext) {
    oNext.focus();
    oNext.bgColor="#336699";
    oNext.style.color="#FFFFFF";
}
}
//获取指定行、列号的单元格
ObjTable.getNode= function(a_iRow, a_iCol) {
    //如果行号或列号小于零, 则返回null
    if (a_iRow<0 || a_iCol<0) {returnnull;}
}

```

```

//初始化数据
var oNode=null;
var aMaps=ObjTable.Map, aTemps=null;
var iRow= a_iRow, iCol=a_iCol;
var sTemp="";
//获取节点
while(true) {
    oNode= aMaps[iRow][iCol];
    sTemp=typeof(oNode);
    if (sTemp.toLowerCase()=="string") {
        aTemps=oNode.split(",");
        iRow= aTemps[0];
        iCol= aTemps[1];
    }
    else{break;}
}
return oNode; //函数返回值
}

```

自定义函数CTable\_getNextNode ()，用于获取下一个单元格的节点位置，其代码如下。

```

function CTable_getNextNode(a_key) {
    var oNode=null;
    var iCol=ObjTable.curCol , iRow=ObjTable.curRow;
    var iRowCount=ObjTable.rowCount,
    iColCount=ObjTable.colCount;
    var aMaps=ObjTable.Map, aTemps=null;
    var iCurCol=iCol, iCurRow= iRow, iTemp;

```

```
var sTemp="", sStr="";
oNode= aMaps[iRow][iCol];
if (typeof(oNode)=="string") {
    aTemps=oNode.split(",");
    iCurRow= aTemps[0];
    iCurCol= aTemps[1];
}
while (true) {
    switch(a_key) {
        case0:
            iRow--;
            break;
        case1:
            iRow++;
            break;
        case2:
            iCol--;
            if (iCol<0) {
                iCol= iColCount-1;
                iRow--;
            }
            break;
        case3:
            iCol++;
            if (iCol>=iColCount) {
                iRow++;
                iCol=0;
            }
            break;
    }
}
```

```

        }
        break;
    }
    if (iRow<0) {iRow=iRowCount-1;}
    if (iRow>=iRowCount) {iRow=0;}
    if ((iCurRow== iRow)&& (iCurCol== iCol)) {continue;}
    oNext= aMaps[iRow][iCol];
    sTemp=typeof(oNext);
    if (sTemp.toLowerCase()=="string") {
        aTemps=oNext.split(",");
        if ((iCurRow!=aTemps[0]) || (iCurCol!=aTemps[1])) {
            oNext=ObjTable.getNode(iRow, iCol);
            break;
        }
    }
    continue;
}
break;
};//endCTable.getNextNode
switch(a_key) {
    case0://上
    case1://下
        if (oNext.colSpan>1) {
            ObjTable.prevCol=CTable.curCol;
            ObjTable.prevRow=ObjTable.curRow;
            ObjTable.curRow= iRow;
        }
    else {

```

```
        ObjTable.prevCol=ObjTable.curCol;
        ObjTable.prevRow=ObjTable.curRow;
        ObjTable.curCol= iCol;
        ObjTable.curRow= iRow;
    }
    break;
case2://左
case3://右
    if (oNext.rowSpan>1) {
        ObjTable.prevCol=ObjTable.curCol;
        ObjTable.curCol= iCol;
        iTemp=ObjTable.curRow;
        ObjTable.curRow= iRow;
        ObjTable.prevRow= iTemp;
    }
    else {
        ObjTable.prevCol=ObjTable.curCol;
        ObjTable.prevRow=ObjTable.curRow;
        ObjTable.curCol= iCol;
        ObjTable.curRow= iRow;
    }
    break;
}
RecallFocus(ObjTable);
prevbool=false;
iCurRow=ObjTable.curCol;
iCurCol=ObjTable.curRow;
```

```
    return oNext;
}
```

自定义函数 RecallFocus(), 当单元格失去焦点时, 使单元格的前景色和背景色恢复正常, 其代码如下。

```
function RecallFocus(c) {
    var oPrev = ObjTable.getNode(c.prevRow, c.prevCol);
    if (oPrev) {
        oPrev.bgColor="#FFFFFF";
        oPrev.style.color="#000000";
    }
}
```

在窗体载入时, 将表格的相关信息保存到自定义对象ObjTable中, 其代码如下。

```
window.onload=function() {ObjTable.Load("table1", "TD");};
</script>
```

举一反三

根据本实例, 读者可以:

任意设置单元格移动的位数;

用键盘中的上下移动键对行的焦点进行移动。

## 7.3 对表格的行、列进行修改

在制作网页时，有些数据库的数据是经常变动的，为了能够更方便地对表格进行修改，就需要以指定的行数和列数动态生成表格，下面将通过几个典型实例进行详细讲解。

### 实例199 动态制作表格

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\07\199

实例说明

本实例是在页面中的行数和列数文本框中输入指定的行数和列数，并在表值文本框中输入单元格的内容，每个单元格中的内容是用“#”号进行分割的，单击“生成表格”按钮，将在页面生成设计好的表格。当预计行数大于实际行数时，将用空单元格补位，反之，将删除多余的单元格。运行结果如图7.10所示。

行数：  列数：

表值：

```
COL1#COL2#COL3#COL4#ROW1#A1#A2#a3#ROW2#B1#B2#B3#ROW3#C1#C2#C3
```

COL1	COL2	COL3	COL4
ROW1	A1	A2	a3
ROW2	B1	B2	B3
ROW3	C1	C2	C3

图7.10 动态制作表格

## 技术要点

本实例主要是将<table>、<tr>、<th>和<td>标记所生成表格的代码以字符串的形式保存起来，再用层的innerHTML属性将表动态生成。首先利用自定义函数Table1()将动态生成表格的首行设计为表头，然后利用自定义函数tablevalue()生成其他的行。

parseInt()方法是将字符型转换成数值型。

## 实现过程

(1) 在页面中添加3个文本框，并设置相应的名称，代码如下。

```
<form name="form1" method="post" action="">
  <p><b>行数: </b>
    <input name="name1" type="text"
    style="width:40px" value="4">
    <b>列数: </b>
    <input name="name2" type="text"
    style="width:40px" value="4">
    <input type="button" name="Submit3" value="生成表
    格" onClick="tableclick
    (document.form1.name1,document.form1.name2,document.for
    m1.name3)">
  </p>
  <p><b align="top">表值: </b></p>
  <p>
    <textareaname="name3"wrap="VIRTUAL"
    style="width:300px ">COL1#COL2#COL3#
    COL4#ROW1#A1#A2#a3#ROW2#B1#B2#B3#ROW3#C1#C2#C3</textare
    a>
  </p>
```

```
</form>
```

```
<div id="aa"></div>
```

(2) 编写用于实现动态制作表格的JavaScript代码。

自定义函数tableclick(), 用于判断生成表格的条件是否完整, 其代码如下。

```
<script language="JavaScript">  
function tableclick(name1, name2, name3) {           //判断生成表  
格的条件是否完整  
    Trow=name1.value;  
    Tcol=name2.value;  
    Tv=name3.value;  
    if ((Trow=="") || (Tcol=="") || (Tv=="")) {  
        alert("请将制作表格的条件填写完整");  
    }  
    else {  
        r=parseInt(Trow); //将文本中的行数和列数转换成数字  
        c=parseInt(Tcol);  
        Table1(r, c, Tv);  
    }  
}
```

自定义函数tablevalue(), 用于生成首行以下的所有行, 其代码如下。

```
function tablevalue(a, ai, rows, col, str) {           //生成首行以  
下的所有行  
    int1=a.length;  
    for (i=0;i<rows;++i) {  
        for (j=0;j<col;++j) {
```

```

    if ((j==0)&&(ai>=int1)) {break;}
    if (ai>=int1) {
        str=str+"<td scope=' col' >&nbsp;</td>";
    }
    else{
        if (j==0) {
            str=str+"<tr><th scope=' col' >&nbsp;<"+(a[ai++])+"
</th>";
        }
        else{
            if (j==col-1) {
                str=str+"<td scope=' col' >&nbsp;<"+(a[ai++])+"
</td>";
            }
            else{
                str=str+"<td scope=' col' >&nbsp;<"+(a[ai++])+"
</td>";
            }
        }
    }
}
str=str+"</tr>";
}
return str;
}

```

自定义函数Table1(), 根据行数和列数, 生成表的首行, 使其成为表头, 其代码如下。

```

functionTable1(row, col, Str1) {           //根据行数和列数生成表
的首行，也就是表头
var str="";
a=newArray();
s=newString(Str1);
a=s.split("#");
int1=a.length;
ai=0;
if (col<=int1) {
str=str+"<tablewidth=' 300' border=' 4' >";
for (i=0;i<col;++i) {
if (i==0) {
str=str+"<tr><th scope=' col' >&nbsp;"+(a[ai++])+"
</th>";
}
else {
if (i==(col-1)) {
str=str+"<th scope=' col' >&nbsp;"+(a[ai++])+"
</th></tr>";
}
else {
str=str+"<th scope=' col' >&nbsp;"+(a[ai++])+"
</th>";
}
}
}
}
if (int1>col) {

```

```

        if (row>1) {
            str=tablevalue(a, ai, row-1, col, str);
        }
    }
    str=str+"</table>";
    aa.innerHTML=str;
}
</script>

```

举一反三

根据本实例，读者可以：

在页面中动态生成一个登录界面；

动态替换表格的内容。

## 实例200 动态生成行或列

本实例可提高工作效率

实例位置：光盘\mingrisoft\07\200

实例说明

本实例是将一个一行一列的表格动态生成一个多行多列的表格，当插入多行后，在插入列时，每列中的行数将与插入后的行数相同，反之，也一样。运行结果如图7.11所示。

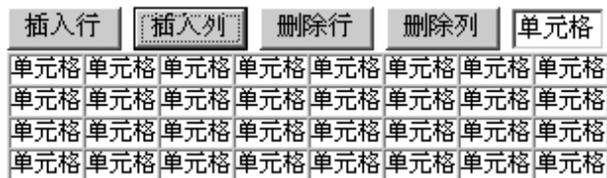


图7.11 动态生成行或列

技术要点

本实例主要应用了表格的insertRow(n)方法在表格的尾部添加行，n表示当前表格的行数，为表格添加列时，实际上是以表格的行数为最大值，用表格的rows[i]的insertCell(n)方法向每行的尾部添加单元格，i表示行数，n表示当前表格的最大列数。用表格的rows[i]的cells[n]的innerText属性来动态向单元格中添加指定的文本，i表示指定的行，n表示指定行中的第几个单元格。

### 实现过程

(1) 在页面中添加3个文本框，并设置相应的名称，代码如下。

```
<input type="button" value="插入行"
onclick="insertrow()">
<input type="button" value="插入列"
onclick="insertcell()">
<input type="button" value="删除行"
onclick="deleterow()">
<input type="button" value="删除列"
onclick="deletecell()">
<input name="text1" size="5" value="单元格" >
<table id="tt" cellpadding=0 cellspacing=0 border="1">
<tr><td>单元格</td></tr>
</table>
```

(2) 编写用于实现动态插入或删除表格行或列的JavaScript代码。

自定义函数insertrow()，用于在已有单元格的后面插入行，其代码如下。

```
<script language="JavaScript">
function insertrow() { //插入行
    text1.value+=""?str=" ":str=text1.value;
```

```

varn=tt.rows.length;//行长度
tt.insertRow(n);//插入一行
for (i=0;i<=tt.rows(0).cells.length-1;i++){
    tt.rows[n].insertCell(i);//在插入的行中插入单元格
    tt.rows[n].cells[i].innerText=str;//放入TEXT
}
}

```

自定义函数insertcell(), 用于在已有单元格的后面插入列, 其代码如下。

```

function insertcell() { //插入列
    text1.value==" "?str=" ":str=text1.value;
    varn=tt.rows[0].cells.length;//0行的单元格长度
    for (i=0;i<=tt.rows.length-1;i++){
        tt.rows[i].insertCell(n);//在某行中插入单元格
        tt.rows[i].cells[n].innerText=str;//放入TEXT
    }
}

```

自定义函数deleterow(), 从表的下边开始向上删除行, 其代码如下。

```

functiondeleterow() { //删除行
    varn=tt.rows.length;//行长度
    if(n==1) return;
    tt.deleteRow(n-1);//删除某一行
}

```

自定义函数deletecell(), 从表的右边开始向左边删除列, 其代码如下。

```

functiondeletecell() { //删除列

```

```
var n=tt.rows[0].cells.length//0行的单元格长度
if(n==1) return;
for (i=0;i<=tt.rows.length-1;i++){
    tt.rows[i].deleteCell(n-1);//删除某行的某一单元格
}
}
```

</script>

举一反三

根据本实例，读者可以：

动态添加单元格；

动态修改单元格中的文本。

## 实例201 删除表中的行

本实例可提高工作效率

实例位置：光盘\mingrisoft\07\201

实例说明

本实例主要实现了在表格中单击一个单元格，当单击“删除行”按钮时，将删除单元格所在的行，该按钮可以将表格中的行全部删除。运行结果如图7.12所示。

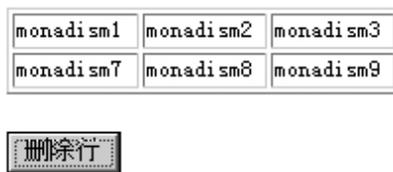


图7.12 删除表中的行

技术要点

在表格中单击鼠标，使单元格获得焦点已在实例196中进行了详细的说明。

本实例主要用单元格的parentElement属性（返回当前标志的父级）返回到表主体，并用表主体的removeChild(r)方法删除单元格所在行，r表示当前单元格的父级（TR）。

### 实现过程

(1) 在页面中添加一个表格及一个按钮，并设置表格的id名称，代码如下。

```
<table id="table1"width="200"border="1">
  <tr>
    <td id="cell-1">monadism1</td>
    <td id="cell-2">monadism2</td>
    <td id="cell-3">monadism3</td>
  </tr>
  <tr>
    <td id="cell-4">monadism4</td>
    <td id="cell-5">monadism5</td>
    <td id="cell-6">monadism6</td>
  </tr>
  <tr>
    <td id="cell-7">monadism7</td>
    <td id="cell-8">monadism8</td>
    <td id="cell-9">monadism9</td>
  </tr>
</table>
<form name="form1"method="post"action="">
  <inputname="Button1" type="button"
id="Button1"value="删除行"
onclick="removeRow()">
```

```
</form>
```

(2) 编写用于实现删除表中行的JavaScript代码。

自定义函数select(), 用来保存发生事件的文档元素信息, 其代码如下。

```
<script language="JavaScript">
var lastSelection=null;
//获取选择行或单元格的参数值
function select(element) {
    var e, r, c;
    if (element==null) {e= event.srcElement;} //用于获取发
生事件的文档元素信息
    else { e= element; }
    if (e.tagName=="TD") {
        c= findCell(e);
        if (c != null) {
            if (lastSelection != null)
            {deselectRowOrCell(lastSelection);}
            selectRowOrCell(c);
            lastSelection= c;
        }
    }
    window.event.cancelBubble= true;
}
table1.onclick=select;
```

自定义函数findCell(), 查找是否为单元格, 其代码如下。

```
function findCell(e) {
    if (e.tagName == "TD") {return e;}
```

```
else if (e.tagName == "BODY") {return null;}
    else{return findCell(e.parentElement);}
}
```

自定义函数selectRowOrCell(), 用于改变选定行或单元格的前景色和背景色, 其代码如下。

```
function selectRowOrCell(r) {
    r.runtimeStyle.backgroundColor="darkblue";
    r.runtimeStyle.color="white";
}
```

自定义函数deselectRowOrCell(), 用于取消对行或单元格的选定, 其代码如下。

```
function deselectRowOrCell(r) {
    r.runtimeStyle.backgroundColor = "";
    r.runtimeStyle.color = "";
}
```

自定义函数findCell(), 判断选中的位置是否是单元格, 其代码如下。

```
function findCell(e) {
    if (e.tagName == "TD") {return e;}
    else if (e.tagName == "BODY") {return null;}
    else{return findCell(e.parentElement);}
}
```

自定义函数removeRow(), 用于删除单元格所在的行, 其代码如下。

```
function removeRow() {
    var r, p, nr;
    if (lastSelection==null) return false;
```

```
r= lastSelection;
if (r.tagName=="TD") {r= r.parentElement;}
p= r.parentElement;
p.removeChild(r);
lastSelection=null;
return r;
}
</script>
```

举一反三

根据本实例，读者可以：

实现删除表中的列；

实现对表中的单元格进行单个删除。

## 7.4 单元格的相关操作

在页面中对表格进行设计时，最多的应该是对单元格的应用，例如，可以对相同的单元格进行合并，使表格更加简捷、易懂，也可以通过单元格将一些不重要的说明性文字进行隐藏。本节将通过几个典型实例来进行详细说明。

### 实例202 隐藏及显示单元格

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\07\202

实例说明

本实例是通过指定单元格的隐藏，可以对表格的说明性文字实现一种动态的显示效果。当鼠标移动到带有三角形的图片时，鼠标会变成手状，单击图片，左边的说明性文字将进行隐藏和显示的切换效果。运行结果如图7.13、图7.14所示。



	姓名	性别	出生年月
一班	张同学	男	1990-2-12
一班	孙同学	女	1991-6-14
一班	王同学	男	1990-11-20

图7.13 隐藏单元格

表格说明: 该表格是对人员信息的相关介绍.			
	姓名	性别	出生年月
一班	张同学	男	1990-2-12
一班	孙同学	女	1991-6-14
一班	王同学	男	1990-11-20

图7.14 显示单元格

### 技术要点

本实例主要是通过单元格style样式中的display属性值none和block来对当前单元格进行隐藏和显示的效果。

当鼠标移入和移出带有图片的单元格时，将通过 document 对象的 body 的 style 的 cursor属性值来改变鼠标的形状。

document对象的body的style的cursor属性值如表7.1所示。

表7.1 document对象的body的style的cursor属性值

可 选 值	说 明	可 选 值	说 明
auto	标准光标	text	I形光标
default	标准箭头	vertical-text	水平I形光标
hand	手形光标	no-drop	不可拖动光标
help	帮助光标	all-ctoll	三角方向光标
move	移动光标	crosshair	十字光标
wait	等待光标	not-allowed	无效光标

### 实现过程

(1) 在页面中添加一个表格，并在表格的第二个单元格添加一个图片，代码如下。

```
<table id="table1"width="408"height="167"border="1"
cellpadding="0"cellspacing="0">
  <tr>
    <tdwidth="119"valign="top" id="tdhide"><p>表格说明:
  </p>
```

```

    <p>该表格是对人员信息的相关介绍.</p></td>
    <tdwidth="1"bordercolor="#FFFFFF"background="1.jpg"bg
color="#FFFFFF"
    id="tdsize"onClick="hidecell()"onMouseMove="move()"on
MouseOut="out()">&nbsp;</td>
    <tdwidth="274" id="tdcen"align="center">
    <tablewidth="257"height="108"border="0"bordercolor=
"#000000"
    bgcolor="#000000">
    <trbgcolor="#FFFFFF">
    <tdwidth="36">&nbsp;</td>
    <tdwidth="51">姓名</td>
    <tdwidth="38">性别</td>
    <tdwidth="104">出生年月</td>
    </tr>
    <trbgcolor="#FFFFFF">
    <td>一班</td>
    <td>张同学</td>
    <td>男</td>
    <td>1990-2-12</td>
    </tr>
    <trbgcolor="#FFFFFF">
    <td>一班</td>
    <td>孙同学</td>
    <td>女</td>
    <td>1991-6-14</td>
    </tr>

```

```

        <tr bgcolor="#FFFFFF">
        <td>一班</td>
        <td>王同学</td>
        <td>男</td>
        <td>1990-11-20</td>
        </tr>
    </table>
</tr>
</table>

```

(2) 编写用于实现单元格隐藏及显示的JavaScript代码。

自定义函数move(), 当鼠标移动到带图片的单元格时, 使鼠标变为手形, 其代码如下。

```

<script language="JavaScript">
function move() { //在移动时修改鼠标的形状
    document.body.style.cursor="hand"; //将鼠标设置为手形
}

```

自定义函数out(), 当鼠标移出带有图片的单元格时, 使鼠标恢复为标准形状, 其代码如下。

```

function out() {
    document.body.style.cursor="auto"; //将鼠标设置为标准
光标
}

```

自定义函数hidecell(), 用于控制指定单元格的隐藏及显示, 同时改变表格的大小, 其代码如下。

```

var bool=true;
var tdl=0;
functionhidecell() {

```

```
if (bool==true){
    tdhide.style.display="none"; //对当前单元格进行显示
    td1=parseInt(tdhide.width);
    table1.width=table1.width-tdhide.width;
    bool=false;
}
else{
    tdhide.style.display="block"; //对当前单元格进行隐藏
    table1.width=parseInt(table1.width)+td1;
    bool=true;
}
}
</script>
```

举一反三

根据本实例，读者可以：

在页面中制作一个隐藏的菜单；

在页面中制作一个下拉式的菜单。

## 实例203 编辑单元格中的文本信息

本实例可提高工作效率

实例位置：光盘\mingrisoft\07\203

实例说明

本实例是在表格中选择指定的单元格，单击“修改单元格信息”按钮，将在按钮的下面显示一个文本框，当对文本框中的信息进行修改时，所选单元格中的文本信息也随之改变。运行结果如图7.15所示。

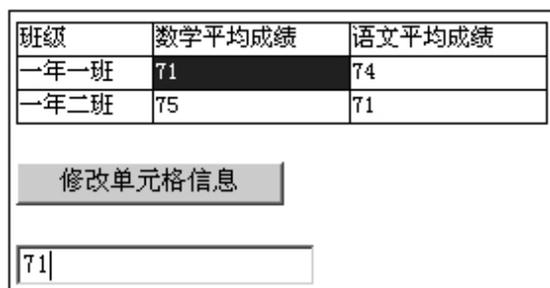


图7.15 显示随机图像

### 技术要点

选择表格中的单元格已在实例196中进行了详细的讲解。

本实例主要是用`setExpression()`方法将文本框中的内容动态传递给已选中的单元格，也就是修改单元格中的`innerHTML`属性，并在修改其他单元格之前用`removeExpression()`方法动态删除`innerHTML`属性。下面对`setExpression()`和`removeExpression()`方法进行详细说明。

`setExpression()`方法的语法格式如下。

```
setExpression(attribute, value, language)
```

参数说明如下。

- `attribute`: 动态属性。
- `value`: 属性值。
- `language`: 可选值，语言，如JavaScript。

功能：用于设置动态属性。

`removeExpression()`方法的语法格式如下。

```
removeExpression(attribute)
```

参数说明如下。

- `attribute`: 动态属性。

功能：删除动态属性。

实现过程

(1) 在页面中添加一个表格、文本框及按钮，并设置表格的id名称，代码如下。

```
<table width="270" cellspacing="1" bordercolor="#000000"
bgcolor="#000000"
id="table1">
  <tr bgcolor="#FFFFFF">
    <td id="cell-1">班级</td>
    <td id="cell-2">数学平均成绩</td>
    <td id="cell-3">语文平均成绩</td>
  </tr>
  <tr bgcolor="#FFFFFF">
    <td id="cell-4">一年一班</td>
    <td id="cell-5">70</td>
    <td id="cell-6">74</td>
  </tr>
  <tr bgcolor="#FFFFFF">
    <td id="cell-7">一年二班</td>
    <td id="cell-8">75</td>
    <td id="cell-9">71</td>
  </tr>
</table>
<form name="form1" method="post" action="">
  <p><input name="Button1" type="button" id="Button1"
value="修改单元格信息" onclick
="editContents()"></p>
  <p><input id="celltext" name="celltext" type="text"
style="display: none "></p>
```

</form>

(2) 编写用于实现修改单元格内容的JavaScript代码。

自定义函数select(), 用来保存发生事件的文档元素信息, 其代码如下。

```
<script language="JavaScript">
```

```
var lastSelection=null;
```

```
//获取选择行或单元格的参数值
```

```
function select(element) {
```

```
    var e, r, c;
```

```
    if (element==null) {e= event.srcElement;} //用于获取发  
生事件的文档元素信息
```

```
    else { e= element; }
```

```
    if (e.tagName=="TD") {
```

```
        c= findCell(e);
```

```
        if (c != null) {
```

```
            if (lastSelection != null)
```

```
{deselectRowOrCell(lastSelection);}
```

```
            selectRowOrCell(c);
```

```
            lastSelection= c;
```

```
        }
```

```
    }
```

```
    window.event.cancelBubble= true;
```

```
}
```

```
table1.onclick=select;
```

自定义函数 findCell(), 查找是否为单元格, 其代码如下。

```
function findCell(e) {
```

```
    if (e.tagName=="TD") {return e;}
```

```
else if (e.tagName=="BODY") {returnnull;}
    else{return findCell(e.parentElement);}
}
```

自定义函数 `selectRowOrCell()`，用于改变选定行或单元格的前景色和背景色，其代码如下。

```
function selectRowOrCell(r) {
    r.runtimeStyle.backgroundColor="darkblue";
    r.runtimeStyle.color="white";
}
```

自定义函数 `deselectRowOrCell()`，用于取消对行或单元格的选定，其代码如下。

```
functiondeselectRowOrCell(r) {
    r.runtimeStyle.backgroundColor= "";
    r.runtimeStyle.color= "";
}
```

自定义函数 `findCell()`，判断选中的位置是否是单元格，其代码如下。

```
function findCell(e) {
    if (e.tagName=="TD") {return e;}
    else if (e.tagName=="BODY") {returnnull;}
    else{return findCell(e.parentElement);}
}
```

自定义函数 `editContents()`，用于编写单元格中的文本信息，其代码如下。

```
function editContents() {
    var c, p, nr;
    if (lastSelection==null) return false;
```

```

c= lastSelection;
if (c.tagName != "TD") { returnnull; }
document.form1.celltext.style.display= "";
document.form1.celltext.value= c.innerHTML;
c.setExpression("innerHTML",
"document.form1.celltext.value");
document.form1.celltext.focus();
document.form1.celltext.onblur=unhookContentsExpression
;
}

```

自定义函数 unhookContentsExpression(), 用于清空文本框中的内容, 其代码如下。

```

functionunhookContentsExpression() {
lastSelection.removeExpression("innerHTML");
document.form1.celltext.value= '';
document.form1.celltext.style.display= "none";
}
</script>

```

举一反三

根据本实例, 读者可以:

制作一个动态修改的关联表。

## 实例204 合并单元格

本实例可以美化界面

实例位置: 光盘\mingrisoft\07\204

实例说明

本实例是将指定列中连续相同的单元格进行合并，这样可以使表格更加简捷、明了，便于阅读。运行结果如图7.16所示。

A	1
	2
B	1
	2
	3
A	1

合并单元格

图7.16 合并单元格

### 技术要点

本实例主要是用表格中的rows的length属性来获得表格的行数，再用表格中的rows[nrow]-cells[ncol]-innerText 属性来获取表格中行的指定单元格的信息，并判断纵向单元格是否相同。如果相同，用表格的rows[i]的deleteCell(ncol)方法，将连续相同的列中的单元格删除。

### 实现过程

(1) 在页面中添加一个表格及一个按钮，并设置表格的id名称，代码如下。

```
<table id="table1"width="200"cellpadding="1"
bordercolor="#000000"
bgcolor="#000000">
  <trbgcolor="#FFFFFF">
    <td width="40%">A</td>
    <td width="60%">1</td>
  </tr>
  <trbgcolor="#FFFFFF">
    <td width="40%">A</td>
    <td width="60%">2</td>
```

```

</tr>
<tr bgcolor="#FFFFFF">
  <td width="40%">B</td>
  <td width="60%">1</td>
</tr>
<tr bgcolor="#FFFFFF">
  <td width="40%">B</td>
  <td width="60%">2</td>
</tr>
<tr bgcolor="#FFFFFF">
  <td width="40%">B</td>
  <td width="60%">3</td>
</tr>
<tr bgcolor="#FFFFFF">
  <td width="40%">A</td>
  <td width="60%">1</td>
</tr>
</table>
<p></p>
<input type="button" value="合并单元格" onclick="merge()">
(2) 编写用于实现合并单元格的JavaScript代码。
<script language="JavaScript">
functionmerge() {
  var rows=table1.rows;
  varnrow=0, nlastrow;
  varncol=0;
  while (nrow<rows.length) {

```

```

        nlastrow=nrow++;
        while(nrow<rows.length&&
rows[nlastrow].cells[ncol].innerText
        ==rows[nrow].cells[ncol].innerText)
            nrow++;
        if(nrow-nlastrow>1) {
            for(var i=nlastrow+1;i<nrow;i++)
{rows[i].deleteCell(ncol);}
            rows[nlastrow].cells[ncol].rowSpan=nrow-nlastrow;
        }
    }
}
</script>

```

举一反三

根据本实例，读者可以：

将表格行中的相同单元格合并；

对表格中的多个列进行合并单元格。

## [实例205 在表格中添加行及单元格](#)

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\07\205

实例说明

在表格中添加行，实际就是在表格的下面动态添加单元格，并在单元格中添加信息。运行结果如图7.17所示。

产品	个数	单价
显示器	20	1950
CPU	30	1430
单元格	单元格	



图7.17 在表格中添加行及单元格

### 技术要点

在表格中选中单元格已在实例196中进行了详细的讲解。

本实例是用 document 对象的 createElement("TR")方法创建表格的行的节点，用表主体 (TBODY) 的 insertBefore(nr, r) 方法在选中单元格的下方添加一个行，nr 为新创建行的节点，r 为选中单元格的相关信息，再用 document 对象的 createElement("TD") 方法在创建的行中创建一个单元格节点，并用 <TD> 标记的 insertBefore(text, null) 方法向创建的单元格中添加信息，用 <TR> 标记的 insertBefore(nc, c) 方法添加单元格。

### 实现过程

(1) 在页面中添加一个表格及两个按钮，并设置表格的 id 名称，代码如下。

```
<tablewidth="200"border="1"cellpadding="0"
cellspacing="0"bordercolor="#000000"
bordercolordark="#FFFFFF"bordercolorlight="#666666"
id="table1">
  <tr>
    <td id="cell-1">产品</td>
    <td id="cell-2">个数</td>
    <td id="cell-3">单价</td>
  </tr>
  <tr>
```

```

        <td id="cell-4">显示器</td>
        <td id="cell-5">20</td>
        <td id="cell-6">1950</td>
    </tr>
    <tr>
        <td id="cell-7">CPU</td>
        <td id="cell-8">30</td>
        <td id="cell-9">1430</td>
    </tr>
</table>
<form name="form1" method="post" action="">
    <input name="Button1" type="button" id="Button1"
value="添加行" onclick="addRow()">
    <input name="Button2" type="button" id="Button2"
value="添加单元格" onclick=
"addCell()">
</form>

```

(2) 编写用于实现在表格中添加行及单元格的JavaScript代码。自定义函数select(), 用于获取选中单元格的原始记录, 其代码如下。

```

<script language="JavaScript">
var lastSelection=null;
function select(element) {
    var e, r, c;
    if (element==null) {e= event.srcElement;}
    else { e= element; }
    if (e.tagName== "TD") {

```

```

    c= findCell(e);
    if (c != null) {
        if (lastSelection != null)
        {deselectRowOrCell(lastSelection);}
        selectRowOrCell(c);
        lastSelection= c;
    }
}
window.event.cancelBubble= true;
}

```

table1.onclick=select;

查找是否有单元格，其代码如下。

```

function findCell(e) {
    if (e.tagName == "TD") {return e;}
    else if (e.tagName == "BODY") {return null;}
    else{return findCell(e.parentElement);}
}

```

自定义函数selectRowOrCell()，用于修改获得焦点的前景色和背景色，其代码如下。

```

function selectRowOrCell(r) {
    r.runtimeStyle.backgroundColor="darkblue";
    r.runtimeStyle.color="white";
}
functiondeselectRowOrCell(r) {
    r.runtimeStyle.backgroundColor= "";
    r.runtimeStyle.color= "";
}

```

自定义函数addRow(), 为表格添加行, 其代码如下。

```
function addRow() {
    var r, p, nr;
    if (lastSelection==null) {r=null;p=table1.children[0];}
    else {
        r= lastSelection;
        if (r.tagName=="TD") {r=r.parentElement;}
        p= r.parentElement;
    }
    nr=document.createElement("TR");
    p.insertBefore(nr, r);
    select(nr);
    addCell();
    returnnr;
}
```

自定义函数addCell(), 用于在添加的行中插入一个单元格, 其代码如下。

```
function addCell() {
    var r, p, c, nc, text;
    if (lastSelection == null) return false;
    r = lastSelection;
    if (r.tagName == "TD") {r = r.parentElement;c =
lastSelection;}
    else {c = null;}
    nc = document.createElement("TD");
    text = document.createTextNode("单元格");
    nc.insertBefore(text, null);
```

```
r.insertBefore(nc, c);
select(nc);
return nc;
}
</script>
```

举一反三

根据本实例，读者可以：

在表格中添加行；

在表格中添加列。

## 实例206 删除表中的单元格

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\07\206

实例说明

本实例是用鼠标选中表中的单元格，然后单击“删除单元格”按钮，将删除表中的单元格，当删除的是前面或中间的单元格时，后面的单元格将向前移位。运行结果如图7.18所示。



图7.18 删除表中的单元格

技术要点

用鼠标选中表中的单元格已在实例196中进行了详细讲解。

本实例主要用<TD>标记的 `parentElement` 属性来获得父级标记<TR>，再用<TR>标记的`removeChild(c)`方法将已选中的单元格删除。

## 实现过程

(1) 在页面中添加一个表格及一个按钮，并设置表格的id名称，代码如下。

```
<table id="table1"width="200"border="1">
  <tr>
    <td id="cell-1">monadism1</td>
    <td id="cell-2">monadism2</td>
    <td id="cell-3">monadism3</td>
  </tr>
  <tr>
    <td id="cell-4">monadism4</td>
    <td id="cell-5">monadism5</td>
    <td id="cell-6">monadism6</td>
  </tr>
  <tr>
    <td id="cell-7">monadism7</td>
    <td id="cell-8">monadism8</td>
    <td id="cell-9">monadism9</td>
  </tr>
</table>
<form name="form1"method="post"action="">
  <inputname="Button1" type="button"
id="Button1"value="删除单元格"onclick=
"removeCell()">
</form>
```

(2) 编写用于删除表中的单元格的 JavaScript 代码。获取选择行或单元格的参数值，其代码如下。

```

<script language="javascript">
var lastSelection=null;
function select(element) {
    var e, r, c;
    if (element==null) {e= event.srcElement;}
    else { e= element; }
    if (e.tagName== "TD") {
        c= findCell(e);
        if (c != null) {
            if (lastSelection != null)
            {deselectRowOrCell(lastSelection);}
            selectRowOrCell(c);
            lastSelection= c;
        }
    }
    window.event.cancelBubble= true;
}
table1.onclick=select;

```

查找是否有单元格，其代码如下。

```

function findCell(e) {
    if (e.tagName== "TD") {return e;}
    else if (e.tagName== "BODY") {returnnull;}
    else{return findCell(e.parentElement);}
}

```

将选择的行或单元格的前景色和背景色进行变色，其代码如下。

```

function selectRowOrCell(r) {
    r.runtimeStyle.backgroundColor="darkblue";
}

```

```
    r.runtimeStyle.color="white";  
}
```

取消对行或单元格的选定，其代码如下。

```
functiondeselectRowOrCell(r) {  
    r.runtimeStyle.backgroundColor= "";  
    r.runtimeStyle.color= "";  
}
```

删除表中的单元格，其代码如下。

```
function removeCell() {  
    var c, p, nr;  
    if (lastSelection==null) return false;  
    c= lastSelection;  
    if (c.tagName != "TD") { returnnull; }  
    p= c.parentElement;  
    p.removeChild(c);  
    lastSelection=null;  
    return c;  
}  
</script>
```

举一反三

根据本实例，读者可以：

删除指定的行；

删除指定的列。

## 7.5 表格的特殊效果

在制作网页时，最常用的就是表格，可以通过表格来设计页面的布局，也可以将表格设计成不同的风格，使页面更加美观。本节将通过几个典型实例来进行详细说明。

### 实例207 透明表格

本实例可以美化界面

实例位置：光盘\mingrisoft\07\207

实例说明

在制作网页时，有时需要将表格进行透明，本实例是通过设置表格滤镜中的透明度，来实现该效果，运行结果如图7.19所示。



厂长	姓名	性别	籍贯
	李先生	男	长春
党部	张先生	男	天津
工人	刘先生	男	四川

图7.19 透明表格

技术要点

本实例主要是设置表格的style的filter的alpha的opacity的值（opacity的取值范围为0~100），使表格具有透明的效果。

实现过程

- (1) 将图片设置为当前页的背景。
- (2) 在页面中添加一个表格，并设置表格的id名称，代码如下。

```

<tablewidth="400"border=5 align=center
cellpadding="4"cellspacing="4"bgcolor
="#FFFFFF" id="table1" style="border-color:#FFCCFF; font-
size:14px">
  <tr>
    <td id="btb"align=center>&nbsp;</th>
    <td id="btb"align=center>姓名</th>
    <td id="btb"align=center>性别</th>
    <td id="btb"align=center>籍贯</th>
  </tr>
  <tr>
    <td id="btb"align=center>厂长</th>
    <td id="btb"align=center>李先生</td>
    <td id="btb"align=center>男</td>
    <td id="btb"align=center>长春</td>
  </tr>
  <tr>
    <td id="btb"align=center>党部</th>
    <td id="btb"align=center>张先生</td>
    <td id="btb"align=center>男</td>
    <td id="btb"align=center>天津</td>
  </tr>
  <tr>
    <td id="btb"align=center>工人</th>
    <td id="btb"align=center>刘先生</td>
    <td id="btb"align=center>男</td>
    <td id="btb"align=center>四川</td>
  </tr>

```

```
</tr>  
</table>
```

(3) 编写用于实现表格透明的JavaScript代码。

```
<script language="javascript">  
table1.style.filter="alpha(opacity=50)";  
</script>
```

举一反三

根据本实例，读者可以：

制作一个渐隐渐现的表格；

实现当鼠标指向表格时，使透明表格清晰化。

## 实例208 限制表格的宽度

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\07\208

实例说明

在向表格输入信息时，当信息大于单元格的宽度时，单元格的宽度将自动向右扩展，这样使表格看上去极不美观，本实例将限制单元格的宽度，当信息超出单元格的宽度时，单元格的高度将增大。运行结果如图7.20所示。

本实例实现了固定表格的大小，在向表格的单元格中添加信息时，单元格只会向下移动，而不会向右扩展。

图7.20 限制表格的宽度

技术要点

本实例主要是应用表格的style样式的table-layout属性的fixed值来固定表格的大小，在用table-layout属性之前，要用表格的width

和height属性来设置表格的宽度和高度。下面对style样式的table-layout属性进行详细说明。

table-layout属性的语法格式如下。

```
table-layout : auto | fixed
```

参数说明如下。

- auto: 默认的自动算法（计算速度慢）。布局将基于各单元格的内容。表格在每一单元格读取计算之后才会显示出来。

- fixed: 固定布局的算法。在这种算法中，水平布局是仅仅基于表格的宽度、表格边框的宽度、单元格间距、列的宽度，而和表格内容无关。也就是说，内容可能被裁切。

功能：设置或检索表格的布局算法。

实现过程

(1) 在页面中添加一个表格，并设置表格的id名称，代码如下。

```
<table
id="table1"border="1px"cellpadding="0px"cellspacing="0px"bordercolor
="#000000"bordercolordark="#FFFFFF"bordercolorlight="#666
666" style="font-size:
12px">
  <tr>
    <td>本实例实现了固定表格的宽度,在向表格的单元格中添加信
息时,单元格只会向下移动,而不会向右扩展.</td>
  </tr>
</table>
```

(2) 编写用于固定单元格宽度的 JavaScript代码。

```
<script language="JavaScript">
varw=180;
```

```
var h=50;
table1.align="Center";
table1.width=w;
table1.height=h;
table1.style.tableLayout="fixed";
</script>
```

举一反三

根据本实例，读者可以：

实现对指定单元格的大小进行限制。

## 实例209 表格的标题

本实例可以美化界面

实例位置：光盘\mingrisoft\07\209

实例说明

在某些网站中，当对一些图片或数据进行操作时，由于页面太过复杂，很难一下子找到要操作的对象，这样，就需要一个标题将其括起来，并进行文字性说明，本实例将用表格来实现这一效果。运行结果如图7.21所示。



图7.21 表格的标题

技术要点

本实例主要是用<fieldset>和<legend>标记将表单元素组合在一起，形成一个表格标题。并用层的 innerHTML 属性将设置好的表的标题以动态形式添加到层中。下面对<fieldset>和<legend>标记进行说明。

- <fieldset>标记：对表单中的元素进行分组。
- <legend>标记：用来标记子表单。

### 实现过程

(1) 在页面中添加一个层，并设置层的id名称，代码如下。

```
<div id="Tdiv" align="center" width="200px"></div>
```

(2) 编写用于显示表格标题的JavaScript代码如下。

```
<script language="JavaScript">  
function caption(name) {  
    var s="<tablewidth=' 240px' height=' 160' align=' center' >";  
    s=s+"<tr><td>";  
    s=s+"<fieldset><legend><font style=' font-  
size:12px' >"+name+"</font></legend>";  
    s=s+"<img src=' 7. jpg' width=' 200' height=' 140' >  
</fieldset>";  
    s=s+"</td></tr></table>";  
    Tdiv.innerHTML=s;  
}  
</script>
```

(3) 在<body>标记中的窗体载入事件 (onLoad) 中调用自定义函数caption(), 其代码如下。

```
<body onLoad="caption(' 要进行锐化的图片')">
```

### 举一反三

根据本实例，读者可以：

随机显示标题文字的颜色；  
用表标题设计一个登录窗口。

## 实例210 表格的外阴影

本实例可以美化界面

实例位置：光盘\mingrisoft\07\210

实例说明

本实例为了能够让表格具有立体效果，在表格的外边框、单元格及文本的后面都加了阴影效果。运行结果如图7.22所示。



图7.22 表格的外阴影

技术要点

本实例主要是应用表格中style的filter的Shadow属性来对表格进行阴影设置。下面对Shadow属性进行详细说明。

Shadow属性的语法格式如下。

Shadow(color, direction, strength)

参数说明如下。

- color: 设置阴影的颜色。
- direction: 设置阴影的显示角度。
- strength: 设置阴影的长度。
- 功能: 对当前组件进行阴影设置。

实现过程

(1) 在页面中添加一个表格，并设置表格的id名称，代码如下。

```
<table id="table1" style="border-color:#0000FF" border=5>  
<tr><td align=center><strong>明日科技</strong></td>
```

```
</tr>
<tr><td align=center>
<pre>明日科技图书</pre>
</table>
```

(2) 编写用于实现表格阴影效果的JavaScript代码。

```
<script langrige="JavaScript">
table1.style.filter="progid:DXImageTransform.Microsoft.Shadow(Color=#333333,
Direction=120, strength=5)";
</script>
```

举一反三

根据本实例，读者可以：

对图片进行阴影效果；

制作一个动态移动的阴影效果。

## 实例211 立体表格

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\07\211

实例说明

在浏览网页时，有些网页中表格的行标记和列标记都具有立体效果，跟 SQL Server 2000中的表格样式差不多。本实例将实现这一效果。运行结果如图7.23所示。

	姓名	性别	籍贯
厂长	李先生	男	长春
党部	张先生	男	天津
工人	刘先生	男	四川

图7.23 立体表格

技术要点

本实例主要是对表格的 borderColorLight（边缘的灯光颜色）、borderColorDark（边缘的模糊颜色）和bgColor（背景颜色）属性进行颜色设置，并设置每行的首单元格的bgColor属性。

### 实现过程

（1）在页面中添加一个表格，并设置表格及行标记的id名称，代码如下。

```
<table id="table1" >
  <tr id="btr" >
    <td align=center >&nbsp;&nbsp;&nbsp;</td>
    <td align=center nowrap >姓名</td>
    <td align=center nowrap >性别</td>
    <td align=center nowrap >籍贯</td>
  </tr>
  <tr>
    <td align=center nowrap id="btb1">厂长</td>
    <td align=center nowrap >李先生</td>
    <td align=center nowrap >男</td>
    <td align=center nowrap >长春</td>
  </tr>
  <tr>
    <td align=center nowrap id="btb2">党部</td>
    <td align=center nowrap >张先生</td>
    <td align=center nowrap >男</td>
    <td align=center nowrap >天津</td>
  </tr>
  <tr>
    <td align=center nowrap id="btb3">工人</td>
```

```
<td align=center nowrap >刘先生</td>
<td align=center nowrap >男</td>
<td align=center nowrap >四川</td>
</tr>
</table>
```

(2) 编写用于实现立体表格的JavaScript代码。

```
<script language="javascript">
table1.border=1;
table1.cellSpacing=0;
table1.borderColorLight="#333333";
table1.borderColorDark="#efefef";
btr.bgColor="#cccccc";
var s="";
for (var i=1;i<4;i++) {
    s=eval("btb"+i).id;
    eval(s).bgColor="#cccccc";
}
</script>
```

举一反三

根据本实例，读者可以：

使整个表格具有立体效果；

使选中的单元格具有立体效果。

## [实例212 虚线边框表格](#)

本实例是一个人性化实例

实例位置：光盘\mingrisoft\07\212

## 实例说明

在用表格输出数据时，可以对表格边框及单元格边框的样式进行设置，这样可以避免表格的单调性。本实例将表格的边框设置为虚线。运行结果如图7.24所示。

	姓名	性别	籍贯
校长	方先生	男	长春
老师	张先生	男	天津
老师	李先生	男	四川

图7.24 虚线边框表格

## 技术要点

本实例主要是用单元格的style的borderTop、borderBottom、borderLeft和borderRight属性来设置单元格的上、下、左、右4条边的颜色、粗细及样式。

本实例的主要思想如下。

(1) 在页面中添加一个表格，并设置表格的外边框及单元格的边框为0。

(2) 用单元格的style样式中的borderTop和borderLeft属性，将所有单元格的上边和左边以虚线显示。

(3) 将右边最后一列的单元格用style样式中的borderRight属性以虚线显示。

(4) 将最后一行的单元格用style样式中的borderBottom属性将下边以虚线显示。

## 实现过程

(1) 在页面中添加一个表格，并设置表格及单元格的id名称，将表格的外边框宽度设为0。代码如下。

```
<table id="table1" align=center width="300" border="0" cellspacing="0">
```

```
<tr>
```

```

    <td id="btb" align=center>&nbsp;  </th>
    <td id="btb" align=center>姓名</th>
    <td id="btb" align=center>性别</th>
    <td id="btb" align=center>籍贯</th>
</tr>
<tr>
    <td id="btb" align=center>校长</th>
    <td id="btb" align=center>方先生</td>
    <td id="btb" align=center>男</td>
    <td id="btb" align=center>长春</td>
</tr>
<tr>
    <td id="btb" align=center>老师</th>
    <td id="btb" align=center>张先生</td>
    <td id="btb" align=center>男</td>
    <td id="btb" align=center>天津</td>
</tr>
<tr>
    <td id="btb" align=center>老师</th>
    <td id="btb" align=center>李先生</td>
    <td id="btb" align=center>男</td>
    <td id="btb" align=center>四川</td>
</tr>
</table>

```

(2) 编写用于实现虚线边框表格的JavaScript代码。

```

<script language="javascript">
var i=0;

```

```
var row=table1.rows.length;
var column=table1.rows(1).cells.length;
for (i=0;i<row*column;i++){
    btb[i].style.borderTop="#000000 1px dotted";
    btb[i].style.borderLeft="#000000 1px dotted";
    for (var j=1;j<=column;j++)
        btb[j*row-1].style.borderRight="#000000 1px
dotted";
    if (i>=(row-1)*column)
        btb[i].style.borderBottom="#000000 1px dotted";
}
</script>
```

举一反三

根据本实例，读者可以：

制作一个线型表格；

制作一个动态变色的虚线表格。

## 实例213 表格作为分割线

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\07\213

实例说明

在浏览网页时，经常会看到网页上用一些分割线将网页分为几个部分，本实例将用表格来作为页面的分割线，并且表格中以渐变颜色进行动态显示。运行结果如图7.25所示。

明日科技图书

---

图7.25 表格作为分割线

## 技术要点

本实例主要是用document对象的write()方法在页面中动态添加一个一行40列的表格，并将表格外边框及单元格的边框设为0，再用setTimeout()方法以指定时间动态修改每个单元格的背景颜色，以实现表格的动态渐变效果。

## 实现过程

编写用于实现表格作为分割线的JavaScript代码。

```
<script language="JavaScript">
changcol=Array(6,7,8,9,'a','b','b','c','d','e','f')
t="<tableheight=4width=100%cellspacing=0 cellpadding=0>
<tr>"
for(x=0;x<40;x++){t+="<td id=td"+x+"></td>"}
document.write(t+"</tr></table>")
function line(y){
for(i=0;i<40;i++){
c=(i+y)%20;
if(c>10){c=20-c}
document.all["td"+
(i)].bgColor=""#00"+changcol[c]+"00"+changcol[c]+"""}
y++;
setTimeout('line('+y+')',1);
}
line(1);
</script>
```

## 举一反三

根据本实例，读者可以：

制作一个线型表格；

制作一个动态变色的虚线表格。

## 实例214 表格向下展开

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\07\214

实例说明

本实例将给表格赋予一种动态的展开效果，当鼠标移动到表格时，表格将自动向下展开，当鼠标移出表格时，表格将恢复到原来的样子。运行结果如图7.26、图7.27所示。



图7.26 表格展开前的效果



图7.27 表格向下展开

技术要点

本实例主要是用`setTimeout()`方法在表格的`onMouseOver`事件（鼠标移入）和`onMouseOut`事件（鼠标移出）以指定的时间动态增减表格的高度（`height`属性），当高度增减到一定程度时，用`clearTimeout()`方法将`setTimeout()`的值进行清除。

实现过程

(1) 在页面中添加一个表格，并设置表格的id名称。代码如下。

```
<table width="200" height="30" border="2" cellpadding="0" cellspacing="0"
```

```

    id="mytd" onMouseOver="over()" onMouseOut="out()"
style="font-size:14px">
    <tr>
    <td>这里是<br>明日科技<br>图书情况反馈主页<br>主页地
址:www.mingrisoft.com</td>
</tr>
</table>>

```

(2) 编写用于实现表格作为分割线的JavaScript代码。

自定义函数over(), 用于将表格的高度自动增加, 其代码如下。

```

function over() {
    var h = parseInt(mytd.height);
    if (h < 120) {
        mytd.height = h+2;
        clearTimeout(act);
        act = setTimeout('over()', 10);
    }
}

```

自定义函数out(), 将增加高度后的表格恢复到原来的高度, 其代码如下。

```

function out() {
    var h = parseInt(mytd.height);
    if (h > 30) {
        mytd.height = h-2;
        clearTimeout(act);
        act = setTimeout('out()', 10);
    }
}

```

</script>

举一反三

根据本实例，读者可以：

实现当鼠标指向表格时，使表格动态放大；

使表格根据滚动条的移动进行缩放。

## 第8章 图形图像与多媒体

图片与鼠标相关操作

图片与时间相关操作

图片的动画效果

选择头像

在页面中播放图片

播放音乐

插入Flash动画

播放视频文件

## 8.1 图片大小

在页面中浏览图片时往往会对图片进行放大、缩小的操作，下面将通过 JavaScript 编写一些能够改变图片大小的实例。

### 实例215 打开自定义大小的图片

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\215

实例说明

本实例通过一些简单的页面设计，使用户能够在文本框中输入图片的宽度和高度，当用户单击“预览”按钮后，程序会根据用户所指定的图片路径以及图片的大小在页面的居中位置显示该图片，如图8.1所示。

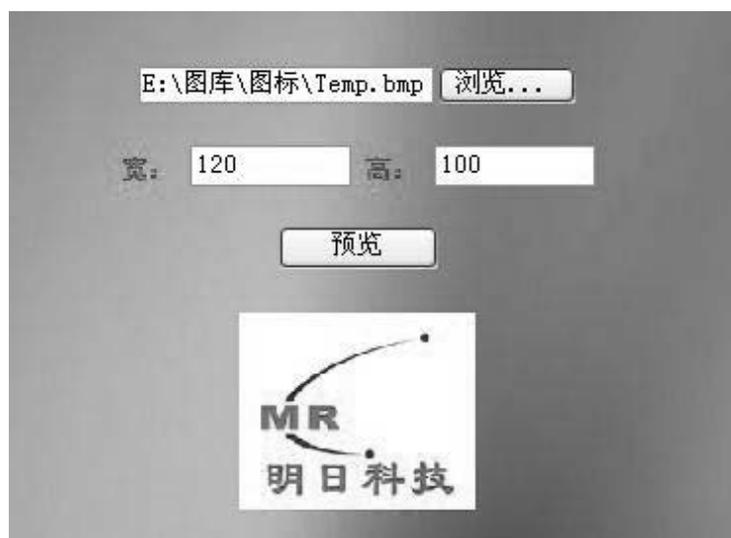


图8.1 打开自定义大小的图片

技术要点

本实例主要通过 `innerHTML()` 方法改变 `<div>` 标记对象内的子标记和内容（不含标记对象本身）。

实现本实例时，首先需要根据用户输入的文件名、宽和高组成显示图片的字符串，然后再通过层的 `innerHTML()` 方法输出该字符串，将指定大小的图片显示在页面上。

### 实现过程

(1) 在 `<body>` 标记中编写页面布局代码，代码如下。

```
<form name="form1" method="post" action="">
<p align="center">
  <input type="file" name="textfield">
</p>
<p align="center"><span class="style3">宽: </span>
  <input name="widthtext" type="text" value="" size="10"
width="60">
  <span class="style3">高: </span>
  <input name="heighttext" type="text" size="10"
width="60">
</p>
<p align="center">
  <input type="button" name="Submit2" value=" 预览 "
onClick="showpic();">
</p>
</form>
```

(2) 编写在页面居中位置显示图片的 JavaScript 代码，在函数体中主要使用字符串的连接动态生成一些 HTML 脚本，最后调用层的 `innerHTML()` 方法在页面中输出指定大小的图片。代码如下。

```
<script language="javascript">
```

```

var str;
function showpic() {
    str = "";
    var ima = form1.textfield.value;
    str = str+"<table WIDTH=' 300' BORDER=' 0'
ALIGN=' CENTER' >";
    str = str+"<tr><td>";
    str = str+"<input NAME=' images1' TYPE=' image'
ALIGN=' MIDDLE' BORDER=' 0' Src = "+ima+" width =
"+form1.widthtext.value+" height =
"+form1.heighttext.value+" >";
    str = str+"</td></tr>";
    str = str+"</table>";
    div1.innerHTML = str;
}
</script>

```

举一反三

根据本实例，读者可以：

制作图片浏览器；

按指定大小布局页面中的所有图片；

实现选择自定义头像时，以指定的图片大小进行浏览。

## [实例216 图片放大缩小](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\216

实例说明

在页面中浏览图片时，有的图片是固定大小的，但是有的图片却不是很清晰，为了方便浏览者浏览图片，使图片应该能够放大缩小。本实例实现了最为简单的图片的放大和缩小，当用户在页面中单击“放大”按钮时，图片会按指定的大小不断增大；单击“缩小”按钮时，图片会按指定的大小不断缩小，如图8.2所示。



图8.2 图片放大缩小

### 技术要点

本实例主要应用了一些简单的条件判断语句，然后根据当前图片的大小来放大或者缩小图片的宽度和高度。

### 实现过程

(1) 编写图片放大的JavaScript代码。

```
<script language="javascript">
functionblowup()//放大
{
    varheight= images1.height;
    varwidth= images1.width;
    if ((height<=height *2) || (width<=width *2))
```

```
{
    images1.height= images1.height+20;
    images1.width= images1.width+20;
}
}
</script>
```

(2) 编写图片缩小的JavaScript代码。

```
<script language="javascript">//缩小
function reduce()
{
    if ((images1.width>100) || (images1.height>100))
    {
        images1.height= images1.height-20;
        images1.width= images1.width-20;
    }
}
</script>
```

(3) 在<body>标记中设置按钮的onclick事件，代码如下。

```
<input type="button" value="放大" onclick="blowup()"> <--
放大-->
```

```
<input type="button" value="缩小" onclick="reduce()"> <--
缩小-->
```

举一反三

根据本实例，读者可以：

实现图片浏览器；

实现图片放大镜。

## 实例217 通过鼠标滚轮放大缩小图片

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\08\217

实例说明

本实例优化了实例216中的功能，实现了使用鼠标滚轮放大缩小图片，当用户将鼠标移动到图片上时，图片将会获取焦点，然后用户可以滚动鼠标滚轮来改变图片的大小，如图8.3所示。

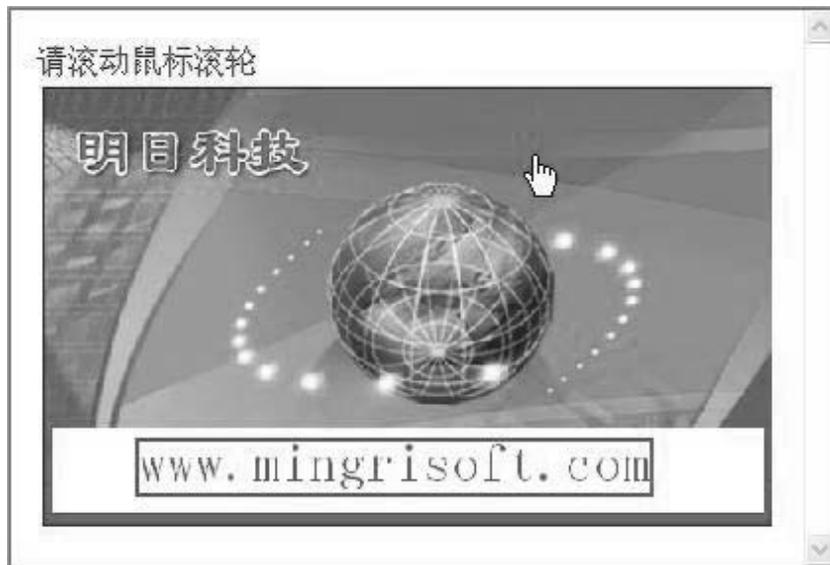


图8.3 通过鼠标滚轮放大缩小图片

技术要点

本实例主要是通过event对象的wheelDelta属性（用于获取鼠标的滚轮键的值）控制图片的放大或缩小的倍数，再根据获取的值改变图片的style属性的zoom属性来实现。下面介绍zoom属性。

功能：设置或检索对象的缩放比例。

语法：

zoom : normal | number

参数说明如下。

- normal：使用对象的实际尺寸。

● number: 百分数|无符号浮点实数。浮点实数值为 1.0 或百分数为 100%时相当于此属性的normal值。

实现过程

(1) 编写滚动鼠标滚轮改变图片大小的函数。代码如下。

```
<script language="javascript">
function bigimg(i)
{
    var zoom = parseInt(i.style.zoom, 10) || 100;
    zoom+= event.wheelDelta/12;
    if(zoom > 0 )
    i.style.zoom=zoom+'%';
    return false;
}
</script>
```

(2) 添加页面设计代码，并将图片的OnMouseWheel事件设置为bigimg(this)函数，以实现在图片上滚动鼠标滚轮时改变图片大小。代码如下。

```
<body>
<tablewidth="100%">
    <tr>
        <td><span class="style1">请滚动鼠标滚轮</span></td>
    </tr>
</table>
<center>
<a href="http://www.mingrisoft.com">

```

```
</a>
```

```
</center>
```

```
</body>
```

举一反三

根据本实例，读者可以：

实现图片浏览器；

使指定的图片能够通过鼠标滚轮改变图片大小。

## 8.2 图片与鼠标相关操作

鼠标是在浏览网页时最主要的工具，一些基本的操作都离不开鼠标，下面将通过JavaScript编写一些图片用鼠标操作的相关实例。

### 实例218 跟随鼠标移动的图片

本实例可以美化界面

实例位置：光盘\mingrisoft\08\218

实例说明

在浏览一些网站时，经常会看到图标跟随着鼠标的移动而移动，这些跟随鼠标移动的图标有的是 cur 鼠标文件，而有的是一些图片。本实例将使用一个GIF格式的图片作为鼠标光标，然后通过JavaScript 实现图片跟随鼠标移动，其效果如图 8.4所示。



图8.4 跟随鼠标移动的图片

技术要点

本实例将图片放在了一个层中，使层的位置与鼠标的位置相等，当层改变位置时，图片也会随之改变位置。

实现过程

(1) 编写实现跟随鼠标移动的图片的JavaScript代码。

```
var x, y
functionhandlerMM() {
    x= window.event.x+document.body.clientLeft;
    y= window.event.y+document.body.clientTop;
}
functionmakesnake() {
    if (document.all) {
        var ob=document.all("tdiv")
        ob.style.posLeft=x
        ob.style.posTop=y
    }
    var timer=setTimeout("makesnake()",10);
}
document.onmousemove=handlerMM;
```

(2) 添加页面设置代码，当页面加载时调用makesnake( )函数。

```
<body onload="makesnake()" background="mr.jpg">
<div id="tdiv" style='position:absolute'>
    <img src='Mymouse.gif' width='32' height='32'
class="spanstyle">
</div>
</body>
```

举一反三

根据本实例，读者可以：

改变鼠标的光标样式；

实现当鼠标移动时使图片慢慢跟随鼠标移动；

实现文字跟随鼠标移动。

## 实例219 可以左右拖动的图片

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\08\219

实例说明

本实例将使用 JavaScript 编写一个可以左右拖动的图片，当用户在图片上按下鼠标左键不放手时，就可以左右拖动图片，当释放鼠标左键时，则将图片放置在释放鼠标左键时的位置上，如图8.5所示。



图8.5 可以左右拖动的图片

技术要点

在制作本实例时，首先编写用于实现左右拖动图片的功能函数，再通过window对象的setInterval()方法每隔1ms执行一次实现左右拖动图片的函数，最终实现可以左右拖动图片的效果。

实现过程

(1) 在<body></body>区域中添加一段CSS样式，其代码如下。

```
<STYLE type=text/css>#floater {  
    LEFT: 445px; POSITION: absolute; TOP:-3px; VISIBILITY:  
visible; WIDTH: 125px; Z-INDEX: 10}  
</STYLE>
```

(2) 编写用于实现可以左右拖动图片的JavaScript代码。

```
<Script language="JavaScript">  
    self.onError=null;  
    currentX=0;
```

```

whichIt=null;
lastScrollX=0;
NS= (document.layers) ?1 :0;
IE= (document.all) ?1:0;
functionheartBeat() {
    if(IE) diffX=document.body.scrollLeft;
    if(NS)diffX= self.pageXOffset;
    if(diffX != lastScrollX) {
        percent= .1 * (diffX-lastScrollX);
        if(percent>0) {
            percent=Math.ceil(percent);
        }else {
            percent=Math.floor(percent);
        }
        if(IE)
document.all.floater.style.pixelLeft+=percent;
        if(NS) document.floater.left+=percent;
        lastScrollX= lastScrollX+percent;
    }
}
function checkFocus(x) {
    stalkerx=document.floater.pageX;
    stalkerwidth=document.floater.clip.width;
    if( (x> stalkerx&&x< (stalkerx+stalkerwidth)) )
        return true;
    else
        return false;
}

```

```

}
function grabIt(e) {
    if(IE) {
        whichIt= event.srcElement;
        while (whichIt.id.indexOf("floater")==-1) {
            whichIt=whichIt.parentElement;
            if (whichIt==null) { return true; }
        }
        whichIt.style.pixelLeft=whichIt.offsetLeft;
        currentX= (event.clientX+document.body.scrollLeft);
    }else{
        window.captureEvents(Event.MOUSEMOVE);
        if(checkFocus (e.pageX)) {
            whichIt=document.floater;
            StalkerTouchedX= e.pageX-document.floater.pageX;
        }
    }
    return true;
}
function moveIt(e) {
    if (whichIt==null) { return false; }
    if(IE) {
        newX= (event.clientX+document.body.scrollLeft);
        distanceX= (newX-currentX);
        currentX=newX;
        whichIt.style.pixelLeft+=distanceX;
    }
}

```

```

        if(whichIt.style.pixelLeft<document.body.scrollLeft)
        whichIt.style.pixelLeft= document.body.scrollLeft;
        if(whichIt.style.pixelLeft>document.body.offsetWidth-
        document.body.scrollLeft-
        whichIt.style.pixelWidth-
        20)whichIt.style.pixelLeft=document.body.offsetWidth-
        whichIt.style.pixelWidth-20;
        event.returnValue= false;
    }else{
        whichIt.moveTo(e.pageX-StalkerTouchedX);
        if(whichIt.left<0+self.pageXOffset)whichIt.left=0+self.pageXOffset;
        if( (whichIt.left+whichIt.clip.width)>=
        (window.innerWidth+self.pageXOffset-17))whichIt.left =
        ((window.innerWidth+self.pageXOffset)-
        whichIt.clip.width)-17;
        return false;
    }
    return false;
}
functiondropIt() {
    whichIt=null;
    if(NS>window.releaseEvents (Event.MOUSEMOVE);
    return true;
}
if(NS) {
    window.captureEvents(Event.MOUSEUP|Event.MOUSEDOWN);

```

```

        window.onmousedown= grabIt;
        window.onmousemove=moveIt;
        window.onmouseup=dropIt;
    }
    if(IE) {
        document.onmousedown= grabIt;
        document.onmousemove=moveIt;
        document.onmouseup=dropIt;
    }
    if(NS || IE)
action>window.setInterval("heartBeat()",1);
</Script>

```

(3) 添加页面设计代码。

```

<DIV align="center" id="floater">
    <TABLEheight="10"width="24">
        <TR>
            <TD
                align="middle"height="6"vAlign="center"width="76">
                    
                </TD>
            </TR>
        </TABLE>
    </DIV>

```

举一反三

根据本实例，读者可以：

实现随意地移动图片。

## 实例220 随意拖动图片

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\08\220

实例说明

本实例与上例相比容易一些，本例实现随意拖动图片的主要方式是使用了div层，在本例中使用JavaScript声明了几个自定义函数，主要是模拟鼠标的基本操作。在页面中用户只需在图片上单击鼠标左键，然后移动鼠标，即可将图片从当前位置移动到另一个位置上，如图8.6所示。



图8.6 随意拖动图片

技术要点

本实例通过 JavaScript 定义的函数，模拟鼠标的相关操作，然后在一个自定义函数中将相对应的函数赋值给页面，使页面中的鼠标事件调用自定义函数，使用event对象的x和y属性来获得当前鼠标的位置。

实现过程

(1) 编写实现随意拖动图片的JavaScript代码。

```
<script language="javascript">
```

```
drag = 0;
```

```
move = 0;
```

声明一个自定义函数mousedown()，使变量X1和Y1获得鼠标到层左上角的距离，代码如下。

```
functionmousedown()
```

```
{
```

```

if (drag)
{
    X1=window.event.x-parseInt(dragimages.style.left);
    Y1=window.event.y-parseInt(dragimages.style.top);
    dragimages.style.Index+=1;
    move=1;
}
}
functionmouseStop()
{
    window.event.returnValue= false;
}
functionmousemove()
{
    if (move)
    {
        dragimages.style.left=window.event.x-X1;
        dragimages.style.top=window.event.y-Y1;
    }
}
functionmouseup()
{
    move=0;
}

```

在id为div1层的OnMouseMove事件中调用mousemove()函数，在OnMouseDown事件中调用mousedown()函数，在OnMouseUp事件中调用

mouseup() 函数，在OnDragStart事件中调用mouseStop() 函数，代码如下。

```
function remove()  
{  
    document.all.div1.onmousemove = mousemove();  
    document.all.div1.onmousedown = mousedown();  
    document.all.div1.onmouseup = mouseup();  
    document.all.div1.ondragstart = mouseStop();  
}  
</script>
```

(2) 添加页面设计代码，当页面加载时调用remove函数，代码如下。

```
<body onLoad="remove()">  
    <div id="div1"  
onMouseOver="dragimages=div1;drag=1;" style="height:77px;left:  
10px;position:absolute;top:10px;width:90px">  
          
    </div>  
</body>
```

举一反三

根据本实例，读者可以：

实现随意拖动的控件；

实现随意拖动的时钟。

## [实例221 当鼠标经过图片时显示图片](#)

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\08\221

实例说明

在一些网站中，为了使页面更加美观，将网站工具栏中的按钮设置为模糊显示状态，当用户将鼠标移动到该按钮上时，此按钮则会十分清晰，这样制作也会使网页更具有个性。本实例将使用JavaScript实现当鼠标经过图片时显示图片，如图8.7、图8.8所示。



图8.7 当图片没有被选中时以模糊的状态显示



图8.8 当图片被选中时以清晰的状态显示

技术要点

本实例主要通过设置CSS样式中的滤镜opacity改变图片的透明度。有关滤镜技术的详细介绍请参见实例127。

实现过程

(1) 编写用于实现当鼠标经过图片时显示图片的JavaScript代码。

声明一个带参数的函数，通过判断参数which的值来设置图片的透明度，代码如下。

```
<script language="JavaScript">  
functionmakevisible(cur, which)  
{
```

```
if (which==0)
    cur.filters.alpha.opacity=100
else
    cur.filters.alpha.opacity=20
}
</script>
```

(2) 添加页面设计代码，使图片的 `onMouseOver` 和 `onMouseOut` 事件调用不同参数的 `makevisible(cur, which)` 函数，代码如下。

```
<body>
    
</body>
```

举一反三

根据本实例，读者可以：

实现当鼠标移动到图片上时，使图片渐渐显示；

使用滤镜技术将图片设置为某种显示状态。

## [实例222 改变图片获取焦点时的状态](#)

这是一个实用的实例

实例位置：光盘\mingrisoft\08\222

实例说明

在一些网站中为了突出某幅图片或链接的焦点，需要设置其获取焦点时的显示状态。在本例中将使用JavaScript编写改变图片获取焦点时的状态的函数。当用户将鼠标移动到图片上时，图片就会以灰度状态显示，如图8.9、图8.10所示。



图8.9 改变图片获取焦点时的状态1



图8.10 改变图片获取焦点时的状态2

### 技术要点

本实例同样应用了CSS样式中的滤镜opacity，通过设置滤镜中的透明度来改变图片获取焦点时的显示状态。

### 实现过程

(1) 编写用于实现改变图片获取焦点时显示状态的JavaScript代码。

```
<script language="JavaScript">
function visible(cursor,i)
{
if (i==0)
    cursor.filters.alpha.opacity=100;
else
    cursor.filters.alpha.opacity=30;
}
</script>
```

(2) 添加页面设计代码，使图片的 onmouseover 事件和 onmouseout 事件调用 visible() 函数，其代码如下。

```
<body>
<table border="0" cellpadding="0" cellspacing="0">
  <tr>
    <td align="center" bgcolor="#CCCCCC">
      <a class="link_image"
href="http://www.mingrisoft.com">
        
      </a>
    </td>
  </tr>
</table>
</body>
```

举一反三

根据本实例，读者可以：

使用CSS样式中的滤镜改变图片的样式；

实现当鼠标移动到图片上时使图片渐隐渐现。

## [实例223 抖动的图片](#)

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\08\223

实例说明

在浏览一些网站时，经常会看到带有抖动效果的图片，将鼠标移动到图片上时，此图片就会开始抖动，当鼠标离开此图片时，图片又

会恢复到原来的状态，使用这样的方法可以优化页面的动态效果。本例使用JavaScript制作一个会抖动的图片，如图8.11所示。



图8.11 抖动的图片

### 技术要点

本实例主要利用if...else语句判断当前的变量a的值，如果a的值符合某一条件，则执行图片向某一方向的抖动。

### 实现过程

(1) 在页面中添加一段CSS样式代码，其代码如下。

```
<style>
.ditheringimg
{
    position : relative
}
</style>
```

(2) 利用JavaScript编写抖动的图片函数，代码如下。

```
<script language="javascript">
var dithering=8;//抖动频率
var stopit=0;
var a=1;
function init(which)
{
    stopit=0;
    shake=which;
    shake.style.left=0;
```

```

    shake.style.top=0;
}
functionditheringimage()
{
    if ((!document.all&&
!document.getElementById) || stopit==1)
        return
    if (a==1)
    {
        shake.style.top=parseInt(shake.style.top)+dithering;/
/图片向上抖动
    }
    else if (a==2)
    {
        shake.style.left=parseInt(shake.style.left)+dithering
;//图片向左抖动
    }
    else if (a==3)
    {
        shake.style.top=parseInt(shake.style.top)-
dithering;//图片向上抖动
    }
    else
    {
        shake.style.left=parseInt(shake.style.left)-
dithering;//图片向左抖动
    }
}

```

```

    if (a<4)
    a++;
    else
    a=1;
    setTimeout("ditheringimage()",50);
}
function stoprattle(which)
{
    stopit=1;
    which.style.left=0;
    which.style.top=0;
}
</script>

```

(3) 在页面中添加图片，使其onMouseOver事件和onMouseOut事件调用JavaScript定义的函数，代码如下。

```

<body>
    
</body>

```

举一反三

根据本实例，读者可以：

实现图片的弹簧效果；

使图片上下抖动。

## 实例224 鼠标移动放大图片

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\08\224

实例说明

在动态网站的人机交互页面中，经常会用到图片。如果要以图片的实际尺寸进行显示，将会给整个网站带来不便，而且图片大小不好控制，使网站页面布局混乱。要解决此问题，可以在页面中加入相关鼠标事件。运行本实例，网页中的图片以缩略图的形式显示，当鼠标移动到指定图片时，该图片将以实际尺寸进行显示；当鼠标离开图片时，该图片又将以缩略图的形式显示。实例运行结果如图8.12所示。

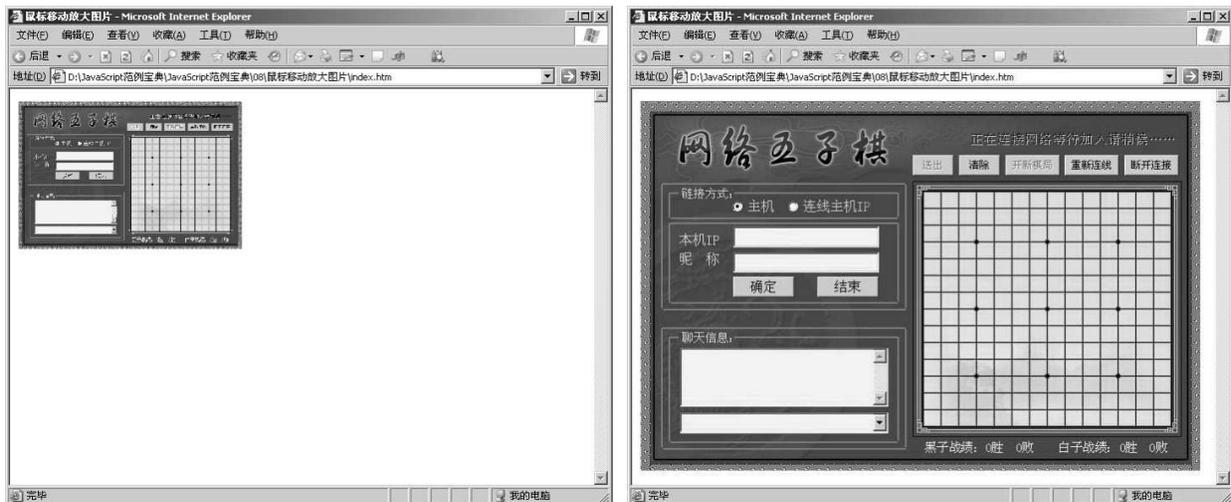


图8.12 鼠标移动放大图片

技术要点

本实例主要应用Math对象实现鼠标移动放大。该对象是一个静态对象，不能使用new关键字创建对象实例，应直接使用“对象名.成员”的格式来访问其属性或方法。

语法格式如下。

Math.method(value[, value])

Math对象的相关属性以及说明如表8.1所示。

表8.1 Math对象的属性

属 性	描 述
e	欧拉常量 (2.718281828459045)
LN2	2 的自然对数 (0.6931471805599453)
LN10	10 的自然对数 (2.3025850994046)
LOG2E	以 2 为底数的 e 的对数 (1.4426950408889633)
LOG10E	以 10 为底数的 e 的对数 (0.4342944819032518)
PI	圆周率常数 (3.141592653589793)
SQRT1/2	0.5 的平方根 (0.7071067811865476)
SQRT2	2 的平方根 (1.4142135623730951)

Math对象的相关方法以及说明如表8.2所示。

表8.2 Math对象的方法

方 法	描 述
abs(x)	返回 x 的绝对值
acos(x)	返回 x 弧度的反余弦
asin(x)	返回 x 弧度的反正弦
atan(x)	返回 x 弧度的反正切
atan2(x,y)	返回坐标 (x,y) 对应的极坐标角度
ceil(x)	返回大于或等于 x 的最小整数
cos(x)	返回 x 的余弦
exp(x)	返回 e 的 x 乘方
floor(x)	返回小于或等于 x 的最大整数
log(x)	返回 x 的自然对数
max(x,y)	返回 x 和 y 中的最大数
min(x,y)	返回 x 和 y 中的最小数
pow(x,y)	返回 x 对 y 的次方
random()	返回 0 和 1 之间的随机数
round(x)	返回最接近 x 的整数
sin(x)	返回 x 的正弦值
sqrt(x)	返回 x 的平方根
tan(x)	返回 x 的正切值

### 实现过程

(1) 应用 JavaScript 脚本创建两个函数，分别是 move() 函数、out() 函数。move() 函数主要用于显示图片的实际大小；out() 函数主要用于当鼠标离开图片时，图片将以缩略图的形式显示，代码如下。

```

<script language="javascript">
var w=images1.width;
var h=images1.height;
images1.height = Math.floor(h*0.4);
images1.width=Math.floor(w*0.4);
functionmove()
{
    images1.height=h;
    images1.width=w;
}
function out()
{
    images1.height=Math.floor(h*0.4);
    images1.width=Math.floor(w*0.4);
}
</script>

```

(2) 将插入图片的Border属性值设为0, 同时当鼠标在图片上移动时, 应用图片的onMouseMove (鼠标移动) 事件调用自定义函数move(); 当鼠标移出图片时, 应用图片的onMouseOut (鼠标移出) 事件调用自定义函数out(), 代码如下。

```

<input NAME="images1" TYPE="image" ID="images1"
SRC="00.jpg" ALIGN="MIDDLE" BORDER="0" onMouseMove= "move()"
onMouseOut="out()">

```

举一反三

根据本实例, 读者可以:

在编写商务网站时应用该技术;

在开发办公自动化管理系统时应用该技术;

在编写宠物网站时应用该技术。

## 8.3 图片与时间相关操作

网页内部的一些效果类的操作都离不开时间，例如定时更换页面背景，使某些文件按指定的方式显示，图片不停闪动等。下面将通过JavaScript编写一些图片与时间相关操作的实例。

### 实例225 定时隐藏图片

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\08\225

实例说明

本实例实现了定时隐藏图片的功能，类似于倒计时功能，当页面打开后程序将会从10秒开始倒计时，并在文本框中显示剩余秒数，当倒计时为0秒时，则设置层的visibility属性为隐藏状态。由于图片在层中，所以图片同时也被隐藏，如图8.13所示。



图8.13 定时隐藏图片

技术要点

本实例首先将图片放置在层中，然后将层的visibility属性设置为hidden，使层被隐藏，由于图片放置在层中，所以图片同时也被隐藏。

实现过程

(1) 编写用于实现定时隐藏图片的JavaScript代码。

```
<SCRIPTLANGUAGE="JavaScript">
var sec=10;
var timer;
functionhidepic()
{
    sec--;
    if (sec==0) {
        textfield.value= "图片被隐藏";
        soccer.style.visibility=(soccer.style.visibility=="hidden") ? "visible" : "hidden";
    }
    else{
        textfield.value= "图片会在 "+sec+"秒后隐藏";
        setTimeout("hidepic()",1000);
    }
}
</SCRIPT>
```

(2) 添加页面设计代码，当页面加载时调用hidepic函数，代码如下。

```
<bodyonLoad= "hidepic();">
<center>
    <inputname="textfield" type="text" size="20"><br>
    <DIV ID="soccer"STYLE="position:absolute; left:333px;
top:43px">
    <imgborder="0" src="bg. jpg">
</DIV>
```

</center>

</body>

举一反三

根据本实例，读者可以：

实现倒数计时器；

使图片在指定的时间内隐藏，然后再次显示。

## 实例226 根据时间变换页面背景

本实例可以美化界面

实例位置：光盘\mingrisoft\08\226

实例说明

有些网站为了丰富页面的显示效果，将页面制作成根据时间变换页面背景的样式，这样一来不但会使浏览者感到该网站制作的非常新颖，也会使网站更具有个性，本例将实现这一功能。在本例中通过调用Date对象的getHours()方法获得当前时间的小时，然后根据不同的时间段来改变页面的背景图片。效果如图8.14所示。



中午好！现在是中午12点

图8.14 根据时间变换页面背景

### 技术要点

本实例中的技术要点是调用Date对象的getHours()方法得到当前小时，然后在一定的时间段内判断当前小时是否符合指定的时间段，如果符合则使用document的write()方法在页面中显示一幅图片并在图片下方输出指定的字符串。

### 实现过程

(1) 在页面中添加一段CSS样式，其代码如下。

```
<style type="text/css">
body {
    background-color: #FFFFFF;
}
</style>
```

(2) 编写用于实现根据当前时间变换页面背景的JavaScript代码。

```
<script language="javascript">
var now=newDate();
var hour=now.getHours();
if (hour>=0&&hour<5)//凌晨时间在0点到5点
{
    document.write("<center><img src='凌晨好.jpg' width='600' height='399'><center>");
    document.write("<p>");
    document.write("<font size=6 face= 黑体 color=#ff9900>凌晨好! 现在是凌晨"+hour+"点</font>");
}
if (hour>=5&&hour<8)//早上时间在5点到8点
```

```
{
    document.write("<center><img src=' 早上
好.jpg' width=' 600' height=' 399' ><center>");
    document.write("<p>");
    document.write("<font size=6 face= 黑体 color=#ff9900>
早上好! 现在是早上"+hour+"点</font>");
}
if (hour>=8&&hour<11)//上午时间在8点到11点
{
    document.write("<center><img src=' 上午
好.jpg' width=' 600' height=' 399' ><center>");
    document.write("<p>");
    document.write("<font size=6 face= 黑体 color=#ff9900>
上午好! 现在是上午"+hour+"点</font>");
}
if (hour>=11&&hour<13)//中午时间在11点到13点
{
    document.write("<center><img src=' 中午
好.jpg' width=' 600' height=' 399' ><center>");
    document.write("<p>");
    document.write("<font size=6 face= 黑体 color=#ff9900>
中午好! 现在是中午"+hour+"点</font>");
}
if (hour>=13&&hour<17)//下午时间在13点到17点
{
    document.write("<center><img src=' 下午
好.jpg' width=' 600' height=' 399' ><center>");
```

```
document.write("<p>");
document.write("<font size=6 face= 黑体 color=#ff9900>
下午好! 现在是下午"+hour+"点</font>");
}
if (hour>=17&&hour<24)//晚上时间在17点到24点
{
document.write("<center><img src='晚上
好.jpg' width='600' height='399'><center>");
document.write("<p>");
document.write("<font size=6 face= 黑体 color=#ff9900>
晚上好! 现在是晚上"+hour+"点</font>");
}
</script>
```

举一反三

根据本实例，读者可以：

实现在指定的时间段内输出指定的字符串。

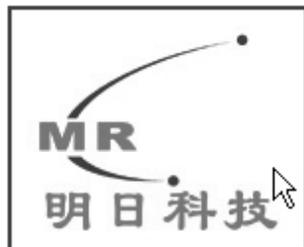
## 实例227 使图片不停闪烁

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\227

实例说明

在一些招商的网站中，页面中包含着大量的广告信息，有时会使用一些网页特效来吸引浏览者。在本例中，为了使图片链接更具吸引力，在图片中增加了不停闪烁的效果，如图8.15所示。



明日科技最新图书

图8.15 使图片不停闪烁

### 技术要点

本例中主要对层进行操作，在glint函数中通过三目运算符设置图片的显示或隐藏状态，然后在指定的时间内重复执行，以达到图片不停闪烁的效果。

### 实现过程

(1) 编写使图片不停闪烁的JavaScript代码，具体代码如下。

```
<SCRIPTLANGUAGE="JavaScript">
var counter=0;
function soccerOnload()//在指定的时间内调用glint函数
{
    setTimeout("glint()", 200);
}
function glint()//设置图片的显示和隐藏状态
{
    div1.style.visibility= (div1.style.visibility=="hidden") ? "visible" : "hidden";
    counter+=1;
    setTimeout("glint()", 200);
}
</SCRIPT>
```

(2) 添加页面设计代码，并在层中加入图片，代码如下。

```
<bodyonload="soccerOnload();">
<DIV ID="div1"STYLE="position:absolute; left:150; top:0">
  <a href="http://www.mingrisoft.com" target="_blank">
    <p></p>
    
    <p></p>
    <font size="3pt" color="#FF0000"> 明日科技最新图书
</font>
  </a>
</DIV>
</body>
```

举一反三

根据本实例，读者可以：

使某一链接快速闪烁；

使图片渐隐渐现。

## 实例228 上下跳动的图片

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\08\228

实例说明

在浏览网站的过程中，经常会看到页面中漂浮着的或者上下跳动的图片链接，一般网站都会采用这种手段做一些广告。本实例将使用JavaScript 制作一个上下跳动的图片。在页面打开后，图片将会从左向右上下跳动，当图片移动到页面的右侧时，图片又会从右向左跳动。效果如图8.16所示。



图8.16 上下跳动的图片

### 技术要点

本实例中的技术要点是实现图片上下跳动，上下跳动的图片在屏幕中有一定的限制，当图片向下移动到指定位置时，图片将会迅速反弹，然后继续跳动。

### 实现过程

(1) 在页面中添加一段CSS样式，其代码如下。

```
<style type="text/css">
#superball {
position:absolute;
left:0;
top:0;
visibility:hide;
visibility:hidden;
width:40;
height:40;
}
</style>
```

(2) 编写用于实现图片上下跳动的JavaScript代码。

```
<script language="JavaScript">
var ballWidth=40;
var ballHeight=40;
varBallSpeed=10;
varmaxBallSpeed=50;
var xMax;
var yMax;
var xPos=0;
var yPos=0;
var xDir= 'right' ;
var yDir= 'down' ;
var superballRunning= true;
var tempBallSpeed;
var currentBallSrc;
var newXDir;
var newYDir;
function initializeBall() {
    xMax=document.body.clientWidth
    yMax=document.body.clientHeight
    document.all("superball").style.visibility= "visible";
    setTimeout('moveBall()',1000);
}
functionmoveBall() {
    if (superballRunning== true) {
        calculatePosition();
        document.all("superball").style.left=xPos+document.bo
dy.scrollLeft;
```

```

        document.all("superball").style.top=
yPos+document.body.scrollTop;
        setTimeout('moveBall()',50);
    }
}

function calculatePosition() {
    if (xDir== "right") {
        if (xPos> (xMax-ballWidth-BallSpeed)) {
            xDir= "left";
        }
    }
    else if (xDir== "left") {
        if (xPos< (0+BallSpeed)) {
            xDir= "right";
        }
    }
    if (yDir== "down") {
        if (yPos> (yMax-ballHeight-BallSpeed)) {
            yDir= "up";
        }
    }
    else if (yDir== "up") {
        if (yPos< (0+BallSpeed)) {
            yDir= "down";
        }
    }
    if (xDir== "right") {

```

```
        xPos=xPos+BallSpeed;
    }
    else if (xDir=="left") {
        xPos=xPos-BallSpeed;
    }
    else {
        xPos=xPos;
    }
    if (yDir=="down") {
        yPos= yPos+BallSpeed;
    }
    else if (yDir=="up") {
        yPos= yPos-BallSpeed;
    }
    else {
        yPos= yPos;
    }
}

window.onload= initializeBall;
window.onresize=newFunction("window.location.reload()");
</script>
```

(3) 页面中添加一个<span>标记, 该标记主要用于插入CSS样式, 然后在层中添加图片, 代码如下。

```
<body>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
```

```
<p>&nbsp;</p>
<p>&nbsp;</p>
<p><a href="http://www.mingrisoft.com"
style="cursor:pointer ">
</a></p>
<a href="http://www.mingrisoft.com" style="cursor:pointer
">
<span id="superball">

</span>
</a>
</body>
```

举一反三

根据本实例，读者可以：

制作在页面中浮动的图片；

制作在页面中随意移动的图片；

制作一个左右跳动的图片。

## [实例229 图片左右晃动](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\08\229

实例说明

本实例使用 JavaScript 制作了一个能够左右晃动的图片，当页面显示后，图片就会从左向右移动，当图片向右移动到一定的区域后

图片又会向左移动，然后重复执行图片左右晃动的操作。效果如图8.17所示。



图8.17 图片左右晃动

### 技术要点

本实例主要通过 JavaScript 所编写的两个函数对图片的位置进行移动，`rect()` 函数主要用于指定图片的位置，`move()` 函数主要设置移动图片的区域。

### 实现过程

(1) 编写实现图片左右晃动的JavaScript代码。

```
<script language="JavaScript">
tempo=0;
imgobj=new Image();
function rect(xx,yy,zz)
{
    imgobj.style.left=x;
    x+= tempo*zz;
    if (x>=(yy+xx)/2)
    {
        if (zz==1)
            tempo--;
        else
            tempo++;
    }
}
```

```

else
{
    if (zz==1)
        tempo++;
    else
        tempo--;
}
if (x>= yy)
{
    x=yy;
    zz=-1;
}
if (x<=xx)
{
    x=xx;
    zz=1;
}
setTimeout('rect('+xx+', '+yy+', '+zz+')', 50);
}
functionmove(picobj, picmovewidth, c)
{
    window_width=document.body.offsetWidth;
    imgobj=document.images[picobj];
    image_width= imgobj.width;
    x1= imgobj.style.left;
    x=Number(x1.substring(0, x1.length-2));
    if (c==0)

```

```

{
  if (picmovewidth==0)
  {
    right_margin=window_width-image_width;
    rect(x, right_margin, 1);
  }
  else
  {
    right_margin=x+picmovewidth-image_width;
    if (picmovewidth<x+image_width)
      window.alert("No space formoving!");
    else
      rect(x, right_margin, 1);
  }
}
else
{
  if (picmovewidth==0)
    right_margin=window_width-image_width;
  else
  {
    x=Math.round((window_width-picmovewidth)/2);
    right_margin=Math.round((window_width+picmovewidt
h)/2)-image_width;
  }
  rect(x, right_margin, 1);
}

```

```
}
```

```
</script>
```

(2) 在页面中添加一幅图片，然后再次写入一段 JavaScript 代码，每间隔 10m/s 调用一次 move( ) 函数，代码如下。

```
<body>
```

```
<a href="http://www.mingrisoft.com">
```

```

```

```
</a>
```

```
<script language="JavaScript">
```

```
setTimeout("move(' picture', 600, 1)", 10);
```

```
</script>
```

```
</body>
```

举一反三

根据本实例，读者可以：

制作在页面中上下晃动的广告链接；

将图片制作成弹球效果。

## 实例230 飘舞的变形图片

这是一个自娱自乐的实例

实例位置：光盘\mingrisoft\08\230

实例说明

本实例所实现的效果是页面中漂浮图片及变形图片的结合，当页面显示后，图片会随意的在页面中飘动，并且在飘动的过程中改变图

片的形状当到达页面的边界时，图片会马上反弹到页面中，然后继续飘动。效果如图8.18、图8.19所示。



图8.18 在页面中飘动的图片



图8.19 改变形状的图片

### 技术要点

本实例实现图片在页面飘舞的效果主要应用了 div 层和 Math 对象的 random() 方法。而实现图片在飘舞的过程中改变形状主要通过一些简单的判断，使图片能够在指定的范围中变大或变小。

本实例中，首先层会向页面的右下角位置开始移动并且改变图片形状，当层的位置等于页面的边界时，程序会根据random()方法获取反弹后的方向，使层的x和y坐标加上指定的值或减少指定的值（层向上x和y的坐标加值，层向下x和y的坐标减值），然后继续运行。

### 实现过程

(1) 编写用于实现飘舞变形图片的JavaScript代码。

下面是使图片随意移动的代码。

```
<SCRIPTLANGUAGE="JavaScript">
var isNS= ((navigator.appName=="Netscape")&&
(parseInt(navigator.appVersion)>=4));
var _all= '';
var _style= '';
var wwidth, wheight;
var ydir= '++';
```

```

var xdir= '++' ;
var id1, id2, id3;
var x=1;
var y=1;
var xl,y1;
if(!isNS)
{
    _all='all.' ;
    _style='.style' ;
}
function getwindowsize()
{
    clearTimeout(id1);
    clearTimeout(id2);
    clearTimeout(id3);
    wwidth=document.body.clientWidth-55;
    wheight=document.body.clientHeight-50;
    id3= setTimeout('randomdir()', 20000);
    animate();
}
function randomdir()
{
    if (Math.floor(Math.random()*2))
    {
        (Math.floor(Math.random()*2)) ?xdir='--' :xdir='++' ;
    }
    else

```

```

    {
        (Math.floor(Math.random()*2)) ?ydir='--' : ydir='++' ;
    }
    id2= setTimeout('randomdir()', 20000);
}
function animate()
{
    eval('x'+xdir);
    eval('y'+ydir);
    if (isNS)
    {
        picture_div.moveTo((x+pageXOffset), (y+pageYOffset))
    }
    else
    {
        picture_div.pixelLeft=x+document.body.scrollLeft;
        picture_div.pixelTop= y+document.body.scrollTop;
    }
    if (isNS)
    {
        if (picture_div.top<=5+pageYOffset) ydir= '++' ;
        if (picture_div.top>=height+pageYOffset) ydir= '--' ;
        if (picture_div.left>=wwidth+pageXOffset)xdir= '--' ;
        if (picture_div.left<=5+pageXOffset)xdir= '++' ;
    }
    else
    {

```

```

        if (picture_div.pixelTop<=5+document.body.scrollTop)
ydir= '++' ;
        if
        (picture_div.pixelTop>=wheight+document.body.scrollTop)
ydir= '--' ;
        if
        (picture_div.pixelLeft>=wwidth+document.body.scrollLeft)
xdir= '--' ;
        if
        (picture_div.pixelLeft<=5+document.body.scrollLeft) xdir=
'++' ;
    }
    id1= setTimeout('animate()', 30);
}

```

(2) 使图片反复地由大变小然后再由小变大，代码如下。

```

var size=1;
var bool= true;
functionpicdistortion()
{
    if (document.all)
    if (bool== true)
    {
        size++;
    }
    if(size==100)
    {
        size--;
    }
}

```

```

        bool=false;
    }
    if(size==10)
    {
        size++;
        bool=true;
    }
    if(bool==false)
    {
        size--;
    }
    imagel.width=130+size;
    imagel.height=120-size;
    setTimeout("picdistortion()",40);
}
</script>

```

(3) 添加页面设计代码，当页面加载时调用getWindowSize()函数和picdistortion()函数，当页面改变大小时调用getWindowSize()函数，代码如下。

```

<bodyOnLoad="getWindowSize(),picdistortion()"OnResize="ge
tWindowSize()">
<div id="picture_div"
style="position:absolute;visibility:visible; left:0px;
top:0px; z-index:-1">
<a href="http://www.mingrisoft.com">
<imgname="imagel" src="rectangle.jpg"border="0">
</a>

```

```
</div>
<script language="javascript">
    varpicture_div=eval('document.'+_all+'picture_div'+_style);
</script>
</body>
```

举一反三

根据本实例，读者可以：

制作在页面中飘舞的文字；

实现多个图片在页面中飘动。

## 8.4 图片的动画效果

本节将通过JavaScript实现一些图片的动画效果。

### 实例231 图片翻转效果

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\231

实例说明

本实例主要使用了CSS样式中滤镜的flipH和flipV样式实现图片的水平翻转和垂直翻转效果。当用户单击“水平翻转”按钮时，“水平翻转”执行结果下的图片变为水平翻转状态；单击“垂直翻转”按钮时，则使“垂直翻转”执行结果下的图片变为垂直翻转状态。效果如图 8.20所示。

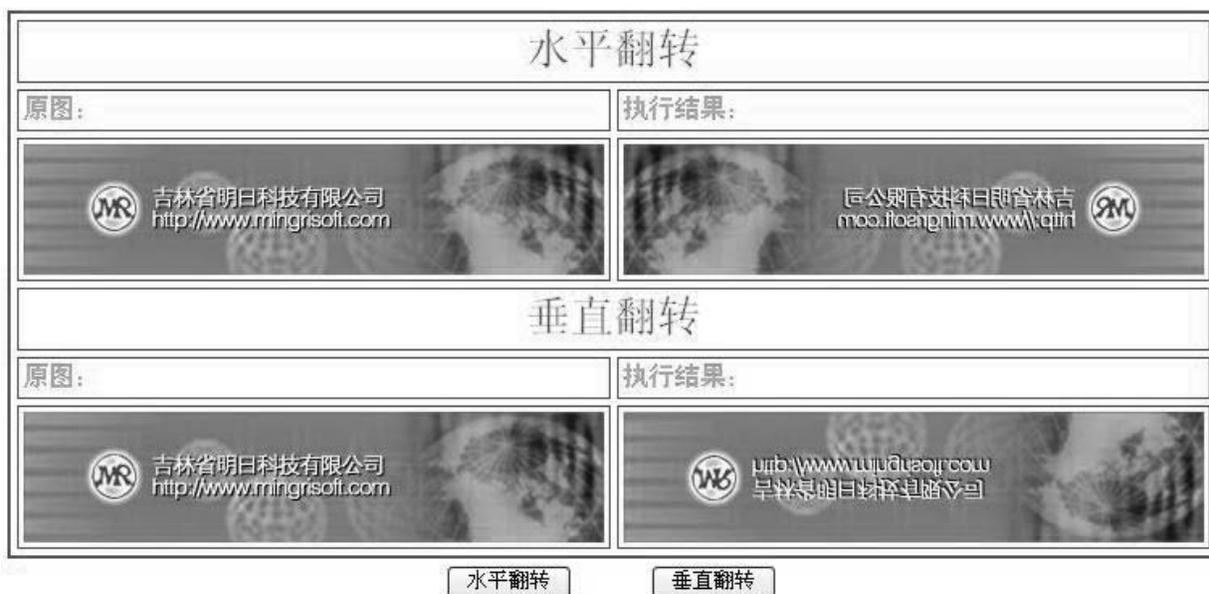


图8.20 图片翻转效果

## 技术要点

使用 CSS 样式中的滤镜 `fliph` 和 `flipV` 技术将图片设置为水平翻转和垂直翻转。有关滤镜技术的详细介绍请参见实例127。

## 实现过程

(1) 在网页中添加表单及相关的表单元素，并设置Form表单的相关属性值，代码如下。

```
<script language="javascript">
functionHturn()//水平翻转
{
    imagell.style.filter=
imagell.style.filter=="fliph"?"":"fliph";
}
functionVturn()//垂直翻转
{
    image22.style.filter=
image22.style.filter=="flipV"?"":"flipV";
}
</script>
```

(2) 添加页面设计代码，使水平翻转按钮调用 `Hturn()` 函数、垂直翻转按钮调用 `Vturn()` 函数，代码如下。

```
<body>
<tablewidth="433"border="2" align="center"cellpadding="3"
cellspacing="4"bordercolor="#006699"bgcolor="#FFFFFF">
<tr>
<td colspan="2"><div align="center"><font
color="#FF0000" size="6">水平翻转</font></div></td>
</tr>
```

```

<tr>
  <tdwidth="165"><font color="#FF9900" face="黑体"
size="4">原图: </font></td>
  <tdwidth="252"><font color="#FF9900" face="黑体"
size="4">执行结果: </font></td>
</tr>
<tr>
  <td></td>
  <td></td>
</tr>
<tr>
  <td colspan="2"><div align="center"><font
color="#FF0000" size="6">垂直翻转</font></div></td>
</tr>
<tr>
  <td><font color="#FF9900" face="黑体" size="4">原图:
</font></td>
  <td><font color="#FF9900" face="黑体" size="4">执行结
果: </font></td>
</tr>
<tr>
  <td></td>
  <td></td>
</tr>
</table>
<p>
<center>

```



为水波倒影(wave)特效。

实现过程

编写用于实现水波倒影图片的JavaScript代码。

```
<body>
<center>

</center>
<br>
<script language="JavaScript">
  functionmapped ()
  {
    setInterval("img1.filters.wave.phase+=10",100);
  }
  if (document.all)
  {
    document.write(' <Center><img id=img1
src="" +document.all.mappedimg.src+' "style="filter:wave(st
rength=3, freq=3,phase=0,lightstrength=30) blur()
flipv()"></Center>');
    window.onload=mapped;
  }
</script>
</body>
```

举一反三

根据本实例，读者可以：

实现有倒影的文字；

实现有倒影的时钟。

## 实例233 图片渐隐渐现

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\233

实例说明

运行本实例，图片将会从模糊状态逐渐变为清晰状态，然后又从清晰状态变为模糊状态。重复执行此操作最后使图片形成渐隐渐现的效果，如图8.22、图8.23所示。



图8.22 图片渐隐渐现1



图8.23 图片渐隐渐现2

技术要点

本实例主要应用了CSS样式中滤镜alpha的opacity属性来改变图片的透明度。本实例中通过一些简单的条件判断，使图片能够在指定的时间内循环渐隐渐现。

实现过程

(1) 编写用于实现图片渐隐渐现的JavaScript代码。

```
<script language="JavaScript">
```

```
var b = 1;
```

```
var c = true;
```

```
function latent()
```

```

{
  if(document.all)
  if(c == true)
    b++;
  else
    b--;
  if(b==100)
  {
    b--;
    c = false
  }
  if(b==10)
  {
    b++;
    c = true;
  }
  myImage.filters.alpha.opacity=b;
  setTimeout("latent()",25);
}
latent();
</script>

```

(2) 在页面中添加一幅图片，代码如下。

```



```

举一反三

根据本实例，读者可以：

实现在浏览图片时使图片渐隐渐现；

使图片渐渐改变亮度。

## 实例234 图片的探照灯效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\08\234

实例说明

本实例将使用JavaScript编写一个图片探照灯的特效，当页面打开后，探照灯将会在图片上重复地左右扫描，将一部分显示为光亮状态，而其他部分显示为灰暗状态，如图8.24、图8.25所示。



图8.24 图片的探照灯效果1



图8.25 图片的探照灯效果2

技术要点

本实例实现的图片的探照灯特效主要应用了 CSS 样式中的滤镜技术，通过滤镜中的 light 属性调用 addPoint() 和 MoveLight() 方法，设置图片中的光源大小，并移动光源。

有关 CSS 样式中 light 的 addPoint() 和 MoveLight() 方法请参见第13章 HTML/CSS 样式。

实现过程

(1) 在页面中添加一段 CSS 样式，并在 <body></body> 标记中加入一幅图片，代码如下。

```
<STYLE type=text/css>
```

```
#imagelight {
  FILTER: light
}
</STYLE>
<center>
<IMG id="imagelight" src="test.jpg" speed="20">
</center>
```

(2) 编写实现图片探照灯效果的JavaScript代码。

```
<script language="javascript">
if (document.all&&window.imagelight) {
  varx=newArray();
  varheading=newArray();
  var y=newArray();
  if (imagelight.length==null) {
    imagelight[0]=document.all.imagelight;
    x[0]=0;
    heading[0]="right";
    y[0]=imagelight[0].height;
    imagelight[0].filters.light.addPoint(100,50,100,255,2
55,255,90);
  }
  else
    for (i=0;i<imagelight.length;i++) {
      x[i]=0;
      heading[i]="right";
      y[i]=imagelight[i].height;
```

```

        imagelight[i].filters.light.addPoint(100, 50, 100, 255, 2
55, 255, 90);
    }
}
function slidelight(cur) {
    imagelight[cur].filters.light.MoveLight(0, x[cur], y[cur]
, 200, -1);
    if (x[cur]
<imagelight[cur].width+200&&heading[cur]=="right")
        x[cur]+=10;
    else if (x[cur]>imagelight[cur].width+200) {
        heading[cur]="left";
        x[cur]-=10;
    }
    else if (x[cur]>-200&&x[cur]<-185) {
        heading[cur]="right";
        x[cur]+=10;
    }
    else {
        x[cur]-=10;
        heading[cur]="left";
    }
}
if (document.all&&window.imagelight) {
    if (imagelight.length==null)
        setInterval("slidelight(0)", imagelight[0].speed);
    else

```

```
    for (t=0;t<imagelight.length;t++){
        var
temp=' setInterval("slidelight(' +t+')",'+imagelight[t].spe
ed+')';
        eval(temp);
    }
}
</SCRIPT>
```

举一反三

根据本实例，读者可以：

制作一个探照灯从图片两边向内侧聚拢的效果；

制作一个探照灯从图片内侧向两边扩散的效果。

## 实例235 雷达扫描图片特效

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\08\235

实例说明

在一些飞机、轮船、坦克等军事武器中都备有雷达扫描装备，当雷达运作时，就会有一个45° 的图片在雷达中旋转。本实例将使用JavaScript编写一个类似于雷达扫描图片的实例，效果如图8.26所示。

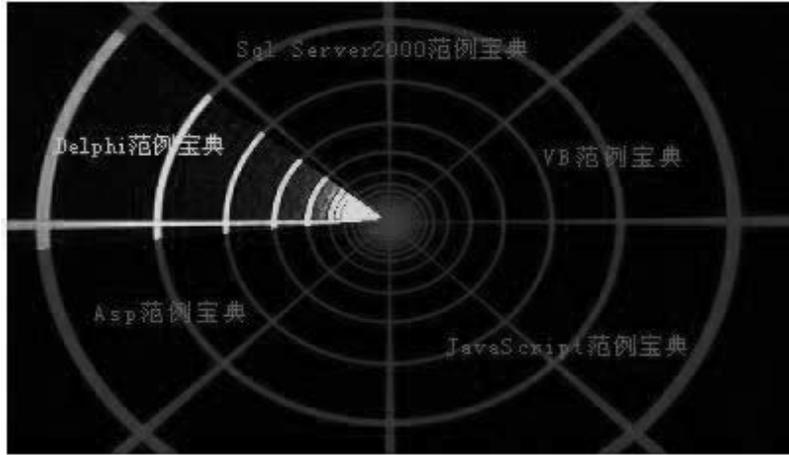


图8.26 雷达扫描图片特效

### 技术要点

本实例主要使用Math 对象的cos( )方法和sin( )方法来获取一个角度，然后使图片以此角度进行旋转。本例形成图片雷达扫描特效的方式是使用图片滤镜的3个方法。

- moveLight: 移动图片的照明位置。
- light.addCone: 设置图片的照明中心点及照明度的大小。
- light.addAmbient: 设置图片的阴影效果。

用户可以在本实例中修改以上方法中的参数，而见到不同的显示效果。

### 实现过程

(1) 编写雷达扫描图片特效的JavaScript代码。

```
<script language="javascript">  
varX=20  
varY=20  
varZ=40  
varxInc=100;  
var yInc=100;  
var r=200;
```

```

varX1=0;
var change= (2 *Math.PI)/360
functionmoveposition()
{
X= r+r*Math.cos(X1*change);
Y= r+r*Math.sin(X1*change);
X1+=10;
if(X1==360)
X1=0;
img1.filters[0].moveLight(0,X,Y,Z,1);
mytimeout=setTimeout('moveposition()',80);
}
functionbeginscan()//开始雷达扫描
{
img1.filters.light.addCone(150,30,0,X,Y,0,255,0,150,20)
;
img1.filters.light.addAmbient(0,255,0,50);
varx=0;
moveposition();
}
</script>

```

(2) 添加页面设计代码，当页面加载时调用beginscan()函数，代码如下。

```

<body onLoad="beginscan();">

</body>

```

举一反三

根据本实例，读者可以：

在图片某一位置上加入超级链接，以实现雷达扫描的动态效果；  
使用滤镜技术将雷达扫描后的图片增加清楚度。

## 实例236 在页面中旋转的图片

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\08\236

实例说明

在网站中经常会见到图片旋转的效果，这些图片一般都是在做一些重复性的操作，例如在页面的某一位置上下移动、左右移动以及做一些规则性的操作。本实例将实现图片在页面中旋转的操作，当页面显示后，图片就会在页面的指定位置上开始旋转，如图8.27所示。



图8.27 在页面中旋转的图片

技术要点

本实例中主要应用了Math对象的sin（返回数的正弦值）和cos（返回数的余弦值），通过取得正弦值和余弦值然后加上一些算法来改变当前层的位置，以使图片在页面中旋转。

实现过程

（1）编写在页面中旋转图片的JavaScript代码。

```
<script language="javascript">
```

```

var x1=200;
var x2=200;
var timer;
var r=60;
var i=0;
function eddy(i)
{
    varob=document.all("rounddiv");
    ob.style.posTop= r*Math.sin((i*Math.PI)/180)+x1;
    ob.style.posLeft= r*Math.cos((i*Math.PI)/180)+x2;
    i=i+1;
    if (r>100)
        {window.clearTimeout(timer);}
    else
        {
            if (i>360)
                {i=0;r= r+1;}
            timer=setTimeout("eddy("+i+")",10);
        }
}
eddy(0);
</script>

```

(2) 在页面中添加一个层，并在层中加入一幅图片，代码如下。

```

<body>
<div id="rounddiv" style="width:50pt; top:198.75pt;
left:256.5pt; position:absolute; z-index:0">


```

</div>

</body>

举一反三

根据本实例，读者可以：

使图片以指定的方式运行，例如图片以三角形形状运行；

实现图片以波浪线形式在页面中来回移动；

将图片制作成滚动字幕的样式在页面中反复移动。

## 实例237 改变形状的图片

这是一个自娱自乐的实例

实例位置：光盘\mingrisoft\08\237

实例说明

本实例将实现一个改变形状的图片。运行本实例，在页面的居中部分将会显示一幅图片，之后图片就会开始改变形状，图片先是由大变小，当图片到达指定的大小时，图片又会由小变大。实例运行结果如图8.28所示。



图8.28 改变形状的图片

技术要点

本实例主要通过一些简单的判断，使图片能够在指定的范围中变大变小。

实现过程

(1) 编写用于实现改变图片形状的JavaScript代码。

```
<script language="javascript">
```

```
var size=1;
```

```
var bool= true;
function imgdistortion()
{
    if (document.all)
    if (bool== true)
    {
        size++;
    }
    if(size==100)
    {
        size--;
        bool=false;
    }
    if(size==10)
    {
        size++;
        bool=true;
    }
    if(bool==false)
    {
        size--;
    }
    imagel.width=130+size;
    imagel.height = 120-size;
    setTimeout("imgdistortion()", 30);
}
</script>
```

(2) 添加页面设计代码，当页面加载时调用

```
<body onLoad="imgdistortion()">  
<center>  
    
</center>  
</body>
```

举一反三

根据本实例，读者可以：

使页面中指定的字体快速变大、变小；

使图片改变成其他形状。

## 实例238 图片在页面浮动

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\08\238

实例说明

在浏览一些网站时，总会发现页面中浮动着许多的图片广告链接，这种浮动的广告已成为网站中不可缺少的工具，它们不但可以做为一种广告的宣传工具，而且还可以丰富网站的显示效果。本实例将使用JavaScript实现这一功能，如图8.29所示。



图8.29 图片在页面浮动

技术要点

本实例实现图片在页面浮动主要应用了 div 层和 Math 对象的 random( )方法。首先层会向页面的右下位置开始移动，当层的位置等于页面的边界时，程序会根据 random 获取反弹后的方向，使层的x和y坐标加上指定的值或减去指定的值（层向上x和y坐标加值，层向下x和y坐标减值），然后继续运行。

### 实现过程

(1) 编写用于实现图片在页面浮动的JavaScript代码。

```
<script language="JavaScript">
var isNS= ((navigator.appName=="Netscape")&&
(parseInt(navigator.appVersion)>=4));
var _all= '';
var _style= '';
var wwidth, wheight;
var ydir= '++';
var xdir= '++';
var id1, id2, id3;
var x=1;
var y=1;
var x1, y1;
if(!isNS)
{
    _all=' all.';
    _style='.style';
}
function getwindowsize()
{
    clearTimeout(id1);
```

```

clearTimeout(id2);
clearTimeout(id3);
    wwidth=document.body.clientWidth-55;
    wheight=document.body.clientHeight-50;
id3= setTimeout('randomdir()', 20000);
animate();
}
function randomdir()
{
    if (Math.floor(Math.random()*2))
    {
        (Math.floor(Math.random()*2)) ?xdir='--':xdir='++';
    }
    else
    {
        (Math.floor(Math.random()*2)) ?ydir='--': ydir='++';
    }
    id2= setTimeout('randomdir()', 20000);
}
function animate()
{
    eval('x'+xdir);
    eval('y'+ydir);
    if (isNS)
    {
        picture_div.moveTo((x+pageXOffset), (y+pageYOffset))
    }
}

```

```

else
{
    picture_div.pixelLeft=x+document.body.scrollLeft;
    picture_div.pixelTop= y+document.body.scrollTop;
}
if (isNS)
{
    if (picture_div.top<=5+pageYOffset) ydir= '++';
    if (picture_div.top>=height+pageYOffset)ydir= '--';
    if (picture_div.left>=wwidth+pageXOffset)xdir= '--';
    if (picture_div.left<=5+pageXOffset)xdir= '++';
}
else
{
    if (picture_div.pixelTop<=5+document.body.scrollTop)
ydir= '++';
    if
(picture_div.pixelTop>=height+document.body.scrollTop)
ydir= '--';
    if
(picture_div.pixelLeft>=wwidth+document.body.scrollLeft)
xdir= '--';
    if
(picture_div.pixelLeft<=5+document.body.scrollLeft) xdir=
'++';
}
    id1= setTimeout('animate()', 10);

```

```

}
</script>
(2) 添加页面设计代码，当页面加载时调用 getwindowsize() 函数，当页面改变大小时调用 getwindowsize() 函数，代码如下。
<body OnLoad="getwindowsize()"
OnResize="getwindowsize()">
<div id="picture_div" style="position:absolute;
visibility:visible; left:0px; top:0px; z-index:-1">
<a href = "http://www.mingrisoft.com">

</a>
</div>
<script language="javascript">
var
picture_div=eval(' document.'+_all+' picture_div'+_style);
</script>
</body>

```

举一反三

根据本实例，读者可以：

制作在页面中浮动的文字；

制作在页面的指定位置中上下浮动的图片。

## [实例239 随机变化的网页背景](#)

本实例可以美化界面

实例位置：光盘\mingrisoft\08\239

实例说明

在浏览某个博客空间或打开某个页面时，有时重复两次打开该页就会显示不同的页面背景，这是网站制作者所实现的一种网页随机背景，使用这种方式可以优化网站的显示效果，不过这只适用于个人空间或博客。本实例将通过JavaScript实现此功能。效果如图8.30、图8.31所示。



图8.30 随机变化的网页背景1



图8.31 随机变化的页面背景2

### 技术要点

本实例主要应用了Math 对象的random( )方法和floor( )方法，使用random( )方法可以产生一个 0 到 1 之间的伪随机数。floor( )方法用于返回小于或等于指定数的最大整数。本实例通过使用这两个方法来取得一个0到4之间的整数。例如一个伪随机数

“0.7399107343159774” 将其乘以 5 之后的结果为 3.699553671579887，然后再使用 floor( )方法取得最大整数，则最后的结果为3。那么在本实例中就会显示一个3.jpg的背景图片。

### 实现过程

- (1) 编写实现随机变化页面背景的JavaScript代码。

```
<script language="javascript">
function randompic()
{
    var i=Math.floor(Math.random()*5);//取整并*5
    var src= "";
    switch(i)
    {
        case0 :
            src= "0. jpg";
            break;
        case1:
            src= "1. jpg";
            break;
        case2:
            src= "2. jpg";
            break;
        case3:
            src= "3. jpg";
            break;
        case4:
            src= "4. jpg";
            break;
    }
    document. body. background=src;
    setTimeout("randompic()", 5000);
}
</script>
```

(2) 当页面加载时调用randompic()函数，代码如下。

```
<body onLoad="randompic();">
```

举一反三

根据本实例，读者可以：

按时间随机变化页面背景；

随机变化页面欢迎词。

## 8.5 选择头像

头像选择在网络程序中经常会用到，它不仅应用起来方便，而且可以使整个网页看起来更有活力。例如在用户留言簿页面，需要选择自己喜欢的头像来模拟个人的肖像，这种接近现实生活中的模拟既形象又生动，着时为网页增色不少。下面就通过两个典型实例来讲解头像选择在页面中的应用。

### 实例240 在列表中选择图片

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\08\240

实例说明

在网站论坛或留言簿的用户注册页面中，加入头像选择功能可以使网站增色不少。运行本实例，在“选择头像”下拉列表中选择某个头像时，将在该下拉列表框的上方显示该项所代表的头像（见图 8.32）。用户也可以将该下拉列表框选中，然后按上下方向键选择想要的头像，此处选择的头像在用户实言时将显示在页面中。

姓名:	<input type="text" value="fly_angel"/>
性别:	<input type="radio"/> 男 <input checked="" type="radio"/> 女
用户昵称:	<input type="text" value="飞行的天使"/>
QQ号码:	<input type="text" value="283313288"/>
E-Mail:	<input type="text" value="*****@163.com"/>
	
请选择显示头像: <input type="text" value="14"/> ▾	
<input type="button" value="提交"/> <input type="button" value="取消"/>	

图8.32 在列表中选择图片

### 技术要点

本实例主要通过下拉列表框的onChange事件调用JavaScript自定义函数实现，在该自定义函数中关键是应用下拉列表框的selectedIndex 属性（用于获取当前选择的选项的 ID 号）和 options[n].value（用于获取由选项ID号指定的选项的value值）来获取选择的头像所对应的文件名，并将其附给Image对象的 src属性，用于在页面中实时显示头像信息。下面将详细介绍onChange事件。

onChange事件的语法格式如下。

onChange="handlerText"

参数说明如下。

● handlerText: 该参数是JavaScript 代码或对一个JavaScript 函数的调用。

功能: 当一个Select、Text 或 Textarea 域失去焦点并更改值时触发 change 事件。

注意：使用onChange 在用户修改数据后进行校验。

实现过程

(1) 准备18 张图片，分别命名为“0.gif”、“1.gif”、…、“17.gif”。

(2) 在网页中添加下拉列表，下拉列表的name属性为“selectIC0”，选项的值为图片文件的名称，下拉列表框的onChange事件执行的操作是调用JavaScript编写的自定义函数showico()，代码如下。

```
<select name="selectIC0" size="1" id="selectIC0"
onChange="showico();">
  <option selected>1</option>
  <option>2</option>
  <option>3</option>
  <option>4</option>
  <option>5</option>
  <option>6</option>
  <option>7</option>
  <option>8</option>
  <option>9</option>
  <option>10</option>
  <option>11</option>
  <option>12</option>
  <option>13</option>
  <option>14</option>
  <option>15</option>
  <option>16</option>
  <option>17</option>
```

```
</select>  

```

(3) 应用JavaScript编写自定义函数showico(), 用于实现在用户改变下拉列表的值时, 显示对应的图片信息, 代码如下。

```
//通过下拉列表选择头像时应用该函数  
<script language="javascript">  
function showico()  
{  
    document.form1.img.src=  
"Images/"+document.form1.selectIC0.selectedIndex+".gif";  
}  
</script>
```

举一反三

根据本实例, 读者可以:

在QQ聊天室中, 实现通过下拉列表选择头像的功能;

在人事档案管理系统中, 通过下拉列表选择头像来选择员工的个人肖像。

## 实例241 在弹出的新窗口中选择图片

这是一个应用比较广泛的实例

实例位置: 光盘\mingrisoft\08\241

实例说明

在用户注册网页中加入头像选择功能可以使网站更具吸引力, 而在选择头像时通过弹出的网页对话框选择, 则更直观、方便。运行本实例, 单击图8.33中的“请选择显示头像”超级链接, 将弹出图8.34

所示的网页对话框，当用户单击某个头像时，则该网页对话框将会关闭，并且该超级链接上将会显示刚刚选择的头像，如果用户在网页对话框中没有选择任何头像，则将显示系统默认的头像。



图8.33 单击“请选择显示头像”链接弹出图像选择对话框



图8.34 在此对话框中选择一个头像

### 技术要点

本实例主要是通过window对象的showModalDialog()方法实现的。但是需要读者注意的是，在实现该实例时，需要考虑用户没有从弹出的头像列表中选择一个想要的头像，而是直接关闭了选择头像的对话框的情况。如果此处不做处理，则页面中的选择头像位置将显示一个无图片的框。为了解决该问题，需要使用JavaScript的Global对象的undefined属性，下面对该属性进行详细介绍。

### 语法：

undefined

功能：返回 undefined 的一个初始值。

说明：undefined属性是Global对象的一个成员，该属性在脚本引擎初始化后可用。如果已声明了一个变量但还没有初始化，那么该变量的值就是 undefined；如果还没有声明变量，那么就不能将其与 undefined进行比较，但是可以将该变量的类型与字符串“undefined”进行比较。

注意：当显式地测试变量或将变量设置为undefined时，undefined属性是很有用的。

实现过程

(1) 在网页的适当位置添加“请选择显示头像”超级链接，该超级链接触发时执行的操作是调用JavaScript编写的自定义函数 deal()。代码如下。

```
<a href="#" onClick="deal();">[请选择显示头像]</a>
```

(2) 应用JavaScript编写自定义函数deal()，该函数用于打开一个网页对话框，要求用户选择自己喜欢的头像，并将返回的值赋给“请选择显示头像”图片的 scr 属性，用于在页面中实时显示选择的头像，代码如下。

```
<script language="javascript">
functiondeal()
{
    var someValue;
    someValue=window.showModalDialog(' ImageList.htm','', 'di
alogWidth=520px;\
dialogHeight=400px;status=no;help=no;scrollbars=no;resi
zable=yes');
}
```

//当用户在弹出的网页对话框中没有选择头像时则使头像默认为0.gif

```
if (someValue==undefined) {
    someValue="0";
}
document.img.src="images/"+someValue+".gif";
}
</script>
```

(3) 制作显示头像列表页面 ImageList.htm, 在该页面中将显示出全部可供用户选择的头像, 并提供将选择的头像返回到用户注册页面和关闭该网页对话框的超级链接。代码如下。

```
<table border="1"
align="center" cellpadding="-2" cellspacing="-2"
bordercolor="#FFFFFF" bordercolorlight="#FFFFFF"
bordercolordark="#8492b9" class="noborder">
<script language="javascript">
function selectICO(ICO)
{
    window.returnValue=ICO;
    window.close();
}
</script>
<tr align="center">
    <td valign="top"><a href="#" onClick="selectICO('0')">
        
    </a></td>
    <td valign="top"><a href="#" onClick="selectICO('1')">
```

```

        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('2')">
        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('3')">
        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('4')">
        
</a></td>
</tr>
<tr align="center">
        <td valign="top"><div align="center" class="style1">头像
1</div></td>
        <td valign="top"><div align="center" class="style1">头像
2</div></td>
        <td valign="top"><div align="center" class="style1">
头像3</div></td>
        <td valign="top"><div align="center" class="style1">
头像4</div></td>
        <td valign="top"><div align="center" class="style1">
头像5</div></td>
</tr>
<tr align="center">
        <td valign="top"><a href="#"onClick="selectIC0('5')">

```

```

        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('6')">
        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('7')">
        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('8')">
        
</a></td>
        <td valign="top"><a href="#"onClick="selectIC0('9')">
        
</a></td>
</tr>
<tr align="center">
        <td valign="top"><div align="center" class="style1">
头像6</div></td>
        <td valign="top"><div align="center" class="style1">
头像7</div></td>
        <td valign="top"><div align="center" class="style1">
头像8</div></td>
        <td valign="top"><div align="center" class="style1">
头像9</div></td>
        <td valign="top"><div align="center" class="style1">
头像10</div></td>
</tr>

```

```
</table>
<hr>
<p>
<center>
<input name="Button" type="button" class="btn_grey"
value="关闭该窗口"
onClick="selectICO('0')">
</center>
```

举一反三

根据本实例，读者可以：

在人事档案管理系统中，通过从网页对话框中选择头像来选择员工的个人肖像；

在网络聊天室中，通过从网页对话框中选择头像来选择人物头像。

## 8.6 在页面中播放图片

在浏览网站时，有时会看到一些网页中有播放图片的效果。通过添加播放图片的效果使整个网站内的信息更加丰富。下面将通过几个具体实例详细介绍如何在页面中播放图片。

### 实例242 幻灯片式播放图片

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\08\242

实例说明

本实例将实现一个以幻灯片的形式播放图片的效果。运行程序，在页面中单击“下一张”按钮会显示下一张图片，单击“上一张”按钮会显示上一张图片。当用户在页面中停留3s后，图片会自动播放，如图8.35所示。



图8.35 幻灯片式播放图片

### 技术要点

本实例主要应用了CSS的revealTrans属性。revealTrans属性提供了更加多变的转换效果。revealTrans属性的详细介绍请参见实例171。

### 实现过程

(1) 设置图片的路径。

```
<script language="javascript">
var sPicArr = new Array();
sPicArr[0] = new Array("images/0.gif");
sPicArr[1] = new Array("images/1.gif");
sPicArr[2] = new Array("images/2.gif");
sPicArr[3] = new Array("images/3.gif");
sPicArr[4] = new Array("images/4.gif");
sPicArr[5] = new Array("images/5.gif");
sPicArr[6] = new Array("images/6.gif");
sPicArr[7] = new Array("images/7.gif");
sPicArr[8] = new Array("images/8.gif");
sPicArr[9] = new Array("images/9.gif");
sPicArr[10] = new Array("images/10.gif");
</script>
```

(2) 定义翻动图片的自定义函数，代码如下。

```
<SCRIPT language=javascript type=text/javascript>
<!--
var plPic=new Image();
var gIndex=0;
functionSlidePic(index) {
```

```

gIndex= index;
if ('Microsoft InternetExplorer'==navigator.appName)
{
document.images["slidePic"].filters.item(0).Apply();
}
document.images["slidePic"].src= sPicArr[index][0];
f ('Microsoft InternetExplorer'==navigator.appName)
{
document.images["slidePic"].filters.item(0).play();
}
}
-->
</SCRIPT>

```

(3) 单击“上一张”或者“下一张”时，调用的自定义函数，代码如下。

```

function NextPic() {
gIndex = ((gIndex+1)>=sPicArr.length?0:(gIndex+1));
SlidePic(gIndex);}
function PrevPic() {
gIndex = ((gIndex-1)<0?(sPicArr.length-1):(gIndex-1));
SlidePic(gIndex)}

```

(4) 自动播放的自定义函数，代码如下。

```

var sid;
function inislide() {
if(sid==null) sid = setInterval('NextPic()',
3000);//fixed by AmourGUO, 051017}

```

(5) 调用自定义函数，并设置图片的样式，代码如下。

```

<bodyonLoad="inislide();">
  <tablewidth="31%"height="178"
border="1"align="center"cellpadding="3"
cellspacing="2"bordercolor="#0066FF">
  <tr>
    <tdwidth="29%"><form
name="form1"method="post"action="">
      <inputname="firstbutton" type="button"
id="firstbutton"value="上一张"onClick="PrevPic();">
    </form></td>
    <tdwidth="42%">
      <div align="center">
    </div></td>
    <tdwidth="29%"><form
name="form2"method="post"action="">
      <inputname="nextbutton" type="button"
id="nextbutton"value="下一张"onClick="NextPic();">
    </form></td>
  </tr>
</table>
</body>

```

举一反三

根据本实例，读者可以：

实现网站中的广告图片幻灯片式播放；

实现网站中新闻栏目的新闻图片幻灯片式播放。

## 实例243 无间断的图片循环滚动效果

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\08\243

实例说明

在网页设计时，通常需要同时考虑页面的美观程度及其下载速度，这时可以对图片进行无间断的循环滚动显示，这样不但可以提高页面的下载速度，而且可以简化界面，同时也达到了宣传的效果。实例的运行结果如图8.36所示。



图8.36 无间断的图片循环滚动效果

技术要点

本实例主要应用`setTimeout()`方法实现图片循环滚动效果，下面将对该方法进行详细介绍。

`setTimeout()`方法的语法格式如下。

```
setTimeout (function, milliseconds, [arguments])
```

参数说明如下。

- **function:** 要调用的JavaScript 自定义函数的名称。
- **milliseconds:** 设置超时时间（以m/s 为单位）。

功能：超过超时时间后，调用函数。此值可以用clearTimeout() 函数清除。

注意：setTimeout() 方法在超过超时时间后只调用一次函数，而 setInterval() 方法是按一定时间重复调用指定的函数。

### 实现过程

(1) 通过 JavaScript 脚本创建 move() 函数，用于实现无间断的图片循环滚动效果。n 数值越大图片滚动的越快，代码如下。

```
<SCRIPT language="javascript">
var n=5
//数值越大图片滚动的越快
td2.innerHTML=td1.innerHTML
varMycheck;
functionmove()
{
    if(td2.offsetWidth-div1.scrollLeft<=0)
        div1.scrollLeft-=td1.offsetWidth
    else
        {div1.scrollLeft++}
    Mycheck=setTimeout(move,n)
}
div1.onmouseover=function() {clearTimeout(Mycheck)}
div1.onmouseout=function() {Mycheck=setTimeout(move,n)}
move();
</SCRIPT>
```

(2) 应用DIV层实现图片的显示与隐藏，代码如下。

```
<DIV id=div1
style="OVERFLOW:hidden;WIDTH:100%;COLOR:#ffffff">
    <TABLE cellSpacing=0 cellPadding=0 align=leftborder=0
cellspace="0">
        <TR>
            <TD id=td1 vAlign=top>
                <tablewidth="1710"height="116" border="0"cellpadding="0"
cellspacing="0">
                    <tr>
                        <tdwidth="171"background="1. jpg"></td>
                        <tdwidth="171"background="2. jpg"></td>
                        <tdwidth="171"background="3. jpg"></td>
                        <tdwidth="171"background="4. jpg"></td>
                        <tdwidth="171"background="5. jpg"></td>
                        <tdwidth="171"background="6. jpg"></td>
                        <tdwidth="171"background="7. jpg"></td>
                        <tdwidth="171"background="8. jpg"></td>
                        <tdwidth="171"background="9. jpg"></td>
                        <tdwidth="171"background="10. jpg"></td>
                    </tr>
                </table>
            </TD>
            <TD id=td2 vAlign=top>&nbsp;</TD></TR></TABLE>
</DIV>
```

举一反三

根据本实例，读者可以：

在开发美食网站时，对特色美食进行展示；

在开发宠物网站时，对最新宠物宝宝进行展示；  
在开发电子商务网站时，对特价商品进行展示。

## 8.7 图片的其他效果

在进行网络程序设计时，可以在网页中添加各种各样的图片效果，例如抖动的图片、鼠标移动放大图片等。下面通过本节实例对图片的其他效果的实现进行详细介绍。

### 实例244 导航地图

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\08\244

实例说明

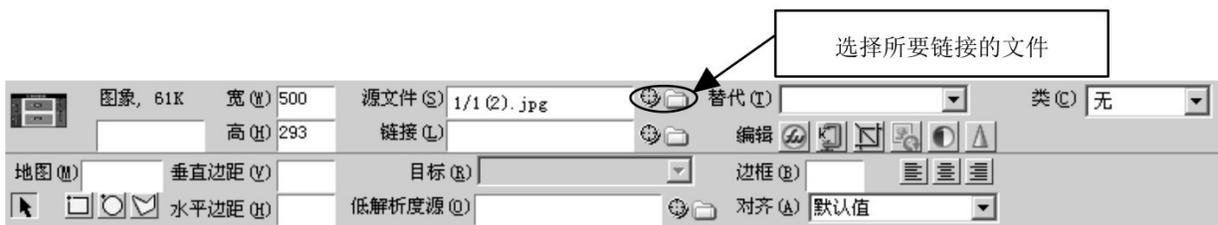
在开发网络程序时，通常需要对网站的功能流程进行详细说明。程序的开发者很重视如何将某个网站的功能清晰地描述出来。解决这一问题的最好办法就是制作导航地图，该地图不仅可以美化网站，同时也可以更加详细地说明该网站的具体功能及操作流程。实例运行结果如图8.37所示。



## 图8.37 导航地图

### 技术要点

本实例主要通过建立图像地图实现导航地图。图像地图通常被用在设置图像链接或图像导航方面。可以通过设置图像地图热区的链接，完成商场内各个商品专区的链接。设置图像地图的“替代”文本可以在图像中起到对商品专区导航的作用。建立图像热区也就是建立图像地图。图像的热区工具如图8.38所示。



## 图8.38 图像地图属性面板



为图像“属性”面板中的图像地图工具，共分为3种，分别是矩形、圆形和多边形。建立图像地图非常简单，只要选择任意一种图形后，在适当位置按住左键拖动鼠标即可建立相应热区。

通过单击图像热区“属性”面板中“链接”域后的浏览图像文件夹图标添加相应图标，在弹出的“选择文件”对话框中选择所要链接的文件，或者直接输入超级链接的路径及文件名。

### 实现过程

(1) 首先在Photoshop中制作需要建立图片热区的图片。

(2) 在Dreamweaver中选择“插入”→“图像”选项，插入一张图片，使用鼠标左键单击选中此图片，此时在“属性”控制面板中可以看到此图片的相关属性。

(3) 初次使用“属性”控制面板时，其默认为“简化”形式，单击右下方的向下箭头可打开其扩展属性模式。

(4) 此时可以通过“属性”控制面板设置图像热区。

### 举一反三

根据本实例，读者可以：

在开发宠物网站程序时，对宠物进行购买时使用导航地图；

在开发医药管理系统时，对药品进行出库/入库管理时使用导航地图；

在开发电子商城网站时，对成功进入商城后进行商品选购时使用导航地图。

## 8.8 播放音乐

随着网络开发技术的不断更新，人们已不再局限于使用简单的网页设计模式。因此程序员在开发程序时，应该为网页设置一些背景音乐，从而使网页的功能更加完善。下面通过具体的几个应用实例进行详细介绍。

### 实例245 为网页设置背景音乐

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\08\245

实例说明

程序员在进行网络程序开发时，经常喜欢在网页中添加一些特殊的效果。例如，为网页设置背景音乐，当浏览者登录网站时，设置的背景音乐将会响起。本实例便实现了这一功能。程序运行结果如图8.39所示。



图8.39 为网页设置背景音乐

### 技术要点

本实例主要应用<bgsound>标记实现为网页设置背景音乐的效果，通过该标记可以嵌入多种格式的音乐文件。

<bgsound>标记的语法格式如下。

```
<BGSOUND SRC=“file_name” LOOP=loop_value>
```

参数说明如下。

- SRC：背景音乐的路径。
- LOOP：播放的循环次数，取值为-1 或者Infinite 时表示无限次循环。

注意：使用标记<bgsound>嵌入背景音乐的网页，在网页最小化的时候，音乐停止。

### 实现过程

(1) 首先应用<bgsound>标记向指定的网页中添加背景音乐，代码如下。

```
<bgsound src="py.MP3" loop="-1">
```

(2) 根据实际情况的需要为指定网页添加表格或表单元素。代码如下。

```
<tablewidth="775"height="741"border="0"align="center"cellpadding="0"cellspacing="0">
```

```
<tr>
```

```
<tdwidth="775"height="433"background="ht_2.jpg">&nbsp; ;</td>
```

```
</tr>
```

```
</table>
```

### 举一反三

根据本实例，读者可以：

在开发游戏网站时，为网页设置背景音乐；

在开发在线点播程序时，为网页设置背景音乐。

## 实例246 随机播放背景音乐

这是一个实用的实例

实例位置：光盘\mingrisoft\08\246

### 实例说明

在编写网络程序时，可以为指定的网页添加背景音乐，设置的背景音乐也可以是随机的。当浏览者浏览网页时每次播放的背景音乐都

是不一样的，这样可以使浏览者有一种新鲜感而不会因为循环地听着同一首音乐而感到腻烦。本实例运行结果如图8.40所示。



图8.40 随机播放背景音乐

### 技术要点

实现本实例时，首先需要应用 Array对象将背景音乐的文件名保存到数组中，然后随机选择要播放的文件名，并通过<bgsound>标记播放背景音乐。下面对Array对象进行详细介绍。

Array 对象用于创建数组，数组是有序数据的集合。每一个数组中的每个元素都是一个独立的值，并且使用new关键字来定义数组，数组的索引值从0开始。

Array对象常用的属性如表8.3所示。

表8.3 Array对象的属性

属 性	描 述
index	对于一个由规则表达式匹配生成的数组，这个属性返回此匹配的索引位置
input	对于一个由规则表达式匹配生成的数组，这个属性返回原始字符串
length	返回数组中的元素个数
prototype	用于在定义数组时添加新的属性和方法，prototype 是数组对象的静态属性

Array对象常用的方法如表8.4所示。

表8.4 Array对象的方法

方 法	描 述
concat()	返回一个新数组，这个新数组是由两个或更多数组组合而成的
join()	返回字符串值，其中包含了连接到一起的数组的所有元素，元素由指定的分隔符分隔开来
toLocaleString	返回一个日期，该日期使用当前区域设置并已被转换为字符串
toString	返回对象的字符串表示
ubound	返回在 VBAArray 的指定维中所使用的最大索引值

### 实现过程

(1) 应用Array对象实现随机背景音乐的播放。代码如下。

```
<script language="javascript">
var arrayname=newArray();
for (var i=0;i<5;i++)
{
    arrayname[i]="music"+(i+1)+".mp3";
}
var x=(Math.floor(Math.random()*4)+1);
document.write(' <bgsound src='+' '+arrayname[x]+' '+
loop="infinite">');
</script>
```

(2) 根据需要在指定的页面中添加表格或表单元素，代码如下。

```
<table width="784" height="668" border="0" align="center"
cellpadding="0" cellspacing="0">
<tr>
```

```
<td width="784" height="668"
background="0.gif">&nbsp;</td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

在在线点播系统中随机播放背景音乐；

在进行在线MP3播放时，随机播放背景音乐。

## 实例247 MIDI音乐选择

本实例是一个实用的程序

实例位置：光盘\mingrisoft\08\247

实例说明

在进行网络程序开发时，有时需要为网站添加一些背景音乐或 MIDI 音乐等。同时也可以向网页中添加多个音乐，如何在网页中对添加的多个音乐进行选择播放？例如本实例主要对网页中的 MIDI 音乐进行选择播放。运行本例，通过下拉列表框选择所要播放的 MIDI 音乐，单击“播放 MIDI 音乐”按钮进行音乐的播放。程序运行结果如图 8.41 所示。

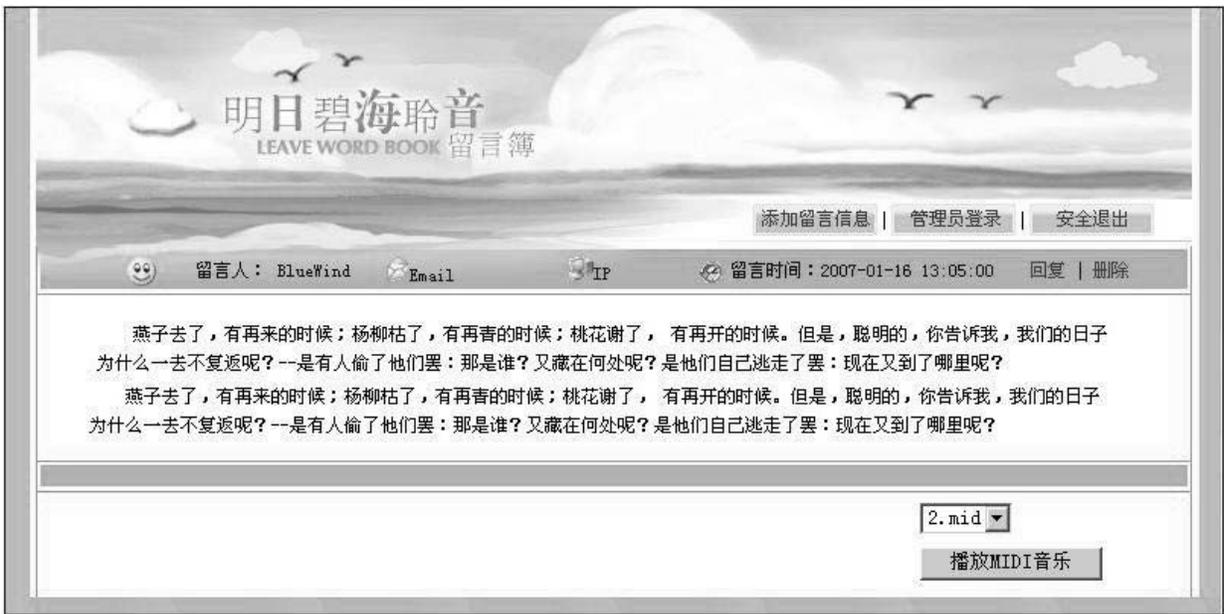


图8.41 MIDI音乐选择

### 技术要点

本实例主要应用Options数组实现MIDI音乐选择，下面将对该数组进行详细介绍。

Options数组是Select对象的一个属性，即选择框中的所有选项（<OPTION>）的一个列表。

Options数组的相关属性如表8.5所示。

表8.5 Options数组的相关属性的说明

属 性	说 明
defaultSelected	选项列表中的缺省选项
index	选项列表中某选项的索引位置
length	选项列表中的选项数（<OPTIONS>）
name	选项列表的名字（NAME 特性）
selected	表示选项列表中某选项<OPTION>是否被选中的一个布尔类型值
selectedIndex	选项列表中已选中的<OPTION>的索引（位置）
text	选项列表中<OPTION>标记后的文本
value	选项列表中的 VALUE 等于特性

### 实现过程

（1）通过JavaScript 脚本定义Mycheck（）函数，用于打开一个窗口。代码如下。

```

<SCRIPT LANGUAGE="JavaScript">
<!--
function Mycheck(sml)
{
    ggg = window.open (sml, "ming",
"toolbar=no, location=no, directories=no, status=no, scrollbars=n
o, resizable=no, copyhistory=no, width=200 , height=30")
}
//-->
</SCRIPT>

```

(2) 向指定页面中添加所需要的表单元素，代码如下。

```

<FORM NAME="form1">
<SELECT NAME="list">
    <option value="1.mid">1.mid</option>
    <option value="2.mid">2.mid</option>
    <option value="3.mid">3.mid</option>
</SELECT>
<P><INPUT TYPE=BUTTON VALUE="播放MIDI音乐"
onClick="Mycheck(form1. list.options [form1. list.
selectedIndex].value)">
</FORM>

```

举一反三

根据本实例，读者可以：

在开发在线论坛程序时，对留言信息进行查看；

在开发在线点播系统时，在线下载MIDI音乐。

## 8.9 插入Flash动画

Flash是一种矢量格式的动画文件，可以包含动画、声音、超文本链接等，而且文件的体积也很小，如果将其嵌入到网页中，可以为整个网页增色不少。本节将通过几个典型实例介绍Flash动画在网页中的应用。

### 实例248 插入Flash动画

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\08\248

实例说明

为了美化网站，使其有一个更好的视觉感受，可以将Flash技术引入到网页中，使网页更具有表现力。打开大多数网页都会不断弹出各种各样用Flash制作的精美动感广告和有趣的Flash游戏或MTV，增强了网页的宣传效果。由此可见，Flash技术在网页中起着举足轻重的作用。图8.42所示就是将Flash动画插入到网页中的效果。



图8.42 插入Flash动画

### 技术要点

本实例主要应用到了<object>标记和<embed>标记。<object>标记里的classid是告诉浏览器插件的类型，codebase可选；安装Flash插件的朋友浏览你的网页时，自动连接到Shockwave的下载网页，自动下载并安装相关插件。<object>和<embed>标记里quality = high 的记号作用是使浏览器以高质量浏览动画。

### 实现过程

定义<object>标签和<embed>标签，代码如下。

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,19,0" width="830" height="471">
```

```
<paramname="movie" value="flash/flash.swf"/>
```

```
<paramname="quality" value="high"/>
<paramname="wmode" value="transparent"/>
<embed
src="flash.swf" width="830" height="471" quality="high" plugins
page="http://www.macromedia.com/go/getflashplayer"
type="application/x-shockwave-flash" wmode="transparent">
</embed>
</object>
```

举一反三

根据本实例，读者可以：

在网页中插入Flash广告；

在网页中插入Flash导航条；

在网页中插入Flash影片。

## 实例249 插入背景透明的Flash动画

这是一个美化网站的实例

实例位置：光盘\mingrisoft\08\249

实例说明

如果说在网页中插入 Flash 是为了美化网站，使网页更具有表现力，那么在网页中插入背景透明的Flash，则会使网站的整体效果更加和谐统一，达到更完美的效果。运行本实例，该网站的顶部的企业文化的背景动画就是插入的透明背景的Flash，如图8.43所示。



图8.43 插入背景透明的Flash动画

### 技术要点

实现在网页中插入背景透明的 Flash 动画，主要通过设置 `<object></object>` 或 `<embed></embed>` 标记的 `wmode` 属性实现。

在 `<object></object>` 标记之间加入如下代码。

```
<param name="wmode" value="transparent">
```

设置 `<embed></embed>` 标记的 `wmode` 属性的代码。

```
wmode="transparent"
```

### 实现过程

(1) 将Flash动画文件添加到页面中的代码。

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,19,0" width="551" height="143">
```

```
<param name="movie" value="tm.swf"/>
```

```
<param name="quality" value="high"/>
<embed src="tm.swf" quality="high"
pluginspage="http://www.macromedia.com/go/getflashplayer"
type="application/x-shockwave-flash" width="551"
height="143"></embed>
</object>
```

(2) 设置Flash的背景透明效果，代码如下。

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-
444553540000" codebase="http://download.macromedia.com/pub/sho
ckwave/cabs/flash/swflash.cab#version=7,0,19,0" width="551"
height="143">
<param name="movie" value="tm.swf"/>
<param name="quality" value="high"/>
<param name="wmode" value="transparent">
<embed src="tm.swf" quality="high"
pluginspage="http://www.macromedia.com/go/getflashplayer"
type="application/x-shockwave-flash" width="551"
height="143" wmode="transparent"></embed>
</object>
```

### 举一反三

根据本实例，读者可以：

在网页中插入背景透明的Flash广告；

在网页中插入背景透明的Flash导航条；

在网页中插入背景透明的Flash影片。

## 8.10 播放视频文件

### 实例250 播放AVI文件

本实例是一个实用的程序

实例位置：光盘\mingrisoft\08\250

实例说明

在开发网络程序时，有时会在网站中播放AVI文件。本实例将详细介绍如何在网站中播放AVI文件。程序运行结果如图8.44所示。

技术要点

本实例主要使用<embed>标记实现AVI文件的播放。下面将对该标记进行详细介绍。



图8.44 播放AVI文件

在HTML文件中可以直接嵌入多媒体文件。多媒体是指利用计算机技术，把多种媒体综合在一起，使之建立起逻辑上的联系，并能对其进行各种处理的一种方法。多媒体主要包括文字、声音、图像和动画等各种形式。在HTML文件中使用标记`<embed>`嵌入多媒体文件。

语法：

```
<embed SRC= "file_url" WIDTH=value HEIGHT=value  
HIDDEN=hidden_value AUTOSTART=auto_value LOOP=loop_value>  
</embed>
```

标记`<embed>`的属性如表8.6所示。

表8.6 标记`<embed>`的属性

属 性	描 述
SRC	多媒体文件路径
WIDTH	播放多媒体文件区域的宽度
HEIGHT	播放多媒体文件区域的高度
HIDDEN	控制播放面板的显示和隐藏，取值为 True 代表隐藏面板，取值为 No 代表显示面板
AUTOSTART	控制多媒体内容是否自动播放，取值为 True 代表自动播放，取值为 False 代表不自动播放
LOOP	控制多媒体内容是否循环播放，取值为 True 代表无限次循环播放，取值为 No 代表仅播放一次

## 实现过程

(1) 在网页中指定的位置处插入<embed>标记，通过该标记实现AVI文件的播放。将<embed>标记中的LOOP参数值设为True时，表示指定的AVI文件进行循环播放，代码如下。

```
<embed src="a.avi" width="185"
height="145",noerror="true" loop="True" autostart="True">
</embed>
```

(2) 根据需要在指定的页面中添加表格或表单元素，代码如下。

```
<tablewidth="803"border="0" align="center"cellpadding="0"
cellspacing="0">
<tr>
<tdwidth="803"height="75"valign="top"background="imag
es/top.jpg">
<tablewidth="100%" border="0"
cellspacing="0"cellpadding="0">
<tr>
<tdheight="30">
<tablewidth="100%"cellspacing="0"cellpadding="0">
<tr align="center"valign="bottom">
<tdwidth="78%"height="29"align="right"class="st
yle2">设为首页</td>
```

```
        <tdwidth="9%" align="right" class="style2"><span
class="style2"> | </span>收藏本站</td>
        <tdwidth="11%"><span class="style2"> | 联系我们
</span></td>
        <tdwidth="2%">&nbsp;</td>
    </tr>
</table>
</td>
</tr>
<tr>
    <td>&nbsp;</td>
</tr>
</table>
</td>
</tr>
<tr>
    <td>
        <tablewidth="100%" border="0"
cellspacing="0" cellpadding="0">
            <tr>
                <tdwidth="803" height="166" background="images/
banner. jpg">&nbsp;</td>
            </tr>
        </table>
    </td>
</tr>
<tr>
```

```
<tdheight="30"><tablewidth="100%"  
border="0"cellspacing="0" cellpadding="0">  
  <tr>  
    <tdwidth="803"height="38"background="images/dh.jpg">  
      <div align="center"></div>  
    </td>  
  </tr>  
</table>  
</td>  
</tr>  
</table>
```

举一反三

根据本实例，读者可以：

在开发企业商务网站时，在网站中加载AVI文件；

在开发游戏网站时，在网站中加载AVI文件；

在开发在线点播系统时，在网站中加载AVI文件。

## [实例251 自制视频播放器](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\08\251

实例说明

AVI 是比较常用的多媒体视频文件格式，利用多媒体制作软件可以制作出许多丰富多彩的视频文件，但是制作好的AVI视频文件如何在网络中播放呢？通常有两种方法，一是应用超级链接标签来播放，应用该方法时，系统会利用默认的播放器播放该文件；另一种方法是应

用HTML 的标记符<embed>来播放。本实例使用的就是后一种方法，结合图层制作了一个多媒体播放器。运行程序，单击“单击观看多媒体演示”图标，将打开图 8.45 所示的窗口播放 AVI文件。



图8.45 自制视频播放器

### 技术要点

实现本实例主要利用HTML 的标记符<embed>直接调用Media Player 播放AVI 视频文件，但是其界面并不是很美观，尤其是底部的播放控制区不可以隐藏，这时可以通过在页面中加入一个浮动的图层，并使用该图层遮盖住系统播放器的灰色控制区来实现隐藏。

在IE浏览器中，层依靠<div></div>和<span></span>来实现，两者基本相同。通常<div>用于较大的层，<span>用于较小的层，并且<div>在实现的层后面加上一个回车键换行，而<span>不加。本实例在网页中加入浮动的图层利用<div>实现。

div的语法如下。

```
<div id=layername style="style definition">
```

```
layer content
</div>
```

其中，style definition 是一组用分号隔开的样式定义。常用的有以下几种。

- position: 其值可以是absolute 和relative, 所谓的absolute (绝对定位) 就是层的内容不必按照出现的先后次序在浏览器上依次排列, 而是可以以像素为单位定位到浏览器用户区域的任意位置; 而relative (相对定位) 则是层按照相邻的内容依次排行。

- left, top, width, height: 指层的左上角坐标以及它的密度和高度。

- z-index: 由于层可以被放置在页面的任何位置, 当它与其他内容重合时, z-index 值大的显示在上面, 所有层的z-index值为1。但是在NS中, 所有的表单元素 (文本框、列表框、按钮等), 只要是可见的, 就无法被遮住, 无论 z-index 的值是多少。因此在设计页面时, 要注意不要在动态出现或隐藏的元素 (例如弹出式菜单) 的位置上放置表单元素。

- visibility: 指明层是否可见, 通过在程序中动态改变这个值, 可以实现层的出现和消失, 比如下拉菜单就要依靠它来实现。它的3个候选值为inherit, 可见性与父元素的可见性相同; visible, 可见 (在NS中是show); hidden, 不可见 (在NS中是hide)。

### 实现过程

(1) 在网页中适当的位置添加单击观看多媒体演示的图片, 并为该图片添加超级链接, 单击超级链接所执行的操作是打开新窗口, 用于播放AVI视频文件, 关键代码如下。

单击观看多媒体演示:

```
<a
href="#"onClick="JavaScript:window.open(' AVIPlay.htm', '', 'wid
```

```
th=400,height=352')"></a>
```

(2) 利用HTML的标记符<embed></embed>播放AVI视频文件的关键代码如下。

```
<embed src="AVI/mrsoft.avi" width="326" height="315" noerror="true">
</embed>
```

(3) 为了隐藏其底部的播放控制区可以应用图层，利用图层将其底部的播放控制区遮盖住，以达到天衣无缝的效果。其关键代码如下。

```
<div id="Layer1" style="position:absolute; width:199px; height:94px; z-index:1; left: 0px; top: 290px;">

</div>
```

举一反三

根据本实例，读者可以：

制作一个可选择路径的视频文件的播放器；

制作一个可以循环播放的视频播放器。

## 第9章 页面特效

页面背景效果

特殊页面

广告页面

页面动画效果

其他

## 9.1 页面背景效果

在打开网页时，经常会看到以一些特殊的动画效果显示页面中的内容，或将背景图案固定在工作区域内，下面将通过几个典型实例进行详细讲解。

### 实例252 背景固定居中

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\09\252

实例说明

在制作网页时，为了使网页更加美观，在页面背景中会添加一个图片，但有时因图片过小，在页面中会重复显示图片，这样，反而破坏了页面的美观性。本实例将使页面中的背景固定居中，当页面内容过多时，无论怎样移动滚动条，背景图片始终固定居中。运行结果如图9.1所示。



图9.1 背景固定居中

### 技术要点

本实例主要通过 body 对象的 style 样式中的 backgroundImage、backgroundPosition、backgroundRepeat 和 backgroundAttachment 属性在页面中添加背景图片，并对图片进行居中显示。下面对这几个属性进行详细说明。

backgroundImage 属性的语法格式如下。

backgroundImage : none | url (url)

参数说明如下。

- none: 无背景图。
- url: 使用绝对或相对地址指定背景图像。

功能：设置或检索对象的背景图像。

backgroundPosition属性的语法格式如下。

backgroundPosition : length || length

backgroundPosition : position || position

参数说明如下。

- length: 百分数 | 由浮点数字和单位标识符组成的长度值。

- position: 为top | center | bottom | left | center |

right。

功能：设置或检索对象的背景图像位置。必须先指定backgroundImage属性。

backgroundRepeat属性的语法格式如下。

backgroundRepeat : repeat | no-repeat | repeat-x |

repeat-y

参数说明如下。

- repeat: 背景图像在纵向和横向上平铺。

- no-repeat: 背景图像不平铺。

- repeat-x : 背景图像在横向上平铺。

- repeat-y: 背景图像在纵向上平铺。

功能：设置或检索对象的背景图像是否及如何铺排。必须先指定对象的背景图像。

BackgroundAttachment属性的语法格式如下。

backgroundAttachment : scroll | fixed

参数说明如下。

- scroll: 背景图像随对象内容滚动。

- fixed: 背景图像固定。

功能：设置或检索背景图像是随对象内容滚动还是固定的。

实现过程

编辑用于实现背景居中的JavaScript代码。

```
<script language="javascript">
document.body.style.backgroundImage="URL(1.jpg)";
document.body.style.backgroundPosition="center";
document.body.style.backgroundRepeat="no-repeat";
document.body.style.backgroundAttachment="fixed";
</script>
```

举一反三

根据本实例，读者可以：

将页面的背景图片纵向平铺；

制用一个随机变换背景图的网页。

## 实例253 背景图片纵向重复显示

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\09\253

实例说明

在浏览网页时，经常会看到一些页面的背景无论怎样移动纵向滚动条，页面的背景图片都相同，但移动滚动条时，背景也跟着移动，这并不是因为页面的背景图片在设计的时候有多大，而是将一个上下连接处相同的图片进行纵向重复显示。运行结果如图9.2所示。

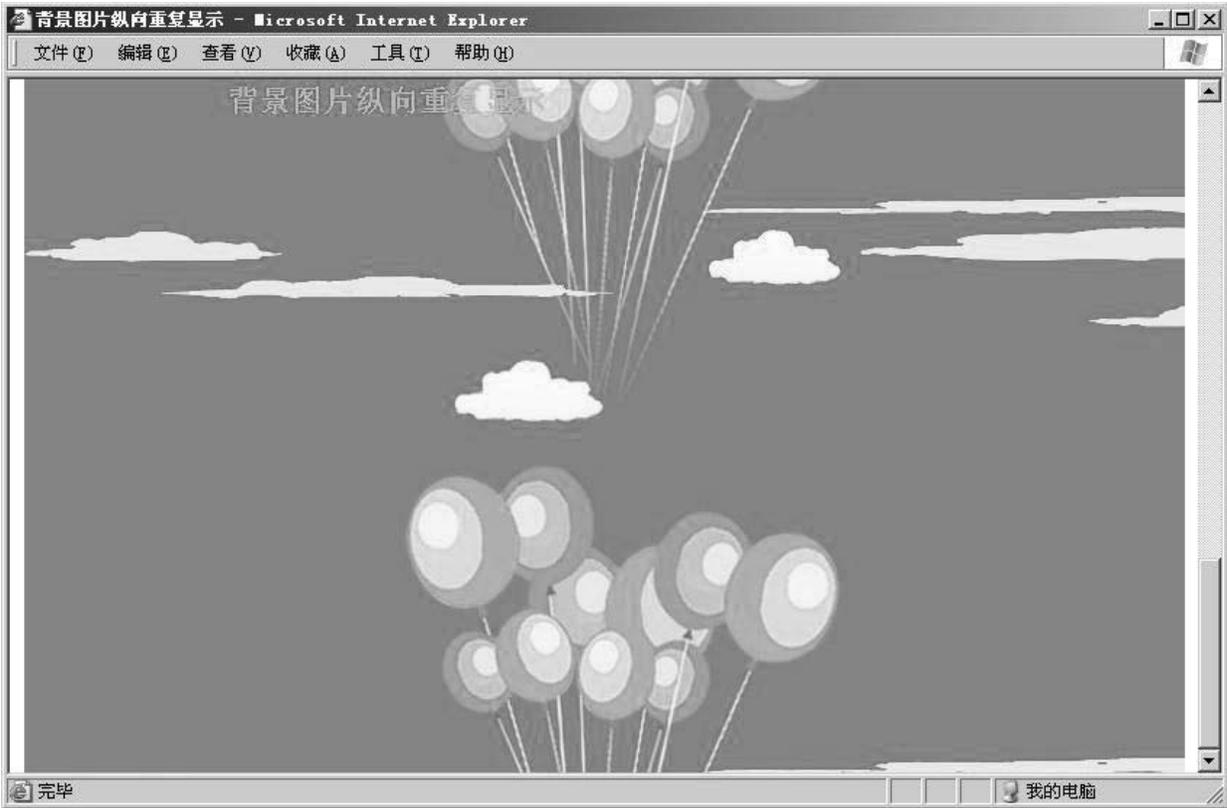


图9.2 背景图片纵向重复显示

### 技术要点

本实例主要通过 body 对象的 style 样式中的 backgroundImage、backgroundPosition 和backgroundRepeat属性将背景图片纵向重复显示。其中最关键的是将 backgroundRepeat属性的参数值设为repeat-y，用于背景图像在纵向平铺。

这几个属性的具体介绍请参考实例252。

### 实现过程

(1) 在页面中添加一个表格，使页面出现滚动条，便于观看效果。

```
<tablewidth="500"height="1538"border="0"
cellpadding="2"cellspacing="0">
```

```
<tr>
```

```
<th scope="col"><p>显示背景图片</p>
```

```
<p>&nbsp;</p>
... //省略部分代码，请读者参见光盘
<p>&nbsp;</p>
<p>&nbsp;</p>
</p>
<div align="center">背景图片纵向重复显示
</div></th>
</tr>
</table>
```

(2) 编辑用于实现背景图片纵向重复显示的JavaScript代码。

```
<script>
document.body.style.backgroundImage="URL(1.jpg)";
document.body.style.backgroundPosition="center";
document.body.style.backgroundRepeat="repeat-y";
</script>
```

举一反三

根据本实例，读者可以：

使背景图片在工作区中置顶；

将页面的背景图片横向平铺。

## 实例254 通过按钮变换背景颜色

本实例是一个美化界面的程序

实例位置：光盘\mingrisoft\09\254

实例说明

每个人对颜色的观赏性都不同，本实例将用一个按钮来动态地改变页面的背景颜色，使浏览者可以根据自己的喜好，以不同的背景颜

色来观看网页内容。运行结果如图 9.3、图9.4所示。

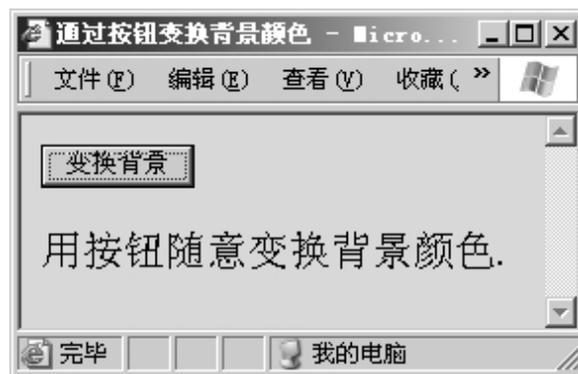


图9.3 背景颜色变换前的效果

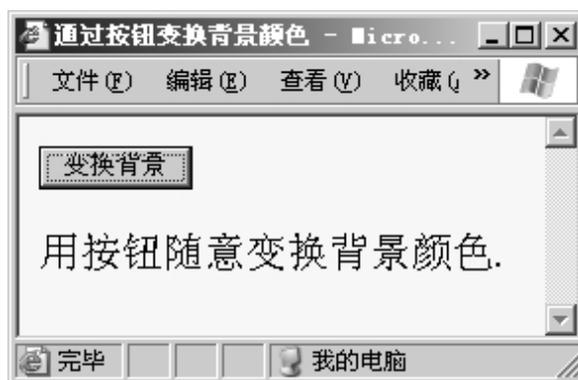


图9.4 背景颜色变换后的效果

### 技术要点

本实例主要是将多个颜色值保存在一个一维数组中，当单击按钮时，按顺序获取数组中的颜色值，并修改document对象的bgColor属性值，bgColor属性用于改变页面的背景颜色。

### 实现过程

(1) 在页面中添加一个按钮，代码如下。

```
<form name="form1" method="post" action="">  
  <input type="button" name="Submit" value="变换背景"  
onclick="turncolors()">  
</form>
```

(2) 编辑用于实现背景颜色变换的JavaScript代码。

```
<script language="javascript">
```

```
var Arraycolor=newArray("olive","teal","red","blue","maroon",
"navy","lime",
"fuschia","green","purple","gray","yellow","aqua","white",
"silver");
var n=0;
function turncolors() {
    if (n==(Arraycolor.length-1))n=0;
    n++;
    document.bgColor=Arraycolor[n];
}
</script>
```

举一反三

根据本实例，读者可以：

动态修改页面中的字体样式；

随机改变页面背景颜色。

## 实例255 背景自动变色

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\09\255

实例说明

在浏览页面时，经常会看到一些页面的背景颜色自动变色，本实例将实现这一效果。运行结果如图9.5、图9.6所示。

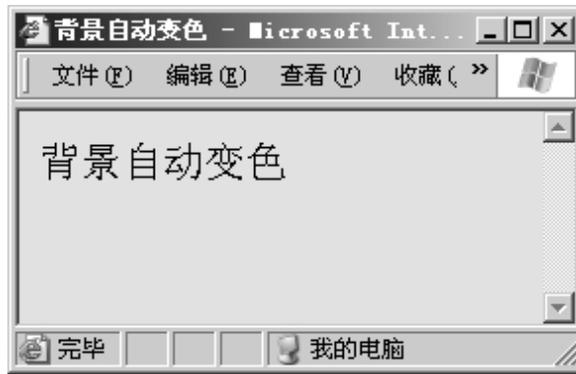


图9.5 背景自动变色前

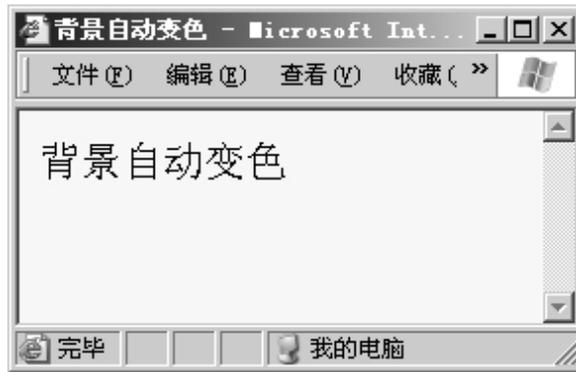


图9.6 背景自动变色后

### 技术要点

本实例主要是用 window 对象的 setTimeout() 方法按一定间隔时间对 document 对象的 bgColor 属性值进行修改。

### 实现过程

编辑用于实现背景自动变色的 JavaScript 代码。

```
<script language="javascript">
var Arraycolor=newArray("#00FF66", "#FFFF99", "#99CCFF", "#FF
CCFF", "#FFCC99",
"#00FFFF", "#FFFF00", "#FFCC00", "#FF00FF");
var n=0;
function turncolors() {
    n++;
    if (n==(Arraycolor.length-1))n=0;
```

```
document.bgColor=Arraycolor[n];
setTimeout("turncolors()",1000);
}
turncolors();
</script>
```

举一反三

根据本实例，读者可以：

制作一个具有渐变背景色的页面；

使页面的字体样式自动变化。

## 实例256 百叶窗

本实例可以美化界面

实例位置：光盘\mingrisoft\09\256

实例说明

在浏览网页时，经常会动态地打开页面，然后再显示页面中的内容。本实例将以百叶窗的形式动态地显示页面。运行结果如图9.7所示。

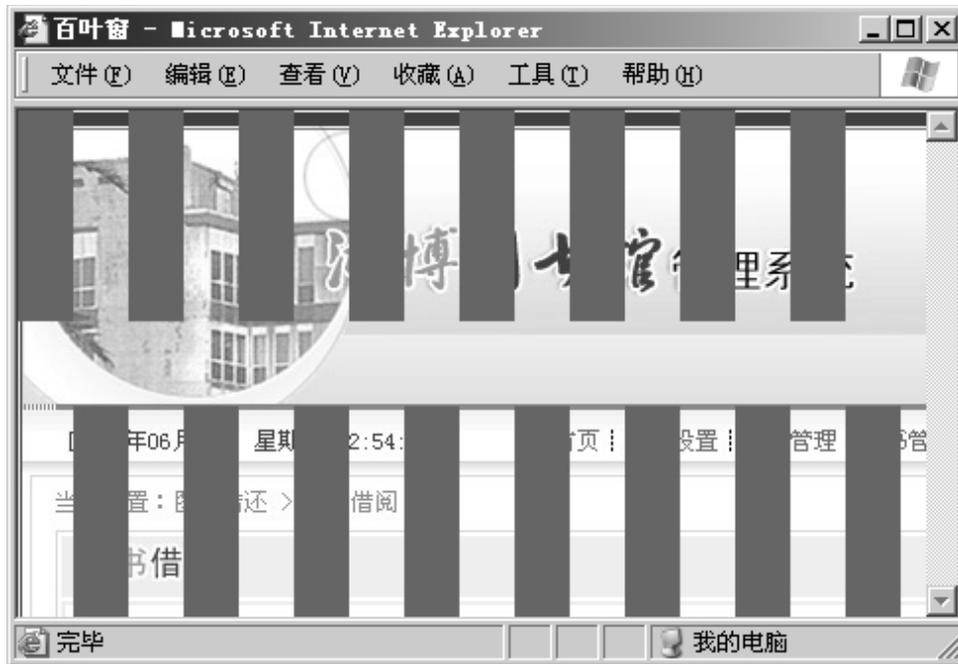


图9.7 百叶窗

### 技术要点

本实例主要是将页面工作区的宽度分成16等份，然后按每份的宽度在页面中添加16个等宽的层，层的高度为页面工作区的高度，将层按顺序平铺在页面中，并用<div>标记的 style 的clip 的rect()方法来实现奇数层的高由大到小进行显示，偶数层的高由小到大进行显示。下面对rect属性进行详细说明。

rect属性的语法格式如下。

clip : auto | rect( number number number number )

参数说明如下。

- auto: 对象无剪切。
- rect( number number number number ) : 依据上-右-下-左的顺序提供以对象左上角为(0,0)坐标计算的4个偏移数值，其中任一数值都可用auto替换，即此边不剪切。

功能：检索或设置对象的可视区域。区域外的部分是透明的。必须将 position 的值设为absolute，此属性方可使用。

## 实现过程

编辑用于实现百叶窗的JavaScript代码。

(1) 在页面中添加16个层，并设置层的大小及位置。

```
<script language="JavaScript">
var s="";
for (i=1;i<=16;++i) {
    s=s+'<div id="i'+i+' " style="position:absolute;
left:0px; top:0px;
    background-color:#0066FF;border:'+"0.1px"+' '+"solid"+'
'+"#0066FF"+' "></div>';
}
document.write(s);
var speed=30;
var temp=newArray();
varHeight=document.body.clientHeight,Top=0;
for (i=1;i<=16;i++) {
    temp[i]=eval("document.all.i"+i+".style");
    temp[i].width=document.body.clientWidth/16;
    temp[i].height=document.body.clientHeight;
    temp[i].left=(i-1)*parseInt(temp[i].width);
}

```

(2) 自定义函数kind(), 使奇数层的高由大到小进行显示, 偶数层的高由小到大进行显示。

```
function kind() {
    Height-=speed;
    for (i=1;i<=16;i=i+2) {
        temp[i].clip="rect(0 auto+"+Height+" 0)";
    }
}

```

```

}
Top+=speed
for (i=2;i<=16;i=i+2) {
    temp[i].clip="rect("+Top+" auto auto auto)";
}
if (Height<=0)
    clearInterval(tim);
}

```

(3) 在窗体载入时，重复调用自定义函数kind()。

```
tim=setInterval("kind()",100)
```

```
</script>
```

举一反三

根据本实例，读者可以：

使页面以飞出的形式显示页面信息；

使页面逐渐缩小，显示网页信息。

## 实例257 渐隐渐现的背景颜色

本实例是一个美化页面实例

实例位置：光盘\mingrisoft\09\257

实例说明

本实例在打开页面后，使页面的背景颜色由黑逐渐变成绿色，再由绿色逐渐变成黑色。运行结果如图9.8、图9.9所示。



图9.8 由黑色渐变成深绿色



图9.9 由深绿色渐变成绿色

### 技术要点

本实例主要是用 window对象的 setTimeout()方法以指定的间隔时间改变 document对象的bgColor属性值。在HTML的颜色属性中，黑色是#000000，纯红色是#FF0000，纯绿色是#00FF00，纯蓝色是#0000FF，而白色是#FFFFFF。

### 实现过程

编辑用于实现渐隐渐现的背景颜色的JavaScript代码。

(1) 自定义一个数组，用于记录渐变颜色的部分值。

```
<script language="JavaScript">
```

```
var Color= new
Array("ff","ee","dd","cc","bb","aa","99","88","77","66","55",
"44",
"33","22","11","00");
```

(2) 自定义函数fadeIn(), 使页面的背景颜色由黑色逐渐变成绿色。

```
function fadeIn(n) {
  if (n>=1) {
    document.bgColor="#"+"00"+Color[n]+"00";
    n-=1;
    setTimeout("fadeIn("+n+")", 100);
  }
  else {setTimeout(' fadeOut(1)', 100);}
}
```

(3) 自定义函数fadeOut (), 使页面的背景颜色由绿色逐渐变成黑色。

```
function fadeOut(n) {
  if (n<=(Color.length-1)) {
    document.bgColor="#"+"00"+Color[n]+"00";
    n+=1;
    setTimeout("fadeOut("+n+")", 100)
  }
  else {setTimeout("fadeIn("+ (Color.length-1) +")", 100);}
}
</script>
```

举一反三

根据本实例，读者可以：

制作一个渐变的页面背景；  
使背景不停地闪动。

## 实例258 页面缩小

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\258

实例说明

本实例是在打开页面时，使页面逐渐地缩小，并显示页面后面的网页信息，这样可以使浏览者在打开页面时，有一种动态的效果。运行结果如图9.10、图9.11所示。



图9.10 页面缩小

技术要点

本实例主要是用document对象中body的clientWidth和clientHeight属性值使层平铺在工作区域内，然后用层的style样式的clip的rect属性使层由大到小进行显示。

rect属性已在实例256中进行了详细的说明。



图9.11 页面基本消失

### 实现过程

(1) 在页面中添加一个层。代码如下。

```
<div id="div1" style="position:absolute;top:0;left:0;background-color:#6699FF;"></div>
```

(2) 编写用于实现页面缩小的JavaScript代码。

自定义函数doit(), 将层居中, 并由大到小进行局部显示。

```
<script language="JavaScript">
var speed=5;
var
reference=document.body.clientWidth/document.body.clientHeight;
t;
var rightclip, leftclip, topclip, bottomclip;
var temp=document.all.div1.style;
topclip=leftclip=0;
rightclip=temp.width=document.body.clientWidth;
```

```
bottomclip=temp.height=document.body.clientHeight;
functiondoit() {
    if (leftclip>document.body.clientWidth/2)
        clearInterval(tim);
    temp.clip="rect( "+topclip+" "+rightclip+"
"+bottomclip+" "+leftclip+" )";
    leftclip+=reference*speed;
    topclip+=speed;
    rightclip-=reference*speed;
    bottomclip-=speed;
}
```

在窗体载入时，用setInterval()方法重复调用自定义函数doit()。

```
tim=setInterval("doit()",100);
</script>
```

举一反三

根据本实例，读者可以：

实现页面的下沉效果；

实现页面向右侧移动的效果。

## [实例259 页面上下打开效果](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\259

实例说明

本实例打开页面时，页面将会横向从中间被分开，并以上下分开的方式来显示页面中的内容。运行结果如图9.12、图9.13所示。



图9.12 页面上下打开效果



图9.13 页面基本完全打开

### 技术要点

本实例是用document对象中body的clientHeight属性来获取当前工作区的高度，通过页面的1/2高度，将两个层以上下的形式平均放在页面中，再用层的style样式中clip的rect属性的top和bottom值来动态获取层的局部大小。

rect属性的介绍请参考实例256。

### 实现过程

(1) 在页面中添加两个层，代码如下。

```
<div id="div1" style="position:absolute; left:0px;
top:0px; background-color:#0066FF;
border:'0.1px' 'solid' '#0066FF' "></div>
<div id="div2" style="position:absolute; left:0px;
top:0px; background-color:#0066FF;
border:'0.1px' 'solid' '#0066FF' "></div>
```

(2) 编写用于实现页面上下打开的JavaScript代码。

设置两个层的大小及位置。

```
<script language="JavaScript">
var speed=20;
var temp=new Array();
var Height=document.body.clientHeight/2, Top=0;
for (i=1;i<=2;i++) {
    temp[i]=eval("document.all.div"+i+".style");
    temp[i].width=document.body.clientWidth;
    temp[i].height=document.body.clientHeight/2;
    temp[i].top=(i-1)*parseInt(temp[i].height);
}

```

自定义函数openit()，根据层的宽度由大到小局部显示两个层。

```
function openit() {
    window.scrollTo(0,0);
    Height-=speed;
    temp[1].clip="rect(0 auto+"+Height+" 0)";
    Top+=speed;
}

```

```
temp[2].clip="rect("+Top+" auto auto)";  
if (Height<=0)  
clearInterval(tim);  
}
```

在窗体载入时，按一定间隔时间调用自定义函数openit()。

```
var tim=setInterval("openit()",100);  
</script>
```

举一反三

根据本实例，读者可以：

实现页面阶梯效果；

实现页面回旋飞出效果。

## 实例260 页面左右打开效果

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\09\260

实例说明

本实例在打开页面时，页面将会纵向从中间分开，并以左右分开的方式来显示页面中的内容。运行结果如图9.14、图9.15所示。



图9.14 页面左右打开效果

### 技术要点

本实例通过document对象中body的clientWidth属性来获取当前工作区的宽度，通过页面的1/2宽度，将两个层以左右的形式平均放在页面中，再用层的style样式中clip的rect属性的right和left值来动态获取层的局部大小。



图9.15 页面基本完全打开

rect属性的介绍请参考实例256。

### 实现过程

(1) 在页面中添加两个层，代码如下。

```
<div id="div1" style="position:absolute; left:0px;
top:0px; background-color:#0066FF;
border:'0.1px' 'solid' '#0066FF' "></div>
<div id="div2" style="position:absolute; left:0px;
top:0px; background-color:#0066FF;
border:'0.1px' 'solid' '#0066FF' "></div>
```

(2) 编写用于实现页面左右打开的JavaScript代码。

设置两个层的大小及位置。

```
<script language="JavaScript">
var speed=20;
var temp=newArray();
var clipright=document.body.clientWidth/2,clipleft=0;
for (i=1;i<=2;i++){
    temp[i]=eval("document.all.div"+i+".style");
    temp[i].width=document.body.clientWidth/2;
    temp[i].height=document.body.clientHeight;
    temp[i].left=(i-1)*parseInt(temp[i].width);
}
}
```

自定义函数openit()，根据层的高度由大到小局部显示两个层。

```
function openit(){
    clipright-=speed;
    temp[1].clip="rect(0 "+clipright+" auto 0)";
    clipleft+=speed;
    temp[2].clip="rect(0 auto auto "+clipleft+)";
}
```

```
    if (clipright<=0)
        clearInterval(tim);
}
```

在窗体载入时，按一定间隔时间调用自定义函数openit()。

```
var tim=setInterval("openit()",100);
</script>
```

举一反三

根据本实例，读者可以：

实现页面盒状展开效果；

实现随机飞出线条效果。

## 实例261 页面溶解效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\261

实例说明

在浏览页面时，很多页面都以溶解的方式来显示页面中的信息。本实例是在打开页面时，页面会以块状形式进行显示，直到将页面中的信息全部显示出来。运行结果如图9.16、图9.17所示。



图9.16 页面溶解效果



图9.17 页面基本溶解完成

### 技术要点

本实例主要通过层的style 样式的width 和height 属性来指定层的大小，left 和top 属性来指定层的位置，再用Math对象的random() 方法来随机获取层，并用层的style样式中的visibility属性来设置层的可视度，它的可选值有visible（设定该元素显示）、hidden（设定

该元素不显示，但仍占据空间）和inherit（由父元素可视度决定），默认值为inherit。

### 实现过程

编写用于实现页面溶解的JavaScript代码。

（1）在页面中动态添加26个层，并将它们按指定位置平铺在页面中。

```
<script>
var s="";
for (var i=1;i<26;i++) {
    s=s+'<div id="div'+i+'" style="position:absolute;
left:0px; top:0px;background-color:
    #0066FF;border:'+"0.1px"+' '+"solid"+' '+"#0066FF"+' ">
</div>';
}
document.write(s);
var espeed=300;
var counter=1;
var temp=newArray();
var oddwidth=parseInt(document.body.clientWidth/5);
var oddheight=parseInt(document.body.offsetHeight/5);
for (i=1;i<=25;i++) {
    temp[i]=eval("document.all.div"+i+".style");
    temp[i].width=oddwidth;
    temp[i].height=oddheight;
}
for (i=1;i<=5;i++)
    temp[i].left=(i-1)*oddwidth;
```

```

for (i=6;i<=10;i++) {
    temp[i].left=(i-6)*oddwidth;
    temp[i].top=oddheight;
}
for (i=11;i<=15;i++) {
    temp[i].left=(i-11)*oddwidth;
    temp[i].top=2*oddheight;
}
for (i=16;i<=20;i++) {
    temp[i].left=(i-16)*oddwidth;
    temp[i].top=3*oddheight;
}
for (i=21;i<=25;i++) {
    temp[i].left=(i-21)*oddwidth;
    temp[i].top=4*oddheight;
}

```

(2) 自定义函数whereat(), 将页面中的层以随机的方式进行隐藏。

```

functionwhereat() {
    varoddfew=Math.round(Math.random()*25);
    if (oddfew==0) oddfew=1;
    if (temp[oddfew].visibility!="hidden")
        temp[oddfew].visibility="hidden";
    else{
        while (temp[oddfew].visibility=="hidden"){
            oddfew=Math.round(Math.random()*25);
            if (oddfew==0)

```

```
        oddfew=1;
    }
    temp[oddfew].visibility="hidden";
}
if (counter==25) {
    clearInterval(beginerase);
}
counter++;
espeed-=10;
}
```

(3) 在窗体载入时，重复调用自定义函数whereat()。

```
beginerase=setInterval("whereat()", espeed)
```

```
</script>
```

举一反三

根据本实例，读者可以：

实现页面逐渐消失的效果；

实现页面旋转飞出的效果。

## 9.2 特殊页面

在浏览网页时，经常会看到一些特殊的页面，例如下载页面、程序加载页面、颜色拾取器等。下面将通过几个典型实例对这些页面的实现进行详细讲解。

### 实例262 页首页尾切换

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\09\262

实例说明

在浏览网页时，为了能够快速浏览，有些网页带有“页首”和“页尾”按钮。当浏览者浏览到页尾时，可以通过“页首”按钮快速回到页面顶端。本实例将实现这一功能。运行结果如图9.18、图9.19所示。



图9.18 切换到页首



图9.19 切换到页尾

### 技术要点

本实例主要是在<img>标记的移动事件（onMouseMove）和移出事件（onMouseOut）中，用<img>标记的src属性来动态修改图片。用document对象中body的scrollTop和scrollHeight属性来控制页面的首尾页切换。

- scrollTop: 设置或获取位于对象最顶端和窗口中可见内容的最顶端之间的距离。

- scrollHeight: 获取对象的滚动高度。

### 实现过程

(1) 在页面中添加一个表格，在该表格的单元格 td2 和 td3 中各添加一个表格，并在表格中添加图片。代码如下。

```
<table width="822px" border="0" align="center"
cellpadding="0" cellspacing="0"
id="table1">
<tr>
<td width="800" rowspan="2" valign="top" id="td1">

```

```

</td>
<td height="512" valign="top" id="td2">
  <table id="T_img2" width="32" height="60px" border="0"
style="top:500 ">
  <tr>
    <td align="center" valign="bottom">
      <img id='img2' src='3.jpg' onMouseMove='move(4)'
onMouseOut='move(3)'
      onClick="DUform(1)">
    </td>
  </tr>
</table>
</tr>
<tr>
  <td width="32" valign="bottom" id="td3">
  <table id="T_img1" width="32" height="60px" border="0"
style="top:0 ">
  <tr>
    <td align="center" valign="top">
      <img id='img1' src='1.jpg' onMouseMove='move(2)'
onMouseOut='move(1)'
      onClick="DUform(2)">
    </td>
  </tr>
</table>
</tr>
</table>

```

(2) 编写用于实现首尾页切换的JavaScript代码。

自定义函数pp(), 使表格table1的大小与页面工作区相一致。

```
<script language="JavaScript">  
functionpp() {  
    document.all.td2.style.width=document.all.td3.style.wid  
th="2%";  
}
```

自定义函数move(), 实现当鼠标移到图片上时, 改变图片; 移出时, 恢复原始图片的功能。

```
functionmove(N) {  
    if ((N==1) || (N==2)) {  
        document.all.img1.src=String(N)+".jpg";  
    }  
    if ((N==3) || (N==4)) {  
        document.all.img2.src=String(N)+".jpg";  
    }  
}
```

自定义函数DUform(), 用于使页面首尾页进行切换。

```
functionDUform(N) {  
    if (N==1)  
{document.body.scrollTop=document.body.scrollHeight}  
    else {document.body.scrollTop=0}  
}  
</script>
```

(3) 在<body>标记中的窗体装入事件 (onLoad) 和窗口缩放事件 (onResize), 调用自定义函数pp(), 代码如下。

```
<body style="margin:0pt" onResize="pp()" onLoad="pp()">
```

举一反三

根据本实例，读者可以：

实现滚屏显示页面内容；

实现以翻页的形式显示页面内容。

## 实例263 调用下载页面

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\263

实例说明

在浏览网站时，经常会对一些文件进行下载，在下载的时候会弹出一个下载页面，在该页面中会显示下载的进度。本实例将制作一个下载页面，当下载完成后，会弹出一个下载完成对话框，单击“确定”按钮，将关闭下载页面，并在主页面的状态栏中显示下载成功。如果没有下载完成，直接关闭下载页面，则在主页面的状态栏中显示下载失败。下载页面的运行结果如图9.20所示。

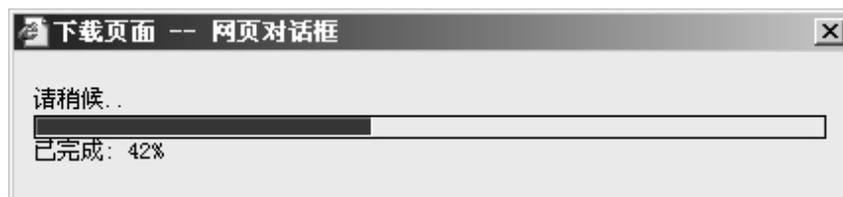


图9.20 调用下载页面

技术要点

本实例主要用window对象的showModalDialog()方法来调用子窗体，该方法可以在子窗体中获得返回值，可以通过返回值，在主窗体的window对象的状态(status)属性中显示下载是否成功。将子窗体的滚动条长度设为400，用load\_line\_i+=(400/500)计算出颜色条每次增长的总长度，并修改<img>标记的width属性，使颜色条不断地增

长，再用`Math.floor(load_line_i/4.00)`来计算颜色条增长的百分比，并在页面中进行显示。

### 实现过程

(1) 主页面的相关代码。

在页面中添加一段文字，并设置成类似超级链接的形式。代码如下。

```
<font id="a1" onclick="pp()"
onMouseMove="a1.style.color='#AA66CC'" onMouseOut
="a1.style.color='#000000'"><u>下载</u></font>
编写用于实现调用下载页面的JavaScript代码。
<script language="javascript">
functionpp() {
    varobj;
    varname=window.showModalDialog("bury.htm", "下载页
面", "dialogHeight:5px;
    dialogWidth:430px;dialogLeft:"+Math.floor((window.screen.
availWidth)*0.5)+"px;
    dialogTop:"+Math.floor((window.screen.availHeight)*0.8)+"
px:center:no;help:no;
    status:no;scroll:no"); //调用下载页面，并将返回的值保存
在name变量中
    if (name=="t") {window.status="下载成功"} //判断是否
下载成功
    else {window.status="下载失败"}
}
</script>
```

(2) 下载页面的相关代码。

在页面中以嵌套形式添加3个层，并在里面的层中添加一个<img>标记，用于显示颜色块。代码如下。

```
<body style="font-size:9pt
"bgcolor="#EDECE9"onload="load_n('已完成:')">
  <div id=load style="font-size:9pt;
cursor:default;position: absolute;
  display:block;width:402;height:20;top:expression(18);le
ft:expression(10);
  display:block; z-index:100;background:#EDECE9
"align="left">请稍候..
  <div style="border-width:1px;border-
color:#000000;border-style:ridge">
    <img id=line style="background:#3300CC"height=10>
  </div>
  <div id=load_txt><div style="font-size:9pt;
cursor:default;position: absolute;
  display:marker;width:402;height:20;top:expression(18)
;left:expression(10);
  display:block; z-index:100;background:#EDECE9 ">
</div>
</div>
</div>
```

编写用于实现下载页面的JavaScript代码。

自定义函数load\_n()，用于按一定的百分比将颜色块不断地变宽。

```
<script language="JavaScript">
var load_line_i=1;
```

```

var load_line_n=500;
var tim;
function load_n(txt) {
    if ((Math.floor(load_line_i/4.00))==100) {           //判断是
否下载完成
        finish();
        window.returnValue=' t';                       //向主窗体传
值
        clearTimeout(tim);
    }
    else{
        load_line_i+=400/load_line_n;                   //计算颜色块
的长度
        //将颜色块的长度计算成百分比，并显示在下载页面中
        load_txt.innerText=txt+"
"+Math.floor(load_line_i/4.00)+"%";
        document.all("line").width=load_line_i;       //改变颜色
块的长度
        tim=setTimeout("load_n(' 已完成:')", 10);    //按一定时
间调用本身
    }
}

```

自定义函数finish()，弹出一个询问式的对话框，当单击“确定”按钮时，关闭下载页面。

```

function finish() {
    if(confirm("已经下载完成, 关闭对话框吗?")) {
        close();
    }
}

```

```
}  
}  
</script>
```

举一反三

根据本实例，读者可以：

制作带有背景颜色的进度条；

根据文件大小，显示进度条进度。

## 实例264 程序加载页面

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\09\264

实例说明

在进入一些游戏网站的时候，通常会先进入一个程序加载页面，等待几秒钟后才能进入程序。本实例将制作一个程序加载的页面，运行结果如图9.21所示。



图9.21 程序加载页面

#### 技术要点

本实例主要是用<div>标记的innerHTML 属性将文本框按照指定的顺序添加到pimg层中，并通过修改层的style样式的pixelLeft属性值，使层在页面中不断移动。

#### 实现过程

(1) 在页面中添加一个层，并将层的align属性设置为center，使其在页面上居中。

```
<div id="div1"
style="position:absolute;width:322;height:14;border:1 #707888
solid;
overflow:hidden">
  <div style="position:absolute;top:-1;left:0" id="pimg">
</div>
</div>
```

```
<div id="div2"
style="position:absolute;top:30;left:120;font-size:9pt;
color:#f4f4f4">
正在加载中.....
</div>
```

(2) 编写用于实现文字伸展的JavaScript代码。

将所在的层在页面中居中，并定义一个数组，用于保存颜色值。

```
<script language="JavaScript">
s=new
Array("#050626", "#0a0b44", "#0f1165", "#1a1d95", "#1c1fa7", "#1c2
0c8", "#060cff");
div1.style.posTop=Math.floor((document.body.clientHeight-
14)/2);
div1.style.posLeft=Math.floor((document.body.clientWidth-
322)/2);
div2.style.posTop=parseInt(div1.style.posTop)+20;
div2.style.posLeft=parseInt(div1.style.posLeft)+120;
自定义函数Larrange(), 用于在pimg层中添加9个文本框，根据数
组元素进行正向排序，并设置不同的颜色。
function Larrange() {
pimg.innerHTML="";
for(i=0;i<9;i++) {
pimg.innerHTML+="<input
style=\"width:15;height:10;border:0;background:
+s[i]+\";margin:1\">";
}
}
```

自定义函数Rarrange(), 用于在pimg层中添加9个文本框, 根据数组元素进行反向排序, 并设置不同的颜色。

```
function Rarrange() {
    pimg.innerHTML="";
    for(i=9;i>-1;i--) {
        pimg.innerHTML+="<input
style=\"width:15;height:10;border:0;background:
+s[i]+\";margin:1\">";
    }
}
```

自定义函数move(), 将文本框按照指定的顺序在pimg层中循环移动。

```
var is=0;size=0;
functionmove() {
    if(pimg.style.pixelLeft<350&&is==0) {
        if(size==0) {Larrange();size=1;}
        pimg.style.pixelLeft+=3;
        setTimeout("move()",1);
        return;
    }
    is=1;
    if(pimg.style.pixelLeft>-200&&is==1) {
        if(size==1) {Rarrange();size=0;}
        pimg.style.pixelLeft-=3;
        setTimeout("move()",1);
        return;
    }
}
```

```
is=0;
move();
}
自定义函数flashes(),使页面中的文字不停地闪动。
```

```
function flashes() {
    if(div2.style.color=="#ffffff") {
        div2.style.color="#707888";
        setTimeout('flashes()',500);
    }
    else {
        div2.style.color="#ffffff";
        setTimeout('flashes()',500);
    }
}
```

在窗体载入时,调用自定义函数Larrange()、flashes()和move()。

```
Larrange();
flashes();
move();
</script>
```

举一反三

根据本实例,读者可以:

在进入个人网站前,应用加载页面;

根据程序大小,控制页面的显示时间。

## 实例265 颜色拾取器

这是一个实用的实例

实例位置：光盘\mingrisoft\09\265

实例说明

在浏览网站时，经常会进入一些讨论性的网站，在这些网站中，可以对一些问题进行讨论，在发送文字的时候，可以利用颜色拾取器对字体的颜色进行设置。本实例将制作一个16\*16\*16色的颜色拾取器。运行结果如图9.22所示。在下拉列表中选择红、绿、蓝、灰4种颜色，在选择其中一种颜色后，在表的左面会显示相应的颜色，用鼠标单击表格左面的颜色块，可以在表格右面看到更多的颜色，当单击表右边的颜色块时，会弹出标有相应颜色值的对话框。

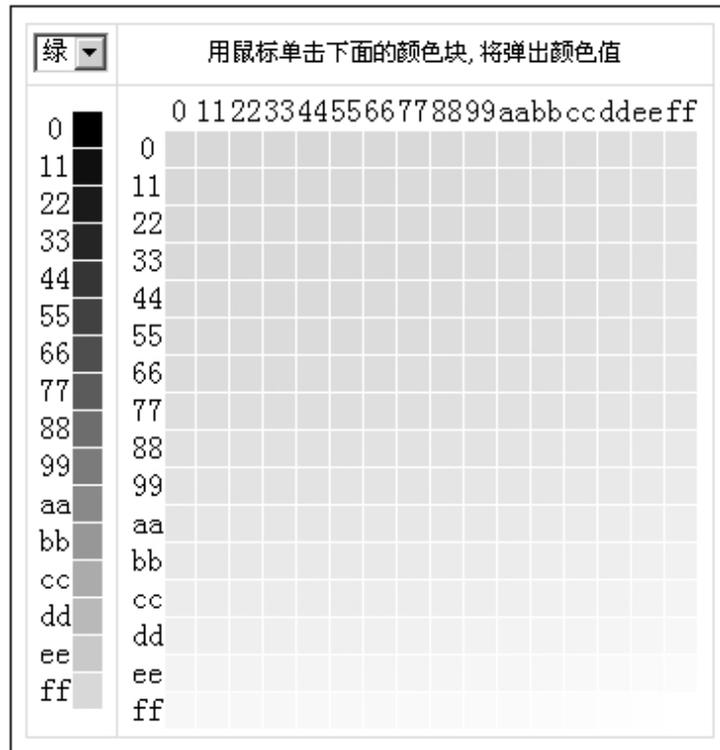


图9.22 颜色拾取器

技术要点

本实例主要应用 document 对象的 write() 方法动态向表格中添加单元格，用 RGB() 函数修改各单元格 bgcolor 属性中的值，使单元格

以不同的颜色进行显示。Number对象的toString(16)方法是将十进制转换成十六进制，该方法的详细介绍请参见实例102。

### 实现过程

(1) 在页面中添加表格，并在表格中以单元格背景来显示颜色块的内容，代码如下。

```
<tableborder="0"
cellspacing="1"cellpadding="3"width="350"align="center"bgcolo
r
="#dddddd">
<trbgcolor="#FFFFFF">
<tdwidth="10%"align="center">
<selectname="select1"onchange="selectmenu(this.value)">
<option value="1" selected>红</option>
<option value="2">绿</option>
<option value="3">蓝</option>
<option value="4">灰</option>
</select>
</td>
<tdwidth="90%"align="center">
<tablewidth="100%"border="0" cellspacing="0"
cellpadding="0">
<tr>
<td align="center" style="font-size:12px">用鼠标单击下面
的颜色块,将弹出颜色值</td>
</tr>
</table>
</td>
```

```
</tr>
<tr bgcolor="#FFFFFF">
<td width="10%" align="center">
  <table id="table1" border="0" cellspacing="1"
cellpadding="0">
```

在表格（table1）中添加JavaScript脚本，用于在表格中添加单元格，并设置单元格的背景颜色。

```
<script language="JavaScript">
for(i=0;i<=15;++i) {
  document.write(' <tr><td align="center"
style="font:menu">' + ishex(i*17) + ' </td>
<td id="Ltd'+i+' " bgcolor="rgb(' +
(i*17) + ', 0, 0)" width="15" height="15" onclick=
"changeright(this.num)"></td></tr>' )
  document.all['Ltd'+i].num=i
}
function ishex(which) {
  return which.toString(16);
}
</script>
```

HTML标记用于在单元格中创建表格。

```
</table></td>
<td align="center" width="90%">
<table id="table2" border="0" cellspacing="1"
cellpadding="0">
```

在表格中添加JavaScript脚本，用于向表格中添加指定的单元格，并设置单元格的背景颜色。

```

<script language="JavaScript">
document.write(' <tr><td></td>' )
for(i=0;i<=15;++i) {
    document.write(' <td align="center"
style="font:menu">' +ishex(i*17)+' </td>' );
}
document.write(' </tr>' )
for(i=0;i<=15;++i) {
    document.write(' <tr>' )
    document.write(' <td align="center"
style="font:menu">' +ishex(i*17)+' </td>' )
    for(j=0;j<=15;++j) {
        document.write(' <td id="Rtd'+i+' and'+j+' "
style="font:menu"bgcolor="rgb
(0,' +(i*17)+' ,'+
(j*17)+' )"width="15"height="15"onclick="clickright(this)"
></td>' );
    }
    document.write(' </tr>' )
}
</script>
HTML标记。
</table>
</td>
</tr>
</table>

```

(2) 编写用于实现更换颜色拾取器的颜色值，并进行显示的JavaScript代码。

自定义函数selectmenu()，用于判断在下拉列表（Menu组件）中，选择了红、绿、蓝、灰哪种颜色。

```
<script language="JavaScript">
function selectmenu(which) {
    switch(which) {
        case '1' :leftR();break;
        case '2' :leftG();break;
        case '3' :leftB();break;
        case '4' :leftA();break;
    }
}
```

自定义函数leftR()，当在下拉列表中选择红色时，改变表格左面的颜色值。

```
function leftR() {
    for(i=0;i<=15;++i) {
        document.all['Ltd'+i].bgColor=' rgb('+(i*17)+' ,0,0)';
    }
    rightR(0)
}
```

自定义函数leftG()，当在下拉列表中选择绿色时，改变表格左面的颜色值。

```
function leftG() {
    for(i=0;i<=15;++i) {
        document.all['Ltd'+i].bgColor=' rgb(0,'+(i*17)+' ,0)';
    }
}
```

```
rightG(0)
}
```

自定义函数leftB(), 当在下拉列表中选择蓝色时, 改变表格左面的颜色值。

```
function leftB() {
    for(i=0;i<=15;++i) {
        document.all['Ltd'+i].bgColor='rgb(0,0,'+(i*17)+' )';
    }
    rightB(0)
}
```

自定义函数leftA(), 当在下拉列表中选择灰色时, 改变表格左面的颜色值。

```
function leftA() {
    for(i=0;i<=15;++i) {
        document.all['Ltd'+i].bgColor='rgb('+(i*17)+','+(i*17)+' ,'+(i*17)+' ,'+(i*17)+' )';
    }
    rightA()
}
```

自定义函数rightR(), 当在左面选颜色块(红色)时, 相应地改变表格右面的颜色值。

```
function rightR(which) {
    for(i=0;i<=15;++i) {
        for(j=0;j<=15;++j) {
            document.all['Rtd'+i+' and'+j].bgColor='rgb('+(which*17)
            +' ,'+(i*17)+' ,'+(j*17)+' )';
        }
    }
}
```

```
    }  
  }  
}
```

自定义函数rightG ()，当在左面选颜色块（绿色）时，相应地改变表格右面的颜色值。

```
function rightG(which) {  
  for(i=0;i<=15;++i) {  
    for(j=0;j<=15;++j) {  
      document.all['Rtd'+i+' and'+j].bgColor=' rgb(' +  
      (i*17)+' ,'+  
      (which*17)+' ,'+(j*17)+' )' ;  
    }  
  }  
}
```

自定义函数rightB ()，当在左面选颜色块（蓝色）时，相应地改变表格右面的颜色值。

```
function rightB(which) {  
  for(i=0;i<=15;++i) {  
    for(j=0;j<=15;++j) {  
      document.all['Rtd'+i+' and'+j].bgColor=' rgb(' +(i*17)  
      +' ,'+(j*17)+' ,'+(which*17)+' )' ;  
    }  
  }  
}
```

自定义函数rightA ()，当在左面选颜色块（灰色）时，相应地改变表格右面的颜色值。

```
function rightA() {
```

```

for(i=0;i<=15;++i) {
    for(j=0;j<=15;++j) {
        document.all['Rtd'+i+' and'+j].bgColor=' rgb(' +
        (i*16+j)
        +','+(i*16+j)+' ,'+(i*16+j)+' )' ;
    }
}
}

```

自定义函数clickright(), 用于获取表格右面颜色块的颜色值。

```

function clickright(which) {
    alert(which.bgColor)
}

```

自定义函数changeright(), 通过对左面颜色块的选择, 相应地改变右面颜色块的颜色。

```

function changeright(which) {
    switch(select1.value) {
        case '1' :rightR(which);break;
        case '2' :rightG(which);break;
        case '3' :rightB(which);break;
    }
}
</script>

```

(3) 在<body>标记中设置字体的样式。代码如下。

```
<body style="font:menu">
```

举一反三

根据本实例, 读者可以:

制作一个可以传值的颜色拾取器;

制作一个可以保存颜色的拾取器。

## 9.3 广告页面

在打开网站时，经常会显示各种各样的广告图片，这样可以在网站中对其他的一些信息进行宣传。下面将通过几个典型实例对广告的显示方式进行详细讲解。

### 实例266 图片总置于顶端

这是一个提高信息传播的实例

实例位置：光盘\mingrisoft\09\266

实例说明

在浏览一些网站时，经常会看到一些广告图片总是置于工作区的右上角，本实例将实现这一效果。运行结果如图9.23所示。



图9.23 图片总置于顶端

### 技术要点

本实例主要通过document对象中body的scrollTop和scrollLeft属性来获取纵向滚动条和横向滚动条所卷去的部分值，实现过程如下。

(1) 用scrollTop属性值来修改层的style样式中的top属性，使层总置于顶端。

(2) 用scrollLeft属性值加上工作区宽度（用document对象中body的clientWidth属性来获取工作区的宽度），再减去图片的宽度，用所得到的值来修改层的 style样式中的left属性，使图片总置于工作区的右面。

(3) 用setInterval()方法以1ms为间隔时间，重复执行步骤(1)和(2)。

### 实现过程

(1) 在页面中添加一个层，并在层中放置图片。代码如下。

```
<div id=Tdiv style="HEIGHT: 45px; LEFT: 0px; POSITION: absolute; TOP: 0px; WIDTH: 45px; Z-INDEX: 25">  
<input name="image1" type="image" id="image1" src="mrsoft.jpg" width="52" height="249" border="0">  
</div>
```

(2) 编写用于实现图片总置于工作区右上角的JavaScript代码。自定义函数permute()，用于设置层的位置。

```
var ImgW=parseInt(image1.width);  
function permute(tfloor, Top, left) {  
    var RealTop=parseInt(document.body.scrollTop);  
    buyTop=Top+RealTop;
```

```
document.all[tfloor].style.top=buyTop;
var buyLeft=parseInt(document.body.scrollLeft)+parseInt
(document.body.clientWidth)-ImgW;
document.all[tfloor].style.left=buyLeft-left;
}
```

在窗体载入时，以1ms的间隔时间重复调用自定义函数permute()。

```
setInterval('permute("Tdiv",2,2)',1);
</script>
```

举一反三

根据本实例，读者可以：

使广告置于页面的左上角；

制作一个在页面中浮动的广告。

## 实例267 随机显示广告

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\267

实例说明

在浏览网站时，会看到一些广告图片以多种多样的形式进行显示，这样会更加吸引浏览者。本实例将用5种样式使图片以不同的方式进行显示。运行结果如图9.24所示。



图9.24 随机显示广告

### 技术要点

本实例主要应用了CSS的revealTrans属性。revealTrans属性提供了更加多变的转换效果。revealTrans属性的详细介绍请参见实例171。

### 实现过程

(1) 在页面中添加一个层，并对层的 filter 滤镜中的 revealTrans 属性进行相应的设置，同时在层中添加一个图片组件。代码如下。

```
<div id="div1" style="position: absolute; text-align:center;visibility:hidden; filter:  
    revealTrans (Transition=8,Duration=1)  
    revealTrans (Transition=9,Duration=3)  
    revealTrans (Transition=10,Duration=1)  
    revealTrans (Transition=11,Duration=2)
```

```
        revealTrans (Transition=12, Duration=3)
revealTrans (Transition=17, Duration=2)
        revealTrans (Transition=18, Duration=3)
revealTrans (Transition=19, Duration=1)
        revealTrans (Transition=20, Duration=2)
revealTrans (Transition=21, Duration=3) ">
<img id="img1"></div>
```

(2) 编写用于实现随机显示广告的JavaScript代码。

自定义一些变量，并设置图片的初始位置。

```
<script language="javascript">
var n=0;
var n1=0;
var is=true;
document.all.div1.style.top=document.body.clientHeight-
135;
document.all.div1.style.left=document.body.clientWidth-
175;
```

自定义函数showdiv()，用于按指定的样式来显示和隐藏图片。

```
function showdiv() {
    n=Math.floor(Math.random()*10);
    document.all.div1.filters[n].apply();
    if (n1==4) {n1=1}
    else {n1=n1+1}
    document.all.img1.src=n1+".jpg";
    if (is==true) {
        document.all.div1.style.visibility="visible";
        is=false;
    }
```

```

}else{
    document.all.div1.style.visibility="hidden";
    is=true;
}
document.all.div1.filters[n].play();
setTimeout("showdiv()",6000);
}

```

自定义函数place(),使图片总置于工作区的左下角。

```

functionplace(){
    window.status=String(n);
    document.all.div1.style.top=parseInt(document.body.scrollTop)+parseInt
(document.body.clientHeight)-135;
    document.all.div1.style.left=parseInt(document.body.scrollLeft)+parseInt
(document.body.clientWidth)-175;
    setTimeout("place()",50)
}

```

在载入窗体时调用自定义函数showdiv()和place()。

```
showdiv();
```

```
place();
```

```
</script>
```

举一反三

根据本实例,读者可以:

制作一个不同效果的公告栏;

使表格以不同的效果进行显示。

## 实例268 广告随滚动条漂移

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\09\268

实例说明

本实例将实现在页面中移动滚动条时，图片跟随滚动条的移动而漂移。这样可以使广告图片有一种动态的移动效果。运行结果如图9.25所示。



图9.25 广告随滚动条漂移

技术要点

本实例主要应用到了body的scrollLeft属性、scrollTop属性和offsetLeft属性、offsetTop属性。具体用法如下。

- scrollLeft：设置或获取对象左边界和窗口中目前可见内容的最左端之间的距离。

● scrollTop: 设置或获取对象最顶端和窗口中目前可见内容的最顶端之间的距离。

● offsetLeft: 获取对象相对于版面或由 offsetLeft 属性指定的父坐标的计算左侧位置。

● offsetTop: 获取对象相对于版面或由 offsetTop 属性指定的父坐标的计算顶端位置。

### 实现过程

编写用于实现广告随滚动条漂移的JavaScript代码。

(1) 自定义函数floaters(), 用于对层进行相应的设置。

```
<script language="JavaScript">
var delta=0.08
var layers;
function floaters() {
    this.items= [];
    this.addItem= function(id, x, y, content) {
        document.write(' <DIV id=' +id+' style="Z-INDEX: 10;
POSITION: absolute;
        width:180px;
height:130px;left:' +x+' ;top:' +y+' ">' +content+' </DIV>');
        var newItem= {};
        newItem.object= document.getElementById(id);
        if(y>10) {y=document.body.clientHeight-140}
        newItem.x= x;
        newItem.y= y;
        this.items[this.items.length]= newItem;
    }
    this.play= function() {
```

```

        layers= this.items;
        setInterval('play()', 10);
    }
}

```

(2) 自定义函数play(), 用于实现图片的漂移效果。

```

functionplay() {
    for(var i=0;i<layers.length;i++) {
        var obj= layers[i].object;
        var obj_x=layers[i].x;
        var obj_y=layers[i].y;
        if(obj.offsetLeft!=(document.body.scrollLeft+obj_x)) {
            vardx=(document.body.scrollLeft+obj_x-
obj.offsetLeft)*delta;
            dx=(dx>0?1:-1)*Math.ceil(Math.abs(dx));
            obj.style.left=obj.offsetLeft+dx;
        }
        if(obj.offsetTop!=(document.body.scrollTop+obj_y)) {
            vary=(document.body.scrollTop+obj_y-
obj.offsetTop)*delta;
            dy=(dy>0?1:-1)*Math.ceil(Math.abs(dy));
            obj.style.top=obj.offsetTop+dy;
        }
        obj.style.display= '';
    }
}

```

(3) 自定义函数excur(), 用于调用自定义函数floaters()和play()。

```
function excur() {  
    var strfloat = new floaters();  
    strfloat.addItem('div1', document.body.clientWidth-  
180, document.body.clientHeight  
-130, '<img src=2.jpg border=1 style="border-  
color:#FFFFFF">');  
    strfloat.play();  
}
```

(4) 在窗体载入时，调用自定义函数excur()。

```
excur();
```

```
</script>
```

举一反三

根据本实例，读者可以：

制作在页面中随意漂移的广告。

## 9.4 页面动画效果

在浏览页面时，经常会看到具有动画效果的页面，如下雪的效果、星空的效果等。下面将通过几个典型实例进行详细讲解。

### 实例269 下雪

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\09\269

实例说明

在浏览网页时，经常会看到一些动画效果，使页面显得更加生动。本实例在页面打开时，页面中将会出现下雪的效果。运行结果如图9.26所示。



图9.26 下雪

### 技术要点

本实例主要运用了 document 对象的 write() 方法在页面中添加层 (<div>), 再用 Math 对象的 random() 方法来随机设置层的字体大小 (使雪花大小不同), 通过 window 对象的 setTimeout() 方法不断地修改层的 style 样式的 top 和 left 属性, 使层在页面中不断地移动, 从而形成下雪的效果。

注意: 如果要使层在页面中移动, 必须将层的 style 样式的 position 属性值设为 absolute。

### 实现过程

- (1) 给页面添加一个背景图片。
- (2) 编写用于实现下雪效果的 JavaScript 代码。

在页面中添加15个层, 并设置层的初始位置以及层的字体大小。代码如下。

```
<script language="JavaScript">
drop=newArray();xx=newArray();yy=
newArray();mv=newArray();
snowsym= "*";
howmany=15;
speed=50;drops=howmany; angle=6;
movex=-speed/angle;movey= speed; count=0;
for(var i=0; i<drops; i++){
    document.write('<div id="drop'+i+' " style="position:
absolute;">' +
    snowsym+'</div>');
    drop[i]= eval("document.all['drop'+i+'"].style");
    maxx=document.body.clientWidth-40;
```

```
maxy=document.body.clientHeight-40;
xx[i]=Math.random()*maxx;
yy[i]=-100-Math.random()*maxy;
drop[i].left=xx[i];
drop[i].top=yy[i];
mv[i]= (Math.random()*5)+speed/16;
drop[i].fontSize= (Math.random()*10)+10;
drop[i].color= "#FFFFFF";
}
```

自定义函数moverain(), 用于实现下雪的动态效果。

```
function moverain() {
    for(move = 0; move < drops; move++) {
        xx[move]+=movex; yy[move]+=mv[move];
        if(xx[move]<0) {xx[move]=maxx+10;}
        if(yy[move]>maxy) {yy[move]=10;}
        drop[move].left=xx[move];
        drop[move].top= yy[move]+document.body.scrollTop;
    }
    setTimeout('moverain()', '1');
}
```

在窗体载入时, 调用自定义函数moverain()。

```
moverain();
```

```
</script>
```

举一反三

根据本实例, 读者可以:

实现蒲公英飞舞的页面。

## 实例270 飘落的枫叶

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\270

实例说明

在浏览网页时，经常会看到一些动画效果，使页面显得更加生动。本实例在页面打开时，枫叶会不断地随风飘落。运行结果如图9.27所示。

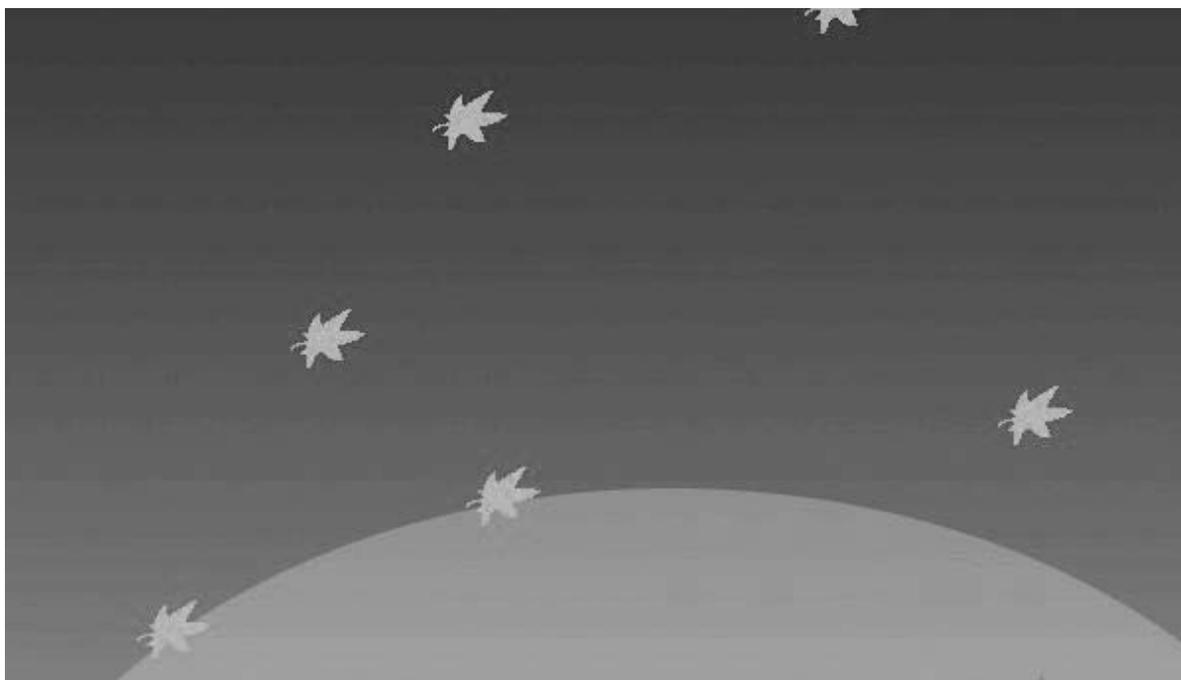


图9.27 飘落的枫叶

技术要点

本实例主要运用了document对象的write()方法在页面中添加层(<div>)，再用window对象的setTimeout()方法不断地修改层的style样式的top和left属性，使层在页面中不断移动，从而形成枫叶飘落的效果。

注意：如果要使层在页面中移动，必须将层的style样式的position属性值设为absolute。

实现过程

编写用于实现枫叶飘落的JavaScript代码。

(1) 自定义全局变量及数组。

```
<script language="javascript">
var no=8;
var speed=100;
var heart="1.gif";
var flag;
var dx, xp, yp;
var am, stx, sty;
var i;
doc_width=document.body.clientWidth;
doc_height=document.body.clientHeight;
dx=newArray();
xp=newArray();
yp=newArray();
amx=newArray();
amy=newArray();
stx=newArray();
sty=newArray();
flag= newArray();
```

(2) 在页面中添加多个层，并设置层的位置及移动大小。

```
for (i=0; i<no;++i) {
    dx[i]=0;
    xp[i]=Math.random()*(doc_width-30)+10;
    yp[i]=Math.random()*(doc_height-30);
    amy[i]=12+Math.random()*20;
```

```

    amx[i]=10+Math.random()*40;
    stx[i]=0.02+Math.random()/10;
    sty[i]=0.7+Math.random();
    flag[i]= (Math.random()>0.5)?1:0;
    document.write("<div id=\"dot"+i+"\" style=\"POSITION:
");
    document.write("absolute;Z-INDEX: "+i+";VISIBILITY: ");
    document.write("visible;TOP:15px;LEFT:15px;\"><img
src=\"");
    document.write(heart+"\"border=\"0\"></div>");
}

```

(3) 自定义函数grue(), 使多个层在页面中进行移动。

```

function grue() {
    for (i=0; i<no;++i) {
        if (yp[i]>doc_height-50) {
            xp[i]=10+Math.random()*(doc_width-amx[i]-30);
            yp[i]=0;
            stx[i]=0.02+Math.random()/10;
            sty[i]=0.7+Math.random();
            flag[i]=(Math.random()<0.5)?1:0;
            doc_width=document.body.clientWidth;
            doc_height=document.body.clientHeight;
            if (doc_height==0)doc_height=768;
        }
        if (flag[i])dx[i]+= stx[i];
        elsedx[i]-= stx[i];
        if (Math.abs(dx[i])>Math.PI) {

```

```

        yp[i]+=Math.abs(amy[i]*dx[i]);
        xp[i]+=amx[i]*dx[i];
        dx[i]=0;
        flag[i]=!flag[i];
    }
    document.all["dot"+i].style.pixelTop= yp[i]+amy[i]*
(Math.abs
(Math.sin(dx[i])+dx[i]));
    document.all["dot"+i].style.pixelLeft=xp[i]+amx[i]*dx
[i];
    }
    setTimeout("grue()", speed);
}

```

(4) 在窗体载入时，调用自定义函数grue()。

```

grue();
</script>

```

举一反三

根据本实例，读者可以：

实现花瓣的飘落效果；

实现羽毛的飘落效果。

## [实例271 下雨](#)

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\09\271

实例说明

在浏览网页时，经常会看到一些动画效果，使页面显得更加生动。本实例将在页面打开时实现下雨的效果。运行结果如图9.28所示。

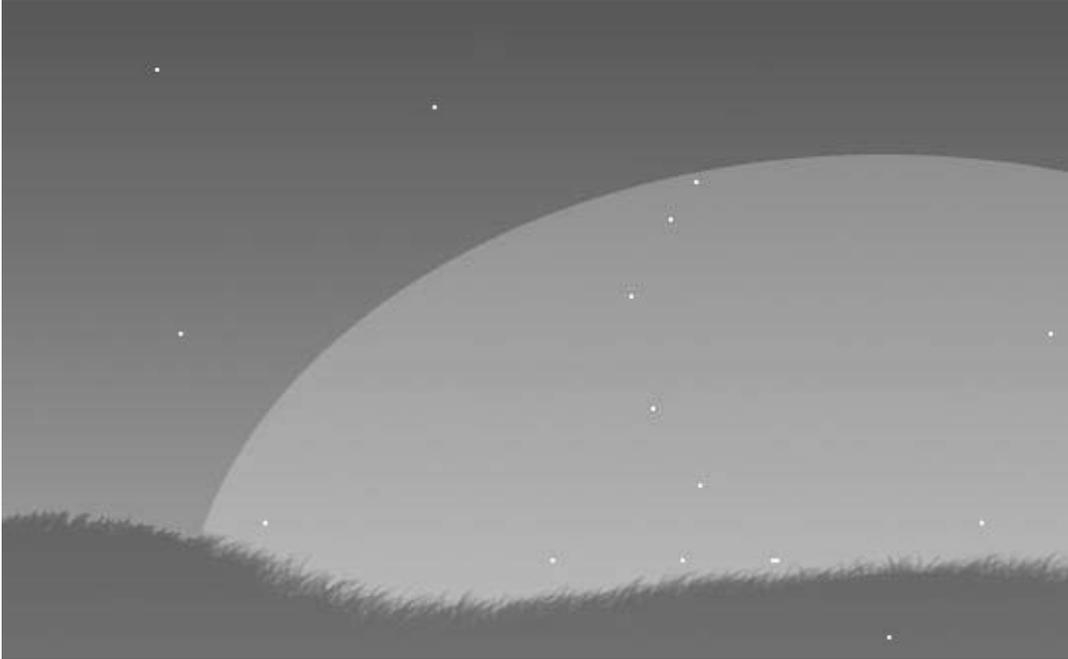


图9.28 下雨

#### 技术要点

本实例的技术要点与实例270相同，只是在本实例中用了多个自定义函数将其功能进行封装，这样可以使思路更加清晰。

#### 实现过程

(1) 在页面中添加背景图片。

(2) 编写用于实现在页面中下雨效果的JavaScript代码。

自定义变量及数组，并获取页面工作区的宽度和高度。

```
<script language="JavaScript">
```

```
var no=50;
```

```
var speed=1;
```

```
var s, x, y, sn, cs;
```

```
var a, r, cx, cy;
```

```
var i;
clientW=document.body.clientWidth;
clientH=document.body.clientHeight;
x=newArray();
y = new Array();
r = new Array();
cx = new Array();
cy = new Array();
s = 20;
```

在页面中添加多个层，并指定其位置及大小。

```
for (i=0; i<no;++i) {
    initRain();
    document.write("<div id=dot"+i+" style=' POSITION:
absolute;Z-INDEX: "+i+"
    VISIBILITY:visible;TOP:15px;LEFT:15px' ><font
color=' #FFFFFF' >.</font></div>");
}
```

自定义函数initRain()，用于设置每个层的移动方向及移动的大小。

```
function initRain() {
    a=6;
    r[i]=1;
    sn=Math. sin(a);
    cs=Math. cos(a);
    cx[i]=Math. random() *clientW+1;
    cy[i]=Math. random() *clientH+1;
    x[i]= r[i] * sn+cx[i];
```

```
    y[i]= cy[i];  
}
```

自定义函数makeRain(), 使层进行移动。

```
functionmakeRain() {  
    r[i]=1;  
    cx[i]=Math.random() *clientW+1;  
    cy[i]=1;  
    x[i]= r[i] * sn+cx[i];  
    y[i]= r[i] *cs+cy[i];  
}
```

自定义函数updateRain(), 判断层是否超出可移动的范围。

```
function updateRain() {  
    r[i]+= s;  
    x[i] = r[i] * sn+cx[i];  
    y[i] = r[i] * cs+cy[i];  
}
```

自定义函数 raindropIE(), 调用自定义函数 updateRain() 和 makeRain(), 并实现层的动态移动。

```
function raindropIE() {  
    for (i=0; i<no;++i) {  
        updateRain();  
        if ((x[i]<=1) || (x[i]>= (clientW-20)) || (y[i]>=  
(clientH-20))) {  
            makeRain();  
            clientW=document.body.clientWidth;  
            clientH=document.body.clientHeight;  
        }  
    }  
}
```

```
        document.all["dot"+i].style.top= y[i];
        document.all["dot"+i].style.left=x[i];
    }
    setTimeout("raindropIE()", speed);
}
```

在窗体载入时，调用自定义函数raindropIE()。

```
raindropIE();
```

```
</script>
```

举一反三

根据本实例，读者可以：

实现滑落的流星效果；

根据网络的天气预报制作不同的页面效果。

## 实例272 背景的烟花效果

本实例可以美化界面

实例位置：光盘\mingrisoft\09\272

实例说明

在浏览网页时，经常会有一些动画效果，使页面显得更加生动。本实例将实现在页面打开时绽放的烟花效果。运行结果如图9.29、图9.30所示。

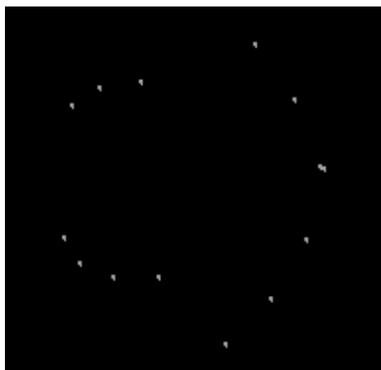


图9.29 烟花效果



图9.30 烟花绽放效果

### 技术要点

本实例主要是用document对象的write()方法将层添加到页面上，并根据Math对象的sin()方法（获取正弦值）来动态改变每个层的style样式的top和left属性值，使层根据正弦值进行扩大和缩小。

### 实现过程

编辑用于实现烟花效果的JavaScript代码。

（1）自定义变量及数组，并用数组来保存颜色值，在页面中添加多个层，设置其大小、颜色和初始位置。

```
<script language="JavaScript">  
var  
col=newArray(' #ffffff', ' #fff000', ' #ffa000', ' #ff00ff', ' #00ff00  
,
```

```

    '#0000ff', '#ff0000');
    var p='<div id="rearDiv"
style="position:absolute;top:0px;left:0px">';
    var n=0;
    for (i=0;i<14;++i) {
        n++;
        if (n=(col.length-1))n=0;
        p=p+'<div
style="position:relative;width:1px;height:1px;background:' +
col[n]+';
font-size:3px">.</div>';
    }
    p=p+"</div>";
    document.write(p);
    varClrs=newArray('ff0000','00ff00','000aff','ff00ff','ffa
500','ffff00','00ff00'
,'ffffff','ffffff');
    var sClrs = new
Array('ffa500','55ff66','AC9DFC','fff000','ffffff');
    var peepY;
    var peepX;
    var step = 5;
    var tallyStep = 0;
    var backColor = 'ffa000';
    var Mtop = 250;
    var Mleft = 250;

```

(2) 自定义函数dissilient(), 调用自定义函数enlarge()和reduce(), 用于实现绽放的烟花效果。

```
functiondissilient() {  
    peepY=window.document.body.clientHeight/3;  
    peepX=window.document.body.clientWidth/8;  
    enlarge();  
    tallyStep+= step;  
    reduce();  
    T=setTimeout("dissilient()",20);  
}
```

(3) 自定义函数enlarge(), 利用正弦值来实现烟花的绽放与缩小。

```
function enlarge() {  
    for ( i=0 ; i< rearDiv.all.length ; i++) {  
        var c=Math.round(Math.random()*(Clrs.length-1));  
        if (tallyStep<90)  
            rearDiv.all[i].style.background=backColor;  
        if (tallyStep>90)  
            rearDiv.all[i].style.background=Clrs[c];  
        rearDiv.all[i].style.top=Mtop+peepY*Math.sin((tallyStep+i*5)/3)  
            *Math.sin(550+tallyStep/100);  
        rearDiv.all[i].style.left=Mleft+peepY*Math.cos((tallyStep+i*5)/3)  
            *Math.sin(550+tallyStep/100);  
    }  
}
```

(4) 自定义函数reduce(), 用于改变烟花的绽放位置。

```
function reduce() {
    if (tallyStep==220) {
        tallyStep=-10;
        vark=Math.round(Math.random()*(sClrs.length-1));
        backColor= sClrs[k];
        Dtop=window.document.body.clientHeight-250;
        Dleft=peepX *3.5;
        Mtop=Math.round(Math.random()*Dtop);
        Mleft=Math.round(Math.random()*Dleft);
        document.all.rearDiv.style.top=Mtop+document.body.scrollTop;
        document.all.rearDiv.style.left=Mleft+document.body.scrollLeft;
        if ((Mtop<20) || (Mleft<20)) {
            Mtop+=90;
            Mleft+=90;
        }
    }
}
```

(5) 在窗体载入时, 调用自定义函数dissilient()。

```
dissilient();
```

```
</script>
```

举一反三

根据本实例, 读者可以:

制作页面的爆炸效果;

实现当鼠标单击时, 出现波纹效果。

## 实例273 变色的圆圈

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\273

实例说明

在浏览网页时，经常会有一些动画效果，使页面显得更加生动。本实例在页面打开时，将动态显示一个变色且翻转的圆圈。运行结果如图9.31、图9.32所示。



图9.31 圆圈变色缩小效果



图9.32 圆圈变色放大效果

技术要点

本实例主要通过Math对象的 $\sin()$ 和 $\cos()$ 的值来修改页面中层的style样式的top和left属性值。当 $\sin()$ 和 $\cos()$ 的值相同时，页面中的层将形成一个圆形。如果按一定的 $\sin()$ 或 $\cos()$ 值再次改变层的top和left属性，那么在页面中的层将会不断翻转。

在对层中的层设置属性、样式及滤镜时，可以用以下语句来实现。

```
div1.all[i].mode
```

其中，div1表示外层。all[i]表示外层中的内层，i为数值型，i的个数不能超出内层的总个数，如果层中只包含一个内层，那么应用div1.all[0]对层进行设置。mode表示层的属性、样式及滤镜。当层中有多个层时，可以用该语句对内层进行设置。

```
div1.all.div2.mode
```

其中，div1表示外层。div2表示外层中的内层，mode表示层的属性、样式及滤镜。

### 实现过程

编写用于实现圆圈变色的JavaScript代码。

(1) 自定义变量及数组，并在数组中保存变换的颜色值，在页面中添加moveDiv层，在该层中添加多个层，并设置其大小和位置。

```
<body>
<script language="JavaScript">
varCoLoUrS=newArray('ff0000','ffffff','0000ff','ffffff');
var dots=16;
var step=0.3;
var a_StEp=0.05;
varRunTime=Xpos=Ypos=count_a=count=currStep=0;
var cntr=70;
varmove_a=move=1;
var curColour;
document.write(' <div id="moveDiv"
style="position:absolute;top:0px;left:0px">');
for (i=0; i<dots; i++) {
document.write(' <div
style="position:absolute;top:0px;left:0px;width:2px;
height:2px;background:#ffffff;font-size:2px"></div>');
```

```

}
document.write('</div>');
var x = moveDiv.all;
var numdiv=new
Array(x[0], x[1], x[2], x[3], x[4], x[5], x[6], x[7], x[8],
x[9], x[10], x[11], x[12], x[13], x[14], x[15]);

```

(2) 自定义函数colourStep(), 用于指定时间变换moveDiv层中各层的前景颜色。

```

function colourStep() {
    count+=move;
    if (count>=numdiv.length) {count=0;count_a+=move}
    if (count_a==CoLoUrS.length) count_a=0;
    numdiv[count].style.background=CoLoUrS[count_a];
    setTimeout('colourStep()', 100)
}

```

(3) 自定义函数movediv(), 用于计算层的移动位置。

```

functionmovediv() {
    Ypos=document.body.scrollTop+document.body.offsetHeight
/2+document.body.
    offsetHeight/2.8*Math.cos((RunTime)/3)*Math.cos(RunTime
/10);
    Xpos=
document.body.scrollLeft+document.body.offsetWidth/2+docume
nt.body.
    offsetWidth/2.4*Math.sin((RunTime)/5)*Math.sin(RunTime/
20);
    RunTime+=a_StEp;
}

```

```
Tma=setTimeout("movediv()", 10);  
}
```

(4) 自定义函数overturn(), 在翻转层的同时, 对层进行移动。

```
function overturn() {  
    for (i=0;i<numdiv.length;i++) {  
        numdiv[i].style.top =  
Ypos+cntr*Math.cos((currStep+i*4)/10.2)*  
        Math.cos(currStep/10)  
        numdiv[i].style.left=  
Xpos+cntr*Math.sin((currStep+i*4)/10.2);  
    }  
    currStep+=step;  
    setTimeout("overturn()", 10);  
}
```

(5) 自定义函数Shrink(), 用于缩小各层的排列位置。

```
function Shrink() {  
    cntr-=0.5;  
    GoShrink=setTimeout('Shrink()', 10);  
    if (cntr <=20) {clearTimeout(GoShrink);Swell()}  
}
```

(6) 自定义函数Swell(), 用于放大各层的排列位置。

```
function Swell() {  
    cntr+=0.5;  
    GoSwell=setTimeout('Swell()', 10);  
    if (cntr >=70) {clearTimeout(GoSwell);Shrink()}  
}
```

(7) 自定义函数transfer(), 用于调用其他的自定义函数。

```
function transfer() {  
    movediv();    //移动层  
    overturn();   //使层进行翻转  
    Shrink();     //改变层的大小  
    colourStep(); //变换层的颜色  
}
```

(8) 在窗体载入时，调用自定义函数transfer()。

```
transfer();
```

```
</script>
```

举一反三

根据本实例，读者可以：

在页面中制作一条动态旋转的线；

制作一个球体的碰撞效果。

## 实例274 滚动的光环

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\09\274

实例说明

在浏览网页时，经常会有一些动画效果，使页面显得更加生动。本实例在页面打开时，将实现滚动的光环效果。运行结果如图9.33、图9.34所示。

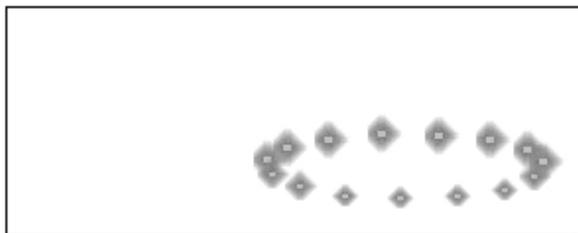


图9.33 光环向右下角滚动

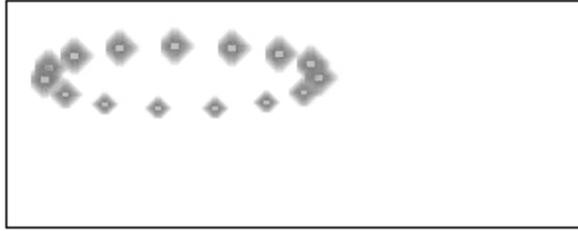


图9.34 光环向左上角滚动

### 技术要点

本实例主要通过层的style样式中的paddingTop属性来设置层的高度，通过style样式的filter（滤镜）中的glow属性来实现光环的发光效果。下面对paddingTop属性和glow属性进行详细说明。

paddingTop属性用于设置对象与其最近一个定位的父对象顶部相关的位置。

语法格式如下。

```
paddingTop:length
```

参数说明如下。

- length: 由浮点数字和单位标识符组成的长度值或者百分数。百分数是基于父对象的宽度。

glow属性用于在对象附近的外边框加发光效果，其语法格式如下。

```
STYLE="filter:glow(Color=color, Strength=strength)"
```

参数说明如下。

- Color: 发光颜色。
- Strength: 强度(0~100)。

### 实现过程

编写用于实现光环向上滚动的JavaScript代码。

(1) 自定义变量，并在页面中添加多个层。

```
<script language="JavaScript">
```

```
amount=ybase=15;
```

```

Color=' #ffaaff' ;
GlowColor=' #ff00ff' ;
xbase=-70;
step=c=0.05;
TrigSplit=360/amount;
xpos=ypos=currStep=Ci=0;
for (i=0; i < amount; i++)
    document.write(' <div id="Div"
style="position:absolute;top:0px;left:0px;
height:20;width:20;text-align:center"><B>.</B></div>');

```

(2) 自定义函数running(), 使光环在页面中上下移动, 并且自身也进行转动, 在转动的同时, 根据远近改变层的大小。

```

function running() {
    var
MY=document.body.scrollTop+document.body.clientHeight/2+doc
ument.body
    .clientHeight/2.8*Math.cos((Ci)/4)*Math.cos(Ci/10);
    var
MX=document.body.scrollLeft+document.body.clientWidth/2+doc
ument.body
    .clientWidth/2.4*Math.sin((Ci)/6)*Math.sin(Ci/20);
    ypos=MY;
    xpos=MX;
    for (i=0; i< amount; i++){
        vard=Div[i].style;
        if (d.pixelTop>ypos+2) {
            d.fontSize=18;

```

```

d.paddingTop=7;
d.filter=' glow(color='+GlowColor+', strength=7)';
if (d.pixelTop>ypos+10) {
    d.fontSize=15;
    d.paddingTop=10;
    d.filter=' glow(color='+GlowColor+', strength=5)';
}
}
else {
    d.fontSize=25;
    d.paddingTop=0;
    d.color=Color;
    d.filter=' glow(color='+GlowColor+', strength=8)';
}
}
d.top=ypos+ybase*Math.sin(currStep+i*TrigSplit*Math
.PI/180);
d.left=xpos+xbase*Math.cos(currStep+i*TrigSplit*Mat
h.PI/180);
}
Ci+=c;
currStep+=step;
setTimeout('running()', 20);
}

```

(3) 在窗体载入时，调用自定义函数running()。

```

running();
</script>

```

举一反三

根据本实例，读者可以：  
制作太阳的发光效果；  
制作飞翔的气球效果。

## 实例275 星空极速飞入效果

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\09\275

实例说明

在浏览网页时，经常会有一些动画效果，使页面显得更加生动，本实例在页面打开时，会产生一种星空的动态效果。运行结果如图9.35所示。

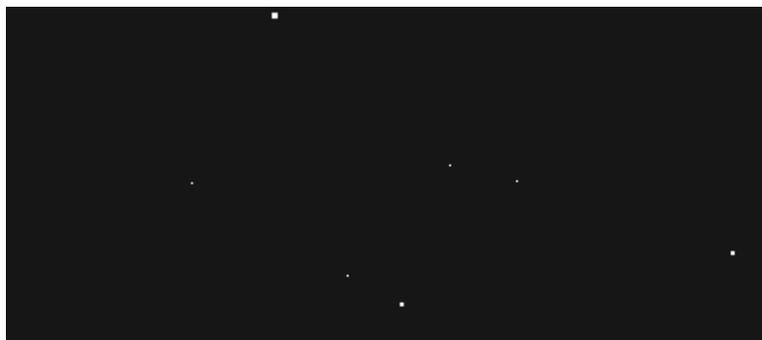


图9.35 星空极速飞入界面

技术要点

本实例主要是对style样式的position属性进行设置，使层可以在页面中进行移动。下面对position属性进行详细说明。

position属性的语法如下。

```
position: static | absolute | relative
```

参数说明如下。

- static: 无特殊定位，对象遵循HTML 定位规则。
- absolute: 将对象从文档流中拖出，使用left、right、top、bottom等属性进行绝对定位。而其层叠通过z-index属性定义。此时，

对象不具有边距，但仍有补白和边框。

● **relative**：对象不可层叠，但将依据left、right、top、bottom等属性在正常文档流中移动位置。

实现过程

(1) 将页面的背景颜色设置为深蓝色。

(2) 编写用于实现星空页面的JavaScript代码。

自定义变量及数组，并在页面中添加多个层。

```
<script language="JavaScript">
xL=8;
xH=xW=xR=xE=xMY=xMX=xWd=xHd=0;
xF=newArray();
xY=newArray();
xX=newArray();
xS=newArray();
xA=newArray();
xB=newArray();
ini=newArray();
document.write(' <div id="sdiv"
style="position:absolute;top:0px;left:0px">
<div style="position:relative">');
for (i=0; i<xL; i++) {
    document.write(' <div id="div"
style="position:absolute;top:0;left:0;
width:5px;height:5px;' + ' font-size:10px;color:#ffffff">
</div>');
}
document.write(' </div></div>');
```

```
functionSet () {
    for (i=0; i<xL; i++) {
        transfer(i)
        xF[i]=xW/14;
    }
}
```

自定义函数Assign(), 用于改变层的位置及大小。

```
functionAssign () {
    sdiv.style.top=document.body.scrollTop;
    for (i=0; i<xL; i++) {
        xF[i]-=xS[i]*25;
        if (xF[i]<4) xF[i]=3;
        div[i].style.top=xY[i];
        div[i].style.left=xX[i];
        div[i].style.fontSize=xF[i];
    }
}
```

自定义函数fly(), 用于动态计算层的位置及大小。

```
function fly () {
    xMY=window.document.body.clientHeight/2;
    xMX=window.document.body.clientWidth/2;
    xWd=Math.round(Math.random()*40+5);
    xHd=Math.round(Math.random()*30+5);
    for (i=0; i<xL; i++) {
        xY[i]=xA[i]+(xMY-xA[i])*(xS[i]);
        xX[i]=xB[i]+(xMX-xB[i])*(xS[i]);
    }
}
```

```

        if ((xX[i]>xMX-xWd)&& (xX[i]<xMX+xWd)&& (xY[i]>xMY-
xHd)&& (xY[i]<xMY+xHd)) {
            transfer(i)
        }
        if ((xX[i]<0) || (xX[i]>xW) || (xY[i]<0) || (xY[i]>xH)) {
            xF[i]=xW/14;
        }
    }
    Assign();
    setTimeout('fly()',1);
}

```

自定义函数transfer()，将层不断地向页面中间移动。

```

function transfer(i) {
    xH=window.document.body.offsetHeight;
    xW=window.document.body.offsetWidth;
    xA[i]=Math.round(Math.random()*xH);
    xB[i]=Math.round(Math.random()*xW);
    xS[i]=Math.random()*0.05+0.05;
    xR=Math.round(Math.random()*3);
    xE=Math.round(Math.random()*50+50);
    if (xR==3) xB[i]=-xE;
    if (xR==2) xB[i]=xW+xE;
    if (xR==1) xA[i]=-xE;
    if (xR==0) xA[i]=xH;
}

```

在窗体载入时，调用自定义函数Set()和fly()。

```
Set();
```

```
fly());  
</script>
```

举一反三

根据本实例，读者可以：  
实现飞行的陨石的效果；  
应用于网络小游戏。

## 实例276 闪烁的星星

这是一个可以提高基础性能的实例

实例位置：光盘\mingrisoft\09\276

实例说明

在浏览网页时，经常会有一些动画效果，使页面显得更加生动，本实例在页面打开时，将会产生一种星星闪烁的效果。运行结果如图9.36所示。

技术要点

本实例主要是用层的style样式的left属性来设定层的左端位置，并用style样式的top属性来随机设定层的顶端位置。

document对象的body的clientHeight属性用来获取对象的高度，不计算任何边距、边框、滚动条或可能应用到该对象的补白。Math对象的random()方法用来获取0~1（包含0，不包含1）之间的随机数。

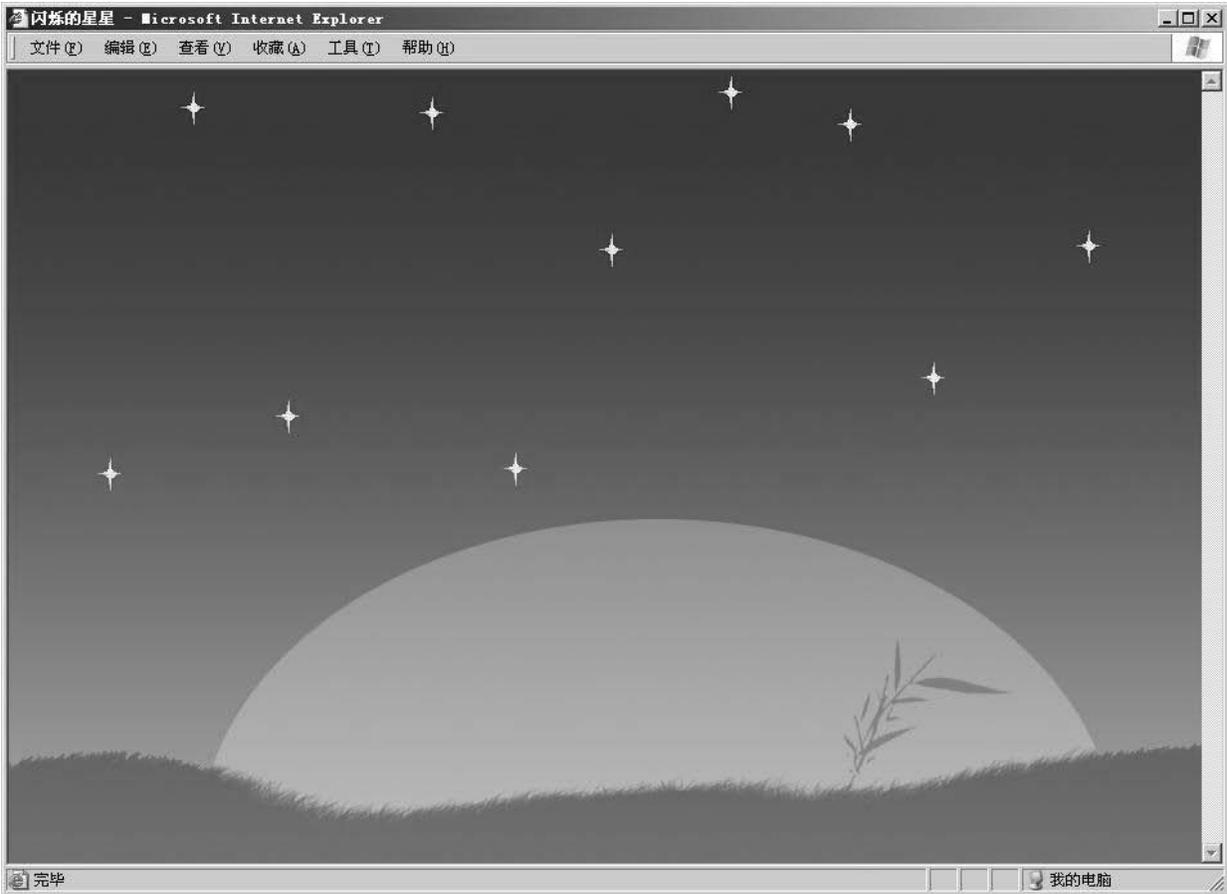


图9.36 闪烁的星星

### 实现过程

(1) 编写用于实现星星闪烁的JavaScript代码。

在页面中添加多个层，并在每个层中添加图片。

```
<script language="JavaScript">
for(i=1;i<=10;i++) {
    varp=Math.floor(Math.random()*5)+9*(i-1);
    document.write("<div id=\"div"+i+\"\"
style=\"position:absolute;top:-30px;
    left:\"+p+\"%;width:21;height:21;visibility:hidden;\">
<dd><img src=\"star.gif\"
    WIDTH=\"21\"HEIGHT=\"21\"></dd></div>");
}

```

自定义函数move(), 用于改变每个层与上边距的距离。

```
function move() {  
    for(i=1;i<=10;i++)  
        eval("div"+i).style.top=gettop();  
}
```

自定义函数gettop(), 用于随机获取层与上边距的距离。

```
function gettop() {  
    var t=String(Math.floor(Math.random()*  
(document.body.clientHeight/2)))+“px”;  
    return(t);  
}
```

自定义函数vision(), 将页面中的层设为显示状态。

```
function vision() {  
    for(i=1;i<=10;i++)  
        eval("div"+i).style.visibility=“visible”;  
}
```

自定义函数start(), 用于调用自定义函数move()和vision(), 并对move()函数进行重复调用。

```
function start() {  
    setInterval(“move()”,1000);  
    vision();  
}
```

```
</script>
```

(2) 在<body>标记中改变页面的背景颜色, 并在窗体载入(onload)事件中调用自定义函数start()。

```
<body bgColor=#000000 onload=start()>
```

举一反三

根据本实例，读者可以：  
根据时间将网页设置成白天、黑夜的状态；  
制作个人网站背景。

## 9.5 其他

在制作网页时，页面的设置多种多样，本节将对页面的其他特效进行详细的讲解，例如带密码的网页、自制页面等。

### 实例277 带密码的网页

本实例可以提高网络安全性

实例位置：光盘\mingrisoft\09\277

实例说明

在浏览网页时，经常会有一些只有特殊用户才能进去的网页，这样可以保护网页中的一些重要资源，本实例将制作一个带有密码的网页，如图9.37所示。

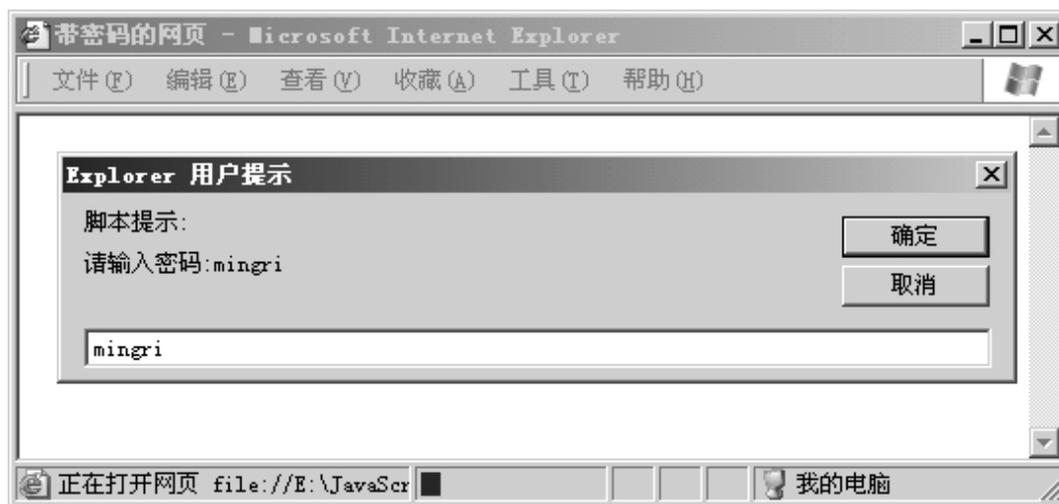


图9.37 带密码的网页

技术要点

本实例主要通过 window 对象的 prompt() 方法在打开页面时弹出一个提示对话框。下面对 prompt() 方法进行详细说明。

prompt()方法的语法格式如下。

```
window.prompt(text, defaultInput)
```

参数说明如下。

- text: 在对话框中显示的提示信息。
- defaultInput: 文本框的默认值。如果在该参数中设置一个字符串,将在文本框中显示默认值;如果不需要默认值,那么在文本框中设置一个空字符串。

功能: 显示提示对话框。

实现过程

(1) 在页面中添加一个表格,用于显示调用的页面。代码如下。

```
<table width="100%" height="100%" border="0"
cellpadding="0" cellspacing="0">
  <tr>
```

JavaScript代码通过对输入密码的判断,在表格中显示指定的语言。

```
<script language="javascript">
  var arr=newArray("你已经通过正确的密码进入本网页","密码错误,请重新输入")
  if (s==true) {
    document.write(' <td id="td1"height="13"
scope="col">' +arr[0]+' </td>')
  }
  else {document.write(' <td id="td1"height="13"
scope="col">' +arr[1]+' </td>')}
</script>
  设置单元格的高度的HTML代码如下。
</tr>
```

```
<tr>
```

```
  <th height="200" valign="top" scope="col">
```

JavaScript代码通过对输入密码的判断，在表格中显示指定的页面。

```
  <script language="javascript">
```

```
    if (s==true) {
```

```
      document.write(' <iframe src="page.htm" name="ifrm"
frameborder="0"
```

```
  scrolling="auto" width="100%"
height="100%">ff</iframe>')
```

```
    }
```

```
  </script>
```

HTML代码如下。

```
    </th>
```

```
  </tr>
```

```
</table>
```

(2) 编写用于实现带密码网页的JavaScript代码。

自定义函数mima()，用于弹出一个判断窗体，并将返回的结果保存在自定义变量s中。

```
<script language="javascript">
```

```
var s=false;
```

```
var str="mingri"
```

```
functionmima() {
```

```
  if (prompt("请输入密码:"+str)==str) {
```

```
    s=true;
```

```
  }
```

```
  else {s=false;}
```

```
}
```

在窗体载入时，调用自定义函数mima()。

```
mima();
```

```
</script>
```

举一反三

根据本实例，读者可以：

制作一个注册网页；

在关闭网页时，询问是否浏览其他网页。

## 实例278 页面左右滚动

这是一个自娱自乐的实例

实例位置：光盘\mingrisoft\09\278

实例说明

在浏览网站时，有的网站因页面过宽，在浏览时非常不便。本实例将实现一个页面左右滚动的效果，实现此效果，在页面中必须出现水平滚动条，用鼠标双击页面，页面将会从左到右进行滚动，当单击鼠标时，页面停止滚动。运行结果如图9.38所示。



图9.38 页面左右滚动

技术要点

本实例主要通过body的scrollLeft属性和clientWidth属性来获取整个工作区的宽度，通过window对象的scrollBy()方法进行移动滚动条。下面对其进行详细说明。

● scrollLeft: 设置或获取对象左边界和窗口中目前可见内容的最左端之间的距离。

● clientWidth: 获取对象的宽度，不计算任何边距、边框、滚动条或可能应用到该对象的补白。

scrollBy()方法的语法格式如下。

```
widnow.scrollBy(x, y)
```

参数说明如下。

● x: 数值型，控制左右移动的个数。

● y: 数值型，控制上下移动的个数。

功能: 相对于上一次滚动条的位置进行滚动条的移动。

实现过程

(1) 在页面中添加一个图片，使页面出现水平滚动条。代码如下。

```

```

(2) 编写用于实现页面左右滚动的JavaScript代码。

自定义函数aboutmove(), 用于实现页面的左右滚动。

```
<script language="javascript">
```

```
functiondocument.body.onselectstart(){return false;}
```

```
var tim=null;
```

```
var bigit=1;
```

```
function aboutmove() {
```

```
    if(document.body.scrollLeft==0 ||
```

```
(document.body.scrollHeight-document.body.
```

```
scrollLeft)==document.body.clientWidth) {
```

```
        bigit*=-1;
    }
    window.scrollBy(bigit,0);
}
```

通过页面的ondblclick()（鼠标双击）事件，重复调用自定义函数aboutmove()。

```
function document.ondblclick() {tim =
setInterval("aboutmove()",100);}
```

用页面的onclick()（鼠标单击）事件，停止对自定义函数aboutmove()的调用。

```
function document.onclick() {clearInterval(tim)}
</script>
```

举一反三

根据本实例，读者可以：

实现页面的上下滚动；

实现以读书的方式滚动页面。

## [实例279 动态移动的层](#)

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\09\279

实例说明

在浏览一些网站时，页面的里面还会有一个窗体，并可以对该窗体进行移动、关闭等操作。本实例将用层在页面中制作一个窗体。运行结果如图9.39、图9.40所示。



图9.39 层的初始位置



图9.40 移动层到右下角

### 技术要点

本实例主要是在鼠标按住要移动的层时，将层的名称赋给一个变量，并用该变量对层进行移动的操作。如果直接用层的名称来对层进行移动，将会选中层后面的一些信息。

通过event对象的x（当前鼠标的x坐标值）和y（当前鼠标的y坐标值）属性在要移动的层上拖动鼠标，从而改变当前层的位置。

CSS 样式的filter 滤镜中的revealTrans 属性的介绍请参考实例267。

### 实现过程

(1) 在页面中添加两个层，并且其中的一个层包含另一个层。在 div1 层的鼠标按下 (onmousedown) 事件中调用自定义函数 DIVDown()。

```
<div id=div1class=yellow style="VISIBILITY:visible
;background-color:#FFFF00;
position:absolute; top:60; left:200;width:360;height:250;
filter:revealTrans
(transition=12, 'duration=0.1)blendTrans(duration=0.1)">
  <div id=title onmousedown=DIVDown("div1")
style="background-color:#30608F;
padding:2px; font-size:13px;text-
indent:5;color:#FFFFFF;cursor:move">明日科技
</div>
  
</div>
```

(2) 编写用于实现动态移动层的JavaScript代码。

自定义函数DIVDown()，获取当前鼠标与层的左边界和上边界的距离。

```
<script language="JavaScript">
var Obj="none";
var pX
var pY
function DIVDown(tag) {
  Obj=tag;
  pX=parseInt(document.all(Obj).style.left)-event.x;
```

```
    pY=parseInt(document.all(Obj).style.top)-event.y;
}
```

自定义函数DIVMove(), 用于移动层的位置。

```
function DIVMove() {
    if(Obj!="none") {
        document.all(Obj).style.left=pX+event.x;
        document.all(Obj).style.top=pY+event.y;
        event.returnValue=false;
    }
}
```

自定义函数DIVUp(), 当松开鼠标时, 释放对当前层的控制。

```
function DIVUp() {Obj="none";}
```

自定义函数Hide(), 用于将层以动态效果隐藏。

```
function Hide(divid) {
    divid.filters.revealTrans.apply();
    divid.style.visibility = "hidden";
    divid.filters.revealTrans.play();
}
```

在页面上移动鼠标时, 通过页面的 onmousemove (鼠标移动) 事件, 调用自定义函数DIVMove()。

```
document.onmousemove=DIVMove;
```

在页面上松开鼠标时, 通过页面的onmouseup (鼠标松开) 事件, 调用自定义函数DIVUp ()。

```
document.onmouseup=DIVUp;
```

```
</script>
```

举一反三

根据本实例, 读者可以:

在页面中制作一个窗体，并对它进行最大化、最小化和还原的操作；

在窗体中移动透明表格。

## 实例280 在页面中显示十字光标

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\09\280

实例说明

本实例是在页面中显示一个十字光标，当鼠标在页面中移动时，十字光标将跟随鼠标进行移动。运行结果如图9.41所示。

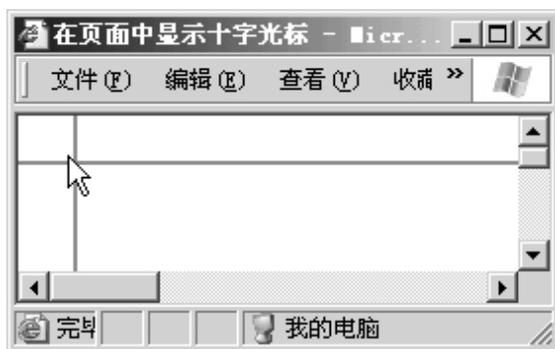


图9.41 在页面中显示十字光标

技术要点

本实例主要应用eval()函数对JavaScript代码进行执行。并通过document对象中body的scrollLeft、scrollTop属性来获取页面中横向滚动条和纵向滚动条的坐标值，通过event对象的clientX、clientY属性值来修改层的当前位置，使层跟随鼠标进行移动。下面将对eval()函数进行详细说明。

eval()函数的语法格式如下。

eval(codeString)

参数说明如下。

● codeString: 必选项, codeString 参数是包含有效 JavaScript 代码的字符串值。这个字符串将由JavaScript 分析器进行分析和执行。

功能: 检查 JavaScript 代码并执行。

实现过程

(1) 在页面中添加两个层。代码如下。

```
<div id="divleft"></div>
```

```
<div id="divtop"></div>
```

(2) 编写用于实现文字阴影的JavaScript代码。

自定义函数move(), 通过对鼠标的移动来改变两个层的位置。

```
<script language="JavaScript">
```

```
function move() {
```

```
    if (event.clientX<document.body.clientWidth-2)
```

```
        divtop.style.pixelLeft=document.body.scrollLeft+event.clientX+1;
```

```
    else divtop.style.pixelLeft=document.body.clientWidth-2;
```

```
    if (event.clientY<document.body.clientHeight-2)
```

```
        divleft.style.pixelTop=document.body.scrollTop+event.clientY+1;
```

```
}
```

自定义函数setstyle(), 对层样式进行初始化设置。

```
function setstyle(n) {
```

```
    varname=eval(n).style;
```

```
    name.position="absolute";
```

```
    name.left="0";
```

```
    name.top="0";
```

```
if (n=="divleft"){
    name.height="2px";
    name.width=document.body.clientWidth;
}
if (n=="divtop"){
    name.width="2px";
    name.height=document.body.clientHeight;
}
name.backgroundColor="#6699cc";
name.fontSize="0px";
}
```

当鼠标在页面上移动时，调用自定义函数move()。

```
document.onmousemove=move;
```

在窗体载入时，调用自定义函数setstyle()。

```
setstyle("divleft");
```

```
setstyle("divtop");
```

```
</script>
```

### 举一反三

根据本实例，读者可以：

用十字光标在地图上进行搜索；

在页面中制作可以移动的网格线。

## 第10章 状态栏特效

状态栏文字动画

其他

## 10.1 状态栏文字动画

不论在应用程序中还是在网页中都会应用到状态栏，状态栏用于为用户提供一些提示信息。如当前系统时间、软件版本、链接地址、系统信息以及一些自定义信息等。下面将通过一些实例来优化状态栏的显示效果。

### 实例281 使状态栏中的文字不停闪烁

本实例是一个人性化的实例

实例位置：光盘\mingrisoft\10\281

实例说明

本实例在状态栏中使用 JavaScript 实现一个文字不停闪烁的效果，当页面显示后状态栏中的文字将会不停闪烁，如图10.1、图10.2所示。



图10.1 使状态栏中的文字不停闪烁1



图10.2 使状态栏中的文字不停闪烁2

### 技术要点

本实例主要使用if语句对一个布尔类型的变量进行判断，当布尔变量的值为true时在状态栏中显示“欢迎来到明日科技软件网”，而当布尔变量的值为false时则在状态栏中显示空。然后调用window对象的setTimeout()方法，每间隔一秒钟就执行一次show()函数。

### 实现过程

(1) 编写在状态栏中文字不停闪烁的效果的程序，其代码如下。

```
<script language="javascript">
var text= " 欢迎来到明日科技软件网 ";
var timer=1000;
var bool= true;
function show()
{
if (bool==true)
{
window.status= showtext;
bool= false;
}
else
{
```

```
    window.status= "";
    bool= true;
}
window.setTimeout("show()", timer);
}
</script>
```

(2) 当页面加载时调用show()函数，其代码如下。

```
<body onLoad="show();">
```

举一反三

根据本实例，读者可以：

实现在指定的时间内使状态栏中显示不同的信息。

## 实例282 文字从右向左依次弹出

本实例可以增强动感效果

实例位置：光盘\mingrisoft\10\282

实例说明

本实例在状态栏中使用JavaScript实现一个文字从右向左依次弹出的效果，当页面显示后状态栏中的文字将会从右边向左边一个一个的弹出，文字在状态栏中全部输出完毕后，程序将会清空状态栏中的文字，然后重复执行文字从右向左依次弹出的操作，如图 10.3、图 10.4所示。



图10.3 文字从右向左依次弹出1



图10.4 文字从右向左依次弹出2

### 技术要点

本实例主要通过window对象的setTimeout()方法动态地改变状态栏中显示文字的位置。

说明：改变状态栏中的文字可以通过window对象的状态(status())方法实现。

### 实现过程

编写用于实现文字从右向左快速依次弹出的效果的程序，其代码如下。

```
<script language="JavaScript">
var message = " 欢迎来到明日科技主页，请您提出宝贵意见！" //状态栏信息
var position=150 //位置
var delay =10 //弹出文字的间隔时间
var statusobj=new statusMessageObject()
function statusMessageObject(p,d)
{
    this.msg=message;
    this.out= " ";
    this.pos=position;
    this.delay=delay;
    this.i=0;
    this.reset= clearMessage;
}
function clearMessage() //清空信息
{
    this.pos=POSITION;
}
function brush()
{
```

```

    for (statusobj.i=0; statusobj.i< statusobj.pos;
statusobj.i++)
    {
        statusobj.out+= " ";
    }
    if (statusobj.pos>=0)
        statusobj.out+= statusobj.msg;
    else
        statusobj.out= statusobj.msg.substring(-
statusobj.pos, statusobj.msg.length);
        window.status= statusobj.out;
        statusobj.out= " ";
        statusobj.pos--;
    if (statusobj.pos<-(statusobj.msg.length))
    {
        statusobj.reset();
    }
    setTimeout ('brush()', statusobj.delay);
}
function outtext(space, position)
{
    varmsg= statusobj.msg;
    varout= "";
    for (var i=0; i<position; i++)
    {
        out+=msg.charAt(i);
    }
}

```

```

for (i=1;i<space;i++)
{
    out+= " ";
}
out+=msg.charAt(position);
window.status= out;
if (space<=1)
{
    position++;
    if (msg.charAt(position)==' '){
        position++; }
    space=100-position;
}
else if (space> 3)
{
    space *= .75;
}
else
{
    space--;
}
if (position !=msg.length)
{
    var cmd= "outtext("+space+", "+position+)";
    scrollID=window.setTimeout(cmd, statusobj.delay);
}
else

```

```
{
window.status="";
space=0;
position=0;
cmd= "outtext("+space+", "+position+")";
scrollID=window.setTimeout(cmd, statusobj.delay);
return false
}
return true
}
outtext(100,0);
</script>
```

举一反三

根据本实例，读者可以：

实现从右向左依次弹出系统时间；

使一段字符串在状态栏中左右弹动。

## 实例283 文字从中间向两边展开

本实例是一个人性化的实例

实例位置：光盘\mingrisoft10\283

实例说明

本实例在状态栏中使用 JavaScript 实现一个文字从中间向两边展开的效果。当页面显示后，状态栏中的文字将会从中间开始向两边展开，然后逐渐显示所有的文字，如图 10.5、图10.6所示。



图10.5 文字从中间向两边展开1



图10.6 文字从中间向两边展开2

### 技术要点

本实例主要应用了字符串处理中的`charAt()`方法。该方法用于返回指定索引位置处的字符。语法格式如下。

`Strobj.charAt(index);`

参数说明如下。

- `Strobj`: 任意 `string` 对象或文字。

● index: 想得到的字符基于0 的索引。有效值是 0 与字符串长度减 1 之间的值。

例如:

```
function charAtTest(n) {  
    var str= "ABCDEFGHJKLMNOPQRSTUVWXYZ";    //初始化变量  
    var s;                                     //声名变量  
    s=str.charAt(n-1);                         //从索引为n  
- 1的位置处  
    //获取正确的字符  
    return(s);                                //返回字  
符  
}
```

实现过程

(1) 编写用于实现状态栏中的文字从中间向两边展开的效果的程序, 其代码如下。

```
<SCRIPTLANGUAGE="JavaScript">  
var showtext= " 欢迎来到明日科技主页, 请您提出宝贵意  
见! "; //状态栏信息  
var textlength= showtext.length; //获得状态栏字符的长度  
var outtext= "";  
var i=0;  
var j=0;  
//使状态栏文字从中间向两边展开  
functiondiffuse () {  
    i++;  
    outtext= "";  
    for (j=1; j<=(textlength/2)-i; j++) {
```

```

        outtext+= " ";
    }
    for (j=1; j<=i; j++) {
        outtext+= showtext.charAt(j-1);
    }
    for (j=i; j>=1; j--) {
        outtext+= showtext.charAt(textlength-j);
    }
    for (j=1; j<=(textlength/2)-i; j++) {
        outtext+= " ";
    }
    window.status= outtext;
    if (i<textlength/2) {
        var tim=setTimeout("diffuse ()",200);
    }
    else{
        i=j=0;
        window.status=outtext= "";
        clearTimeout(tim);
        diffuse ();
    }
}
</SCRIPT>

```

(2) 当页面加载时调用diffuse ()函数。其代码如下。

```
<body OnLoad="diffuse()">
```

举一反三

根据本实例，读者可以：

实现文字从两边向中间收缩。

## 实例284 文字跑马灯特效

本实例可以展现滚动的文字

实例位置：光盘\mingrisoft\10\284

实例说明

文字跑马灯的特效与屏幕中的滚动字幕十分类似，例如在网站的公告栏中就会显示一个这样的特效。本实例将在状态栏中实现这一功能，当页面显示后状态栏中的文字将会从右向左连续滚动，然后重复执行，如图10.7、图10.8所示。



图10.7 文字跑马灯特效1



图10.8 文字跑马灯特效2

### 技术要点

本实例主要应用了字符串处理中的substring()方法。该方法用于返回string对象中指定位置的子字符串。

语法格式如下。

```
StrObj.substring(start , end);
```

参数说明如下。

- start: 指明子字符串的起始位置, 该索引从 0 开始起算。
- end: 指明子字符串的结束位置。

说明: substring()方法将返回一个包含从 start 到最后 (不包含 end) 的子字符串。如果 start或 end 为 NaN 或者负数, 那么将其替换为0。

例如:

```
functionSubstringDemo() {  
    var str; //声明变量  
    var str= "The rain inSpain fallsmainly in theplain..";  
    str= str.substring(12,17); //取子字符串
```

```
    return(ss);                //返回子字符串
}
```

## 实现过程

(1) 编写用于实现文字跑马灯特效的程序，其代码如下。

```
<script language="JavaScript">
function runtext(s)
{
var text= " 你好！欢迎来到明日图书销售网，请您提出宝贵意见。 E-mail:mingrisoft@mingrisoft.com  "
var outtext= " ";
var i=1;
if (s>100)
{
    s--;
    var out="runtext("+s+"");
    var timerTwo=window.setTimeout(out,90);
}
else if (s<=100&& s>0)
{
    for (i=0;i<s;i++)
    {
        outtext+=" ";
    }
    outtext+=showtext;
    s--;
    window.status=outtext;
    out="runtext("+s+"");
```

```

        timerTwo=window.setTimeout(out, 90);
    }
    else if (s<=0)
    {
        if (-s< showtext.length)
        {
            outtext+=showtext.substring(-s, showtext.length);
            s--;
            window.status=outtext;
            out="runtext("+s+")";
            timerTwo=window.setTimeout(out, 90);
        }
        else
        {
            window.status=" ";
            timerTwo=window.setTimeout("runtext(100)", 90);
        }
    }
}
</SCRIPT>

```

(2) 当页面加载时调用runtext()函数，其代码如下。

```
<body onLoad="runtext(100);">
```

举一反三

根据本实例，读者可以：

制作滚动的字幕。

## 实例285 文字依次显示后快速收缩

本实例是一个人性化的实例

实例位置：光盘\mingrisoft\10\285

实例说明

当页面显示后，状态栏中的文字将会依次出现，首先在状态栏中将会显示“欢”字，然后依次显示其他文字。当文字全部显示完毕之后，全部文字将会向左快速收缩，然后重复执行此操作，如图10.9、图10.10所示。



图10.9 文字依次显示后快速收缩1



## 图10.10 文字依次显示后快速收缩2

### 技术要点

本实例中主要应用了字符串对象的 `length` 属性和 `substring()` 方法。本实例中调用 `length` 属性来获取字符串的长度，然后利用 `substring()` 方法以指定的个数输出字符串。有关 `substring()` 方法的参数说明请参见本章实例284。

### 实现过程

(1) 编写用于实现文字依次显示后快速收缩的效果的程序，其代码如下。

```
<script language="javascript">
var showtext="欢迎您来WL物流信息网 ";
var point=1;
function input()
{
    window.status= showtext.substring(0,point);
    if (point>= showtext.length)
    {
        point=1;
        window.setTimeout("output()",300);
    }
    else {
        point++;
        window.setTimeout("input()",100);
    }
}
function output()
{
```

```
        window.status=showtext.substring(point,
showtext.length);
        if (point>= showtext.length)
        {
            point=1;
            window.setTimeout("input()", 100);
        }
        else {
            point++;
            window.setTimeout("output()", 100);
        }
    }
</script>
```

(2) 当向页面加载时调用output()函数，其代码如下。

```
<body onLoad="output();">
```

举一反三

根据本实例，读者可以：

实现状态栏的打字效果。

## 实例286 文字的展开与收缩

本实例是一个人性化的实例

实例位置：光盘\mingrisoft\10\286

实例说明

运行本实例，状态栏中的文字将会依次出现。当显示完毕之后，全部文字将会从右向左快速收缩，然后重复执行此操作，如图10.11、图10.12所示。



图10.11 文字的展开与收缩1



图10.12 文字的展开与收缩2

### 技术要点

本实例的设计思想是先声明两个变量 `comeback` 和 `cometo`，主要用于存储在状态栏中显示字符的长度。然后程序会判断 `comeback` 的值

是否等于 1。如果等于1则 cometo的值每次减 1，否则 cometo 的值每次加 1。最后使用 substring( )方法按cometo 的值取得在状态栏中显示的字符。

### 实现过程

编写用于实现在状态栏中收缩显示文字的效果的程序，其代码如下。

```
<ScriptLanguage="JavaScript">
var showtext="欢迎访问明日科技主页：www.mingrisoft.com";
comeback=0;
cometo=0;
function rolltext() {
window.status=showtext.substring(0,cometo);
if(comeback==1) {
    cometo--;
    if(cometo==0) {comeback=0}
}
else {
    cometo++;
    if(cometo==showtext.length)
        {comeback=1}
}
window.status=showtext.substring(0,cometo)+" \ (^0^ )";
if(cometo==showtext.length) {
window.setTimeout("rolltext()",1000)}
else {
window.setTimeout("rolltext()",80)}
}
```

```
rolltext();
```

```
</Script>
```

举一反三

根据本实例，读者可以：

使状态栏收缩显示系统日期；

使状态栏从两边收缩显示信息。

## 实例287 状态栏中的文字依次弹出

本实例可以增强动感效果

实例位置：光盘\mingrisoft\10\287

实例说明

当页面显示后，状态栏中的文字将会依次显示在状态栏中。运行本实例，如图 10.13、图10.14所示。



图10.13 状态栏中的文字依次弹出1



图10.14 状态栏中的文字依次弹出2

### 技术要点

本实例首先使变量textlength获取字符串的长度，然后用另一个变量getlength与其对比，如果变量getlength的长度不大于或等于textlength的值，则按次序输出所有字符串。

### 实现过程

编写用于实现状态栏中的文字依次弹出的效果的程序，其代码如下。

```
<script language="JavaScript">
    var text = "欢迎来到明日科技主页，请您提出宝贵意见！";
    varTimer=200;
    vargetlen=0;
    functionScroll()
    {
        window.status= "";
        var len= text.length;
        window.status= text.substring(0, getlen+1);
```

```
getlen++;  
if ( getlen>= len )  
{  
getlen=0;  
window.status= '' ;  
window.setTimeout("Scroll()",Timer );  
}  
else  
window.setTimeout("Scroll()",Timer );  
}  
Scroll();  
</script>
```

### 举一反三

根据本实例，读者可以：

实现页面中逐个显示广告信息的效果；

制作文本浏览器时应用本技巧浏览文本内容。

## 10.2 其他

### 实例288 在状态栏显示固定自定义信息

本实例是一个固定状态栏信息的实例

实例位置：光盘\mingrisoft\10\288

实例说明

默认情况下，当鼠标移动到页面中的超级链接时，状态栏中就会显示链接地址或是其他信息。如果用户只想在状态栏中显示指定的自定义信息，那么可以通过本例来解决这一问题。运行本实例，无论将鼠标移动到哪个超级链接上，在状态栏中都将显示“<==吉林省明日科技==>”，如图10.15所示。



图10.15 在状态栏显示固定自定义信息

技术要点

本实例中主要应用了window对象的状态属性，通过该属性可以设置状态栏中的显示信息。

## 实现过程

(1) 实现在状态栏中显示固定的自定义信息，其代码如下。

```
<script language="javascript">
function text()
{
    window.status= "<==吉林省明日科技==>";
}
document.onmouseover= text;
</script>
```

(2) 添加页面设计代码，并在超级链接的onMouseOver事件中调用text()函数，其代码如下。

```
<bodyvlink="#0066FF">
<div align="center">
<p><a href="#" style="text-decoration:none
"onMouseMove="text();">首页&nbsp;|</a>
    <a href="#" style="text-decoration:none
"onMouseMove="text();">图书介绍&nbsp;|</a>
    <a href="#" style="text-decoration:none
"onMouseMove="text();">新书预告&nbsp;|</a>
    <a href="#" style="text-decoration:none
"onMouseMove="text();">勘误发布&nbsp;|</a>
    <a href="#" style="text-decoration:none
"onMouseMove="text();">技术支持&nbsp;|</a>
    <a href="#" style="text-decoration:none
"onMouseMove="text();">明日论坛</a>
</p>

```

</p>

</div>

</body>

举一反三

根据本实例，读者可以：

实现当鼠标移动到链接上时，在状态栏中显示不同的提示信息。

## 实例289 在状态栏显示鼠标坐标

本实例是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\10\289

实例说明

我们制作一些图形图像类的程序时，有时可能会用到鼠标的坐标位置，例如提取某个坐标位置上的颜色值等。下面在状态栏中使用JavaScript 编写一个显示鼠标坐标的实例。运行本实例，当鼠标在页面中移动时，状态栏中将会显示鼠标的坐标位置，如图10.16所示。



图10.16 在状态栏显示鼠标坐标

技术要点

本实例主要调用window对象中event的clientX属性和clientY属性来获取当前的鼠标坐标。

- clientX：返回鼠标单击点相对于窗口工作区的水平位置。

● clientY: 返回鼠标单击点相对于窗口工作区的垂直位置。

实现过程

(1) 在函数体中声明两个变量, 用其获得鼠标的当前坐标, 代码如下。

```
<script language="javascript">
function getpoint()
{
    varx,y;
    if (document.all)
    {
        x=window.event.clientX;
        y=window.event.clientY;
    }
    window.status= "X坐标为: "+x+" Y坐标为: "+y;
}
document.onmousemove=getpoint;
</script>
```

(2) 添加页面设计代码, 并在<body>标记中设置超级链接的颜色, 其代码如下。

```
<body alink="#FF0000" vlink="#3333FF">
<font color="#FFCC00">
    <a href="#" style="text-decoration:none ">首页 |
</a>
    <a href="#" style="text-decoration:none ">图书介绍
 |</a>
    <a href="#" style="text-decoration:none ">新书预告
 |</a>
```

```
<a href="#" style="text-decoration:none ">勘误发布  
&nbsp;|</a>  
<a href="#" style="text-decoration:none ">技术支持  
&nbsp;|</a>  
<a href="#" style="text-decoration:none ">明日论坛</a>  
</font>  
</body>
```

举一反三

根据本实例，读者可以：

获取页面中某一位置上的x和y坐标；

得到页面背景中的指定位置上的颜色值。

## 实例290 在状态栏显示特定的超级链接信息

本实例是一个应用较为广泛的实例

实例位置：光盘\mingrisoft\10\290

实例说明

默认情况下，当鼠标移动到一个超级链接上时，在状态栏上就会显示该超级链接的URL，而这些信息并不一定是浏览者想要知道的。为了有效利用资源，可以在状态栏上显示特定的说明信息，帮助浏览者快速了解超级链接的具体内容。本实例将介绍如何在状态栏上显示特定的超级链接信息。运行本实例，将鼠标移动到“软件下载”超级链接上，在状态栏上将显示说明信息“进入明日科技编程者之家[软件下载]页面！”，如图10.17所示。



图10.17 在状态栏显示特定的超级链接信息

### 技术要点

本实例主要通过 `onMouseMove` 和 `onMouseOut` 事件控制状态栏上的显示信息实现。`onMouseMove` 事件当鼠标在指定元素上移动时触发，`onMouseOut` 事件当鼠标离开指定元素时触发。控制状态栏上的显示信息可以通过window对象的状态属性实现。

### 实现过程

在页面的适当位置添加超级链接文字和图片，并为其设置超级链接，同时通过`onMouseMove`和`onMouseOut`事件控制状态栏的显示信息，关键代码如下。

```
<tdwidth="12%"align="center">
<a href="void(0);"onMouseMove="window.status=' 返回明日科技编程者之家首页!' "
="window.status=' 欢迎访问明日科技编程者之家网站!' ">
<br>
```

```
    首页</a></td>
    <tdwidth="7"align="center"></td>
    <tdwidth="12%"align="center">
    <ahref="void(0);"onMouseMove="window.status=' 进入明日科技
编程者之家 [软件下载 ]页面!' "
onMouseOut="window.status=' 欢迎访问明日科技编程者之家网
站!' ">
    <br>
    软件下载</a></td>
```

举一反三

根据本实例，读者可以：

在状态栏上显示图片的说明信息；

在状态栏上显示下载说明信息。

# 第11章 报表与打印

Web打印

利用Word打印报表

利用Excel打印报表

利用CSS样式打印

套打邮寄产品单

## 11.1 Web打印

Web打印是一种常用的打印方式，其使用方法简单、方便、快捷，在浏览网页的同时就可以实现打印的功能。

### 实例291 调用IE自身的打印功能实现打印

本实例是一个提高工作效率的程序

实例位置：光盘\mingrisoft\11\291

实例说明

通过JavaScript调用IE自身的打印功能实现打印，这种方法比较简单，也是常用的打印方式。使用该方法只须将要打印的页面设计好，再通过JavaScript的window对象的print()方法调用IE的打印功能即可。运行本实例，单击“打印”超级链接，会弹出打印对话框，如图11.1所示，然后进行相应的设置，并进行打印。



图11.1 调用IE自身的打印功能实现打印

### 技术要点

本实例主要通过调用window对象的打印方法print()来实现打印功能。

window对象的print()方法的语法格式如下。

window.print();

例如：

<a href="#" onClick="window.print()">打印</a>

### 实现过程

(1) 设计要打印的页面。

(2) 调用window对象的打印方法实现打印功能，代码如下。

<a href="#" onClick="window.print()">打印</a>

### 举一反三

根据本实例，读者可以：

开发打印库存统计报表；  
打印校友录中的同学联系方式。

## 实例292 打印指定框架中的内容

本实例是一个技巧性较强的程序

实例位置：光盘\mingrisoft\11\292

实例说明

在实例292中介绍了如何通过JavaScript调用IE自身的打印功能实现打印，但是该方法将打印页面中的全部内容，包括“打印”超级链接，有时这是不需要的。本实例将介绍如何通过打印指定框架中的内容实现页面部分内容的打印。运行本实例，如图11.2所示，单击“打印”超级链接，会弹出打印对话框，进行相应的设置后，单击“打印”按钮即可对图11.2中的表格内容进行打印。

鸿博图书馆管理系统

刷新页面 关闭系统

当前登录用户：mr

[2007年03月02日 星期五 16:25:02] 首页！系统设置！读者管理！图书管理！图书借还！系统查询！排行榜！更改口令！退出系统

当前位置：系统查询 > 借阅信息打印 >>> 打印

图书条形码	图书名称	读者条形码	读者名称	借阅时间	应归时间	是否归还
7115145474	JSP数据库系统开发完全手册	123456789	无*语	2007-01-01	2007-02-01	已归还
7115145474	JSP数据库系统开发完全手册	123456789	无*语	2007-02-04	2007-03-04	未归还
7115148732	ASP程序开发范例宝典	123456789	无*语	2006-12-20	2007-01-20	已归还
7115148732	ASP程序开发范例宝典	123456789	无*语	2007-02-01	2007-03-01	未归还

Copyright © 2007 www.mingrisoft.com 吉林省明日科技有限公司  
本站请使用IE6.0或以上版本 1024\*768为最佳显示效果

图11.2 打印指定框架中的内容

技术要点

在实现打印指定框架中的内容时，首先让需要打印的框架获得焦点，然后再调用 window对象的print()方法实现打印。

语法：

```
parent.mainFrame.focus();
```

```
window.print();
```

参数说明如下。

● mainFrame：表示框架的名称。

实现过程

(1) 设计content.htm页面，将要打印的内容放置在该页面中。

(2) 设计 index.htm 页面，在该页面的适当位置添加浮动框架（应用<iframe>标记创建），并将该浮动框架的src属性指定为步骤(1)中创建的content.htm文件，代码如下。

```
<iframe name="contentFrame" src="content.htm"
frameborder="0" width="100%" height="100%"></iframe>
```

(3) 在index.htm页面中添加“打印”超级链接，打印指定浮动框架中的内容，代码如下。

```
<a href="#"
onClick="parent.contentFrame.focus();window.print();">打印
</a>
```

举一反三

根据本实例，读者可以：

开发打印库存盘点报表；

实现在办公自动化系统中加入打印考勤明细表功能。

## [实例293 利用WebBrowser打印](#)

本实例是一个方便操作、提高效率的程序

实例位置：光盘\mingrisoft\11\293

## 实例说明

WebBrowser是IE内置的浏览器控件，无须用户下载。它的优点是客户端独立完成打印目标文档的生成，减轻服务器负荷；缺点是源文档的分析操作复杂，并且要对源文档中要打印的内容进行约束。运行本实例，单击“打印预览”超级链接，即可打开“打印预览”对话框，如图11.3所示，单击“打印”超级链接，即可打开打印对话框进行打印。

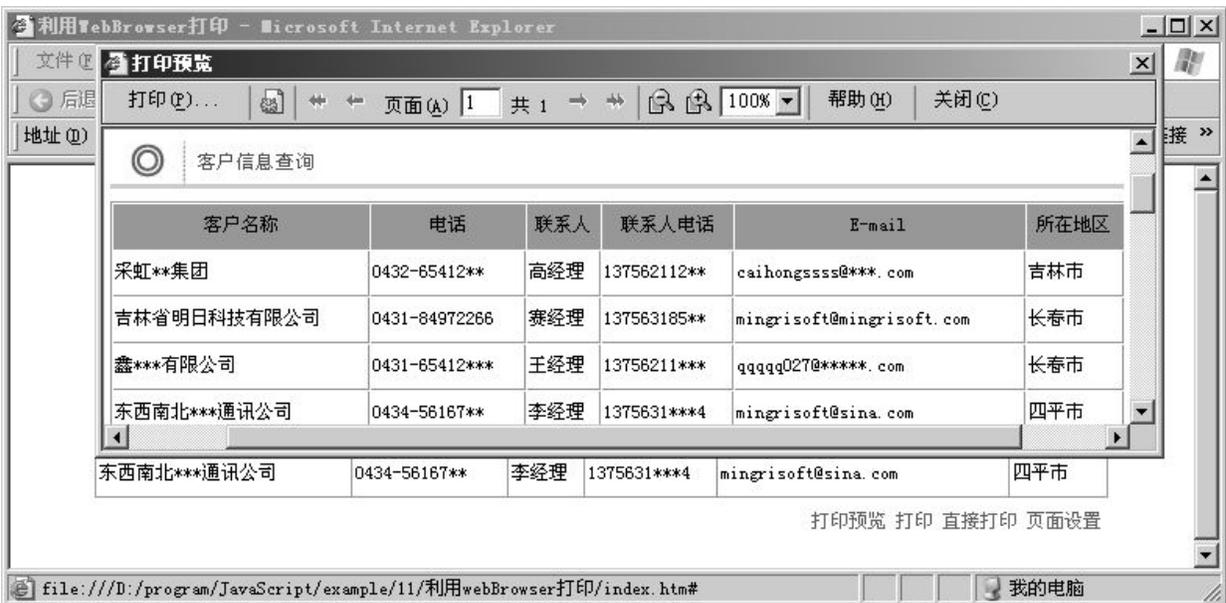


图11.3 利用WebBrowser打印

## 技术要点

本实例主要应用IE内置的WebBrowser控件实现，该控件的具体参数如下。

- document.all.WebBrowser.Execwb(7,1)：表示打印预览。
- document.all.WebBrowser.Execwb(6,1)：表示打印。
- document.all.WebBrowser.Execwb(6,6)：表示直接打印。
- document.all.WebBrowser.Execwb(8,1)：表示页面设置。

## 实现过程

(1) 建立HTML的Object标签，调用WebBrowser控件，代码如下。

```
<object id="WebBrowser" classid="CLSID:8856F961-340A-11D0-A96B-00C04Fd705A2" width="0" height="0">
</object>
```

(2) 建立相关的打印超级链接，并调用 WebBrowser 控件的相应参数实现打印预览、打印等功能。代码如下。

```
<a href="#"
onClick="document.all.WebBrowser.Execwb(7,1)">打印预览</a> <a
href="#"
onClick="document.all.WebBrowser.Execwb(6,1)">打印</a> <a
href="#" onClick="document.all.WebBrowser.Execwb(6,6)">直接打
印</a> <a href="#"
onClick="document.all.WebBrowser.Execwb(8,1)">页面设置</a>
```

举一反三

根据本实例，读者可以：

实现在网站中打印各种报表功能；

实现在网站中打印单据功能。

## 实例294 设置页眉页脚

本实例是一个方便操作、人性化的程序

实例位置：光盘\mingrisoft\11\294

实例说明

在系统默认的情况下，无论是利用 IE 自身的打印功能进行打印，还是利用 IE 自带的WebBrowser控件进行打印，在打印的文档顶部和底部会包括页眉和页脚，有时并不需要打印默认的页眉和页脚。本实例将介绍如何清空页眉页脚和恢复页眉页脚。运行本实例，单击

“清空页眉页脚”超级链接，即可清空IE默认的页眉页脚，这时再单击“打印预览”超级链接，在打开的“打印预览”窗口中将不显示IE默认的页眉页脚，如图11.4所示，单击“恢复页眉页脚”超级链接即可恢复页眉页脚的显示。



图11.4 设置页眉页脚

### 技术要点

本实例主要通过WshShell对象的相关方法实现。WshShell对象是WSH（WSH是Windows Scripting Host的缩写，内嵌于Windows操作系统中的脚本语言工作环境）的内建对象，主要负责程序的本地运行、处理注册表、创建快捷方式、获取系统文件夹信息及处理环境变量等工作。WshShell对象的相关方法如表11.1所示。

表11.1 WshShell对象的相关方法

方 法	说 明
CreateShortcut	创建并返回 WshShortcut 对象
ExpandEnvironmentStrings	扩展 PROCESS 环境变量并返回结果字符串
Popup	显示包含指定消息的消息窗口
RegDelete	从注册表中删除指定的键或值
RegRead	从注册表中返回指定的键或值
RegWrite	在注册表中设置指定的键或值
Run	创建新的进程，该进程用指定的窗口样式执行指定的命令

本实例中主要应用了 RegWrite 方法，下面将对该方法进行详细介绍。

RegWrite方法用于在注册表中设置指定的键或值，其语法格式如下。

```
WshShell.RegWrite strName, anyValue, [strType]
```

参数说明如下。

- strName: 用于指定注册表的键或值，若strName以一个反斜杠(在JavaScript中为\\) 结束，则该方法设置键，否则设置值。strName 参数必须以根键名“HKEY\_CURRENT\_USER”、“HKEY\_LOCAL\_MACHINE”、“HKEY\_CLASSES\_ROOT”、“HKEY\_USERS”或“HKEY\_CURRENT\_CONFIG”开头。

- AnyValue: 用于指定注册表的键或值的值。当strType 为 REG\_SZ 或REG\_EXPAND\_SZ时，RegWrite 方法自动将 anyValue 转换为字符串。若 strType 为 REG\_DWORD，则anyValue被转换为整数。若strType为REG\_BINARY，则anyValue必须是一个整数。

- StrType: 用于指定注册表的键或值的数据类型。RegWrite 方法支持的数据类型为REG\_SZ、REG\_EXPAND\_SZ、REG\_DWORD和REG\_BINARY。其他的数据类型被作为strType 传递，RegWrite 返回E\_INVALIDARG。

实现过程

(1) 编写自定义JavaScript函数PageSetup\_del()和PageSetup\_set(), 用于实现清空页眉页脚和恢复页眉页脚的功能。具体代码如下。

```
<script language="JavaScript">
    varHKEY_RootPath="HKEY_CURRENT_USER\\Software\\Microsoft\\
\\InternetExplorer\\PageSetup\\";
    functionPageSetup_del() { //清空页眉页脚
        try{
            varWSc=newActiveXObject("WScript.Shell");
            HKEY_Key="header";
            WSc.RegWrite(HKEY_RootPath+HKEY_Key, "");
            HKEY_Key="footer";
            WSc.RegWrite(HKEY_RootPath+HKEY_Key, "");
        } catch(e) {}
    }
    function PageSetup_set() { //恢复页眉页脚
        try{
            varWSc=newActiveXObject("WScript.Shell");
            HKEY_Key="header";
            WSc.RegWrite(HKEY_RootPath+HKEY_Key, "&w&b页
码, &p/&P");
            HKEY_Key="footer";
            WSc.RegWrite(HKEY_RootPath+HKEY_Key, "&u&b&d");
        } catch(e) {}
    }
</script>
```

(2) 建立HTML的Object标签, 调用WebBrowser控件, 代码如下。

```
<object id="WebBrowser" classid="CLSID:8856F961-340A-11D0-A96B-00C04Fd705A2" width="0" height="0">
```

```
</object>
```

(3) 创建“清空页眉页脚”和“恢复页眉页脚”的超级链接，并调用自定义函数PageSetup\_del()和PageSetup\_set()实现相应功能。代码如下。

```
<a href="#" onClick="PageSetup_del()">清空页眉页脚</a>
```

```
<a href="#" onClick="PageSetup_set()">恢复页眉页脚 </a>
```

(4) 建立相关的打印超级链接，并调用 WebBrowser 控件的相应参数实现打印预览、打印等功能。代码如下。

```
<a href="#"
```

```
onClick="document.all.WebBrowser.Execwb(7,1)">打印预览</a>
```

```
<a href="#"
```

```
onClick="document.all.WebBrowser.Execwb(6,1)">打印</a>
```

```
<a href="#"
```

```
onClick="document.all.WebBrowser.Execwb(6,6)">直接打印</a>
```

```
<a href="#"
```

```
onClick="document.all.WebBrowser.Execwb(8,1)">页面设置</a>
```

举一反三

根据本实例，读者可以：

实现在任何打印页面中添加设置页眉页脚功能；

实现自定义页眉页脚的内容功能。

## 11.2 利用Word打印报表

Microsoft Word 是Microsoft 提供的文档处理软件，它在处理、打印文档和资料过程中所显现出来的强大功能是有目共睹的。本节将介绍如何将网页中的数据直接导出到 Word 中，直接进行打印或处理后再进行打印。

### 实例295 将页面中的表格导出到Word并打印

本实例是一个可以提高效率的程序

实例位置：光盘\mingrisoft\11\295

实例说明

在开发动态网站时，经常会遇到打印页面中的指定表格的情况，这时可以将要打印的表格导出到Word中，然后再打印。本实例将介绍如何将页面中的订单列表导出到Word并打印。运行本实例，在页面中将显示订单信息列表，单击“打印”超级链接后，将把Web页中的数据导出到Word的新建文档中，如图11.5所示，并保存在Word的默认文档保存路径中，最后调用打印机打印该文档。

技术要点

本实例主要应用JavaScript 的ActiveXObject() 构造函数创建一个OLE Automation(ActiveX)对象的实例，并应用该实例的相关方法实现。

ActiveXObject() 构造函数的一般语法格式如下。

```
var objectVar = new ActiveXObject(class[, servername]);
```

参数说明如下。

- objectVar: 用于指定引用对象的变量。
- class: 用于指定应用程序的名字或包含对象的库，并且指定要创建的对象类型。采用library.object的语法格式，如“Word.Application”表明要创建的是Word对象。
- servername: 可选参数，用于指定包含对象的网络服务器的名字。



图11.5 将页面中的表格导出到Word并打印

说明：每个支持自动化的应用程序都至少提供一种对象类型。例如一个字处理应用程序可能会提供Application对象、Document对象以及Toolbar对象等。

### 实现过程

(1) 将显示订单信息的表格的id设置为order，因为要打印该表格中的数据。关键代码如下。

```
<table id="order" width="100%" height="48" border="1"
cellpadding="0" cellspacing="0"
bordercolor="#FFFFFF"bordercolordark="#CCCCCC"
bordercolorlight="#FFFFFF">
```

(2) 编写自定义JavaScript函数outDoc(), 用于将Web页面中的订单信息导出到Word中, 并进行自动打印, 代码如下。

```
<script language="javascript">
function outDoc() {
    var table=document.all.order;
    row=table.rows.length;
    column=table.rows(1).cells.length;
    var wdapp=newActiveXObject("Word.Application");
    wdapp.visible=true;
    wddoc=wdapp.Documents.Add(); //添加新的文档
    thearray=newArray();
//将页面中表格的内容存放在数组中
for(i=0;i<row;i++){
    thearray[i]=newArray();
    for(j=0;j<column;j++){
        thearray[i][j]=table.rows(i).cells(j).innerHTML;
    }
}
var range=wddoc.Range(0,0);
range.Text="订单信息列表"+"\\n";
wdapp.Application.ActiveDocument.Paragraphs.Add(range);
wdapp.Application.ActiveDocument.Paragraphs.Add();
```

```

        rngcurrent=wdapp.Application.ActiveDocument.Paragraphs(3)
        .Range;
        var
objTable=wddoc.Tables.Add(rngcurrent,row,column)    //插入表
格
        for(i=0;i<row;i++){
            for(j=0;j<column;j++){
                objTable.Cell(i+1,j+1).Range.Text= thearray[i]
[j].replace("&nbsp;","");
            }
        }
        wdapp.Application.ActiveDocument.SaveAs("orderInfo.doc",0
,false,"",true,"",false,false,false,false,false);    //保存到
Word的默认文档保存路径中
        wdapp.Application.Printout();    //自动打印
        wdapp=null;
    }
</script>

```

技巧：在Word中查看并修改默认文档保存路径的方法如下：选择“工具”→“选项”选项，在弹出的对话框中选择“文件位置”选项卡，在该选项卡中选中“文档”列表项，单击“更改”按钮，在弹出的对话框中选择默认文档保存路径，连续单击“确定”按钮即可。

(3) 通过单击“打印”超级链接调用自定义JavaScript函数outDoc()。关键代码如下。

```
<a href="#" onClick="outDoc();">打印</a>
```

举一反三

根据本实例，读者可以：

在企业进销存管理系统中打印各种报表；  
将Web页面中的数据自动保存到Word中。

## 实例296 打开指定的Word文档并打印

本实例是一个提高工作效率的程序

实例位置：光盘\mingrisoft\11\296

实例说明

在制作网站时，有时需要打开指定的Word文档并打印该文档。运行本实例，单击“浏览”按钮，打开“选择文件”对话框，在该对话框中选择要打印的Word文档，单击“打开”按钮，返回到图11.6所示的“打开指定的Word文档并打印”页面，单击“打开Word并打印”按钮，将调用Word并自动打印选择的文档。

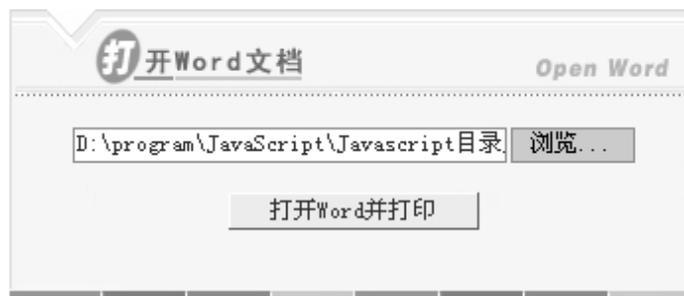


图11.6 打开指定的Word文档并打印

技术要点

实现利用JavaScript打开指定的Word文档并打印的思路如下。

- (1) 应用JavaScript的ActiveXObject()构造函数创建一个Word.Application对象的实例。
- (2) 激活刚刚创建的Word.Application对象的实例。
- (3) 通过Word.Application对象的Documents集合的Open()方法打开指定的Word文档。

(4) 调用 `wdapp.Application.Printout()` 实现自动打印Word文档。

### 实现过程

(1) 编写打开Word文档的JavaScript自定义函数 `openWord()`。代码如下。

```
<script language="javascript">
function openWord(filename) {
    try {
        var wrd=newActiveXObject("word.Application");
        wrd.visible=true;
        wrd.Documents.Open(filename);
        wrd.Application.Printout();
        wrd=null;
    }
    catch(e) {}
}
</script>
```

(2) 在页面的适当位置添加一个用于选择文件的文件域，名称为 `file1`，代码如下。

```
<input name="file1" type="file" class="textarea"
id="file1" size="35">
```

(3) 在“打开Word并打印”按钮的 `onClick` 事件中调用自定义JavaScript函数 `openWord()`，打开指定的Word文档并打印。关键代码如下。

```
<input name="Submit2" type="button" class="btn_grey"
onClick="openWord(file1.value)"
value="打开Word并打印">
```

举一反三

根据本实例，读者可以：

打开指定的Word文档修改后再打印。

## 实例297 在JSP中利用Word自动打印指定格式的会议记录

这是一个JSP与JavaScript结合的实例

实例位置：光盘\mingrisoft\11\297

实例说明

在开发网络应用程序时，有时需要对输入的信息按指定的格式进行打印。例如，在办公自动化系统中，录入的会议记录信息就需要按指定的格式打印。本实例将介绍如何在JSP中实现利用 Word 自动打印指定格式的会议记录。运行本实例，在页面中输入相应的会议信息，单击“Word 打印”按钮，即可将录入的会议信息导出到指定的 Word 文档中，并自动按该文档指定的格式打印。实例运行结果如图11.7所示。

技术要点

在JSP中实现利用Word自动打印指定格式的会议记录的步骤如下。

- (1) 应用JavaScript的ActiveXObject()构造函数创建一个Word.Application对象的实例。
- (2) 打开指定的Word文档，这里需要应用JSP的内置对象request的getRequestURL()方法获取模板文档所在路径。
- (3) 通过Word.Application对象的Bookmarks集合的相应方法将表单内容写入到指定的Word文档中。
- (4) 调用wdapp.Application.Printout()实现自动打印Word文档。

## 实现过程

(1) 创建一个 Word 文档，在该文档中设计好要打印的会议记录的格式，并将其保存到实例根目录下，名称为meetingDot.doc。

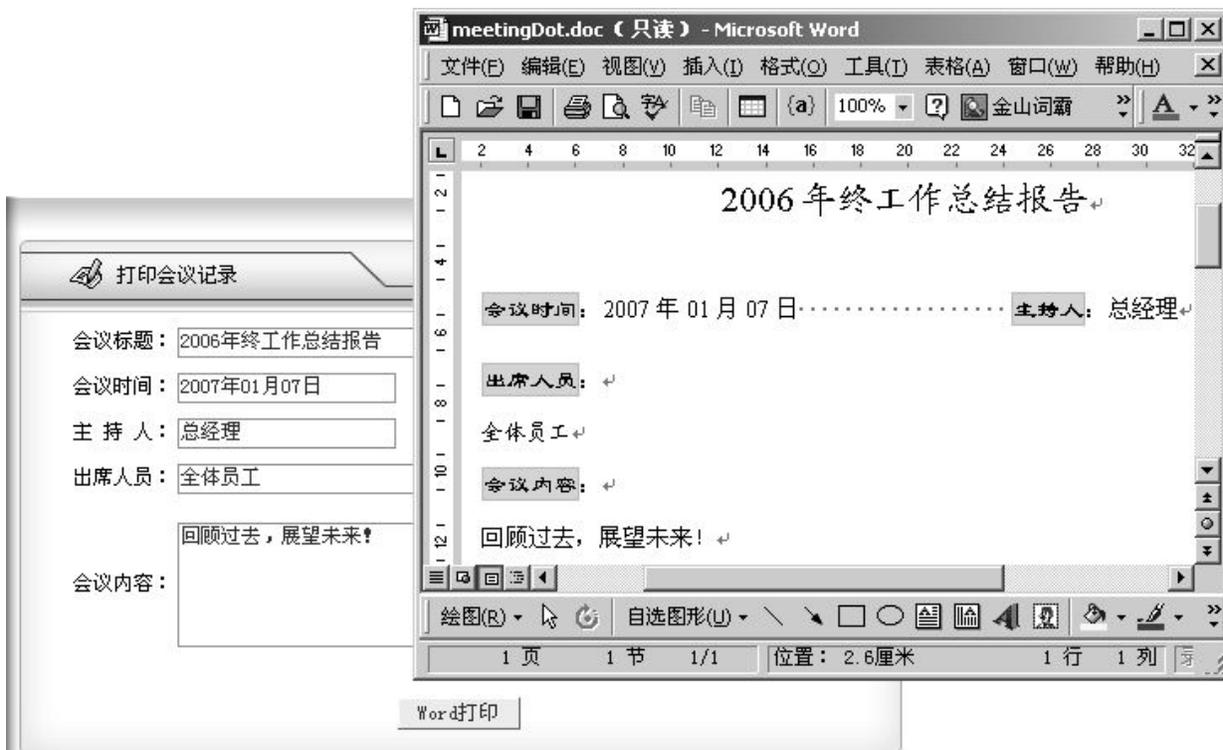


图11.7 在JSP中利用Word自动打印指定格式的会议记录

(2) 在创建好的 Word 文档中的指定位置插入书签。插入书签的方法如下：首先选中需要替换的文本，然后选择“插入”→“书签”选项，在打开的对话框中输入书签名，并单击“添加”按钮即可。

(3) 在实例主页面中添加用于收集会议信息的表单及表单元素，具体设置如表11.2所示。

表11.2 页面中所涉及的表单元素

名称	元素类型	重要属性	含义
form1	form	Method="post" action=""	表单
title	text	size="50"	会议标题
meetingTime	text		会议时间
compere	text		主持人
attend	text	size="58"	出席人员
content	text		会议内容

(4) 编写自定义JavaScript函数outDoc(), 用于将表单收集的数据导出到Word中, 并进行自动打印, 代码如下。

```
<%
//获取模板文档所在的路径
int at=request.getRequestURL().lastIndexOf("/");
Stringpath_part=request.getRequestURL().substring(0, at+1)
;
%>
<script language="javascript">
function outDoc() {
    varwdapp=newActiveXObject("Word.Application");
    wdapp.visible=true;
    wddoc=wdapp.Documents.Open("
<%=path_part%>meetingDot.doc");    //打开指定的文档
    var form=document.all.form1;
    title=form.title.value;
    meetingTime=form.meetingTime.value;
    compere=form.compere.value;
    attend=form.attend.value;
    content=form.content.value;
    //输出会议标题
    range=wdapp.ActiveDocument.Bookmarks("title").Range;
    range.Text=title;
    //输出会议时间
    range=wdapp.ActiveDocument.Bookmarks("meetingTime").Range;
ge;
    range.Text=meetingTime;
```

```
//输出会议主持人
range=wdapp.ActiveDocument.Bookmarks("compere").Range;
range.Text=compere;
//输出出席人员
range=wdapp.ActiveDocument.Bookmarks("attend").Range;
range.Text=attend;
//输出会议内容
range=wdapp.ActiveDocument.Bookmarks("content").Range;
range.Text=content;
wddoc.Application.Printout();
wdapp=null;
}
</script>
```

(5) 通过单击“Word打印”按钮调用自定义JavaScript函数outDoc()。关键代码如下。

```
<input name="Submit" type="button" class="btn_grey"
onClick="outDoc()" value="Word打印">
```

举一反三

根据本实例，读者可以：

在JSP中开发打印客户订单；

在JSP中开发打印各种样式的报表。

## 实例298 在ASP中利用Word自动打印指定格式的会议记录

这是一个ASP与JavaScript结合的实例

实例位置：光盘\mingrisoft\11\298

## 实例说明

在开发网络应用程序时，有时需要对输入的信息按指定的格式进行打印。例如，在办公自动化系统中，录入的会议记录信息就需要按指定的格式打印。本实例将介绍如何在ASP中实现利用 Word 自动打印指定格式的会议记录。运行本实例，在页面中输入相应的会议信息，单击“Word 打印”按钮，即可将录入的会议信息导出到指定的 Word 文档中，并自动按该文档指定的格式打印。实例运行结果如图11.8所示。



图11.8 在ASP中利用Word自动打印指定格式的会议记录

## 技术要点

在ASP中实现利用Word自动打印指定格式的会议记录的步骤如下。

(1) 应用JavaScript的ActiveXObject()构造函数创建一个Word.Application对象的实例。

(2) 打开指定的Word文档，这里需要应用ASP的内置对象request的ServerVariables()方法获取模板文档所在路径。

(3) 通过 Word.Application 对象的 Bookmarks 集合的相应方法将表单内容写入到指定的Word文档中。

(4) 调用wdapp.Application.Printout()实现自动打印Word文档。

### 实现过程

(1) 创建一个 Word 文档，在该文档中设计好要打印的会议记录的格式，并将其保存到实例根目录下，名称为meetingDot.doc。

(2) 在创建好的 Word 文档中的指定位置插入书签。插入书签首先选中需要替换的文本，然后选择“插入”→“书签”选项，在打开的对话框中输入书签名，并单击“添加”按钮即可。

(3) 在实例主页面中添加用于收集会议信息的表单及表单元素，具体设置如表11.3所示。

表11.3 页面中所涉及的表单元素

名称	元素类型	重要属性	含义
form1	form	Method="post" action=""	表单
title	text	size="50"	会议标题
meetingTime	text		会议时间
compere	text		主持人
attend	text	size="58"	出席人员
content	text		会议内容

(4) 编写自定义JavaScript函数outDoc()，用于将表单收集的数据导出到Word中，并进行自动打印，代码如下。

```
<%  
' 获取模板文档所在的路径  
path=mid(request.ServerVariables("URL"), 1, InStrRev(request.  
t.ServerVariables("URL"), "/"))  
path_part="HTTP://"&request.ServerVariables("LOCAL_ADDR")  
&":"&_
```

```

request.ServerVariables("SERVER_PORT")&path&"meetingDot.d
oc"
%>
<script language="javascript">
function outDoc() {
    var wdapp=newActiveXObject("Word.Application");
    wdapp.visible=true;
    wddoc=wdapp.Documents.Open("<%=path_part%>");    //打开
指定的文档
    var form=document.all.form1;
    title=form.title.value;
    meetingTime=form.meetingTime.value;
    compere=form.compere.value;
    attend=form.attend.value;
    content=form.content.value;
    //输出会议标题
    range=wdapp.ActiveDocument.Bookmarks("title").Range;
    range.Text=title;
    //输出会议时间
    range=wdapp.ActiveDocument.Bookmarks("meetingTime").Ran
ge;
    range.Text=meetingTime;
    //输出会议主持人
    range=wdapp.ActiveDocument.Bookmarks("compere").Range;
    range.Text=compere;
    //输出出席人员
    range=wdapp.ActiveDocument.Bookmarks("attend").Range;

```

```
range.Text=attend;
//输出会议内容
range=wdapp.ActiveDocument.Bookmarks("content").Range;
range.Text=content;
wddoc.Application.Printout();
wdapp=null;
}
</script>
```

(5) 通过单击“Word打印”按钮调用自定义JavaScript函数outDoc()。关键代码如下。

```
<input name="Submit" type="button" class="btn_grey"
onClick="outDoc()" value="Word打印">
```

举一反三

根据本实例，读者可以：

在ASP中开发打印客户订单；

在ASP中开发打印各种样式的报表。

## 实例299 在PHP中调用Word自动打印指定格式的会议记录

这是一个PHP与JavaScript结合的实例

实例位置：光盘\mingrisoft\11\299

实例说明

在开发网络应用程序时，有时需要对输入的信息按指定的格式进行打印。例如，在办公自动化系统中，录入的会议记录信息就需要按指定的格式打印。本实例将介绍如何在PHP中实现调用 Word 自动打印指定格式的会议记录。运行本实例，在页面中输入相应的会议信息，

单击“Word 打印”按钮，即可将录入的会议信息导出到指定的 Word 文档中，并自动按该文档指定的格式打印，本实例运行结果如图11.9 所示。

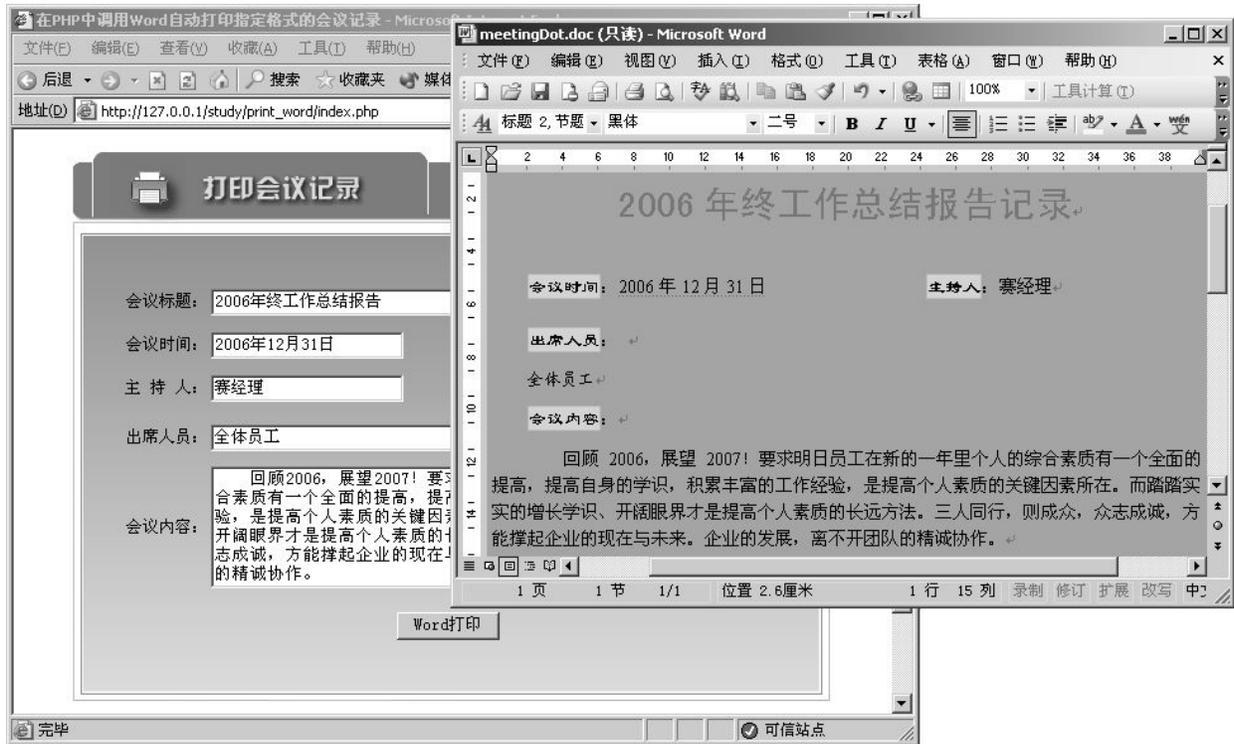


图11.9 在PHP中调用Word自动打印指定格式的会议记录

### 技术要点

在PHP中实现利用Word自动打印指定格式的会议记录的步骤如下。

(1) 应用JavaScript的ActiveXObject()构造函数创建一个Word.Application对象的实例。

(2) 打开指定的Word文档，这里需要应用PHP的函数获取模板文档所在路径，代码如下：

```
$path="http://". $HTTP_HOST. dirname($PHP_SELF). "/meetingDot.doc";
```

其中，\$HTTP\_HOST变量用于获取服务器名称，dirname(\$PHP\_SELF)用于获取路径中的路径名。

(3) 通过 Word.Application 对象的 Bookmarks 集合的相应方法将表单内容写入到指定的Word文档中。

(4) 调用wdapp.Application.Printout()实现自动打印Word文档。

### 实现过程

(1) 创建一个 Word 文档，在该文档中设计好要打印的会议记录的格式，并将其保存到实例根目录下，名称为meetingDot.doc。

(2) 在创建好的 Word 文档中的指定位置插入书签。插入书签首先选中需要替换的文本，然后选择“插入”→“书签”选项，在打开的对话框中输入书签名，并单击“添加”按钮即可。

(3) 在实例主页面中添加用于收集会议信息的表单及表单元素，具体设置如表11.4所示。

表11.4 页面中所涉及的表单元素

名称	元素类型	重要属性	含义
form1	form	Method="post" action=""	表单
title	text	size="50"	会议标题
meetingTime	text		会议时间
compere	text		主持人
attend	text	size="58"	出席人员
content	text		会议内容
Submit	button	class="btn_grey" onClick="outDoc()" value="Word 打印"	“Word 打印”按钮

(4) 利用PHP获取模板文档所在的路径，代码如下。

```
<?php
//获取模板文档所在的路径
$path="http://". $HTTP_HOST. dirname($PHP_SELF). "/meetingDot.doc";
?>
```

(5) 编写自定义JavaScript函数outDoc(), 用于将表单收集的数据导出到Word中, 并进行自动打印, 代码如下。

```
<?php
//定义新型字符串
$str=<<<word
<script language="javascript">
function outDoc() {
    var wdapp=newActiveXObject("Word.Application");
    wdapp.visible=true;
    wddoc=wdapp.Documents.Open("$path");    //打开指定的文
档
    var form=document.all.form1;
    title=form.title.value;
    meetingTime=form.meetingTime.value;
    compere=form.compere.value;
    attend=form.attend.value;
    content=form.content.value;
    //输出会议标题
    range=wdapp.ActiveDocument.Bookmarks("title").Range;
    range.Text=title;
    //输出会议时间
    range=wdapp.ActiveDocument.Bookmarks("meetingTime").Range;
ge;
    range.Text=meetingTime;
    //输出会议主持人
    range=wdapp.ActiveDocument.Bookmarks("compere").Range;
    range.Text=compere;
```

```
//输出出席人员
range=wdapp.ActiveDocument.Bookmarks("attend").Range;
range.Text=attend;
//输出会议内容
range=wdapp.ActiveDocument.Bookmarks("content").Range;
range.Text=content;
wddoc.Application.Printout();
wdapp=null;
}
</script>
word;
echo$str;
?>
```

(6) 通过单击“Word打印”按钮调用自定义JavaScript函数outDoc()。关键代码如下。

```
<input name="Submit" type="button" class="btn_grey"
onClick="outDoc()" value="Word打印">
```

举一反三

根据本实例，读者可以：

在PHP中开发打印客户订单；

在PHP中开发打印各种样式的报表。

## 11.3 利用Excel打印报表

Microsoft Excel 是Microsoft 提供的用于办公管理的应用软件。其强大的表格统计功能比其他同类型软件无法比拟的。本节将介绍如何将Web页中的数据信息导出到Excel中并打印，以提高用户的办公效率。

### 实例300 将Web页面中的数据导出到Excel

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\11\300

实例说明

为了方便用户操作，有时需要将页面中的数据导出到 Excel 中进行处理，处理后还可以利用其强大的打印功能实现页面数据的打印。下面将介绍如何将页面中的数据导出到 Excel。运行本实例，页面中将显示2007年1月份工资列表，单击“导出到Excel”超级链接，可以将工资列表导出到Excel中（用户可以对导入后的数据进行处理），数据处理后通过Excel自身的打印功能实现数据打印。程序运行结果如图11.10、图11.11所示。

技术要点

本实例主要应用JavaScript的ActiveXObject()构造函数创建一个Excel.Application对象的实例，并应用该实例的相关方法实现。ActiveXObject()构造函数的详细介绍可参见实例295。



图11.10 将Web页面中的数据导出到Excel

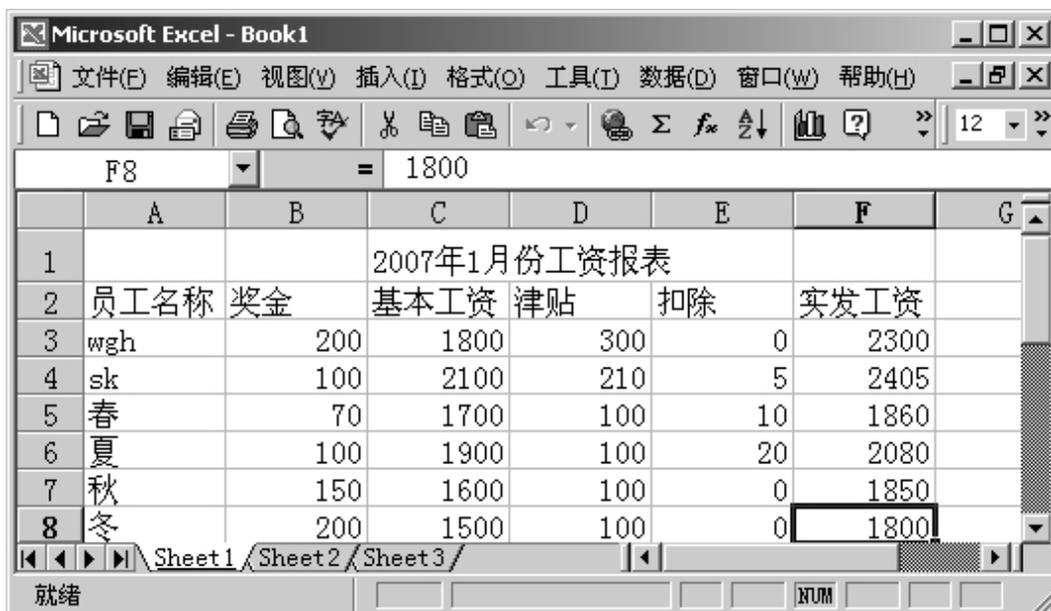


图11.11 导出到Excel中的结果

### 实现过程

(1) 将显示工资信息的表格的id设置为wage，因为要将该表格中的数据导出到Excel中。关键代码如下。

```
<table width="90%" border="0" cellspacing="1"
bgcolor="#000000" id="wage">
```

(2) 编写自定义JavaScript函数outExcel(), 用于将Web页面中的工资信息列表导出到Excel, 代码如下。

```
<script language="javascript">
function outExcel() {
    var table=document.all.wage;
    row=table.rows.length;
    column=table.rows(1).cells.length;
    var excelapp=newActiveXObject("Excel.Application");
    excelapp.visible=true;
    objBook=excelapp.Workbooks.Add();//添加新的工作簿
    varobjSheet=objBook.ActiveSheet;
    title=objSheet.Range("D1").MergeArea; //合并单元格
    title.Cells(1,0).Value=doctitle.innerHTML.replace("&nbsp;
p;",""); //输出标题
    title.Cells(1,1).Font.Size=16;
    for(i=1;i<row+1;i++) {
        for(j=0;j<column;j++) {
            objSheet.Cells(i+1,j+1).value=table.rows(i-
1).cells(j).innerHTML.replace("&nbsp;","");
        }
    }
    excelapp.UserControl = true;
}
</script>
```

(3) 通过单击“导出到Excel”超级链接调用自定义JavaScript函数outExcel()。关键代码如下。

```
<p align="center"><a href="#" onClick="outExcel();">导出到Excel</a></p>
```

举一反三

根据本实例，读者可以：

开发打印全年的工资报表；

开发打印某部门的工资报表。

## 实例301 将Web页面中的数据导出到Excel并自动打印

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\11\301

实例说明

在实际项目开发中，经常需要将Web页面中的数据导出到Excel并自动打印。下面将介绍如何将Web页面中的电话费列表导出到Excel中，并保存，同时实现自动打印功能。运行本实例，在页面中将显示电话费列表，单击“打印”超级链接，即可将电话费列表导出到Excel中，并保存到Excel的默认工作目录中，同时自动调用打印机进行打印。实例运行结果如图11.12、图11.13所示。



图11.12 将Web页面中的数据导出到Excel并自动打印

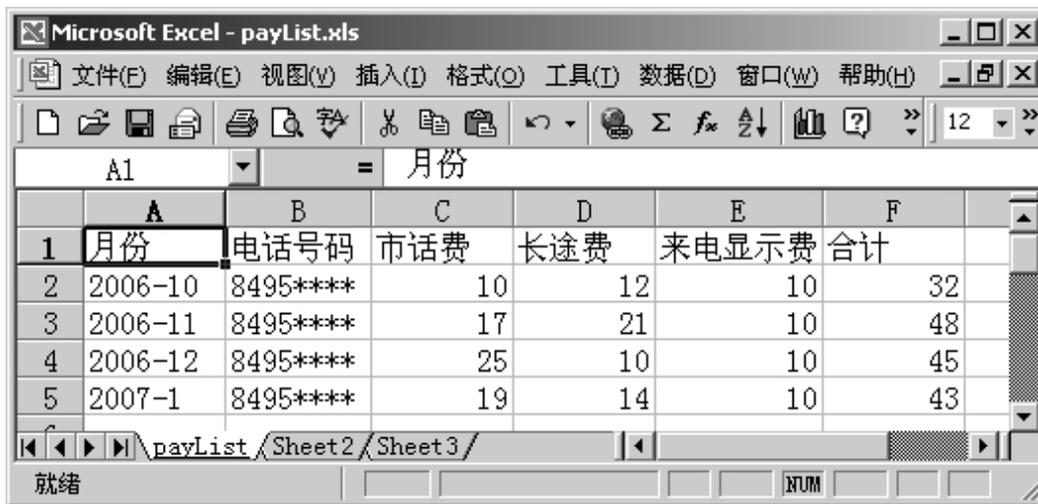


图11.13 导出到Excel中的结果

### 技术要点

本实例主要通过 `PrintOut()` 方法实现自动打印 Excel 工作表中的内容。`PrintOut()` 方法用于打印指定对象。其语法格式如下。

```
expression.PrintOut(From, To, Copies, Preview,
ActivePrinter, PrintToFile, Collate, PrToFileName)
```

参数说明如下。

- `expression`: 必选项，用于返回“Chart 对象”、“Charts 集合对象”、“Range 对象”、“Sheets集合对象”、“window对

象”、“Workbook对象”、“Worksheet 对象”或“Worksheets 集合对象”中的某个对象。

● From: 可选项, 用于指定打印的开始页号。如果省略该参数, 将从起始位置开始打印。

● To: 可选项, 用于指定打印的终止页号。如果省略该参数, 将打印至最后一页。

● Copies: 可选项, 用于指定要打印的份数。如果省略该参数, 将只打印一份。

● Preview: 可选项, 值为 true 或 false。如果为 true, 则 Microsoft Excel 打印指定对象之前进行打印预览, 如果为 false 或者省略此参数, 则立即打印该对象。

● ActivePrinter: 可选项, 用于设置活动打印机的名称。

● PrintToFile: 可选项, 值为 true 或 false。如果为 true, 则打印输出到文件。如果没有指定 PrToFileName, 则 Microsoft Excel 将提示用户输入要输出文件的文件名。

● Collate: 可选项, 当值为 true 时, 则逐份打印每份副本。

● PrToFileName: 可选项。如果将 PrintToFile 设置为 true, 则本参数指定要打印到的文件名。

说明: From参数和To参数所描述的“页”指的是要打印的页, 并非指定工作表或工作簿中的全部页。

### 实现过程

(1) 将显示电话费信息的表格的 id 设置为 pay, 因为要打印此表格中的数据。关键代码如下。

```
<table width="643" border="0" cellspacing="1"
bgcolor="#000000" id="pay">
```

(2) 编写自定义 JavaScript 函数 outExcel(), 用于将 Web 页面中的电话费列表信息导出到 Excel, 并实现自动打印, 代码如下。

```

<script language="javascript">
function outExcel() {
    var table=document.all.pay;
    row=table.rows.length;
    column=table.rows(1).cells.length;
    var excelapp=newActiveXObject("Excel.Application");
    excelapp.visible=true;
    objBook=excelapp.Workbooks.Add();//添加新的工作簿
    varobjSheet=objBook.ActiveSheet;
    for(i=0;i<row;i++){
        for(j=0;j<column;j++){
            objSheet.Cells(i+1,j+1).value=table.rows(i).cells(j).
innerHTML.replace("&nbsp;","");
        }
    }
    objBook.SaveAs("payList.xls");
    objSheet.Printout;    //自动打印
    excelapp.UserControl= true;
}
</script>

```

技巧：在 Excel 中查看并修改默认文档保存路径的方法如下：选择“工具”→“选项”选项，在弹出的对话框中选择“常规”选项卡，在该选项卡的“默认工作目录”文本中将显示默认的工作目录，读者可以将其修改为其他可用路径。

(3) 通过单击“打印”超级链接调用自定义JavaScript函数 outExcel()。关键代码如下。

```
<p align="center"><a href="#" onClick="outExcel();">打印  
</a></p>
```

举一反三

根据本实例，读者可以：

利用Excel打印全体学生报表；

利用Excel打印某年级学生报表。

## 11.4 利用CSS样式打印

利用CSS样式打印是经常使用的一种打印方法，利用它可以非常方便地实现打印页面中的指定内容和分页打印，下面将通过具体实例介绍如何利用CSS样式打印。

### 实例302 利用CSS样式打印页面中的指定内容

本实例是一个方便操作、人性化的实例

实例位置：光盘\mingrisoft\11\302

实例说明

普通的Web打印，将会打印页面中的全部内容，但在开发实际网站时，通常只需要打印页面中指定的内容，为了解决该问题，可以应用CSS样式对打印内容进行控制。运行本实例，单击“打印”超级链接即可按用户的设置进行打印。实例运行结果如图11.14、图11.15所示。



图11.14 利用CSS样式打印页面中的指定内容

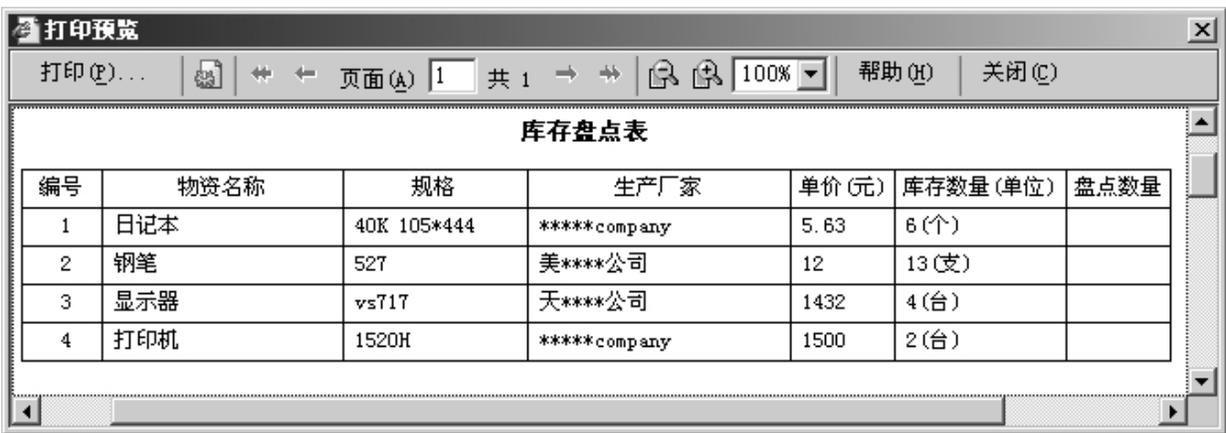


图11.15 打印预览效果

技巧：在进行 Web 打印时，可以通过以下操作控制是否打印背景颜色和图像。在 IE 窗口中，选择“工具”→“Internet选项”选项，在弹出的“Internet选项”对话框中，选择“高级”选项卡，在“设置”列表中设置“打印背景颜色和图像”前面的复选框是否选中，如果选中，代表打印背景颜色和图像，否则不打印背景颜色和图像。

技术要点

本实例主要应用了CSS样式的media类型。下面进行详细介绍。

media类型是CSS属性媒体类型，用于直接引入媒体的属性。其语法格式如下。

```
@media screen | print | projection | braille | aural | tv  
| handheld | all
```

参数说明如下。

- screen: 指计算机屏幕。
- print: 指用于打印机的不透明介质。
- projection: 指用于显示的项目。
- braille: 盲文系统，指有触觉效果的印刷品。
- aural: 指语音电子合成器。
- tv: 电视类型的媒体。
- handheld: 指手持式显示设备。
- all: 用于所有媒体。

实现过程

(1) 编写用于控制指定内容不打印的CSS样式，代码如下。

```
<style>  
@media print {  
div {display:none}  
.bgnoprint {  
background:display:none;  
}  
.noprint {  
display:none  
}  
}  
</style>
```

(2) 为不需要打印的元素设置CSS样式，关键代码如下。

```
<tablewidth="757"height="174"border="0"align="center"cell  
padding="0"  
cellspacing="0">  
  <tr class="noprint">  
    <tdheight="133"align="center"valign="top">  
        
    </td>  
  </tr>
```

…… //此处省略了其他HTML代码，具体代码请参见光盘  
举一反三

根据本实例，读者可以：

实现打印库存明细；

实现打印库存数量达到预警线的库存商品信息。

### 实例303 利用CSS样式分页打印

本实例是一个技巧性较强的程序

实例位置：光盘\mingrisoft\11\303

实例说明

在制作数据打印程序时，对于多页数据（指的是一页纸不能全部打印完毕的数据）通常采用分页打印。这里的分页打印是指在每一页数据的顶端都打印表头信息。下面将通过具体实例介绍如何利用CSS样式实现分页打印。运行本实例，如图11.16所示，单击“打印预览”超级链接，可以查看打印效果，如图11.17所示，单击“打印”超级链接即可进行分页打印。



客户名称	电话	联系人	联系人电话	E-mail	所在地区
采虹**集团	0432-65412**	高经理	137562112**	caihongsssss@***.com	吉林市
吉林省明日科技有限公司	0431-84972266	赛经理	137563185**	mingrisoft@mingrisoft.com	长春市
鑫***有限公司	0431-65412***	王经理	13756211***	qqqqq027@*****.com	长春市
东西南北***通讯公司	0434-56167***	李经理	1375631***4	mingrisoft@sina.com	四平市
明*有限责任公司	0431-849722**	张经理	1305531***6	mingxing@ming**.com	长春市

打印预览 打印

图11.16 利用CSS样式分页打印



图11.17 打印预览效果

### 技术要点

本实例主要应用了thead标记、tfoot标记和page-break-after属性。下面进行详细介绍。

①thead 标记：用于设置表格的表头。

②tfoot 标记：用于设置表格的表尾。

③page-break-after属性：在打印文档时发生作用，用于进行分页打印。但是对于<br>和<hr>对象不起作用。

语法：

```
page-break-after: auto | always | avoid | left | right |
null
```

参数说明如下。

- page-break：打印时在样式控制的对象前后换页。
- after：设置对象后出现页分割符。设置为always 时，始终在对象之后插入页分割符。
  - auto：在对象之后自动插入页分割符。
  - always：始终在对象之后插入页分割符。
  - avoid：未支持，避免在对象后面插入分割符。
  - left：未支持，在对象后面插入页分割符，直到它到达一个空白的左页边。
  - right：未支持，在对象后面插入页分割符，直到它到达一个空白的右页边。
  - null：空白字符串，取消了分割符设置。

实现过程

(1) 在要打印的页面中添加用于显示客户信息的表格，并设置好表头、表尾及打印分页，关键代码如下。

```
<tablewidth="650"border="1"
cellpadding="0"align="center"cellspacing="0"
bgcolor="#FE7529"
id="pay"bordercolor="#FE7529"bordercolordark="#FE7529"
bordercolorlight="#FFFFFF" style="border-bottom-
style:none;">
<!--设置表头-->
```

```

    <thead style="display:table-header-group;font-
weight:bold">
    <tr align="center"bgcolor="#FE7529">
        <tdwidth="155"height="30">客户名称</td>
        <tdwidth="99">电话</td>
        <tdwidth="59">联系人</td>
        <tdwidth="84">联系人电话</td>
        <tdwidth="175">E-mail</td>
        <tdwidth="64">所在地区</td>
    </tr>
</thead>
<tr>
    <tdheight="30"bgcolor="#FFFFFF">采虹**集团</td>
    <tdwidth="99">电话</td>
    <tdwidth="59">联系人</td>
    <tdwidth="84">联系人电话</td>
    <tdwidth="175">E-mail</td>
    <tdwidth="64">所在地区</td>
</tr>
<tr>
    <tdheight="30"bgcolor="#FFFFFF">吉林省明日科技有限公
司</td>
    ..... //此处省略了显示客户其他信息的HTML代码
</tr>
<tr>
<!--控制分页-->

```

```

        <tdheight="30"bgcolor="#FFFFFF" style="page-break-
after:always">鑫***有限公司</td>
        ..... //此处省略了显示客户其他信息的HTML代码
    </tr>
    <tr>
        <tdheight="30"bgcolor="#FFFFFF">东西南北***通讯公司
    </td>
        ..... //此处省略了显示客户其他信息的HTML代码
    </tr>
    <tr>
        <tdheight="30"bgcolor="#FFFFFF">明*有限责任公司</td>
        ..... //此处省略了显示客户其他信息的HTML代码
    </tr>
    <!--设置表尾-->
    <tfoot style="display:table-footer-group;border:none;">
<tr><td></td></tr></tfoot>
    </table>

```

(2) 控制“打印”和“打印预览”超级链接在打印时不显示。关键代码如下。

```

<style>
@mediaprint{
.noprint{display:none}
}
</style>

<tablewidth="647"align="center"class="noprint">
    <tr align="center"bgcolor="#FFFFFF">
        <tdheight="27"colspan="3"align="right">

```

```
<a href="#" onClick="document.all.WebBrowser.Execwb(7, 1)
">打印预览</a>
```

```
<a href="#" onClick="document.all.WebBrowser.Execwb(6, 1)
">打印</a>
```

```
<a href="#" onClick="document.all.WebBrowser.Execwb(8, 1)
"></a></td>
```

```
</tr>
```

```
</table>
```

举一反三

根据本实例，读者可以：

实现打印库存明细；

实现打印库存数量达到预警线的库存商品信息。

## 11.5 套打邮寄产品单

在程序中加入打印邮寄产品单的功能，不但可以使用户非常方便地操作程序，提高工作效率，而且更能使程序适应人性化的潮流。下面通过几个实例介绍应用 WebBrowse+CSS 套打邮寄产品单的方法。

### 实例304 打印汇款单

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\11\304

实例说明

在开发办公自动化等系统时，可能会遇到汇款单打印的情况，这时就需要在系统中加入该功能。本实例将介绍如何实现打印汇款单的功能。运行本程序，在页面中将显示要打印的汇款单及与打印相关的超级链接，如图 11.18 所示，单击“打印”超级链接可以打印该汇款单，单击“打印预览”超级链接可以预览打印结果，如图11.19所示。

1	3	0	0	3	1
---	---	---	---	---	---

邮政汇款通知

请按照背面注意事项携带本通知和本人有效证件名章到 _____ 局取款		汇票号码	
		收汇局	
金额要紧贴人民币三字大写,数字间不留空隙	汇款金额	人民币 <b>壹仟伍佰元整</b> (大写)	
	收款人详细地址	<b>吉林省长春市***街105号</b>	
	收款人姓名	<b>无语</b>	
	汇款人详细地址	<b>吉林省长春市东盛大等89号</b>	
	汇款人姓名	<b>wgh</b>	
		收汇日期	
		兑日期	
		接收专号	
		邮政编码	

汇款人简短附言

2007 年  
第 1 期  
第 10 页  
稿费

汇款通知附单	汇票号码	汇款金额	受汇日期	收汇员
	汇款人详细地址	<b>吉林省长春市东盛大等89号</b>		
	汇款人姓名	<b>wgh</b>		

打印预览  
打印  
直接打印  
页面设置

图11.18 打印汇款单页面运行结果

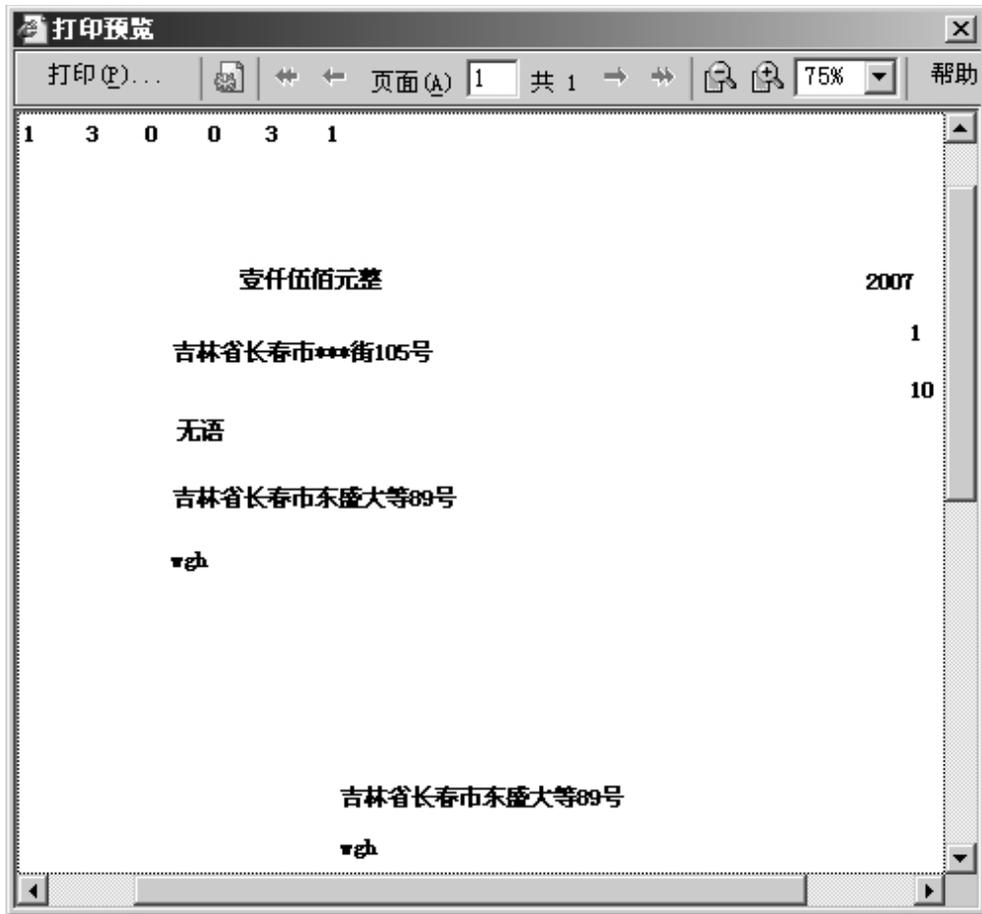


图11.19 打印预览效果

### 技术要点

本实例主要通过CSS样式的media类型属性设置不打印表格背景，及调用WebBrowser控件实现汇款单的打印。关于CSS样式的media类型属性的详细介绍可参见实例302，WebBrowser控件的详细介绍可参见实例293。

### 实现过程

- (1) 在页面中插入一个表格，将该表格的背景设置为空的汇款单图片，并在表格中插入新的表格用于在指定位置显示汇款信息。
- (2) 在页面的指定位置填写汇款信息。
- (3) 在页面的相应位置加入“打印预览”、“打印”、“直接打印”、“页面设置”超级链接。关键代码如下。

```

<div>
  <tablewidth="81"height="111"border="0"align="center"cellp
adding="0"cellspacing="0">
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a
href="#"onClick="document.all.WebBrowser.Execwb(7,1)">打
印预览</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a
href="#"onClick="document.all.WebBrowser.Execwb(6,1)">打
印</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a
href="#"onClick="document.all.WebBrowser.Execwb(6,6)">直
接打印</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a
href="#"onClick="document.all.WebBrowser.Execwb(8,1)">页
面设置</a></td>
    </tr>
  </table>
</div>

```

(4) 应用CSS样式控制表格背景及打印等超级链接在打印时不显示，代码如下。

```
<style>
@media print {
div {display:none}
td, table {
background:display:none;
}
}
</style>
```

举一反三

根据本实例，读者可以：

实现网站中的打印汇款单模块；

实现办公自动化管理系统中的打印汇款单模块。

## 实例305 打印快递单

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\11\305

实例说明

在开发物流网站时，经常会遇到打印快递单的情况。本实例将介绍如何实现打印快递单。运行本实例，在页面中将显示要打印的快递单及与打印相关的超级链接，如图 11.20 所示，单击“打印”超级链接可以打印该快递单，单击“打印预览”超级链接可以预览打印结果，如图11.21所示。

寄件地		2007 年 2 月 12 日 9 时				收件地					
寄件站		帐号				派件站		帐号			
寄件客户/地址： <b>长春市东盛大街89号</b>						收件客户/地址： <b>四平市*****街86号</b>					
电话：0431-84972266 联络人：wgh						电话：0434-5597*** 联络人：无语					
品名： <b>图书</b>		非货样 货 样		付款地点： 寄件单位 收件单位		付款方式： 付清 到付 月结		应收金额： <b>500元</b>		寄件人的声明及签名：我/我们 同意本运单背面的契约条款 寄件人签名：	
数量： <b>5</b>	件 箱	重量： <b>4千克</b>	价值： <b>600元</b>								
以材料计费：A 快递、空运(长 cm×宽 cm×高 cm) ÷6,000 = kgs B 卡车、火车(长 cm×宽 cm×高 cm) ÷3,000 = kgs								取件员		客服员	
其他费用备考								收件客户签名： 月 日 时			

打印预览 打印 直接打印 页面设置

图11.20 打印快递单页面运行结果

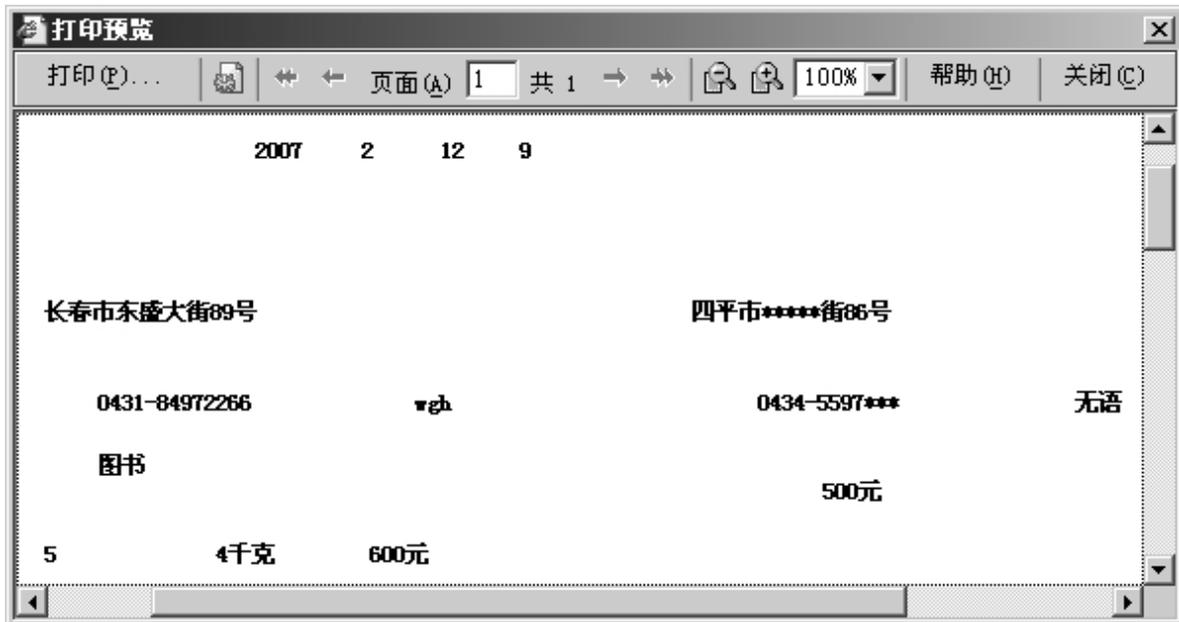


图11.21 打印预览效果

### 技术要点

本实例主要通过CSS样式的media类型属性设置不打印表格背景，及调用WebBrowser控件实现快递单的打印。关于CSS样式的media类型

属性的详细介绍可参见实例302，WebBrowser控件的详细介绍可参见实例293。

### 实现过程

(1) 在页面中插入一个表格，将该表格的背景设置为空的快递单图片，并在表格中插入新的表格用于在指定位置显示快递信息。

(2) 在页面的指定位置填写快递信息。

(3) 在页面的相应位置加入“打印预览”、“打印”、“直接打印”、“页面设置”超级链接。关键代码如下。

```
<div>
  <tablewidth="81"height="111"border="0"align="center"cellpadding="0"cellspacing="0">
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a href="#"onClick="document.all.WebBrowser.Execwb(7,1)">打印预览</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a href="#"onClick="document.all.WebBrowser.Execwb(6,1)">打印</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
      <td colspan="3"><a href="#"onClick="document.all.WebBrowser.Execwb(6,6)">直接打印</a></td>
    </tr>
    <tr align="center"bgcolor="#FFFFFF">
```

```
<td colspan="3"><a  
href="#"onClick="document.all.WebBrowser.Execwb(8,1)">页  
面设置</a></td>  
</tr>  
</table>  
</div>
```

(4) 应用CSS样式控制表格背景及打印等超级链接在打印时不显示,代码如下。

```
<style>  
@media print {  
div {display:none}  
td, table {  
background:display:none;  
}  
}  
</style>
```

举一反三

根据本实例,读者可以:

实现话费缴费单模块;

实现存款单模块。

## [实例306 打印信封](#)

本实例是一个提高效率、人性化的程序

实例位置: 光盘\mingrisoft\11\306

实例说明

在报社或电台日常业务中，最频繁的工作就是给客户或听众发信。如果每一封信都手工填写，不仅会降低工作效率，而且容易出现错误。为了解决该问题，可以在网页中加入打印信封的功能。运行本实例，在页面中将显示要打印的信封及与打印相关的超级链接，如图 11.22 所示，单击“打印”超级链接可以在信封的指定位置打印相关内容，单击“打印预览”超级链接可以预览打印结果，如图11.23所示。



图11.22 打印信封页面运行结果

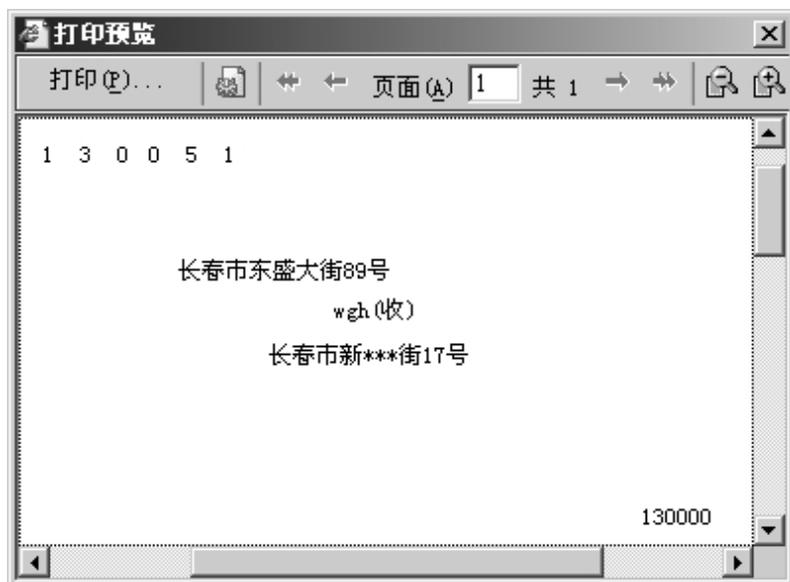


图11.23 打印预览效果

### 技术要点

本实例主要通过CSS样式的media类型属性设置不打印表格背景，及调用WebBrowser控件实现信封的打印。关于CSS样式的media类型属性的详细介绍可参见实例302，WebBrowser控件的详细介绍可参见实例293。

### 实现过程

(1) 在页面中插入一个表格，该表格的尺寸和一个标准的信封大小相同，并在表格中插入新的表格，用于在指定位置显示收信人和发信人信息。

(2) 在页面的指定位置填写收信人和发信人信息。

(3) 在页面的底部加入“打印预览”、“打印”、“直接打印”、“页面设置”超级链接。关键代码如下。

```
<div align="center">  
    <tablewidth="542"border="0"align="center"cellpadding="0"  
"cellspacing="0">  
        <tr>
```

```

        <td><div align="center">
<a href="#" onClick="document.all.WebBrowser.Execwb(7,1)"
>打印预览</a>
        <a href="#" onClick="document.all.WebBrowser.Execwb
(6,1)">打印</a>
        <a href="#" onClick="document.all.WebBrowser.Execwb
(6,6)">直接打印</a>
        <a href="#" onClick="document.all.WebBrowser.Execwb(8
,1)">页面设置</a></div></td>
    </tr>
</table>
</div>

```

(4) 应用CSS样式控制表格背景及打印等超级链接在打印时不显示，代码如下。

```

<style>
@mediaprint {
div {display:none}
td, table {
background:display:none;
}
}
</style>

```

举一反三

根据本实例，读者可以：  
实现批量打印信封。

## 第12章 网站安全

禁止用户复制网页内容

禁止用户刷新屏幕

登录页面

其他

## 12.1 禁止用户复制网页内容

在开发网络程序时，通常为其设置相关禁止用户操作的程序，以确保Web应用程序的安全运行。本节将介绍如何禁止用户复制网页内容及查看程序的相关源码。

### 实例307 禁止用户复制网页内容（方法一）

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\307

实例说明

随着网络技术的迅速发展，越来越多的企业或者个人都拥有自己的网站。为了更好地建立和维护网站中的许多相关重要信息，如技术内容、新闻内容等，可以采用一些维护网站安全的技术。本实例就是维护网站中重要信息的一种最简单的方法。禁止用户复制网页内容，只要在<body>标记中加入相关程序代码就可以实现该功能。程序运行结果如图 12.1所示。



图12.1 禁止用户复制网页内容（方法一）

### 技术要点

本实例主要应用<body>中的相关事件，下面介绍<body>标记中的常用事件及其说明。

- onactivate: 当对象设置为活动元素时触发。
- onclick: 在用户用鼠标左键单击对象时触发。
- ondrop: 当鼠标按钮在拖动操作过程中释放时在目标对象上触发。
- onkeydown: 当用户按下键盘按键时触发。
- onkeypress: 当用户按下并释放字母数字键时触发。
- onkeyup: 当用户释放键盘按键时触发。
- onload: 当浏览器装入对象后立即触发。
- onmousedown: 当用户用任何鼠标按键单击对象时触发。
- onmousemove: 当用户将鼠标划过对象时触发。
- onmouseout: 当用户将鼠标指针移出对象边界时触发。
- onmouseover: 当用户将鼠标指针移动到对象内时触发。

● onmouseup: 当用户在鼠标位于对象之上释放鼠标按键时触发。

● onmove: 当对象移动时触发。

● onmoveend: 当对象停止移动时触发。

● onmovestart: 当对象开始移动时触发。

● onscroll: 当用户滚动对象的滚动条时触发。

● onselect: 当前选中区改变时触发。

● onselectstart: 对象将要被选中时触发。

● onunload: 在对象卸载前立即触发。

实现过程

通过代码实现禁止用户复制网页内容，其关键代码如下。

```
<body onselectstart="return false">
```

举一反三

根据本实例，读者可以：

实现在新闻网站中禁止复制新闻信息；

实现在线论坛中禁止复制主题及回复信息。

## 实例308 禁止用户复制网页内容（方法二）

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\308

实例说明

通过网络传输数据时，数据信息安全保密问题极其重要。在网页中有一部分信息只供浏览者进行浏览而不允许被复制、粘贴。本实例主要应用<body>中的相关事件来禁止用户复制网页中的重要信息。程序运行结果如图12.2所示。



图12.2 禁止用户复制网页内容（方法二）

### 技术要点

本实例主要应用<body>中的相关事件，因为在上一实例中已对其进行了介绍，在此不再赘述，具体请参见实例307。

### 实现过程

将代码添加到<body>标记中，禁止用户复制网页内容。其关键代码如下。

```
<body onmouseup=document.selection.empty()  
oncontextmenu="return false" onselectstart="return false"  
ondragstart="return false" onbeforecopy="return false"  
oncopy=document.selection.empty()>
```

### 举一反三

根据本实例，读者可以：

实现在电子商务网站中禁止复制用户订单信息；

实现在企业门户网站中禁止复制商品销售信息。

## [实例309 禁止网页另存为](#)

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\309

实例说明

在一些商用网站中，经常需要在网站上发布一些相关重要信息。用户可以对记载信息的网页进行浏览，而不能进行下载或是将整个网页另存为。本实例主要应用 JavaScript 脚本中的 `<noscript>` 标记防止网页被另存为。运行程序，打开网页进行浏览时，选择“文件”→“另存为”选项，然后选择文件所要保存到的指定位置，单击“确定”按钮，此时将弹出一个提示对话框，显示“无法保存该网页”。程序运行结果如图12.3所示。



图12.3 禁止网页另存为

技术要点

本实例主要应用JavaScript脚本中的`<noscript>`标记防止网页被另存为。下面将对`<noscript>`标记进行详细介绍。

`<noscript>`标记的语法:

`<noscript>...</noscript>`

<noscript>标记用来与 <script>...</script>标记平衡。假如浏览器关闭了JavaScript，<script>标记就被忽略，转而处理<noscript>标记。<noscript>标记可以用来显示HTML内容，让用户知道JavaScript是关闭的，除非打开JavaScript，否则就不能显示页面的全部高级内容。

### 实现过程

将代码添加到<body>...</body>标记之间，禁止网页被另存为。其关键代码如下。

```
<noscript><iframe src="*.html"></iframe></noscript>
```

### 举一反三

根据本实例，读者可以：

实现在电子商务网站中禁止网页被另存为；

实现开发博客、论坛类网站时禁止网页被另存为。

## 12.2 禁止用户刷新屏幕

在开发网络程序时，为了维护程序的正常运行，可以禁止用户在任何页面中进行其他不规则的相关操作。因此在本节中将对禁止用户的相关操作进行详细介绍。

### 实例310 屏蔽IE主菜单

本实例是一个提高网站安全性、人性化的程序

实例位置：光盘\mingrisoft\12\310

实例说明

在开发电子商务网站时，当用户填写商品订单时，不可以再进行刷新屏幕或后退等操作。因此，程序设计员应将窗口中所有IE菜单屏蔽掉，避免用户重复提交订单信息而导致网站接收错误的订单信息。本实例主要通过JavaScript脚本中的window对象来屏蔽IE主菜单。程序运行结果如图12.4所示。

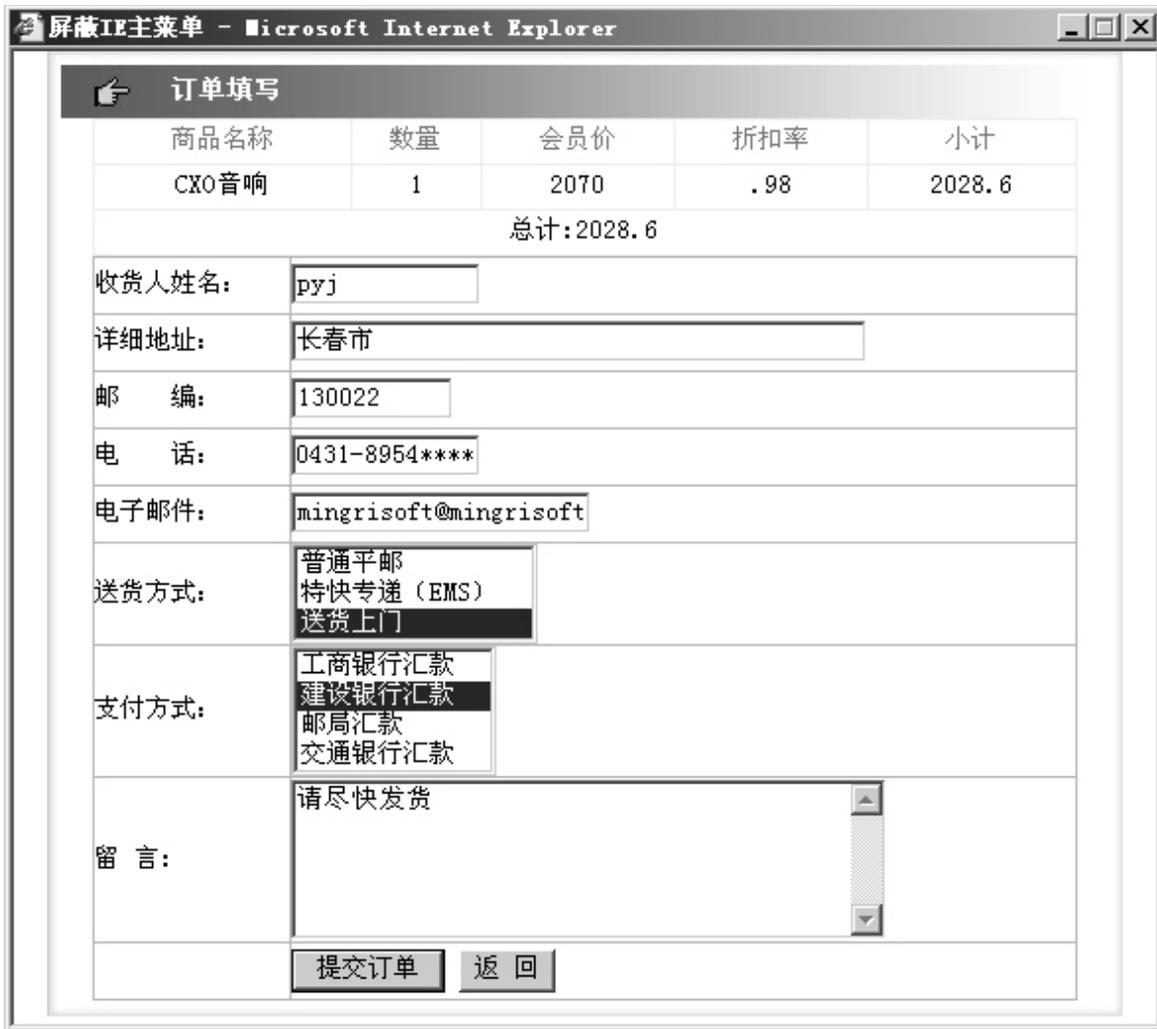


图12.4 屏蔽IE主菜单

### 技术要点

本实例主要应用JavaScript脚本中的window对象的open()方法实现。应用window对象的open()方法可以打开一个基于特定条件的新窗口，例如，可以指定新窗口的大小以及窗口中可用的选项，并且可以为引用它的窗口指派名字。

Open()方法的语法格式如下。

```
window.open('index.htm', '', 'width=790,height=430',
menubar='no')
```

其中，通过width指定窗口的宽度，height指定窗口的高度，menubar指定是否显示菜单条（一般包括文件、编辑及其他一些菜单项）。

### 实现过程

通过JavaScript脚本实现屏蔽IE主菜单，其关键代码如下。

```
<script language="javascript">
function deal() {
window.opener=null;
window.close();
window.open('order.htm','','width=580,height=510');
}
</script>
<a href="#" onClick="deal();">收银台</a>
```

### 举一反三

根据本实例，读者可以：

实现在企业门户网站后台管理系统中屏蔽IE主菜单；

实现在新闻网站中的新闻浏览页面中屏蔽IE主菜单。

## 实例311 屏蔽键盘相关事件

本实例是一个提高网站安全性、人性化的程序

实例位置：光盘\mingrisoft\12\311

### 实例说明

在进行提交商品表单时，通常情况不允许用户刷新屏幕、后退或新建文档，否则可能造成不可估计的损失，这可以通过屏蔽键盘的回车键、退格键、F5 键、Ctrl+N 组合键、Shift+F10组合键来实现。本实例主要通过 JavaScript 脚本屏蔽键盘相关事件。运行程序，在填

写订单信息页面中按键盘的回车键、退格键、F5 键、Ctrl+N 组合键、Shift+F10 组合键，此时系统将给予相关提示信息。程序运行结果如图12.5所示。



图12.5 屏蔽键盘相关事件

### 技术要点

本实例主要应用 JavaScript 脚本中的 Event 对象的相关属性实现。其中，keyCode 属性表示按下按键的数字代号。下面将常用的 keyCode 属性值以表格的形式列出，如表 12.1 所示。

表12.1 keyCode属性值

值	描 述
8	退格键
13	回车键
116	F5 刷新键
37	Alt+方向键“←”或方向键“→”
78	Ctrl+N 新建 IE 窗口
121	Shift+F10

## 实现过程

屏蔽键盘的相关事件代码如下：

```
<script language=javascript>
function keydown() {
    if (event.keyCode==8) {
        event.keyCode=0;
        event.returnValue=false;
        alert("当前设置不允许使用退格键");
    } if(event.keyCode==13) {
        event.keyCode=0;
        event.returnValue=false;
        alert("当前设置不允许使用回车键");
    } if(event.keyCode==116) {
        event.keyCode=0;
        event.returnValue=false;
        alert("当前设置不允许使用F5刷新键");
    } if((event.altKey)&&((window.event.keyCode==37) ||
(window.event.keyCode==39))) {
        event.returnValue=false;
        alert("当前设置不允许使用Alt+方向键←或方向键→");
    } if((event.ctrlKey)&&(event.keyCode==78)) {
```

```
    event.returnValue=false;
    alert("当前设置不允许使用Ctrl+n新建IE窗口");
}if((event.shiftKey)&&(event.keyCode==121)){
    event.returnValue=false;
    alert("当前设置不允许使用shift+F10");
}
}
</script>
<script language=javascript>
    function click(){
        event.returnValue=false;
        alert("当前设置不允许使用右键!");
    }
    document.oncontextmenu=click;
</script>
<bodyonkeydown="keydown()">
```

举一反三

根据本实例，读者可以：

实现编写网站用户注册程序；

实现编写用户登录网站程序。

## [实例312 屏蔽鼠标右键](#)

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\312

实例说明

当用户浏览网站时，经常会利用鼠标右键进行快捷方式的操作，例如“查看源文件”、“刷新”等。但是有些网站并不想让用户执行这些操作，这时就需要屏蔽鼠标的右键操作。那么如何禁止浏览者使用鼠标右键呢？这可以通过 JavaScript 脚本调用鼠标事件

(onmousedown 事件)覆盖鼠标右键快捷菜单实现。运行本实例，当用户单击鼠标右键时，将会弹出图12.6所示的提示警告对话框。



图12.6 屏蔽鼠标右键

### 技术要点

在事件处理过程中，有时需要使用 event 事件对象。通过 event 对象，可以访问键盘按键、鼠标动作等事件的状态，从而实现对键盘和鼠标动作的实时控制。实际上，event 对象是window对象的子对象，可以在事件处理程序中直接使用。本实例主要应用event对象的button属性。

event对象的常用属性如下。

- altKey: 指示Alt 键的状态，当Alt 键按下时返回值为True。
- ctrlKey: 指示Ctrl 键的状态，当Ctrl 键按下时返回值为True。
- shiftKey: 指示Shift 键的状态，当Shift 键按下时返回值为True。
- button: 指示哪一个鼠标键被按下（0 表示无键被按下，1 表示左键被按下，2 表示右键被按下，4表示中键被按下）。
- keyCode: 指示引起键盘事件的按键的Unicode 码值。

- `offsetY`: 鼠标光标相对于事件所在对象（或称容器）的相对垂直位置。
- `reason`: 指示数据传递的结果，用于数据源对象。
- `fromElement`: 指示`onmouseover` 和`onmouseout` 事件从哪个元素移来。
- `srcElement`: 指示激发事件的对象。
- `toElement`: 指示`onmouseover` 和`onmouseout` 事件移向哪个元素。

### 实现过程

通过JavaScript脚本，屏蔽鼠标右键，关键程序代码如下。

```
<script language="javascript">
function click() {
    if (event.button==2) {
        alert('禁止使用鼠标右键!!!')
    }
}
document.onmousedown=click
</script>
```

### 举一反三

根据本实例，读者可以：

实现在中介信息网站中屏蔽鼠标右键；

实现在新闻类网站中屏蔽鼠标右键。

## 12.3 登录页面

在设计和建设网站的过程中，网络的安全技术显得越来越重要。本节将通过介绍用户安全登录网站系统的登录页面，向大家展示如何提高网站的安全性。

### 实例313 具有浏览器检测功能的登录页面

本实例可以限制用户的登录环境

实例位置：光盘\mingrisoft\12\313

实例说明

Web 编程的一个重要工作是确定目标浏览器和操作系统，无论是构建一个简单的网站还是一个复杂的Web应用程序，在工作开始之前都必须确定这些重要的信息。因为，不同的浏览器支持不同级别的 HTML 和 JavaScript，而且操作系统不同支持的级别也不同，如果事先了解这些信息，可以保证程序中不会包含那些用户不可用的特性。例如，在Windows操作系统下用户可以使用 Internet Explorer（以下简称 IE）、Mozilla和Opera浏览器，在Macintosh操作系统下用户可以使用 IE、Mozilla 和 Safari，在 UNIX 操作系统下用户可以使用 Mozilla 和Konqueror浏览器。

当创建Web应用程序时，用户看到的第一个页面往往是登录页面。在登录页面中加入浏览器检测，在用户未能输入用户名和密码之前就进行检测，判断用户使用的浏览器是否能保证应用程序的正常运行。运行本实例，如果用户使用的浏览器是IE 6.0或者更高版本，则正常

显示登录页面，如图12.7所示，否则显示错误提示信息页面，如图12.8所示。



图12.7 管理员登录页面



图12.8 错误提示信息页面

### 技术要点

本实例主要应用JavaScript脚本中的navigator对象的userAgent属性和appVersion属性、global对象的parseFloat()方法、string对

象的indexOf()方法、RegExp对象的test()方法以及window对象的onload事件，下面分别进行介绍。

(1) 使用navigator对象的userAgent属性获取用户代理头字符串，使用appName属性获取浏览器版本信息的字符串。

(2) 调用global 对象的parseFloat()方法将指定的字符串转换为一个浮点数。

(3) 调用string 对象的indexOf()方法，将返回字符串中包含指定参数所代表的字符串第一次出现的位置值，如果没有找到指定字符串则返回-1。

(4) RegExp 对象的 test()方法用来判断指定的字符串是否匹配 RegExp 对象确定的模式，如果匹配则返回True，否则返回False。

(5) 当浏览器装入对象后，将立即触发window对象的onload事件。

### 实现过程

本实例首先通过获取到的用户代理头字符串判断使用的是否为Internet Explorer 浏览器，然后再确定使用的 Internet Explorer 浏览器的版本。如果使用的浏览器版本符合要求，则正常显示登录页面，否则显示错误提示信息。以下代码使登录页面具有浏览器检测的功能，其关键代码如下。

```
<script language="javascript">
var sUagent=navigator.userAgent;
var fAVersion=parseFloat(navigator.appVersion);
var isOpera=sUagent.indexOf("Opera")>-1;
var isIE=sUagent.indexOf("compatible")>-1&&
sUagent.indexOf("MSIE")>-1&& !isOpera;
var isIE4=isIE5=isIE5_5=isIE6=false;
if (isIE) {
```

```
var reIE=newRegExp("MSIE (\\d+\\.\\d+);");
reIE.test(sUagent);
var getIEVersion=parseFloat(RegExp["$1"]);
isIE4= getIEVersion>=4;
isIE5= getIEVersion>=5;
isIE5_5=getIEVersion>=5.5;
isIE6= getIEVersion>=6;
}
window.onload=function () {
    if (isIE6) {
    }
};
    document.getElementById("divLogin").style.visibility=
"visible";
    document.getElementById("divError").style.visibility=
"hidden";
</script>
```

举一反三

根据本实例，读者可以：

实现在游戏网站登录页面中添加浏览器检测功能；

实现在网络考试系统中的考生登录页面中添加浏览器检测功能。

## [实例314 防止SQL注入的登录页面](#)

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\314

实例说明

当应用程序使用输入内容来构造动态SQL语句访问数据库时，会产生SQL注入攻击，SQL注入成功后，攻击者就可以随意在数据库中执行命令的漏洞。所以，在程序代码中对一些SQL字符进行屏蔽，可以防止注入漏洞产生。不法用户通常会在登录页面使用SQL注入，从而进入管理系统进行操作。本实例是在用户登录页面中对用户输入的用户名和密码进行判断，屏蔽掉危险字符，防止非法用户登录网站。程序运行结果如图12.9所示。



图12.9 防止SQL注入的登录页面

### 技术要点

使用JavaScript脚本语言中的indexOf()方法来查找字符串中是否包含指定字符串。

语法:

```
str.indexOf(substr)
```

参数说明如下。

- str: 表示在该字符串中进行查找。
- substr: 表示要查找的字符串。

功能: 从字符串str的左面开始查找，并返回第一次出现字符串substr的位置；若没有找到，返回值小于0。

查找字符串的另一种方法:

```
str.lastIndexOf(substr)
```

功能：从字符串str的右面开始查找，并返回第一次出现字符串substr的位置；若没有找到，返回值小于0。

### 实现过程

通过代码判断用户输入的内容是否包含危险字符，防止SQL注入，其关键代码如下。

```
<script type="text/javascript">
functionMycheck(str) {
    varmess="不允许输入的字符：\r\n";
    varmark="yes";
    if(str.indexOf(";")>=0) {
        mark="no";
        mess+=" ; ";
    }
    if(str.indexOf("&")>=0) {
        mark="no";
        mess+="& ";
    }
    if(str.indexOf("<")>=0) {
        mark="no";
        mess+="< ";
    }
    if(str.indexOf(">")>=0) {
        mark="no";
        mess+="> ";
    }
    if(str.indexOf("--")>=0) {
        mark="no";
```

```
        mess+="--";
    }
    if(str.indexOf("/")>=0) {
        mark="no";
        mess+="/";
    }
    if(str.indexOf("%")>=0) {
        mark="no";
        mess+=" %";
    }
    if(str.indexOf(" ")>=0) {
        mark="no";
        mess+=" ' ";
    }
    if(mark=="no") {
        alert(mess);
        return false;
    }
    else return
        return true;
}
</script>
```

### 举一反三

根据本实例，读者可以：

实现编写电子商务网站登录模块；

实现编写网上考试系统登录模块。

## 实例315 带验证码的登录页面

本实例是一个提高网站安全性、人性化的程序

实例位置：光盘\mingrisoft\12\315

实例说明

带验证码的用户登录模块是指当用户登录时，除了要填写用户名和密码外，必须还要填写随机生成的验证码，这样可以加强网站的安全性。运行本实例，在用户登录页面中可以看到随机生成的验证码。程序运行结果如图12.10所示。



图12.10 带验证码的登录页面

技术要点

本实例主要应用JavaScript脚本中的Math对象的random()方法来生成一个随机数，其语法格式如下。

Math.random()

功能：获取0~1之间的随机数。

若要获得0~100之间的数字可以将该数字乘以100，即Math.random()\*100。若要获取0~50之间的随机数可以再与50进行求余运算，即(Math.random()\*100)%50。

实现过程

在页面载入时，调用生成随机数的 JavaScript 脚本函数，将获取到的随机数赋予变量，在页面文本框中显示该随机数。其关键代码如下。

```
<script language="javascript">
function veri() {
var sourcenum="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";
var siglenum="";
var checknum="";
var index=0;
for(i=0;i<8;i++) {
    index=(Math.random()*100)%35;
    siglenum=sourcenum.substring(index,index+1);
    checknum+=siglenum;
    i++;
}
document.form1.txt_ver.value=checknum;
}
</script>
<bodyonLoad="veri();">
```

举一反三

根据本实例，读者可以：

实现在办公自动化系统的登录页面中使用验证码；

实现在客户留言页面中使用验证码。

## 12.4 其他

为了保证网站正常运行，除了在页面中加入适当代码防止非法用户的操作外，还可以使用加密工具对页面进行加密来保护源代码。

### 实例316 使用Script Encoder加密工具加密

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\12\316

实例说明

为了防止自己写的脚本或者源代码被别人盗取，通常采用加密的方法来保护自己的源代码。本实例使用Script Encoder 加密工具进行加密，加密前后如图12.11、图12.12 所示。

```
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>使用Script Encoder加密工具加密</title>
<script language="javascript">
alert("请输入正确的数据格式")
</script>
```

图12.11 加密前的代码

```
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>使用Script Encoder加密工具加密</title>
<script language="JScript.Encode">
#&^GwAAAA==@##&lsDD`J请输入正确的数据格式r##&2wIAAA==^#^@
</script>
```

图12.12 加密后的代码

技术要点

Script Encoder 是Microsoft 出品的一个script 加密工具，它是一个简单的命令行工具，可以对 HTML 文件进行加密，只是对其中的 JavaScript 和 VBScript 部分进行加密处理，其他的文本部分均保持不变，加密后对 script 的功能并无影响，仅使其代码变成密文，用源文件方式查看只能看到一些乱码。该程序除了可对HTML文件加密外，也可以对ASA、ASP、CDX、JS、SCT、VDS文件进行加密。该程序很小，只有64KB，使用非常简单，可在DOS命令行下执行，也可在Windows下带参数运行，执行语句语法如下。

SCRENC [参数] 源文件 加密后文件

说明：简单的运行可不带参数。

实现过程

(1) 安装Script Encoder 加密工具。

(2) 选择“开始”→“程序”→“附件”→“命令提示符”选项。

(3) 把html文件复制到相应的目录下，然后输入相应的命令。

```
screnc *.htm c:\temp
```

举一反三

根据本实例，读者可以：

实现在企业门户网站使用脚本的各页面中使用加密工具。

# 第13章 HTML/CSS样式

页面效果

表格样式

鼠标及滚动条样式

文字及列表样式

文字滤镜特效

图片滤镜特效

## 13.1 页面效果

CSS 样式的用处非常广泛，它有很好的兼容性，只要是识别串接样式表的浏览器就可以应用CSS样式表，利用CSS样式可以改变页面的显示风格。下面将通过几个典型实例进行详细介绍。

### 实例317 统一站内网页风格

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\13\317

实例说明

虽说个性突出的网站能吸引用户的眼球，但是如果将一个网站的每个页面都制作得花花绿绿，就会给人很“晕”的感觉，所以在设计网站时，一定要在突出个性的同时，统一风格。本实例将介绍如何在设计网站时，应用CSS样式统一网站风格。运行结果如图13.1所示。



图13.1 统一站内网页风格

### 技术要点

本实例主要应用CSS样式的字体属性、颜色属性、边框属性实现，下面对这些属性进行详细的介绍。

#### (1) 字体属性

字体属性主要包括字体综合设置、字体名称、字体大小、字体风格、字体加粗、字体英文大小写转换等，如表13.1所示。

表13.1 字体属性

字体属性	描述
font	设置或者检索对象中文本特性的复合属性
font-family	一个指定的字体名或者一个种类的字体名称

续表

字体属性	描述
font-size	字体显示的大小
font-style	以3种方式中的一种来显示字体：normal(普通)、italic(斜体)或者 oblique(倾斜)
font-weight	使字体加粗或者变细
font-variant	设置英文大小写转换

## (2) 颜色和背景属性

CSS的颜色属性允许设计者指定页面元素的颜色，背景属性用于指定页面的背景颜色或者背景图像的属性。颜色和背景属性如表13.2所示。

表13.2 颜色和背景属性

颜色和背景属性	描述
color	设定页面元素的前景色
background-color	设定页面元素的背景色
background-image	设定页面元素的背景图像
background-repeat	设定一个指定的背景图像被重复的方式
background-attachment	设定背景图像是否跟随页面内容滚动
background-position	设定水平和垂直方向上的位置
background	背景属性的综合设定

## (3) 边框属性

边框属性用于设置一个元素边框的宽度、样式和颜色。边框属性如表13.3所示。

表13.3 边框属性

边框属性	描述
border	边框复合属性
border-top	上边框
border-left	左边框
border-right	右边框
border-bottom	下边框
border-color	边框颜色
border-style	边框样式
border-width	边框宽度
border-top-color	上边框颜色
border-left-color	左边框颜色
border-right-color	右边框颜色
border-bottom-color	下边框颜色
border-top-style	上边框样式
border-left-style	左边框样式

续表

边框属性	描述
border-right-style	右边框样式
border-top-width	上边框宽度
border-left-width	左边框宽度
border-right-width	右边框宽度
border-bottom-width	下边框宽度

下面对边框样式的值进行详细的说明，如表13.4所示。

表13.4 边框样式属性值

边框样式属性值	描 述
none	无边框
hidder	隐藏边框 IE 不支持
dotted	边框由点组成
dashed	边框由短线组成
solid	边框是实线
double	边框是双实线
groove	边框带有立体感的沟槽
ridge	边框成脊形
inset	边框内嵌一个立体边框
outset	边框外嵌一个立体边框

注意：在 CSS 样式中的 `td{ font-size: 9pt;}` 表示应用于页面中所有的 `<td>` 标记；`.botton1{ font-size: 9pt;}` 表示自定义的 CSS 样式，只有将标记的 class 属性值设置为 `botton1`，才可以调用 CSS 样式。

### 实现过程

(1) 定义 CSS 样式。为了更方便地统一网页的整体风格，可以根据需要将各种风格的样式定义为单独的外部样式表文件，并根据需要直接调用外部 CSS 样式表文件。本实例中定义了一个“`css1.css`”外部样式表文件，详细代码如下。

```
<style type="text/css">
```

```
<!--
```

设置所有单元格中的字体大小和颜色。

```
td{
    font-size: 9pt;
    color: #007766;
}
```

设置所有文本框的样式及字体的大小、颜色等。

```
input {
    font-size: 9pt;
```

```
color: #003399;
font-family: 宋体;
border: '1px' 'dashed' '#999999';
background: #EEEEEE;
}
```

设置单元格的字体及文字大小。

```
.td1 {
font-size: 12pt;
font-family: 黑体;
}
```

设置按钮的背景颜色、宽度等。

```
.button1 {
background: #FFFFFF;
font-weight: bold;
width: 50px;
}
```

-->

</style>

(2) 在页面中引用外部样式表文件css1.css，并将其id属性设置为myCss，代码如下。

```
<link id="myCss" href="css1.css" rel="stylesheet">
```

(3) 编写页面代码，并引用相应的CSS样式。代码如下。

```
<tablewidth="100%"height="20"border="1"
cellpadding="0"cellspacing="0"
borderColor="#F3F3F3"bgcolor="#FFFFFF">
<tr>
<tdwidth="5%"></td>
```

```

        <tdwidth="95%"bgcolor="#F9F9F9">&nbsp;明日科技→用户
注册</td>
    </tr>
</table>
<tablewidth="100%"height="20">
    <tr>
        <td><imgwidth="100%"height="25" src="1.jpg"></td>
    </tr>
</table>
<br>
<br>
<tablewidth="100%"height="316"border="0"cellpadding="0"
cellspacing="0"
bgcolor="#FFFFFF">
    <tr>
        <td class="td1"height="239"><div align="center">
            <p>用户姓名:
                <input type="text"name="textfield">
                用户年龄:
                <input type="text"name="textfield2">
            </p>
            <p>用户别名:
                <input type="text"name="textfield3">
                出生年月:
                <input type="text"name="textfield4">
            </p>
            <p>登录密码:

```



</div>

</form>

举一反三

根据本实例，读者可以：

统一网站中的报表风格；

统一网站中的鼠标风格。

## 实例318 设置超级链接文字的样式

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\13\318

实例说明

通过CSS样式可以随意改变超级链接文字的颜色及样式。在本实例中，当鼠标移动到超级链接文字“明日科技”时，将在文字的下面显示一个下划线，并改变文字的颜色，同时将文字改为粗体。运行结果如图13.2所示。

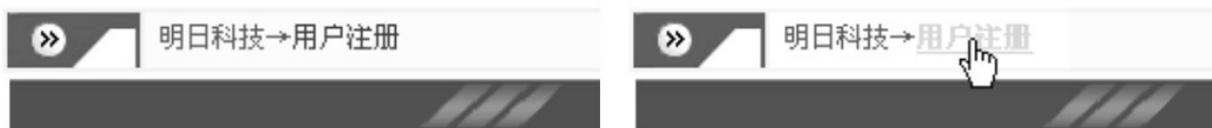


图13.2 设置超级链接文字的样式

技术要点

本实例应用到了CSS样式中组合标签定义层叠样式表中的一个类型：伪类。

伪类是特殊的类，能自动被支持CSS的浏览器识别。伪类可以区别标记的不同状态。伪类的一个最常见的应用是指定超级链接<a>标记以显示不同的状态。

伪类的语法格式：

选择符：伪类{规则1；规则2； ..}

在CSS样式中<a>标记的书写格式有以下几种。

(1) 设置对象在未被访问前的样式表属性，例如：

```
a:link{font-size:16px;text-decoration:
none;color:#CCCCCC;}
```

(2) 设置对象在其鼠标悬停时的样式表属性，例如：

```
a:hover{font-size:16px;text-decoration:
underline;color:#FF0000; }
```

(3) 设置对象在被用户激活（在鼠标单击与释放之间发生的事件）时的样式表属性，例如：

```
a:active{font-size:16px;text-decoration:
none;color:#6699FF;}
```

(4) 设置对象在其链接地址已被访问过时的样式表属性，例如：

```
a:visited{font-size:16px;text-decoration:
none;color:#00FFFF;}
```

在<a>标记中link、hover、active和visited全是伪类，其中，link为<a>标记的省略值，可以写成如下代码。

```
a {font-size:16px;text-decoration: none;color:#CCCCCC;}
```

本实例主要应用了文本属性中的text-decoration属性对文字进行修饰。

text-decoration属性的语法格式：

```
text-decoration: underline | overline | line-through |
blink | none
```

参数说明如下。

- underline：文字加下划线。
- overline：文字加上划线。
- line-through：文字加删除线。
- blink：闪烁文字，只有Netscape 浏览器支持。

● none: 默认值。

### 实现过程

(1) 编写更改超级链接文字的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

设置所有单元格中的字体大小及颜色。

```
td{  
    font-size: 9pt;  
    color: #007766;  
}
```

设置对象在未被访问时的样式表属性。

```
a{  
    font-size:9pt;  
    color:#0000FF;  
    text-transform: uppercase;  
    text-decoration: none;  
}
```

设置对象在鼠标悬停时的样式表属性。

```
a:hover{  
    font-size:9pt;  
    color:#00FF00;  
    font-weight:bold;  
    text-decoration:underline;  
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加的代码与实例 317 中的一样，只是将“明日科技”用<a>标记设为超级链接状态。关键代码如下。

```
<tablewidth="100%"height="20"border="1"
cellpadding="0"cellspacing="0"
borderColor="#F3F3F3"bgcolor="#FFFFFF">
  <tr>
    <tdwidth="5%"></td>
    <tdwidth="95%"bgcolor="#F9F9F9">&nbsp;明日科技
    →<ahref="Index.htm">用户注册</a>
  </td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

当激活超级链接后，改变其字体颜色；

当超级链接被用户激活时，使字体变粗。

## 实例319 网页换肤

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\13\319

实例说明

在 Internet 上很多网页都具有换肤功能。所谓网页换肤就是指用户可以根据自己的喜好选择页面的风格。例如，在Go购电子商城的首页中的网页换肤功能。运行本实例，单击“[秋叶飘香/经典]”超级链接，整个页面将以橘黄色系为主，如图13.3所示。单击“[灰色畅想/舒畅]”超级链接，整个页面将以灰色系为主，如图13.4所示。



图13.3 “秋叶飘香”风格的页面

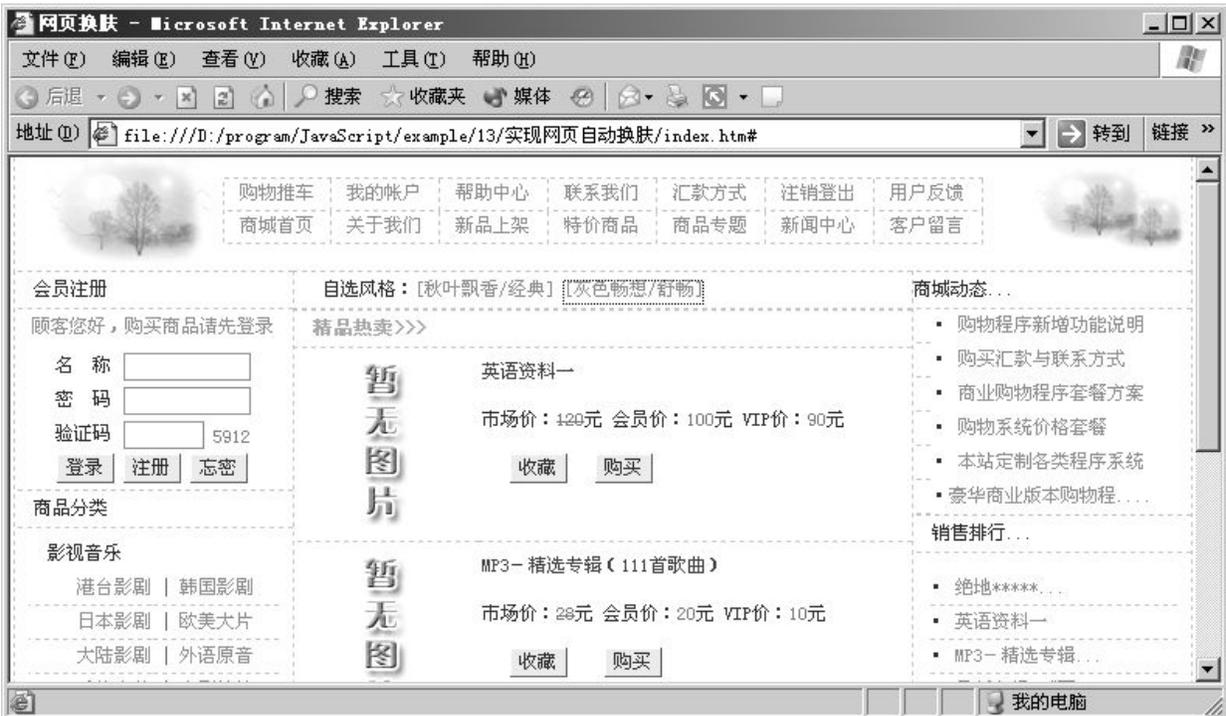


图13.4 “灰色畅想”风格的页面

### 技术要点

本实例主要应用样式表对象的href属性实现。通过样式表对象的href属性可以很方便地改变网页中应用的外部样式表文件，具体代码如下。

```
document.getElementById("myCss").href= "css1.css" ;
```

参数说明如下。

- myCss: 为外部样式表对象的id，用于指定外部样式表对象。
- css1.css: 为外部样式表文件名。

实现过程

(1) 定义 CSS 样式。由于应用外部 CSS 样式表文件可以很方便地修改网页的整体风格，所以为了动态改变网页的整体风格，可以根据需要将各种风格的样式定义为单独的外部样式表文件。本实例中共定义了两个外部样式表文件，分别是“css1.css”（“秋叶飘香”风格）和“css.css”（“灰色畅想”风格），详细代码请参见本书附带的光盘。

(2) 在页面中引用外部样式表文件css1.css，并将其id属性设置为myCss，代码如下。

```
<link id="myCss" href="css1.css" rel="stylesheet">
```

(3) 编写自定义函数writeCookie ( )，用于将指定的值写入到Cookie中，Cookie的有效期为30天。writeCookie ( )函数只有一个用于指定外部样式表文件名的参数csspath，无返回值，代码如下。

```
<script language="javascript">  
functionwriteCookie(csspath) {  
var today=newDate();  
var expires=newDate();  
expires.setTime(today.getTime()+1000*60*60*24*30); //
```

有效期为30天

```

var
str="cssPath="+csspath+";expires="+expires.toGMTString()+";"
;
document.cookie=str;
}
</script>

```

(4) 编写自定义函数 readCookie(), 用于从 Cookie 中读取使用的外部样式文件名。Read Cookie() 函数只有一个用于指定 Cookie 名称的参数, 无返回值, 代码如下。

```

<script language="javascript">
function readCookie(cookieName) {
var search= cookieName+"=";
if (document.cookie.length>0) {
offset=document.cookie.indexOf(search);
if (offset !=-1) {
offset+= search.length;
end=document.cookie.indexOf(";", offset);
if (end==-1) {
end=document.cookie.length;
returnunescape(document.cookie.substring(offset,
end));
}
}
}
}
</script>

```

(5) 编写自定义的函数ifCookie(), 用于初始化页面风格, 关键代码如下。

```
<script language="javascript">
function ifCookie() {
    if(readCookie("cssPath")==undefined) {    //当标记页面网
格的Cookie不存在时
        writeCookie("css1.css");    //调用writeCookie()函数将
默认值写入Cookie
    }
    document.getElementById("myCss").href=readCookie("cssPa
th");
}
</script>
```

(6) 编写自定义的函数change(), 用于改变页面的整体风格, 关键代码如下。

```
<script language="javascript">
function change(myCss, csspath) {
    writeCookie(csspath);    //调用writeCookie()函数将指定
的页面风格写入Cookie
    document.getElementById(myCss).href=readCookie("cssPath
");
}
</script>
```

(7) 在<body>标记的onLoad事件中调用ifCookie()方法初始化页面风格, 关键代码如下。

```
<body leftmargin="0" topmargin="0" onLoad="ifCookie();">
```

(8) 在页面的适当位置加入改变页面风格的超级链接，并在其onClick事件中调用change()方法改变页面整体风格，代码如下。

自选风格：

```
<a href="#" onClick="change('myCss','css1.css')">[秋叶飘香/经典]</a>
```

```
<a href="#" onClick="change('myCss','css.css')">[灰色畅想/舒畅]</a>
```

举一反三

根据本实例，读者可以：

实现更多风格样式的网页换肤；

实现只在初次运行页面时，进行网站风格设定的网站初始化功能。

## 实例320 滚动文字

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\13\320

实例说明

在页面中加入滚动文字，不但可以增强网站的动感，而且可以节省页面空间。本实例将介绍如何实现滚动文字。运行本实例，如图13.5所示，页面中的最新公告将自下而上缓慢滚动，当鼠标移动到滚动文字上时，滚动的速度将降低，当在滚动文字上按住鼠标按键时，滚动文字将向下快速滚动。



图13.5 滚动文字

### 技术要点

本实例主要应用<marquee>标记实现。

<marquee>标记用于实现文字或图片等HTML元素的滚动效果，其语法格式如下。

<marquee>滚动内容</marquee>

<marquee >标记的常用属性如表13.5 所示。

表13.5 <marquee>标记的属性

属 性	描 述
align	滚动内容的对齐方式，可以取值为 top: 顶端对齐 middle: 居中对齐 bottom: 向下对齐
direction	滚动的方向，分为 up (向上)、down (向下)、left (向左) 或 right (向右)
behavior	滚动的方式，可以取值为 scroll: 循环滚动 slide: 一次滚动 alternate: 交替滚动

续表

属 性	描 述
loop	循环滚动的次数，取值为-1 或者 Infinite 表示无限次滚动
scrollamount	滚动的速度，单位为像素，值越大滚动速度越快
scrolldelay	两次滚动的间隔时间，即每一次滚动的间隔产生的时间延迟
width	滚动区域的宽度，单位为像素或者以百分比形式表示
height	滚动区域的高度，单位为像素或者以百分比形式表示
bgcolor	滚动区域的背景颜色，使用颜色的英文名称或者以十六进制形式表示
hspace	滚动区域与浏览器边界的水平距离
vspace	滚动区域与浏览器边界的垂直距离

## 实现过程

首先将要显示的滚动文字放置在表格中，然后将该表格放置在<marquee >和</marquee >标记之间，并通过<marquee>标记的相关属性控制滚动方式，关键代码如下：

```
<marquee direction="up"
onMouseOver="this.scrollAmount='1'"
onMouseOut="this.scrollAmount='2'" onMouseDown=" this.
scrollAmount='4';this.direction='down'"
onMouseUp="this.scrollAmount='1';this.direction='up'"
scrollAmount="2" height="291">
```



实现电子商城中的最新公告栏目的滚动效果；  
实现网上书店中的新书公告的滚动效果；  
实现网上论坛中的版主公告的滚动效果。

## 实例321 制作渐变背景

本实例可以美化界面

实例位置：光盘\mingrisoft\13\321

实例说明

在浏览网页时，经常会看见页面背景的渐变效果，它可以使页面更加美观。本实例将通过CSS样式来设置页面的渐变效果。运行结果如图13.6所示。



图13.6 制作渐变背景

技术要点

本实例主要应用CSS样式的Alpha滤镜实现。

Alpha滤镜的语法格式如下。

```
{filter: Alpha(Opacity=value, Finishopacity=value,  
Style=value, Startx=value,  
Starty=value, Finishx=value, Finishy=value)}
```

参数说明如下。

● **Opacity**: 代表透明水准, 默认的范围为0~100, 0代表完全透明, 100代表完全不透明。

● **Finishopacity**: 是一个可选参数。设置渐变的透明效果时, 就用这个参数来指定结束时的透明度, 范围为0~100。

● **Style**: 指定透明区域的形状特征, 0 代表统一形状, 1 代表线形, 2 代表放射状, 3 代表长方形。

● **Startx**: 渐变透明效果的开始x 坐标。

● **Starty**: 渐变透明效果的开始y坐标。

● **Finishx**: 渐变透明效果的结束x 坐标。

● **Finishy**: 渐变透明效果的结束y坐标。

功能: 设置透明层次。Alpha 属性是把一个目标元素与背景混合。可以通过指定数值来控制混合的程度。

实现过程

使页面背景渐变的CSS样式。代码如下。

```
<style type="text/css">
<!--
body {
    background:blue;/*设置页面的背景颜色*/
    filter:Alpha(Opacity=20,Finishopacity=80,Style=3,Startx
=100,Starty=100,
    Finishx=0,Finishy=0); /*设置透明的样式及透明的大小*/
}
-->
</style>
```

举一反三

根据本实例, 读者可以:

使页面背景水平渐变;

使文字背景渐变。

## 13.2 表格样式

在网页制作时，经常会用 CSS 样式对表格的样式进行设置，使用 CSS 样式可以很方便地对表格进行各种样式的转换。下面将通过几个典型实例进行详细介绍。

### 实例322 只有外边框的表格

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\13\322

实例说明

在浏览网页时，经常会看见一些表格只具有外边框。本实例将通过CSS样式来实现这一效果。运行结果如图13.7所示。



图13.7 只有外边框的表格

### 技术要点

本实例主要应用CSS样式的边框复合属性border实现。

border属性的语法格式如下。

```
border: border-width || border-style || border-color
```

参数说明如下。

- border-width: 用于设置表格的边框宽度。
- border-style: 用于设置表格的边框样式。
- border-color: 用于设置表格的边框颜色。

功能: 对表格边框的宽度、样式和颜色进行设置。

### 实现过程

(1) 使表格只有外边框的CSS样式。代码如下。

```
<style type="text/css">
. tableBorder {
    border: #407D2A 1px solid
}
</style>
```

(2) 在需要只有外边框效果的表格的class属性中应用CSS样式tableBorder，关键代码如下。

```
<table width="100%" height="492" border="0"
cellpadding="0" cellspacing="0"
class="tableBorder">
```

举一反三

根据本实例，读者可以：

在数据统计的页面中设置只有外边框的表格；

在大型网站的数据表中设置只有外边框的表格。

## 实例323 彩色外边框的表格

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\323

实例说明

在制用网站时，为了使网站中的数据表更加明显，可以将表格的4个边框设为不同的颜色。本实例将通过CSS样式来设置表格的外边框颜色。运行结果如图13.8所示。

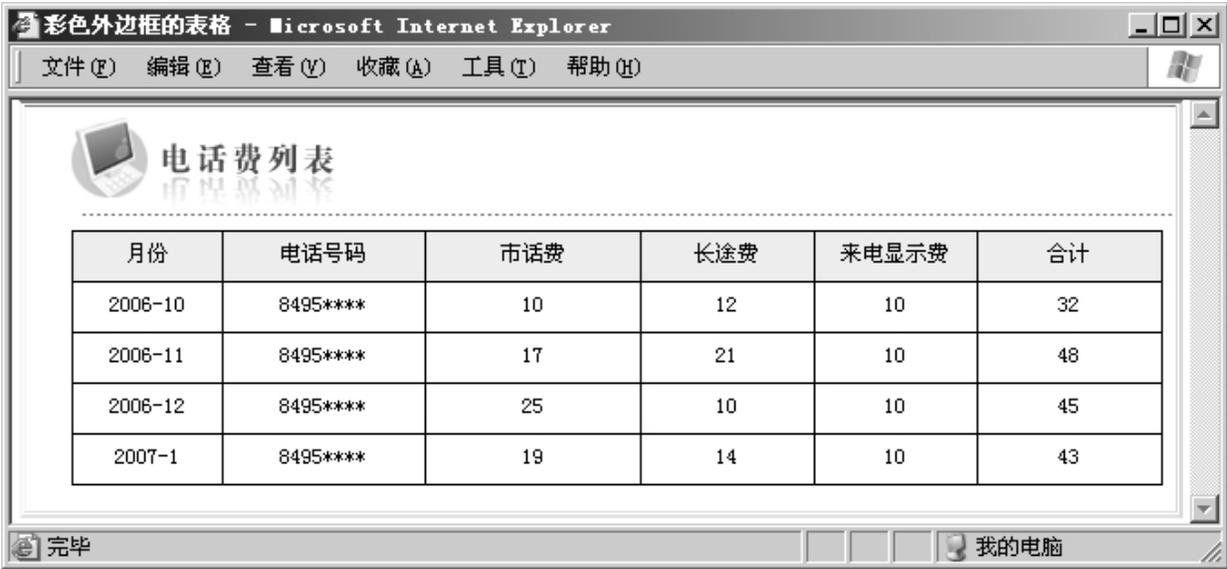


图13.8 彩色外边框的表格

### 技术要点

本实例主要应用CSS样式的边框属性实现。

边框属性用于设置一个元素边框的宽度、样式和颜色。边框属性如表13.6所示。

表13.6 边框属性

边框属性	描述
border	边框复合属性
border-top	上边框
border-left	左边框
border-right	右边框
border-bottom	下边框
border-color	边框颜色
border-style	边框样式
border-width	边框宽度

续表

边框属性	描述
border-top-color	上边框颜色
border-left-color	左边框颜色
border-right-color	右边框颜色
border-bottom-color	下边框颜色
border-bottom-style	下边框样式
border-top-style	上边框样式
border-left-style	左边框样式
border-right-style	右边框样式
border-top-width	上边框宽度
border-left-width	左边框宽度
border-right-width	右边框宽度
border-bottom-width	下边框宽度

下面对border-style的属性值进行详细的介绍。表13.7中的border-top-style、border-bottom-style、border-left-style和border-right-style的属性值与border-style的属性值相同。

border-style用来设置边框样式，其语法格式如下。

border-style: none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset

border-style的属性值如表13.7所示。

表13.7 border-style属性值

border-style 属性值	描 述
none	无边框
hidder	隐藏边框 IE 不支持
dotted	边框由点组成
dashed	边框由短线组成
solid	边框是实线
double	边框是双实线
groove	边框带有立体感的沟槽
ridge	边框成脊形
inset	边框内嵌一个立体边框
outset	边框外嵌一个立体边框

### 实例说明

(1) 使表格具有彩色外边框的CSS样式。代码如下。

```
<style type="text/css">
<!--
. table1{
border-bottom-color:#00FFFF;
border-left-color:#FFFF00;
border-top-color:#FF0000;
border-right-color:#00FF00;
border-style:double;
}
-->
</style>
```

(2) 在需要设置成彩色外边框效果的表格的class属性中应用CSS样式table1, 关键代码如下。

```
<table width="643" border="0" align="center"
cellpadding="0" cellspacing="0"
class="table1">
```

举一反三

根据本实例，读者可以：

使表格的外边框具有霓虹灯效果；

选中单元格时，使单元格的边框变色。

## 实例324 控制表格指定外边框不显示

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\13\324

实例说明

在浏览网页时，会看到各种样式的表格。本实例将通过表格边框的显示与否来对表格进行样式的控制。图13.9所示的表（1）只显示表格的外边框，表（2）将去除表格的所有边框。

商品ID	商品名称	单位	单价
0001	苹果	箱	30
0002	XX香烟	条	50
0003	XX方便面	箱	20

表(1)

商品ID	商品名称	规格	单价
0001	苹果	箱	30
0002	XX香烟	条	50
0003	XX方便面	箱	20

表(2)

图13.9 控制表格指定外边框不显示

技术要点

本实例主要应用CSS样式的边框属性实现，下面对各个表格的设置情况进行说明。

（1）表1主要是用CSS样式tablecss1来设置表格的外边框样式、宽度和颜色。

（2）表2主要是用CSS样式tablecss2来设置表格的单元格边框为0，并设置单元格的背景颜色。

CSS样式的边框属性已在实例323中进行了详细的说明。

## 实现过程

(1) 编写格边框是否显示的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

设置表格单元格的字体大小（对所有表格的字体进行设置）。

```
table {  
font-size:12px;}
```

设置表格的外边框样式、宽度及颜色（设置表1中的外边框）。

```
. tablecss1 {  
border-top-style: double;  
border-right-style:double;  
border-bottom-style:double;  
border-left-style:double;  
border-width:2px;  
border-color:#6633FF;  
}
```

设置单元格的外边框宽度及背景颜色（对表2的单元格进行设置）。

```
. tablecss2 {  
background: #DDDDDD;  
border-width:0px;  
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加表格，并在表格中添加层。代码如下。

```
<tablewidth="800"height="340"border="0"cellpadding="0"  
cellspacing="0">
```

```
<tr>
  <tdwidth="400"height="160"><div align="center">
    <table
class="tablecss1"width="328"height="147"border="0"
cellpadding="0"
cellspacing="0">
  <tr>
    <td>商品ID</td>
    <td>商品名称</td>
    <td>单位</td>
    <td>单价</td>
  </tr>
  <tr>
    <td>0001</td>
    <td>苹果</td>
    <td>箱</td>
    <td>30</td>
  </tr>
  <tr>
    <td>0002</td>
    <td>XX香烟</td>
    <td>条</td>
    <td>50</td>
  </tr>
  <tr>
    <td>0003</td>
    <td>XX方便面</td>
```

```

        <td>箱</td>
        <td>20</td>
    </tr>
</table>
<p>表(1)</p>
</div></td>
<tdwidth="400"height="160"><div align="center">
    <table
class="tablecss2"width="328"height="147"border="0"
cellpadding="0"
    cellspacing="0">
        .
        .
        /*与表1中的单元格设置相同，由于篇幅有限，在这里省
略*/
        .
        .
    </table>
    <p>表(2)</p>
</div></td>
</tr>
</table>

```

### 举一反三

根据本实例，读者可以：

在网站中以不同的表格风格进行显示数据；

用无边框的表格作页面摆放的容器。

## 实例325 背景颜色渐变的表格

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\325

实例说明

在制作网页时，为了使页面中的表格更加醒目，可以将表格的背景颜色设置为渐变效果。本实例将通过CSS样式来使表格的背景颜色具有渐变的效果。运行结果如图13.10所示。



	姓名	职务
LBFF公司	张先生	总经理
RTT公司	王小姐	公关部主任

图13.10 背景颜色渐变的表格

技术要点

本实例主要应用CSS样式的progid滤镜实现。

progid滤镜的语法格式如下。

filter :

```
progid:DXImageTransform.Microsoft.Gradient(enabled=bEnabled, startColorStr=  
=iWidth, endColorStr=iWidth )
```

参数说明如下。

● enabled: 可选项，布尔值 (Boolean)，置或检索滤镜是否激活。True 为默认值，滤镜激活。False滤镜被禁止。

● startColorStr: 可选项，字符串 (String)，设置或检索色彩渐变的开始颜色和透明度，其格式为#AARRGGBB。AA、RR、GG、BB为十六进制正整数，取值范围为00~FF。RR指定红色值，GG指定绿色值，BB指定蓝色值。AA指定透明度，00是完全透明，FF是完全不透

明。超出取值范围的值将被恢复为默认值，取值范围为#00000000～#FFFFFFF，默认值为#FF0000FF，即不透明蓝色。

● endColorStr: 可选项，字符串（String），设置或检索色彩渐变的结束颜色和透明度。参阅startColorStr属性。其默认值为#FF000000，不透明黑色。

### 实现过程

(1) 使表格背景颜色渐变的CSS样式。代码如下。

```
<style type="text/css">
<!--
设置页面中所有表格的背景颜色渐变。
table{
border-width:2px;
font-size:12px;
filter:progid:dximageTransform.Microsoft.Gradient(fradien
tType=1,startColorStr=
#46BEFF,endColorStr=#FFFFFF);
}
将单元格的边框宽度设为0，使单元格边框不可见。
td{
border-width:0px;
}
-->
</style>
```

(2) 在页面中添加表格，代码如下。

```
<tablewidth="264"height="100"border="1">
<tr>
<tdwidth="57">&nbsp;   </td>
```

```

        <tdwidth="85"><div align="center">姓名</div></td>
        <tdwidth="100"><div align="center">职务</div></td>
</tr>
<tr>
    <td>LBFF公司</td>
    <td><div align="center">张先生</div></td>
    <td><div align="center">总经理</div></td>
</tr>
<tr>
    <td>RTT公司</td>
    <td><div align="center">王小姐</div></td>
    <td><div align="center">公关部主任</div></td>
</tr>
</table>

```

举一反三

根据本实例，读者可以：

实现层的背景颜色渐变。

## 实例326 表格隔行变色

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\326

实例说明

在浏览网页时，经常会有一些表格以特殊的效果进行显示。本实例将通过CSS样式对表格进行隔行变色。运行结果如图13.11所示。



图13.11 表格隔行变色

### 技术要点

本实例主要应用split ()方法来修改CSS 样式中表格的background-color (背景色)和color (前景色)属性值。

split ()方法的语法格式如下。

```
stringObj.split(str)[Index]
```

参数说明如下。

- stringObj: 必选项, 要被分解的String 对象或文字。该对象不会被 split()方法修改。

- str: 必选项, 是一个字符串或正则表达式对象, 描述用来定义分解位置的字符。

- index: 索引, 下一个匹配从该索引处开始。

功能: split()方法的结果是一个字符串数组, 在 stingObj 中每个出现 str 的位置都要进行分解。

expression用来把CSS属性和Javascript表达式关联起来, 这里的CSS属性可以是元素固有的属性, 也可以是自定义属性。也就是说 CSS

属性后面可以是一段 Javascript 表达式，CSS属性的值等于 Javascript表达式计算的结果。在表达式中可以直接引用元素自身的属性和方法，也可以使用其他浏览器对象。这个表达式就好像是这个元素的一个成员函数一样。

### 实现过程

(1) 使表格隔行变色的CSS样式。代码如下。

```
<style>
<!--
.DoubleColorTable tr{
    font-size:12px;
    background-
color:expression("#DEF1FE,#FFFFFF".split(",")[rowIndex%2]);
    color:expression("#FF3300,#0000FF".split(",")
[rowIndex%2])
}
-->
</style>
```

(2) 在需要设置成隔行变色的表格的class属性中应用CSS样式 DoubleColorTable，关键代码如下。

```
<table class="DoubleColorTable" width="100%" border="0"
cellspacing="0"
cellpadding="0" height="189">
```

### 举一反三

根据本实例，读者可以：

使表格隔列变色；

当选中单元格时，使单元格所在行变色。

## 13.3 鼠标及滚动条样式

在浏览网页时，当鼠标经过不同的组件或进行操作时，鼠标的样式会进行相应的改变，网页开发者也可将鼠标设置成 Flash 动画的形式。为了使页面更加美观，开发者还可以改变页面的滚动条颜色。

### 实例327 显示自定义鼠标形状

这是一个实用的实例

实例位置：光盘\mingrisoft\13\327

实例说明

在制作网站时，可以利用鼠标的形状来判断对表格、图片等控件的相关操作。本实例中当鼠标移入图片、表格和超级链接时，鼠标的形状将改变，当移出时，恢复原始状态。运行结果如图13.12所示。

技术要点

本实例主要应用CSS样式的cursor属性实现。



图13.12 显示自定义鼠标形状

cursor属性的语法格式如下。

```
cursor: value
```

具体的cursor属性值如表13.8所示。

表13.8 鼠标光标属性值

鼠标光标属性值	描 述
Hand	手
Crosshair	交叉十字
Text	文本选择符号
Wait	Windows 的沙漏形状
Default	默认的鼠标形状
Help	带问号的鼠标
Move	移动的箭头
e-resize	向东的箭头
ne-resize	向东北的箭头
n-resize	向北的箭头
nw-resize	向西北的箭头
w-resize	向西的箭头
sw-resize	向西南的箭头
s-resize	向南的箭头
se-resize	向东南的箭头

功能：通过设置CSS样式鼠标光标属性，当将鼠标移动到链接上时，可以看到多种不同的效果。

实现过程

(1) 更改编写鼠标显示形状的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

设置页面的字体大小。

```
body {font-size:12px;}
```

当鼠标移入表格时，将鼠标效果改为移动的箭头。

```
table {
```

```
font-size:12px;
```

```
cursor:move;
}
```

当鼠标移入单元格时，将鼠标效果改为文本选择符号。

```
tr {cursor:text;}
```

当鼠标移入图片时，将鼠标效果改为手状。

```
img {cursor:hand;}
```

当鼠标移入超级链接时，将鼠标效果改为带问号的鼠标。

```
a {cursor:help;}
```

```
-->
```

```
</style>
```

(2) 在页面中添加图片、表格和超级链接。代码如下。

```

```

```
<br>
```

```
<br>
```

```
<tablewidth="308"height="110"border="1">
```

```
<tr>
```

```
<tdwidth="70"height="26">&nbsp;</td>
```

```
<tdwidth="70"height="26">姓名</td>
```

```
<tdwidth="70"height="26">年龄</td>
```

```
<tdwidth="70"height="26">职务</td>
```

```
</tr>
```

```
<tr>
```

```
<tdwidth="52" rowspan="3">RTT公司</td>
```

```
<tdwidth="52"height="26">李先生</td>
```

```
<tdwidth="52"height="26">28</td>
```

```
<tdwidth="52"height="26">经理</td>
```

```
</tr>
```

```
<tr>
  <tdheight="26">王先生</td>
  <td>31</td>
  <td>科长</td>
</tr>
<tr>
  <tdheight="26">孙小姐</td>
  <td>22</td>
  <td>主任</td>
</tr>
</table>
<br>
<a href="www.mingrisoft.com">明日科技</a>
```

举一反三

根据本实例，读者可以：

设计当鼠标移到表格的左右边框时，将鼠标改为左右箭头，当鼠标移动到表格的上下边框时，将鼠标改为上下箭头；  
用鼠标来判断表格中的数据是否能修改。

## 实例328 动画光标

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\13\328

实例说明

在浏览网页时，一些网站上经常会有鼠标以动画的形式进行显示。本实例实现当鼠标在页面工作区内时将改为指定的动画效果。运行结果如图13.13所示。

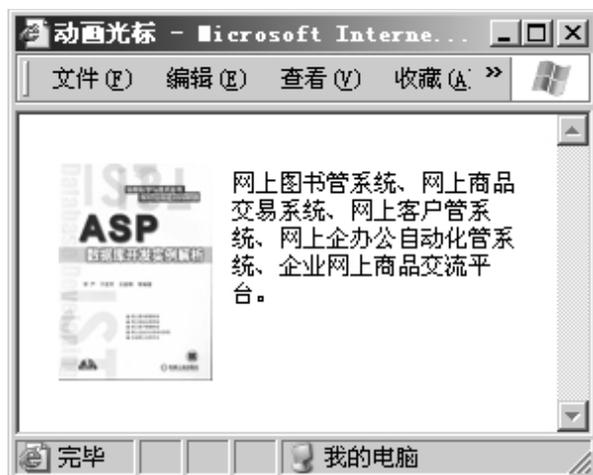


图13.13 动画光标

### 技术要点

本实例主要应用CSS样式的cursor属性实现。

cursor属性的语法格式如下。

cursor:url (cur/ani文件)

参数说明如下。

- url: 光标文件的链接地址。
- cur: 是Window中的光标 (cursor) 文件, 光标文件与图标 (ICON) 文件除了文件头有一个位置的值不同外, 实际是一样的。
- ani: 是Window中的动画光标 (图标) 文件。

功能: 用于改变Window中的光标文件。

### 实现过程

编辑鼠标动画效果的CSS样式。代码如下。

```
<style type="text/css">
<!--
Body {cursor: url('2.ani');}
table {font-size:12px;}
-->
</style>
```

举一反三

根据本实例，读者可以：

在页面中任意选择鼠标的形状；

当鼠标移入一定的区域内，使鼠标显示为动画状态。

## 实例329 制作彩色滚动条

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\329

实例说明

在制作网页时，为了让页面更加美观、更具有个性化，可以对页面的滚动条颜色进行相应的修改。本实例将通过CSS样式来改变滚动条的效果。运行结果如图13.14所示。



图13.14 制作彩色滚动条

技术要点

本实例主要应用CSS样式的滚动条属性实现，滚动条属性如表13.9所示。

表13.9 滚动条属性

属 性	描 述
scrollbar-Face-Color	设置或检索滚动条 3D 表面 (ThreedFace) 的颜色
scrollbar-Highlight-Color	设置或检索滚动条 3D 界面的亮边 (ThreedHighlight) 颜色
scrollbar-Shadow-Color	设置或检索滚动条 3D 界面的暗边 (ThreedShadow) 颜色
scrollbar-3dLight-Color	设置或检索滚动条亮边框颜色
scrollbar-Arrow-Color	设置或检索滚动条方向箭头的颜色, 当滚动条出现但不可用时, 此属性失效
scrollbar-Track-Color	设置或检索滚动条的拖动区域 (TrackBar) 颜色
scrollbar-Dark-shadow-Color	设置或检索滚动条暗边框 (ThreedDarkShadow) 颜色

注意: 滚动条属性只适用于IE浏览器。

实现过程

更改编辑页面滚动条颜色的CSS样式。代码如下。

```
<style type="text/css">
<!--
body {
    font-size:9pt;
    margin-left:0px;
    margin-top:0px;
    margin-right:0px;
    margin-bottom:0px;
    background-image:url(images/bg.gif);
    scrollbar-Face-Color:#107483;
    scrollbar-Highlight-Color:#ffffff;
    scrollbar-Shadow-Color:#fcfcfc;
    color:#000000;
    scrollbar-3dLight-Color:#ecec;
    scrollbar-Arrow-Color:#ffffff;
    scrollbar-Track-Color:#ecec;
    scrollbar-Dark-shadow-Color:#107483;
```

```
background-color: #fcfcfc;  
}  
-->  
</style>
```

举一反三

根据本实例，读者可以：

设置当鼠标移动到滚动条时，改变滚动条的颜色。

## 13.4 文字及列表样式

在网页制作时，有时需要对文字及列表样式进行更改，使用CSS样式可以很方便地对其进行修改。本节将通过几个具体实例介绍如何应用CSS样式改变文字及列表的样式。

### 实例330 应用删除线样式标记商品特价

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\13\330

实例说明

在浏览一些网络购物的网站时，商家为了能够更好地推销商品，一般对商品进行打折销售。为了能让顾客清楚看到省了多少钱，商家一般在商品的价位栏中显示两个价位，一个是原价，一个是现价。原价需要用删除线将其删除。运行结果如图13.15所示。



图13.15 应用删除线样式标记商品特价

### 技术要点

本实例主要应用CSS样式的字体属性中的text-decoration属性实现。

text-decoration属性的语法格式如下。

```
text-decoration: underline | overline | line-through |  
blink | none
```

参数说明如下。

- underline: 文字加下划线。
- overline: 文字加上划线。
- line-through: 文字加删除线。
- blink: 闪烁文字, 只有Netscape 浏览器支持。
- none: 默认值。

功能：设定文字修饰。

实现过程

(1) 使文字具有删除线效果的CSS样式。代码如下。

```
<style type="text/css">
<!--
font {
    text-decoration:line-through;
    color:#FF0000;
    font-size:9pt;
}
-->
</style>
```

(2) 将设置带有删除线的文字放入<font>标记中，详细代码请参见本书附带的光盘。

举一反三

根据本实例，读者可以：

使文字具有下划线效果；

将文字设置成自制的超级链接效果。

## 实例331 在文字上方标注说明标记

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\13\331

实例说明

在设计网站时，有时需要在文字的上方加入标注说明文字，这可以通过在文字上方加入标注说明标记实现。本实例将介绍如何应用该

标记在文字上方加入标注说明。运行本实例，在页面的左侧将显示英文语句及其中文说明，如图13.16所示。

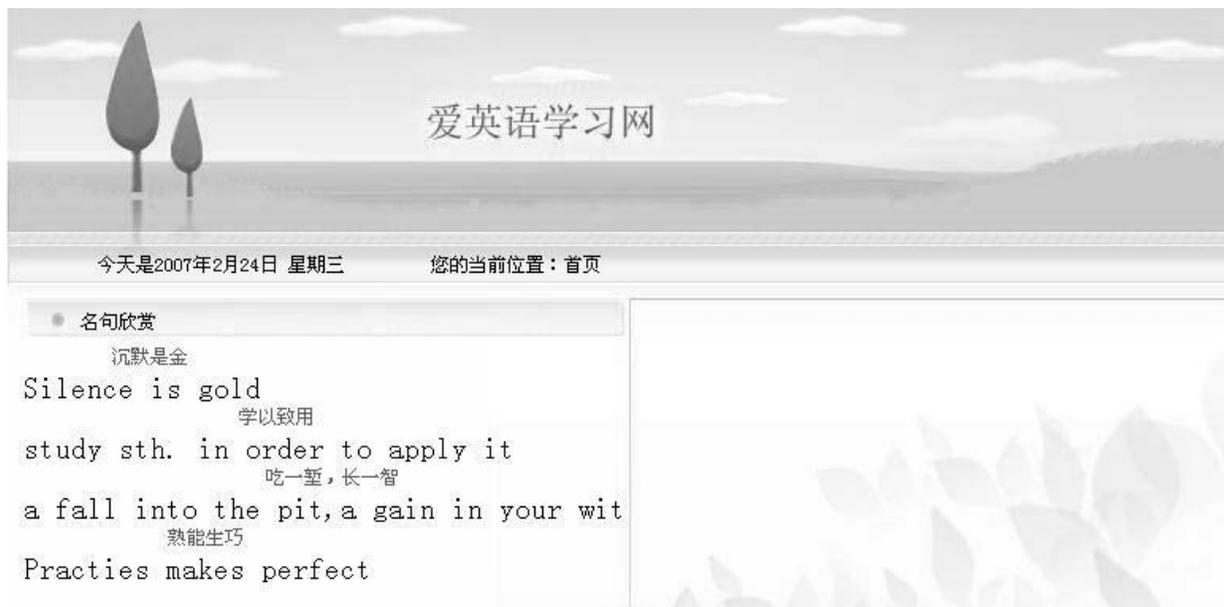


图13.16 在文字上方标注说明标记

### 技术要点

本实例主要通过文字上方标注说明标记<rt>和<ruby>实现。在HTML语言中，被<rt>标记括起来的文字将以缩小的字号形式显示在以<ruby>标记括起来的文字上方，用来说明这段文字。在文字上方标注说明标记<rt>和<ruby>的语法格式如下。

```
<ruby>
```

```
  被说明的文字
```

```
  <rt>说明文字</rt>
```

```
</ruby>
```

说明：在应用<ruby>和<rt>标记时，说明文字将比被说明的文字的字号小一级。为了更突出被说明的文字，可以将说明文字的颜色标记为灰色。

### 实现过程

在页面中加入在文字上方标注说明的标记，并设置相应的说明文字，关键代码如下。

```
<ruby>
  Silence is gold
  <rt><font color="#666666">沉默是金</font></rt>
</ruby>
<ruby>
  study sth. inorder to apply it
  <rt><font color="#666666">学以致用</font></rt>
</ruby>
<ruby>
  a fall into thepit,a gain inyourwit
  <rt><font color="#666666">吃一堑，长一智</font></rt>
</ruby>
<ruby>
  Practiesmakesperfect
  <rt><font color="#666666">熟能生巧</font></rt>
</ruby>
```

举一反三

根据本实例，读者可以：

在小学生网站中为每个栏目名称加上拼音标注；

实现中英文双语网站。

### 实例332 指定图标的列表项

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\13\332

## 实例说明

将同一页面中的并列信息通过列表项显示，不但可以统一页面整体风格，而且可以提高制作速度。在默认情况下，通过列表标记实现的列表项只能有“●”、“○”和“■”3种符号。本实例将介绍如何通过CSS样式为列表项指定图标。运行本实例，在页面的左侧将以带图标的列表项显示关于注册的帮助信息，如图13.17所示。

## 技术要点

本实例主要通过<ul>标记的style属性的list-style-image属性实现。list-style-image属性用于设置列表的图像样式，此规则将影响到列表的所有列表项，语法格式如下。

```
style="list-style-image:url(ico.gif)"
```



The screenshot shows a registration form on the M-R Blog website. The form is titled "注册帮助" (Registration Help) and includes the following fields and instructions:

- 用户名:** 为用户登录网站的通行证，长度控制在3-20个字符之内。
- 密码:** 请设定在6-20位之间。
- 真实姓名:** 真实姓名由汉字组成。
- Email:** 为方便与您沟通，请填写真实有效的Email地址。

The form also includes fields for "密码:", "确认密码:", "性别:" (with radio buttons for "男" and "女"), "真实姓名:", "生日:" (with a date format instruction: "日期格式为: yyyy-mm-dd"), "OICQ号码:", and "Email:". The form is styled with a light background and a decorative banner at the top.

图13.17 指定图标的列表项

参数说明如下。

- ico.gif: 为设置列表项图标的图片名称。

## 实现过程

在页面的适当位置添加列表项并设置其Style属性，关键代码如下。

```
<ul style="list-style-image:url(ico.gif)">
  <li>
    用户名：为用户登录网站的通行证，长度控制在3-20个字符之内。<br>
  </li>
  <li>密码：请设定在6-20位之间。<br>
  </li>
  <li>真实姓名：真实姓名由汉字组成。<br></li>
  <li>E-mail：为方便与您沟通，请填写真实有效的E-mail地址。</li>
</ul>
```

## 举一反三

根据本实例，读者可以：

实现电子商城中的商品类别列表；

实现通过列表项显示中介信息。

## 13.5 文字滤镜特效

在浏览网页时，经常会有一些特殊效果的文字，它们大多数都是用CSS样式进行设置的。使用 CSS 样式不但可以降低代码的复杂性，而且可以多次调用同一个 CSS 样式，也可以将多个CSS样式相结合，形成新的效果。下面将通过几个典型实例进行详细讲解。

### 实例333 文字的发光效果

这是一个实用的实例

实例位置：光盘\mingrisoft\13\333

实例说明

在浏览网页时，经常会有一些文字具有发光的效果。本实例将通过CSS样式来设置文字的发光效果。运行结果如图13.18所示。



图13.18 文字的发光效果

技术要点

本实例主要应用CSS样式的glow滤镜实现，下面对glow滤镜进行详细的说明。

glow滤镜的语法格式如下。

```
{filter: glow(color=color_value, Strength=value)}
```

参数说明如下。

- Color：设定边缘光晕的颜色。

● Strength: 设定边缘光晕的强度大小, 范围是1~255。

功能: 设置边缘光晕。

实现过程

(1) 使文字发光的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

使文字发光。

```
.fontcss {  
    filter: glow(color=#33FF00,direction=2)  
}
```

设置层中文字的大小及样式。

```
.spancss {  
    font-size: 40px;  
    font-style: italic;  
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加表格, 并在表格中添加层。代码如下。

```
<table align="center" width="200" height="60"  
class="fontcss" border="0">  
    <tr>  
        <td height="60" align="center"><span class="spancss">  
<FONT face="Arial Black"  
        color=#33CC00>mingrisoft</FONT></span></td>  
    </tr>  
</table>
```

举一反三

根据本实例，读者可以：  
实现文字的闪烁发光效果；  
使文字的发光效果越来越强。

## 实例334 文字的阴影效果

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\334

实例说明

在浏览网页时，经常会有一些文字具有阴影效果。本实例将通过CSS样式来设置文字的阴影效果。运行结果如图13.19所示。



图13.19 文字的阴影效果

技术要点

本实例主要应用CSS样式的dropshadow滤镜实现。

dropshadow滤镜的语法格式如下。

```
{filter: dropshadow(Color=value, Offx=value, Offy=value,  
Positive=value)}
```

参数说明如下。

- Color: 阴影的颜色。
- Offx: x 轴方向阴影的偏移量，正数表示沿x 轴右方向偏移，负数表示沿x 轴左方向偏移。
- Offy: y 轴方向阴影的偏移量，正数表示沿y 轴向下方向偏移，负数表示沿 y 轴向上方向偏移。
- Positive: 取值为 True 或者 False。True 表示为不透明像素建立可见的阴影；False 表示为透明的像素建立可见的阴影。

功能：阴影效果。

实现过程

(1) 使文字具有阴影效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

文字的阴影效果。

```
.fontcss {
```

```
    filter:
```

```
    dropshadow(color=#AAAAAA, offx=2, offy=2, positive=4)
```

```
}
```

设置层中文字的大小及样式。

```
.spancss {
```

```
    font-size: 40px;
```

```
    font-style: italic;
```

```
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加表格，并在表格中添加层。代码如下。

```
<table align="center" width="200" height="60"
```

```
class="fontcss" border="0">
```

```
    <tr>
```

```
        <td height="60" align="center"><span class="spancss">
```

```
<FONT face="Arial Black">
```

```
    mingrisoft</FONT></span></td>
```

```
    </tr>
```

```
</table>
```

举一反三

根据本实例，读者可以：  
实现文字阴影的动态旋转；  
实现文字阴影从浅到深的动态效果。

## 实例335 文字的渐变阴影效果

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\335

实例说明

在浏览网页时，经常会看见一些文字具有渐变的阴影效果。本实例将通过CSS样式来设置文字的渐变阴影效果。运行结果如图13.20所示。



图13.20 文字的渐变阴影效果

技术要点

本实例主要应用CSS样式的Shadow滤镜实现。

Shadow滤镜的语法格式如下。

```
{filter: Shadow(Color=color_value, Direction=value)}
```

参数说明如下。

- Color: 设定渐变阴影的颜色。
- Direction: 设定渐变阴影的方向。渐变阴影是按顺时针方向进行的，0° 代表垂直向上，然后每45° 为一个单位。默认值是向左270° 。

功能：渐变阴影效果。

实现过程

- (1) 使文字具有渐变阴影效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

文字的渐变阴影效果。

```
.fontcss {
```

```
  FILTER: shadow(color=#66cc99,direction=135)
```

```
}
```

设置层中文字的大小及样式。

```
.spancss {
```

```
  font-size: 40px;
```

```
  font-style: italic;
```

```
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加表格，并在表格中添加层。代码如下。

```
<table align="center" width="200" height="60"
```

```
class="fontcss" border="0">
```

```
<tr>
```

```
<td height="60" align="center"><span class="spancss">
```

```
<FONT face="Arial Black"
```

```
color=#33CC00>mingrisoft</FONT></span></td>
```

```
</tr>
```

```
</table>
```

举一反三

根据本实例，读者可以：

改变阴影渐变色的范围；

使渐变阴影动态旋转。

## 实例336 文字的图案填充效果

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\13\336

实例说明

所谓文字的图案填充效果是指用指定的图案填充文字，在页面中采用这种效果，不但可以美化页面，而且可以方便用户随时更新页面信息。例如，将电子商城中的网站公告采用图案填充效果，可以随时更新公告内容，而不需要再去修改图片。运行本实例，在页面中将显示以“”为基准图片的公告信息“欢迎访问E-SHOP! ”，如图13.21所示。



图13.21 文字的图案填充效果

技术要点

本实例主要应用了CSS样式的chroma和dropshadow滤镜，其中，dropshadow滤镜的详细介绍请参见实例334，下面详细介绍chroma滤镜。

chroma滤镜用于设置指定颜色透明，其语法格式如下。

```
{filter: chroma(Color=color_value)}
```

参数说明如下。

● Color: 要透明的颜色, 如 Color 的值为#CCCCCC, 则滤镜作用范围内所有颜色为“#CCCCCC”的像素都变为透明, 包括图片的像素。

### 实现过程

(1) 设计一个作为文字填充图案的图片, 大小为8×8像素, 文件名为 bg.gif。为了方便读者观看, 笔者将其放大到48×48, 如图13.22所示。



图13.22 48×48像素填充图案效果图

(2) 在页面的适当位置添加一个<div>标记, 并通过其style属性设置背景图像及宽度, 关键代码如下。

```
<div style="background-image:url(bg.gif);width:100%;">  
</div>
```

(3) 在步骤(2)中添加的<div>标记中间添加一个<div>标记, 并通过其 style 属性设置文字的字体、字号、背景颜色及阴影效果, 并将文字的背景设为透明效果, 这样可以通过该文字看到背景图片, 以达到图案填充效果。关键代码如下。

```
<div style="FILTER: chroma( color=#CCCCCC)  
    dropShadow(Color=#777777, offX=-1, offY=-1, positive=2);  
    color:#CCCCCC;background-color:#FFFFFF;width:100%;  
font:bold50pt黑体; ">
```

欢迎访问E-SHOP!

</div>

举一反三

根据本实例，读者可以：

通过修改基准图片，实现其他样式的图案填充效果；

实现镂空文字。

## 实例337 文字的探照灯效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\337

实例说明

在页面设计中，为了使页面更加生动有趣，可以使一些文字具有探照灯的效果，当鼠标在文字上移动时，文字上方将会有类似于灯光的照明效果。本实例将用CSS样式来实现这一效果。运行结果如图13.23所示。

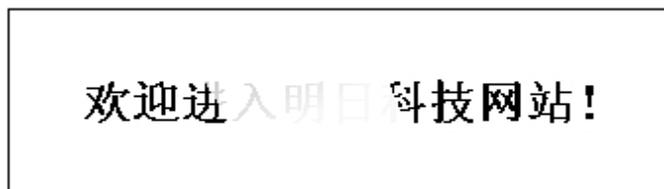


图13.23 文字的探照灯效果

技术要点

本实例主要应用CSS样式的light滤镜实现。

light 滤镜的语法格式如下。

```
{filter:light}
```

功能：这个属性模拟光源的投射效果。一旦为对象定义了light滤镜属性，那么就可以调用它的方法（Method）来设置或者改变属性。light滤镜的相关方法如表13.10所示。

表13.10 light滤镜的相关方法

方 法	描 述
AddAmbient	加入包围的光源
AddCone	加入锥形光源
AddPoint	加入点光源
Changcolor	改变光的颜色
Changstrength	改变光源的强度
Clear	清除所有的光源
MoveLight	移动光源

下面对light滤镜的AddAmbient、AddCone和MoveLight方法进行详细的说明。

AddAmbient方法的语法格式如下。

```
callobject.style.filters.Light(n).addAmbient(R, G, B, strength)
```

参数说明如下。

- R、G、B：决定了Ambient 光的颜色数。
- strength：决定了光投射的数量。

功能：加入包围的光源。

AddCone方法的语法格式如下。

```
callobject.style.filters.Light(n).AddCone(x1, y1, z1, x2, y2, R, G, B, strength, spread)
```

参数说明如下。

- x1、y1、z1：代表光源的位置。
- x2、y2、z2：代表目标点的位置。
- R、G、B：代表光的颜色数。
- strength：代表光投射的数量。

功能：加入锥形光源。

MoveLight方法的语法格式如下。

```
callobject.style.filters.Light(n).MoveLight(lightnumber, x, y, z, fAbsolute)
```

参数说明如下。

- **lightnumber**: 标识光源的数字。
- **x、y、z**: 光源的三维坐标。
- **fAbsolute**: 表示移动是相对的还是绝对的。

功能: 移动光源。

实现过程

(1) 使文字具有探照灯效果的CSS样式。代码如下。

```
<style>
<!--
  divcss {
    COLOR:white;FILTER:light;HEIGHT:300px;font-size:24px;
    LEFT:10px;POSITION:relative;TOP:10px;WIDTH:400px;
    FILTER: light;
  }
-->
</style>
```

(2) 在页面中添加一个层。代码如下。

```
<div id=Tdiv class="divcss"><P align=center>欢迎进入明日科技网站!</P></div>
```

(3) 编写用于实现文字探照灯效果的JavaScript代码。

```
<script language="javascript">
function illume() {
  document.all.Tdiv.filters.light.addambient(0, 0, 20, 30);
  document.all.Tdiv.filters.light.addcone(400, 400, 120, 160, 100,
```

```
255, 255, 80, 120, 5);}  
functionMove() {  
    document.all.Tdiv.filters.light.moveLight(1, window.event.  
t.x-20,  
window.event.y, 0, 1)  
}  
</script>
```

(4) 在窗体载入时，调用自定义函数illuminate()，当鼠标在工作区内移动时，调用自定义函数Move()，代码如下。

```
<body onload="illuminate();" onmousemove="Move()" >
```

举一反三

根据本实例，读者可以：

在页面中制作一个照明灯。

## 实例338 文字的闪烁效果

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\13\338

实例说明

在浏览页面时，经常会有一些文字具有动态的闪烁效果。本实例将实现：在窗体载入后，页面中的文字不断地闪烁，当鼠标指向文字时，停止文字的闪烁效果，如图13.24所示。

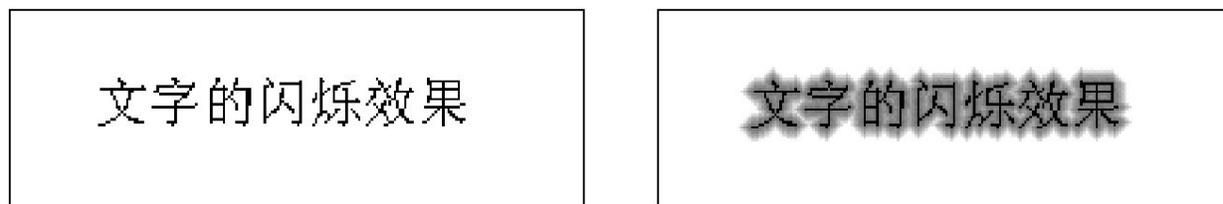


图13.24 文字的闪烁效果

技术要点

本实例主要应用CSS样式的glow滤镜实现，glow滤镜已在实例333中进行了详细介绍，下面主要介绍glow滤镜的enabled属性。

glow滤镜的enabled属性用来激活发光效果。当enabled属性为True时，激活glow滤镜；当enabled属性为False时，则关闭glow滤镜。enabled属性的默认值为True。在CSS样式中可以用1来表示激活状态。代码如下。

```
filter:glow(color=#FF0000, strength=2, 1)
```

#### 实现过程

(1) 使文字具有发光效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

实现文字的发光效果。

```
.glow1 {  
    position:absolute; width:296px; height:42px; z-index:1;  
    filter:glow(color=#FF0000, strength=2)  
}
```

设置层的相对位置。

```
.inside {  
    position:relative;top:10px;  
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加层及闪烁的文字。代码如下。

```
<div id="div1" align="center" class="glow1"  
onmouseover="stopflash(this)"  
onmouseout="init()"><div class="inside">文字的闪烁效果  
</div></font></div>
```

(3) 编写用于实现文字闪烁效果的JavaScript代码。

自定义函数init(), 用于调用自定义函数makeflash(), 代码如下。

```
<script language="javascript">  
function init() { //光晕开始闪烁  
    makeflash(div1);  
}
```

自定义函数makeflash(), 用于以1秒为间隔时间, 动态显示发光效果, 代码如下。

```
functionmakeflash(obj) { //在这里用1秒为间隔的时间  
    obj.flashTimer=setInterval("div1.filters.glow.enabled=!  
div1.filters.  
    glow.enabled",1000)  
}
```

自定义函数stopflash(), 用于清除setInterval()方法, 即停止文字的动态发光, 代码如下。

```
function stopflash(obj) { //光晕停止闪烁  
    clearInterval(obj.flashTimer)  
}  
</script>
```

举一反三

根据本实例, 读者可以:

使文字以不同的颜色进行动态的发光;

制作一个红绿灯发光效果。

## 实例339 文字的空心效果

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\13\339

实例说明

在浏览页面时，经常会看到一些文字以空心的效果进行显示。本实例将通过CSS样式来实现这一效果。运行结果如图13.25所示。



图13.25 文字的空心效果

技术要点

本实例主要应用CSS样式的glow、mask和chroma滤镜实现，下面对mask滤镜进行详细说明。

mask滤镜的语法格式如下。

```
{filter: mask(Color=color_value)}
```

参数说明如下。

● Color：遮罩所用的颜色。

功能：遮罩效果，mask滤镜将可看见的像素遮蔽，将看不见的像素以指定的颜色显示。

glow滤镜的使用请参考实例333。

chroma滤镜的使用请参考实例336。

实现过程

(1) 使文字具有空心效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

实现文字的空心效果。

```
.fontcss {
```

FILTER:

```
glow(strength=1)mask(color=blue)chroma(color=blue)
}
```

对文字的大小及样式进行设置。

```
.spancss {
    font-size: 40px;
    font-style: italic;
}
```

-->

```
</style>
```

(2) 在页面中添加一个表格。代码如下。

```
<table align="center" width="200" height="60"
class="fontcss" border="0">
    <tr>
        <td height="60" align="center"><span class="spancss">
<FONT face="Arial Black"
        >mingrisoft</FONT></span></td>
    </tr>
</table>
```

举一反三

根据本实例，读者可以：

使文字具有实体的金属效果。

## 实例340 文字的浮雕效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\340

## 实例说明

在浏览页面时，经常会有一些具有浮雕效果的文字。本实例将通过多种滤镜效果来自定义一个文字的浮雕效果。运行结果如图13.26所示。



图13.26 文字的浮雕效果

## 技术要点

本实例主要应用CSS样式的mask、shadow、dropshadow和chroma滤镜实现。先用mask滤镜将可看见的像素遮蔽，将看不见的像素以指定的颜色显示。然后用 shadow 滤镜使文字的右下方有渐变的阴影效果，用dropshadow滤镜使文字的左上方有阴影效果，最后用chroma滤镜将指定的颜色进行透明处理。这样就实现了文字的浮雕效果。

## 实现过程

(1) 使文字具有浮雕效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

实现文字的浮雕效果。

```
.fontcss {
```

```
    FILTER:
```

```
    mask(color=#EEEEEE) shadow(color=#66AAFF, direction=135)
```

```
    dropshadow(color=#6655FF, offx=-3, offy=-3, positive=2) chroma
```

```
    a(color=#EEEEEE)
```

```
}
```

设置层中文字的大小及样式。

```
.spancss {
```

```
font-size: 40px;
font-style: italic;
}
-->
</style>
```

(2) 在页面中添加一个表格。代码如下。

```
<table align="center" width="200" height="60"
class="fontcss" border="0">
  <tr>
    <td height="60" align="center"><span class="spancss">
<FONT face="Arial Black">
  mingrisoft</FONT></span></td>
  </tr>
</table>
```

举一反三

根据本实例，读者可以：

实现文字的空心粗体效果。

## 实例341 文字的阳文效果

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\13\341

实例说明

在浏览页面时，经常会看到一些具有阳文效果的文字。本实例将通过多种滤镜效果来实现文字的阳文效果。运行结果如图13.27所示。



图13.27 文字的阳文效果

### 技术要点

本实例主要应用CSS样式的mask滤镜实现文字的阳文效果。mask滤镜的介绍请参考实例339。

### 实现过程

(1) 使文字具有阳文效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

实现文字的阳文效果。

```
.fontcss {
```

```
    FILTER:
```

```
mask(color=white)dropshadow(color=royalblue, offx=-3, offy=-3  
, positive=1)
```

```
    chroma(color=white)
```

```
}
```

设置层中文字的大小及样式。

```
.spancss {
```

```
    font-size: 40px;
```

```
    font-style: italic;
```

```
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加一个表格。代码如下。

```
<table align="center" width="200" height="60"
class="fontcss" border="0">
  <tr>
    <td height="60" align="center"><span class="spancss">
<FONT face="Arial Black">
  mingrisoft</FONT></span></td>
</tr>
</table>
```

举一反三

根据本实例，读者可以：

设置一个带颜色的阳文效果；

设置文本的阴文效果。

## 实例342 文字的雪雕效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft1\3\342

实例说明

在浏览页面时，经常会看到文字的边缘具有雪花的效果。本实例将通过发光滤镜和阴滤镜来实现文字的雪雕效果。运行结果如图13.28所示。

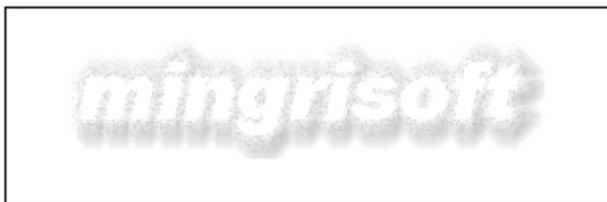


图13.28 文字的雪雕效果

技术要点

本实例主要应用CSS样式的glow和dropshadow滤镜实现。先用glow滤镜使文字以淡灰色发光，然后用dropshadow滤镜使文字出现淡蓝色的阴影效果，将这两个滤镜效果进行重叠，便形成了雪雕效果。

### 实现过程

(1) 使文字具有雪雕效果的CSS样式。代码如下。

```
<style type="text/css">
<!--
实现文字的雪雕效果。
.fontcss {
    filter:
glow(color=#FFFFFF, strength=12) dropshadow(color=#86BFFF, off
x=3,
    offy=3, positive=1);
}
```

设置层中文字的大小及样式。

```
.spancss {
    font-size: 40px;
    font-style: italic;
}
-->
</style>
(2) 在页面中添加一个表格。代码如下。
<table align="center" width="200" height="70"
class="fontcss" border="0">
<tr>
<td height="60" align="center"><span class="spancss">
<FONT face=
```

```
    "Arial Black" color="#FFFFFF">mingrisoft</FONT>  
</span></td>  
</tr>  
</table>
```

举一反三

根据本实例，读者可以：

实现雪雕效果并具有风吹动的效果。

## 实例343 火焰字

本实例是一个美化页面的程序

实例位置：光盘\mingrisoft\13\343

实例说明

在浏览页面时，经常会看到文字的火焰效果。本实例用glow滤镜使文字发出红光，然后用Blur 滤镜使文字向上模糊，最后用 Wave 滤镜使文字扭曲实现文字的燃烧效果。运行结果如图13.29所示。

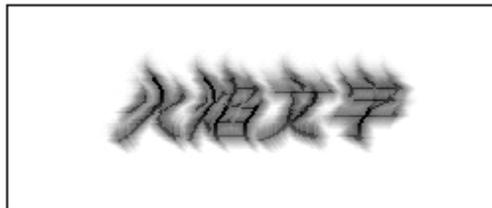


图13.29 火焰字

技术要点

本实例主要应用CSS样式的glow、Blur和Wave滤镜实现，下面对这些滤镜进行详细的介绍。

Blur滤镜的语法格式如下。

```
{filter: Blur (Add=value, Direction=value,  
Strength=value) }
```

参数说明如下。

● Add: 指定图片是否被改变成印象派的模糊效果, 取值为True或者False。

● Direction: 用来设置模糊的方向。模糊效果是按顺时针方向进行的, 0° 代表垂直向上, 然后每45° 为一个单位。默认值是向左270° 。

● Strength: 只能使用整数来指定, 代表有多少像素的宽度将受到模糊影响, 默认是 5 个像素。

功能: 模糊效果。

Wave滤镜的语法格式如下。

```
{filter: Wave(Add=value, Freq=value,  
LightStrength=value, Phase=value, Strength=value)}
```

参数说明如下。

● Add: 是否把元素按照波形样式打乱。

● Freq: 波纹的频率, 即指定在一个元素上需要产生多少个完整的波纹。

● LightStrength: 增强波纹光影的效果, 范围为0~100。

● Phase: 设置正弦波的偏移量。

● Strength: 设置振幅大小。

功能: 波浪变形效果。

glow滤镜的应用请参考实例333。

实现过程

(1) 使文字具有火焰效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

实现文字的火焰效果。

```
.tablecss {
```

```
    filter:
glow(color=red, strength=6)Blur (Add=true, Direction=0, Strength=10)
Wave (Add=add, Freq=3, LightStrength=10, Phase=5, Strength=5)
}
```

设置文字的大小。

```
.fontcss {
    font-size:36px;
}
```

-->

</style>

(2) 在页面中添加一个表格。代码如下。

```
<table align="center" width="200" height="100"
class="tablecss" border="0">
    <tr>
        <td height="100" align="center"><FONT
class="fontcss">火焰文字</FONT></td>
    </tr>
</table>
```

举一反三

根据本实例，读者可以：

制作一个图片扭曲的效果；

制作一个燃烧的火球。

## 实例344 文字扭曲动画

本实例是一个增强页面动画效果的程序

实例位置：光盘\mingrisoft\13\344

实例说明

在浏览页面时，经常会看到文字具有动态的扭曲效果。本实例将通过CSS样式来实现文字的这一效果。运行结果如图13.30所示。

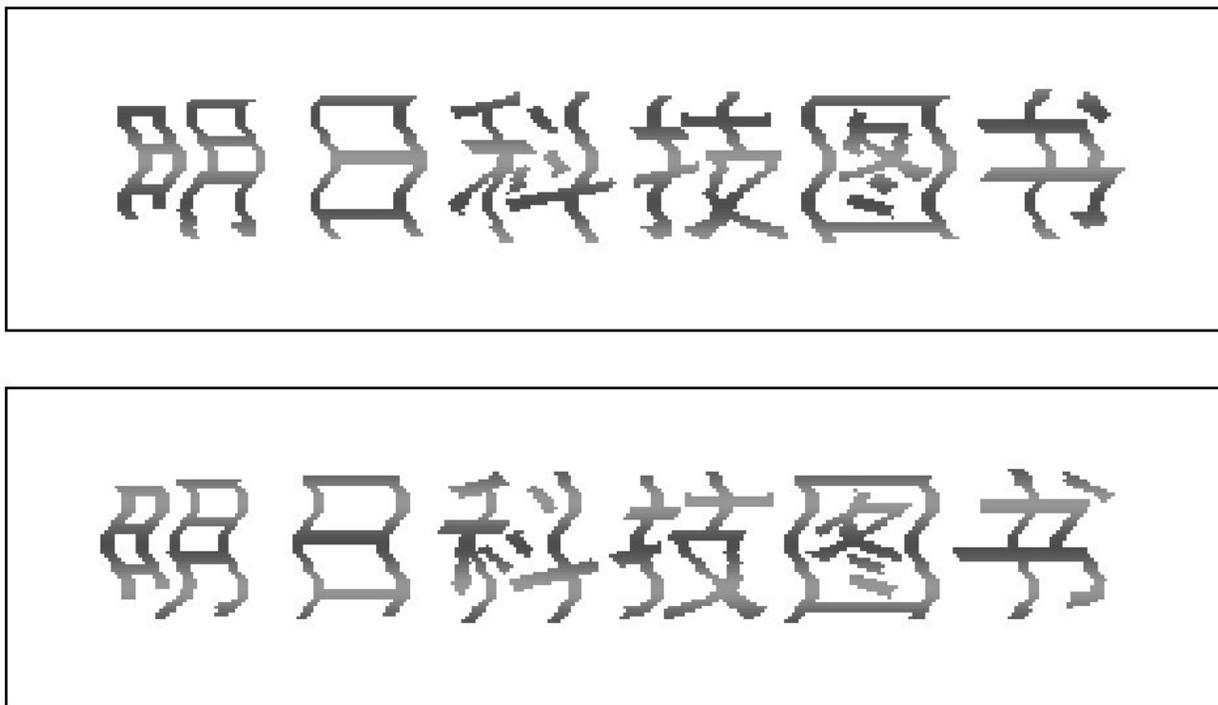


图13.30 文字扭曲动画

技术要点

本实例主要应用CSS样式的wave滤镜实现。通过wave滤镜的相关属性可以设置波纹的数量及幅度大小等，wave滤镜的介绍请参考实例343。

实现过程

(1) 使文字具有扭曲效果的CSS样式。代码如下。

```
<style type="text/css">  
<!--  
.skewcss {
```

```
width:300px;height:150px;font-family:黑体;font-size:50px;color:#00AAFF;
    filter:wave(freq=6, strength=4, phase=5,
lightstrength=45, add=0, enabled=1)
}
-->
</style>
```

(2) 在页面中添加一个层。代码如下。

```
<div id="div1" class="skewcss">明日科技图书</div>
```

(3) 编写用于实现文字扭曲动画的JavaScript代码。

```
<script language="javascript">
var p=2;
function sport() {
    div1.filters.wave.phase=p;
    p=p+2;
    setTimeout("sport()",10)
}
sport();
</script>
```

举一反三

根据本实例，读者可以：

实现图片扭曲效果的动画。

## 13.6 图片滤镜特效

在网页制作时，经常会用CSS样式对图片进行滤镜效果的处理，这样可以让同一图片以不同的效果进行显示，使图片多样化。下面将通过几个典型实例进行详细讲解。

### 实例345 图片的半透明效果

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\345

实例说明

在网页中浏览图片时，经常会看到一些半透明的图片，有些则将两张图片进行重叠，形成更加美观的图片。本实例将用 CSS 样式来实现图片半透明效果。运行结果如图 13.31所示。



图13.31 图片的半透明效果

技术要点

本实例主要应用CSS样式的Alpha滤镜实现。

Alpha滤镜的语法格式：

```
{filter: Alpha(Opacity=value, Finishopacity=value,  
Style=value, Startx=value,  
Starty=value, Finishx=value, Finishy=value)}
```

参数说明如下。

● **Opacity**: 代表透明水准。默认的范围为 0~100, 0 代表完全透明, 100 代表完全不透明。

● **Finishopacity**: 是一个可选参数。设置渐变的透明效果时就用这个参数来指定结束时的透明度, 范围为0~100。

● **Style**: 指定透明区域的形状特征, 0 代表统一形状, 1 代表线形, 2 代表放射状, 3 代表长方形。

● **Startx**: 渐变透明效果的开始x 坐标。

● **Starty**: 渐变透明效果的开始y坐标。

● **Finishx**: 渐变透明效果的结束x 坐标。

● **Finishy**: 渐变透明效果的结束y坐标。

功能: 设置透明层次。Alpha 属性是把一个目标元素与背景混合。可以通过指定数值来控制混合的程度。

实现过程

(1) 使图片具有半透明效果的CSS样式。代码如下。

```
<style type="text/css">  
<!--  
.clarity {/*使图片透明*/  
  filter:alpha(opacity=30)  
}  
-->  
</style>
```

(2) 在页面中添加图片及表格。代码如下。

```

```

```
<table width="200" border="0" cellpadding="0"
cellspacing="0" background="1.JPG">
  <tr>
    <td></td>
  </tr>
</table>
```

举一反三

根据本实例，读者可以：

使图片从中间到四周逐渐不透明；

制作一个带有透明风景的照片；

制作渐隐渐现的图片。

## 实例346 图片的模糊效果

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\346

实例说明

在设计网页时，如果将一些刮风的图片设计成按一定方向模糊的图片，那么，就会使图片更加逼真。本实例将用CSS样式来实现这一效果。运行结果如图13.32所示。

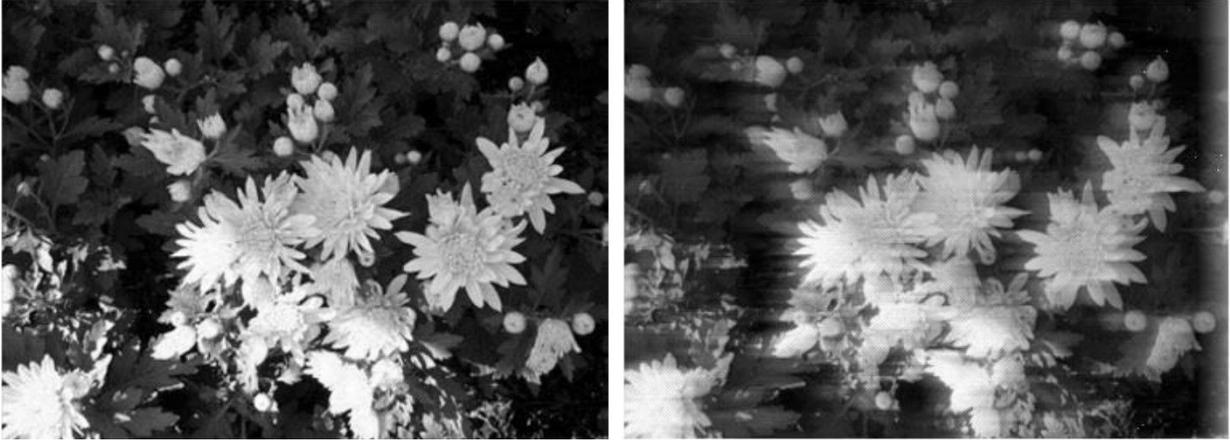


图13.32 图片的模糊效果

### 技术要点

本实例主要应用CSS样式的Blur滤镜实现。通过Blur滤镜的相关属性可以按一定的方向使对象产生模糊效果。Blur滤镜的介绍请参考实例343。

### 实现过程

(1) 使图片具有模糊效果的CSS样式。代码如下。

```
<style>
<!--
.style1 {
    filter:Blur(Add=true,Direction=270,Strength=20)
}
-->
</style>
```

(2) 在页面中添加图片。代码如下。

```


```

### 举一反三

根据本实例，读者可以：

用模糊效果作文字的阴影；

将夜晚的图片设置成朦胧的效果。

## 实例347 图片的水波纹特效

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\13\347

实例说明

在浏览页面时，经常会看到图片水中倒影的效果。本实例将用CSS样式来实现这一效果。运行结果如图13.33所示。

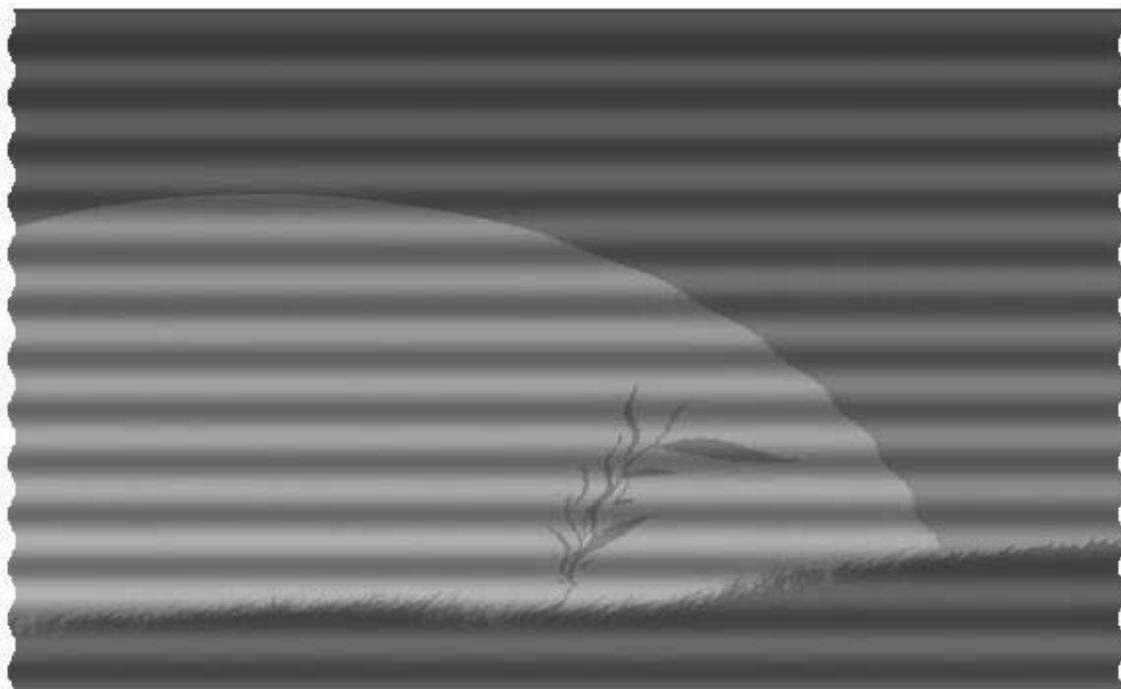


图13.33 图片的水波纹特效

技术要点

本实例主要应用CSS样式的wave滤镜实现。通过wave滤镜的相关属性可以设置波纹的数量及幅度大小等，wave滤镜的介绍请参考实例343。

实现过程

(1) 使图片具有水波纹效果的CSS样式。代码如下。

```
<style type="text/css">
<!--
.skewcss {
width:650px;height:450px;left:100px;top:100px;
filter:wave(freq=17, strength=3, phase=7,
lightstrength=35, add=0, enabled=1);
}
-->
</style>
```

(2) 在页面中添加图片。代码如下。

```
<div id="div1" class="skewcss"><center></center></div>
```

举一反三

根据本实例，读者可以：

实现动态的水波纹效果。

## 实例348 图片的灰度效果

这是一个应用性较强的实例

实例位置：光盘\mingrisoft\13\348

实例说明

在一些个人网页中，经常会看到一些黑白照片，给人一种怀旧的感觉，那么，在IE浏览器中如何将彩色图片转换成灰度效果呢？本实例将用 CSS 样式来解决这一问题。运行结果如图13.34所示。



图13.34 图片的灰度效果

#### 技术要点

本实例主要应用CSS样式的Gray滤镜实现。

Gray滤镜的语法格式：

```
{filter: Gray}
```

功能：灰度效果。将一张彩色图片变成灰度图。

#### 实现过程

(1) 将彩色图片转换成灰度效果的CSS样式。代码如下。

```
<style type="text/css">
```

```
<!--
```

```
img{
```

```
    filter: Gray;
```

```
}
```

```
-->
```

```
</style>
```

(2) 在页面中添加图片。代码如下。

```

```

```
</body>
```

举一反三

根据本实例，读者可以：

动态切换图片的彩色与灰度效果；

将彩色照片设置成旧的黑白照片。

## 实例349 图片的动态说明文字

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\13\349

实例说明

在浏览页面时，有时将鼠标移动到图片上时，会动态地弹出图片的说明性文字，并将图片以半透明方式显示。本实例将用CSS样式动态地在图片上显示文字。运行结果如图13.35、图13.36所示。



图13.35 鼠标没有移动到图片上之前效果



图13.36 鼠标移动到图片上时效果

### 技术要点

本实例主要应用CSS样式的字体属性、颜色属性、定位属性等实现，下面对相关属性进行详细的说明。

(1) 定位属性。CSS提供了用于定义元素二维、三维空间定位以及可见性的定位属性。设置定位属性可以将元素定位于相对其他元素的相对位置或者绝对位置。可以使用表 13.11 所示的属性来定位网页的元素位置。

表13.11 定位属性

定位属性	描述
Position	static: 无特殊定位, 元素遵循 HTML 定位规则 absolute: 绝对定位 relative: 相对位置, 元素不可重叠
Top	距离顶点纵坐标的距离
Left	距离顶点横坐标的距离
Width	元素的宽度
Height	元素的高度
z-index	决定元素的先后顺序和覆盖关系, 值高的元素覆盖值低的元素
Clip	限定只显示剪切出来的区域, 剪切出来的区域为矩形剪切设定两个点: 一个是矩形左上角的顶点, 由 Top 和 Right 两项设置; 另一个是右下角的顶点, 由 Bottom 和 Right 两项设置
Overflow	当元素内的内容超出元素所能容纳的范围时 visible: 无论元素容纳范围的大小, 内容都会显示出来 hidden: 隐藏超出元素范围的内容 scroll: 不管内容是否超出范围, 选中此项都会添加滚动条 auto: 只在内容超出范围时才显示滚动条
Visibility	这一项是针对嵌套元素的设置。嵌套的元素称为子层, 被嵌套的元素称为父层 inherit: 子层继承父层的可见性。父层可见, 子层也可见; 父层不可见, 子层也不可见 visible: 无论父层可见与否, 子层都可见 hidden: 无论父层可见与否, 子层都隐藏

(2) 字体属性、颜色属性和背景属性的介绍请参考实例317。

### 实现过程

(1) 对层进行隐藏/显示的控制, 并设置层的背景颜色、文字大小的CSS样式。代码如下。

```
<style>
```

```
<!--
```

设置说明性文字的默认位置。

```
.style0 {position: absolute;}
```

设置图片的显示位置。

```
.style1 { position:relative;position: absolute;}
```

设置说明性文字的显示位置。

```
.style2 { position:relative;}
```

将图片恢复正常。

```
.style3{
filter:alpha(opacity=80,finishOpacity=20,style=3,startX=10,star
tY=5,
finishX=90,finishY=20);}
```

设置图片为半透明状态。

```
.style4{
filter:alpha(opacity=100,finishOpacity=100,style=3,startX=10,
startY=5,
finishX=90,finishY=20);}
```

-->

</style>

(2) 在页面中添加图片和层。代码如下。

```
<div
onmouseover="document.all.img1.className='style3';document.al
l.text1
.className='style1'; "
onmouseout="document.all.img1.className='style4';
document.all.text1.className='style1';" >
<span id="text1" class="style0" ><font color="#0000FF">
RTT液晶电视的出产日期为:<br>2007年1月1日<br></font>
</span>
<span id="img1" class="style2" >

</span>
</div>
```

举一反三

根据本实例，读者可以：  
实现对一段英文进行动态的中文解释；  
实现对某些网站进行动态说明。

# 第14章 JavaScript与XML

读取XML文件

显示XML文档

操作XML

## 14.1 读取XML文件

随着XML的流行，JavaScript开发者也迫切希望在客户端Web开发中应用XML。本节将介绍如何通过XML DOM 对象和XMLHttpRequest 对象读取XML 文件。

### 实例350 使用XML DOM对象读取XML文件

本实例是一个方便、快捷的程序

实例位置：光盘\mingrisoft\14\350

实例说明

在制作网站时，有时需要在页面上显示商品或其他信息的列表。这时，如果是动态网站可以将列表信息保存到数据库中，但如果是静态网站，制作和维护起来就很麻烦。解决该问题的方法是将要显示的信息保存到XML文件中，然后再通过JavaScript读取并显示该XML文件中的内容。运行本实例，在页面中将以表格的形式显示 XML文件中的商品信息，如图14.1所示。



图14.1 使用XML DOM对象读取XML文件

### 技术要点

本实例主要应用XML DOM 技术实现读取XML 文件。虽然XML 和DOM 已经变成Web开发的重要组成部分，但是目前仅有 IE和 Mozilla两个浏览器支持客户端处理 XML。下面将分别介绍在IE和Mozilla中创建DOM并载入XML的方法。

#### 1. 在IE中创建DOM并载入XML

(1) 创建XML DOM 对象的实例。Microsoft 在JavaScript 中引入了用于创建ActiveX 对象的ActiveXObject 类，通过该类可以创建XML DOM 对象的实例，代码如下。

```
var xmlDoc = new ActiveXObject("Microsoft.XMLDOM");
```

(2) 载入XML。Microsoft 的 XML DOM 有两种载入XML 的方法：load()和 loadXML()，本实例中使用的是load()。

load()方法用于从服务器上载入XML文件，load()方法的语法格式如下。

```
xmlDoc.load(url);
```

参数说明如下。

- xmlDoc：为XML DOM 对象的实例。

● url: 为XML 文件的名称。需要注意的是, load() 方法只可以载入同包含JavaScript 的页面存储于同一服务器上的文件。

在载入时还可以采用同步或异步两种模式。默认情况下, 文件按照异步模式载入。如果需要进行同步载入, 可以设置async属性为False。本实例采用的是默认模式, 即异步模式。

在异步载入文件时, 还需要使用 readyState属性和 onreadystatechange事件处理函数, 这样可以保证在 DOM完全载入后执行其他操作(例如本例调用自定义的 JavaScript函数 createTable()将载入到DOM中的XML取出来, 并以表格的形式显示在页面中), 代码如下。

```
xmlDoc.onreadystatechange = function() {  
    if(xmlDoc.readyState == 4) createTable(xmlDoc);  
}
```

loadXML()方法可直接向XML DOM 输入XML 字符串, 例如:

```
xmlDoc.loadXML("<root><son/></root>");
```

## 2. 在Mozilla中创建DOM并载入XML

(1) 创建 XML DOM 对象的实例。DOM 标准指出使用 document.implementation 对象的createDocument()方法可以创建XML DOM 对象的实例, 代码如下。

```
var xmlDoc = document.implementation.createDocument("",  
"", null);
```

createDocument()方法包括3个参数, 第1个参数用于指定文件的命名空间URL; 第2个参数用于指定文件元素的标签名; 第3个参数用于指定文档类型对象(因为 Mozilla中还没有对文档类型对象的支持, 所以总是null)。

(2) 载入XML。Mozilla只支持一种载入XML的方法: load()。Mozilla中的load()方法和IE 中的 load()方法工作方式一样, 只要指

定载入的 XML 文件，以及是同步还是异步模式载入即可。

Mozilla 的XML DOM 会在文件完全载入后触发load 事件，也就是说必须使用onload 事件处理函数来判断 DOM何时完全载入，这样可以保证在 DOM完全载入后执行其他操作（例如本例，调用自定义的JavaScript函数createTable()将载入到DOM中的XML取出来，并以表格的形式显示在页面中），代码如下。

```
xmlDoc.onload=function() {  
    xmlDoc.onload = createTable(xmlDoc);  
}
```

### 实现过程

(1) 编写自定义的JavaScript函数createTable()，用于将载入到DOM中的XML取出来并以表格的形式显示在页面中。该函数只包括一个参数xmlDoc，用于指定载入到DOM中的XML，无返回值。代码如下。

```
<script language="javascript">  
function createTable(xmlDoc) {  
    var table=document.createElement("table");  
    table.setAttribute("width","100%");  
    table.setAttribute("border","1");  
    table.borderColor="#FFFFFF";  
    table.cellSpacing="0";  
    table.cellpadding="0";  
    table.borderColorDark="#FFFFFF";  
    table.borderColorLight="#AAAAAA";  
    parentTd.appendChild(table);    //在指定位置创建表格  
    var header= table.createTHead();  
    header.bgColor="#EEEEEE";    //设置表头背景  
    var headerrow=header.insertRow(0);
```

```

    headerrow.height="27"; //设置表头高度
    headerrow.insertCell(0).appendChild(document.createTextNode("商品名称"));
    headerrow.insertCell(1).appendChild(document.createTextNode("类别"));
    headerrow.insertCell(2).appendChild(document.createTextNode("单位"));
    headerrow.insertCell(3).appendChild(document.createTextNode("单价"));
    var goodss=xmldoc.getElementsByTagName("goods");
    for(var i=0;i<goodss.length;i++) {
    var g=goodss[i];
    var name=g.getAttribute("name");
    var type= g.getElementsByTagName("type")
[0].firstChild.data;
    var goodsunit= g.getElementsByTagName("goodsunit")
[0].firstChild.data;
    var price= g.getElementsByTagName("price")
[0].firstChild.data;
    var row= table.insertRow(i+1);
    row.height="27"; //设置行高
    row.insertCell(0).appendChild(document.createTextNode(name));
    row.insertCell(1).appendChild(document.createTextNode(type));
    row.insertCell(2).appendChild(document.createTextNode(goodsunit));

```

```

        row.insertCell(3).appendChild(document.createTextNode
(price));
    }
}
</script>

```

(2) 编写自定义的JavaScript函数readXML(), 用于读取XML文件并显示在页面中。在该函数中, 首先实现在IE或Mozilla浏览器中创建DOM, 然后把指定的XML文件载入到DOM中, 最后调用自定义的JavaScript函数createTable()在页面的指定位置显示XML文件的内容, 代码如下。

```

<script language="javascript">
function readXML() {
var url= "goodss.xml";
    if(window.ActiveXObject) { //IE
        varxmlDoc=newActiveXObject("Microsoft.XMLDOM");
        xmlDoc.onreadystatechange= function() {
            if(xmlDoc.readyState==4) createTable(xmlDoc);
        }
        xmlDoc.load(url);
    }
    //Mozilla.....
    else
if(document.implementation&&document.implementation.createDoc
ument){
        var xmlDoc=document.implementation.createDocument("", "",
null);
        xmlDoc.onload=function() {

```

```
xmlDoc.onload= createTable(xmlDoc);
    }
xmlDoc.load(url);
    }
}
</script>
```

(3) 将用于显示新创建表格的单元格的id属性设置为parentTd, 关键代码如下。

```
<td valign="top" id="parentTd">&nbsp;&nbsp;&nbsp;</td>
```

(4) 在<body>标记的onLoad事件中调用自定义函数readXML() 读取XML文件并显示在页面中, 关键代码如下。

```
<body onLoad="readXML()">
```

举一反三

根据本实例, 读者可以:

将XML文件以表格形式显示, 并改变表格的样式;

读取XML文件并以下拉列表框显示。

## [实例351 使用XMLHttpRequest对象读取XML文件](#)

本实例是一个方便、实用的程序

实例位置: 光盘\mingrisoft\14\351

实例说明

实例 350 已经介绍过如何使用 XML DOM 对象读取 XML 文件, 本实例将介绍如何使用XMLHttpRequest对象读取XML。运行本实例, 在页面中将以表格的形式显示 XML文件中的客户信息, 如图14.2所示。



图14.2 使用XMLHttpRequest对象读取XML

### 技术要点

本实例主要应用AJAX技术中的XMLHttpRequest对象实现。

XMLHttpRequest是XMLHTTP组件的一个对象，通过对该对象的调用可以实现像桌面应用程序一样的与服务器进行数据层面的交换，而不需要每次请求都刷新整个页面，也不需要每次的数据操作都交付给服务器去完成。在使用XMLHttpRequest对象进行异步处理之前，需要对该对象进行初始化操作，下面将介绍如何在IE浏览器和Mozilla浏览器中初始化XMLHttpRequest对象。

在IE中初始化XMLHttpRequest对象的代码如下。

```
try {  
    http_request = new ActiveXObject("Msxml2.XMLHTTP");  
} catch (e) {  
    try {  
        http_request = new ActiveXObject("Microsoft.XMLHTTP");  
    } catch (e) {}  
}
```

在Mozilla中初始化XMLHttpRequest对象的代码如下。

```
http_request = new XMLHttpRequest();
if (http_request.overrideMimeType) {
http_request.overrideMimeType("text/xml");
}
```

### 实现过程

(1) 编写自定义的JavaScript函数createTable(), 用于将载入到DOM中的XML取出来, 并以表格的形式显示在页面中。该函数只包括一个参数xmlDoc, 用于指定载入到DOM中的XML, 无返回值。代码如下。

```
<script language="javascript">
function createTable(xmlDoc) {
var table=document.createElement("table");
table.setAttribute("width","620");
table.setAttribute("border","1");
table.borderColor="#FF0000";
table.cellSpacing="0";
table.cellpadding="0";
table.borderColorDark="#FFFFFF";
table.borderColorLight="#000000";
parentTd.appendChild(table); //在指定位置创建表格
var header= table.createTHead();
header.bgColor="#EEEEEE"; //设置表头背景
var headerrow=header.insertRow(0);
headerrow.height="27"; //设置表头高度
headerrow.insertCell(0).appendChild(document.createTextNode("客户名称"));
```

```

    headerrow.insertCell(1).appendChild(document.createTextNode("联系地址"));
    headerrow.insertCell(2).appendChild(document.createTextNode("电话"));
    headerrow.insertCell(3).appendChild(document.createTextNode("邮政编码"));
    headerrow.insertCell(4).appendChild(document.createTextNode("开户银行"));
    headerrow.insertCell(5).appendChild(document.createTextNode("银行账号"));
    var customers=xmlDoc.getElementsByTagName("customer");
    for(var i=0;i<customers.length;i++) {
    var cus= customers[i];
    var name= cus.getAttribute("name");
    var address= cus.getElementsByTagName("address")
[0].firstChild.data;
    var tel= cus.getElementsByTagName("tel")
[0].firstChild.data;
    var postcode= cus.getElementsByTagName("postcode")
[0].firstChild.data;
    var bank= cus.getElementsByTagName("bank")
[0].firstChild.data;
    var bankcode= cus.getElementsByTagName("bankcode")
[0].firstChild.data;
    var row= table.insertRow(i+1);
    row.height="27";    //设置行高

```

```

        row.insertCell(0).appendChild(document.createTextNode
(name));
        row.insertCell(1).appendChild(document.createTextNode
(address));
        row.insertCell(2).appendChild(document.createTextNode
(tel));
        row.insertCell(3).appendChild(document.createTextNode
(postcode));
        row.insertCell(4).appendChild(document.createTextNode
(bank));
        row.insertCell(5).appendChild(document.createTextNode
(bankcode));
    }
}
</script>

```

(2) 搭建AJAX开发框架，具体代码如下。

```

<script language="javascript">
var http_request= false;
function createRequest(url) {
//初始化对象并发出XMLHttpRequest请求
http_request= false;
if (window.XMLHttpRequest) {//Mozilla.....
http_request=newXMLHttpRequest();
if (http_request.overrideMimeType) {
http_request.overrideMimeType("text/xml");
}
} else if (window.ActiveXObject) {//IE浏览器

```

```

try{
http_request=newActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
try{
http_request=newActiveXObject("Microsoft.XMLHTTP");
} catch (e) {}
}
}
if (!http_request) {
alert("不能创建XMLHTTP实例!");
return false;
}
http_request.onreadystatechange=dealresult;    //指定响应
方法
//发出HTTP请求
http_request.open("GET", url, true);
http_request.send(null);
}
</script>

```

(3) 编写自定义的JavaScript函数dealresult(), 用于处理服务器返回的信息。在该函数中, 将调用自定义的JavaScript函数createTable()在页面的指定位置显示 XML文件的内容, 代码如下。

```

<script language="javascript">
functiondealresult() {    //处理服务器返回的信息
if (http_request.readyState==4) {
if (http_request.status==200) {
var xmldoc=http_request.responseXML;

```

```
createTable(xmlDoc);  
} else{  
alert('您请求的页面发现错误');  
}  
}  
}  
</script>
```

(4) 将用于显示新创建表格的单元格的id属性设置为parentTd, 关键代码如下。

```
<td width="96%" id="parentTd">
```

(5) 在<body>标记的onLoad事件中调用自定义函数createRequest()读取XML文件并显示在页面中, 关键代码如下。

```
<body onLoad="createRequest('customers.xml')">
```

举一反三

根据本实例, 读者可以:

将XML文件以表格形式显示, 并改变表格的样式;

实现基于XML的留言簿中的留言显示信息页面。

## 14.2 显示XML文档

要显示XML文档内容，不但可以借助外部CSS样式表或外部XSL样式表实现，而且也可以借助一些客户程序实现。本节将介绍如何通过CSS、XSL 或IE XML 数据岛显示XML 文档。

### 实例352 使用CSS显示XML文档

本实例是一个规范显示文档的程序

实例位置：光盘\mingrisoft\14\352

实例说明

XML 文档在IE 5.0 及以上版本的浏览器中可以像普通Web 页面一样直接打开。如果XML文档没有引用外部CSS文件或是XSL文件，那么IE将只显示整个文档的文本，相反加入引用样式表文件，将按样式表文件指定的样式显示 XML 文档。例如本实例将 index.xml 文件中的“`<?xml-stylesheet href="style.css" type="text/css"?>`”语句删除后，在IE 中打开的结果如图14.3所示；将“`<?xml-stylesheet href="style.css" type="text/css"?>`”语句加入后，在IE 中打开的结果如图14.4所示。



图14.3 没有引用样式表文件的运行结果

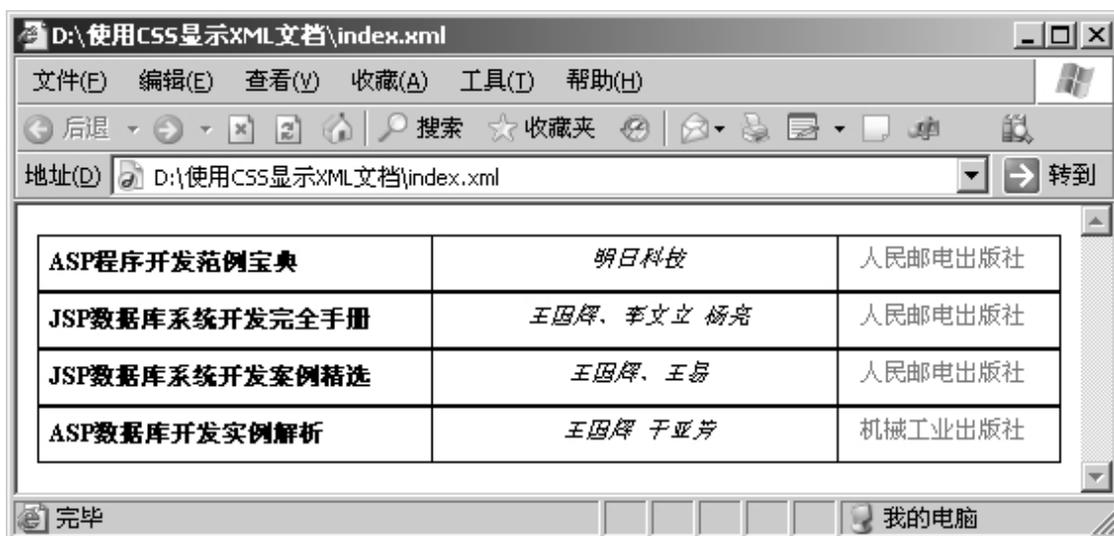


图14.4 引用样式表文件的运行结果

技术要点

在XML文档中引用外部CSS样式表文件的语句如下。

```
<?xml-stylesheet href="style.css" type="text/css"?>
```

其中，style.css代表CSS样式表文件的名称。

实现过程

(1) 编写CSS样式表文件style.css，在该样式表文件中指定XML文档中各元素的显示样式，代码如下。

```
book{
border:1px solid;
width:100%;
font-size:12px;
}
ISBN{
display: none;
}
bookname{
font-size:12px;
display: inline;
font-weight: bold;
width: 200px;
border-right:1px solid;
height:27px;
padding:5px;
}
author{
font-size:12px;
display: inline;
font-style: italic;
```

```
width: 200px;
border-right: 1px solid;
height: 27px;
text-align: center;
padding: 5px;
}
publishing {
font-size: 12px;
display: inline;
color: red;
width: 100px;
height: 27px;
padding: 5px;
}
```

(2) 编写index.xml文件，在该文件中创建一个books根元素，该元素由多个book元素组成，代码如下。

```
<?xml version="1.0" encoding="gb2312" ?>
<books>
<book>
<ISBN>7-115-14873-2</ISBN>
<bookname>ASP程序开发范例宝典</bookname>
<author>明日科技</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-115-14547-4</ISBN>
<bookname>JSP数据库系统开发完全手册</bookname>
```

```
<author>王国辉、李文立 杨亮</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-115-14689-6</ISBN>
<bookname>JSP数据库系统开发案例精选</bookname>
<author>王国辉、王易</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-111-15531-9</ISBN>
<bookname>ASP数据库开发实例解析</bookname>
<author>王国辉 于亚芳</author>
<publishing>机械工业出版社</publishing>
</book>
</books>
```

(3) 在index.xml文件的第二行加入如下引用CSS样式表文件的代码。

```
<?xml-stylesheet href="style.css" type="text/css"?>
```

举一反三

根据本实例，读者可以：

将保存商品信息的XML文档格式化输出；

将保存留言信息的XML文档格式化输出。

## 实例353 使用XSL显示XML文档

本实例是一个方便、快捷的程序

实例位置：光盘\mingrisoft\14\353

## 实例说明

在实例352中已经介绍了如何使用CSS外部样式表文件显示XML文档。本实例将介绍如何使用XSL显示XML文档。运行本实例，在页面中将按XSL文件指定样式显示XML文档的内容，如图14.5所示。

书号	书名	作者	出版社
7-115-14873-2	ASP程序开发范例宝典	明日科技	人民邮电出版社
7-115-14547-4	JSP数据库系统开发完全手册	王国辉、李文立 杨亮	人民邮电出版社
7-115-14689-6	JSP数据库系统开发案例精选	王国辉、王易	人民邮电出版社
7-111-15531-9	ASP数据库开发实例解析	王国辉 于亚芳	机械工业出版社

图14.5 使用XSL显示XML文档

## 技术要点

在XML文档中引用外部XSL样式表文件的语句如下。

```
<?xml-stylesheet href="style.xsl" type="text/xsl"?>
```

其中，style.xsl代表XSL样式表文件的名称。

## 实现过程

(1) 编写XSL样式表文件style.css，在该文件中将XML的<book>标签转换为表格的TR标记；<book>标签下的子元素转换为表格的TD标记，并将整个文档作为HTML输出，代码如下。

```
<?xmlversion='1.0' encoding='gb2312' ?>
```

```
<xsl:stylesheet
```

```
version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>
```

```
<xsl:templatematch="/">
```

```

<html>
<head>
<title>使用XSL显示XML文档</title>
<style>
td {
font-size:9pt; color:#000000;
}
body{
margin:0px;
}
.tableBorder{
border:#aaaaa1px solid
}
</style>
</head>
<body>
<tablewidth="780" border="0"
align="center"cellpadding="0"cellspacing="0"
background="Images/bg.jpg"class="tableBorder">
<tr>
<tdwidth="26%"height="112"valign="top"></td>
</tr>
<tr>
<tdheight="248"align="center"valign="top">
<tablewidth="90%"border="1"
bordercolor="#FFFFFF"bordercolordark="#FFFFFF"bordercolor
light="#999999"

```

```

cellpadding="0" cellspacing="0">
<trheight="27">
<td align="center">书号</td>
<td align="center">书名</td>
<td align="center">作者</td>
<td align="center">出版社</td>
</tr>
<xsl:for-each select="books/book">
<trheight="27">
<td><xsl:value-of select="ISBN"/></td>
<td><xsl:value-of select="bookname"/></td>
<td><xsl:value-of select="author"/></td>
<td><xsl:value-of select="publishing"/></td>
</tr>
</xsl:for-each>
</table></td>
</tr>
<tr>
<tdheight="69" valign="top"></td>
</tr>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

(2) 编写index.xml文件，在该文件中创建一个books根元素，该元素由多个book元素组成，代码如下。

```
<?xmlversion="1.0" encoding="gb2312" ?>
<books>
<book>
<ISBN>7-115-14873-2</ISBN>
<bookname>ASP程序开发范例宝典</bookname>
<author>明日科技</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-115-14547-4</ISBN>
<bookname>JSP数据库系统开发完全手册</bookname>
<author>王国辉、李文立杨亮</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-115-14689-6</ISBN>
<bookname>JSP数据库系统开发案例精选</bookname>
<author>王国辉、王易</author>
<publishing>人民邮电出版社</publishing>
</book>
<book>
<ISBN>7-111-15531-9</ISBN>
<bookname>ASP数据库开发实例解析</bookname>
<author>王国辉 于亚芳</author>
<publishing>机械工业出版社</publishing>
</book>
</books>
```

(3) 在index.xml文件的第二行加入如下引用XSL样式表文件的代码。

```
<?xml-stylesheet href="style.xsl" type="text/xsl"?>
```

举一反三

根据本实例，读者可以：

将保存客户信息的XML文档格式化输出；

将保存员工信息的XML文档格式化输出。

## 实例354 使用IE XML数据岛输出XML文档

本实例是一个方便、快捷的程序

实例位置：光盘\mingrisoft\14\354

实例说明

在实例 353 中已经介绍了如何使用 CSS XSL 显示 XML 文档。本实例将介绍如何使用 IEXML数据岛输出XML文档。运行本实例，在页面中将以表格的形式显示通过XML数据岛获取的朋友圈信息，如图14.6所示。



The screenshot shows a web page header with the logo 'M·R 博客' and the URL 'HTTP://MINGRISOFT.COM'. Navigation links include '首页', '博客文章', '博客注册', and '帮助'. Below the header is a search bar. The main content area displays a table titled '朋友圈信息列表' (Friends Information List) with the following data:

名称	联系地址	邮政编码	联系电话	QICQ号码
无语	长春市***街66号	130031	32768***	623912076
辉儿	长春市***街77号	130031	36895****	2124***12

图14.6 使用IE XML数据岛输出XML文档

技术要点

使用 IE XML 数据岛输出 XML 文档的基本思路是：首先使用一个 XML 数据岛载入外部XML文档，然后将一个HTML的表格绑定到此数据岛上，最后将HTML表格内部的<span>标记使用datafld属性和XML文档对应的XML元素相互绑定。

### 实现过程

(1) 编写friends.xml文件，在该文件中创建一个friends根元素，该元素由多个friend元素组成，代码如下。

```
<?xml version="1.0" encoding="gb2312"?>
<friends>
<friend>
<name>无语</name>
<address>长春市***街66号</address>
<postcode>130031</postcode>
<tel>32768***</tel>
<qq>623912076</qq>
</friend>
<friend>
<name>辉儿</name>
<address>长春市***街77号</address>
<postcode>130031</postcode>
<tel>36895****</tel>
<qq>2124***12</qq>
</friend>
</friends>
```

(2) 使用一个XML数据岛 (id=f) 载入friends.xml文档，并将其绑定到HTML表格 (datasrc=#f) 上，再将<span>标记的datafld属性和XML文档对应的XML元素相互绑定，关键代码如下。

```

<xml id="f" src="friends.xml"></xml>
<table width="90%" border="1" datasrc="#f"
cellpadding="0" cellspacing="0" 00
bordercolor="#FFFFFF" bordercolordark="#FFFFFF"
bordercolorlight="#999999">
<thead>
<tr>
<td align="center">名称</td>
<td align="center">联系地址</td>
<td align="center">邮政编码</td>
<td align="center">联系电话</td>
<td align="center">OICQ号码</td>
</tr>
</thead>
<tr>
<td><span datafld="name"></span></td>
<td><span datafld="address"></span></td>
<td><span datafld="postcode"></span></td>
<td><span datafld="tel"></span></td>
<td><span datafld="qq"></span></td>
</tr>
</table>

```

举一反三

根据本实例，读者可以：

将保存主题或回复信息的XML文档以表格形式输出；

将保存留言信息的XML文档以表格形式输出。

## 14.3 操作XML

通过JavaScript可以对XML文档进行控制。本节将介绍如何用JavaScript控制XML文档的分页显示和通过XML实现留言簿。

### 实例355 用JavaScript控制XML文档的分页显示

本实例是一个应用较强的程序

实例位置：光盘\mingrisoft\14\355

实例说明

本实例将介绍如何使用JavaScript控制XML文档的分页显示。运行本实例，如图14.7所示，在页面中将显示第一篇从XML文档中获取的博客文章，单击“下一篇”超级链接，即可查看下一篇博客文章，单击“上一篇”超级链接，即可查看上一篇文章。



## 图14.7 用JavaScript控制XML文档的分页显示

### 技术要点

本实例主要通过XML数据岛的recordset对象的absolutePosition属性、recordcount属性、moveNext()方法和movePrevious()方法实现数据的分页导航功能。下面分别进行介绍。

- absolutePosition: 该属性用于返回当前记录的记录号。
- recordcount: 该属性用于返回总记录数。
- moveNext(): 该方法用于将记录指针移动到下一条记录。
- movePrevious(): 该方法用于将记录指针向前移动一条记录。

### 实现过程

(1) 编写doc.xml文件, 在该文件中创建一个docs根元素, 该元素由多个doc元素组成, 代码如下。

```
<?xmlversion="1.0"encoding="gb2312"?>
<docs>
<doc>
<id>1</id>
<author>无语</author>
<datetime>2007-03-0311: 50: 12</datetime>
<topic>祝福</topic>
<content>你现在好吗? 今天快乐吗? 我从远方送你的祝福收到了
吗? </content>
</doc>
<doc>
<id>2</id>
<author>wgh</author>
<datetime>2007-03-1510: 20: 42</datetime>
<topic>春天</topic>
```

```
<content>东风来了，春天的脚步近了.....</content>
</doc>
</docs>
```

(2) 首先使用一个XML数据岛 (id=d) 载入doc.xml文档，然后使用<span>标记的datasrc属性与id为d的XML数据岛进行绑定，再使用<span>标记的数据fld属性与XML文档对应的XML元素进行绑定，关键代码如下。

```
<xml id="d" src="doc.xml" async="false"></xml>
<tablewidth="90%" border="1" cellpadding="0"
cellspacing="0" bordercolor="#FFFFFF"
bordercolordark="#FFFFFF" bordercolorlight="#999999">
<tr>
<tdheight="25" colspan="2">博客ID号: <span data-src="#d"
data-fld="id"></span></td>
<tdwidth="35%">作者: <span data-src="#d" data-fld="author">
</span></td>
<tdwidth="43%">发表日期:
<span data-src="#d" data-fld="datetime"></span></td>
</tr>
<tr>
<tdheight="25" colspan="4">博客主题: <span data-src="#d"
data-fld="topic"></span></td>
</tr>
<tr>
<tdwidth="11%" height="25">文章内容</td>
<tdheight="25" colspan="3">
<span data-src="#d" data-fld="content"></span></td>
```

```
</tr>
</table>
```

(3) 编写自定义的JavaScript函数moveNext(), 用于向后移动一条记录, 代码如下。

```
<script type="text/javascript">
function moveNext() {
x=d.recordset;
if (x.absolutePosition < x.recordcount) {
x.movenext();
}
}
</script>
```

(4) 编写自定义的JavaScript函数movePrevious(), 用于向前移动一条记录, 代码如下。

```
function movePrevious() {
x=d.recordset;
if (x.absolutePosition > 1) {
x.moveprevious();
}
}
</script>
```

(5) 在页面的适当位置添加“上一篇”和“下一篇”超级链接, 并在其 onClick 事件中调用相应方法, 代码如下。

```
<a href="#" onClick="movePrevious()">上一篇</a>&nbsp;
<a href="#" onClick="moveNext()">下一篇</a>
```

举一反三

根据本实例, 读者可以:

实现网上社区中的文章浏览;

实现分页显示留言信息。

## 实例356 通过操作XML数据岛实现添加、删除留言信息

本实例是一个提高工作效率的程序

实例位置：光盘\mingrisoft\14\356

实例说明

本实例将介绍如何通过操作 XML 数据岛实现添加、删除留言信息。运行本实例，在“留言人”文本框中输入“wgh”，在“表情”文本框中输入“微笑”，在“电话”文本框中输入“495\*\*\*\*”，在“留言内容”文本框中输入“祝大家新年快乐！”，单击“添加留言”按钮，在页面中将以列表形式显示留言信息，如图14.8所示。单击“删除第一条留言”按钮，将删除第一条留言信息，单击“删除最后一条留言”超级链接，即可删除最后一条留言信息。



图14.8 通过操作XML数据岛实现添加、删除留言信息  
技术要点

本实例主要通过 XML DOM 对象实现。XML DOM 对象的全称是 XML Document Object Model, 即XML 文档对象模型。XML DOM 定义了访问和操作XML 文档的标准方式。下面将对本实例中应用的XML DOM 对象的属性和方法进行介绍。

### 1. createElement() 方法

createElement() 方法用于创建一个指定名称的元素, 其语法格式如下。

```
xmlDocument.createElement(tagName);
```

参数说明如下。

- tagName: 用于指定新元素的名称, 区分大小写。

### 2. createAttribute() 方法

`createAttribute()` 方法用于创建一个指定名称的属性，其语法格式如下。

```
xmlDocument.createAttribute(name);
```

参数说明如下。

- **name**: 用于指定属性名称。

### 3. `createNode()` 方法

`createNode()` 方法用于创建一个指定类型、名称及命名空间的新节点，其语法格式如下。

```
xmlDocument.createNode(type, name, nameSpaceURI);
```

参数说明如下。

- **type**: 用于指定将要创建的节点的类型。
- **name**: 用于指定新节点的名称。
- **nameSpaceURI**: 用于指定一个定义命名空间的URI。

### 4. `createTextNode()` 方法

`createTextNode()` 方法用于创建一个新的text节点，并包含指定的数据，其语法格式如下。

```
xmlDocument.createTextNode(data);
```

参数说明如下。

- **data**: 用于指定一个代表新text节点的字符串。需要注意的是，此时的text节点并没有添加到文件树中。若要将该节点添加到文件树中，必须使用插入方法（例如：`insertBefore`、`replaceChild`或`appendChild`）。

### 5. `insertBefore()` 方法

`insertBefore()` 方法用于在指定的节点前插入一个子节点，其语法格式如下。

```
objDocumentNode=xmlDocumentNode.insertBefore(newChild, refChild);
```

参数说明如下。

- `newChild`: 用于指定一个包含新子节点地址的对象。
- `refChild`: 用于指定参照节点的地址。新的子节点将被插到参照节点之前。如果`refChild`参数没有包含在内, 新的子节点会被插到子节点列表的末端。

#### 6. `removeChild()` 方法

`removeChild()` 方法用于将指定的节点从节点列表中移除, 其语法格式如下。

```
objDocumentNode=xmlDocumentNode.removeChild(oldChild);
```

参数说明如下。

- `oldChild`: 用于指定一个要被移除的节点对象。

#### 7. `appendChild()` 方法

`appendChild()` 方法用于在当前节点的最后加入一个新的节点, 其语法格式如下。

```
xmlDocumentNode.appendChild(newChild);
```

参数说明如下。

- `newChild`: 用于指定一个要添加新子节点的地址。

#### 8. `hasChildNodes()` 方法

如果指定的节点有一个或多个子节点, `hasChildNodes()` 方法将返回 `True`, 否则将返回`False`, 其语法格式如下。

```
boolValue=xmlDocumentNode.hasChildNodes();
```

#### 9. `lastChild`属性

`lastChild` 属性为只读属性, 用于返回一个对象。如果节点中没有包含最后子元素, 将返回`null`, 其语法格式如下。

```
objLastChild = xmlDocumentNode.lastChild;
```

实现过程

(1) 编写board.xml文件，在该文件中创建一个boards根元素，该元素由多个board元素组成，代码如下。

```
<?xml version="1.0" encoding="gb2312"?>
<boards>
<board>
<person>无语</person>
<humor>微笑</humor>
<tel>13634449161</tel>
<message>你现在好吗？今天快乐吗？我从远方送你的祝福收到了
吗？</message>
</board>
</boards>
```

(2) 使用一个 XML 数据岛 (id=board) 载入 board.xml 文档，并将其绑定到 HTML 表格 (datasrc=# board) 上，再将<span>标记的 datafld 属性和XML 文档对应的XML 元素相互绑定，关键代码如下。

```
<xml id="board" src="board.xml"></xml>
<tablewidth="100%"border="1" cellpadding="0"
cellspacing="0"datasrc="#board"
bordercolor="#FFFFFF"bordercolordark="#FFFFFF"bordercolor
light="#96E2FA">
<thead>
<tr>
<thwidth="12%"height="27"bgcolor="#6AC7EC">留言人</th>
<thwidth="12%"bgcolor="#6AC7EC">表情</th>
<thwidth="21%"bgcolor="#6AC7EC">电话</th>
<thwidth="55%"bgcolor="#6AC7EC">留言内容</th>
</tr>
```

```

</thead>
<tr>
<tdheight="27"><spandatafld="person"></span></td>
<td><spandatafld="humor"></span></td>
<td><spandatafld="tel"></span></td>
<td><spandatafld="message"></span></td>
</tr>
</table>

```

(3) 编写自定义的JavaScript 函数addElement (), 用于在id 为board 的XML 数据岛中添加一条留言信息, 代码如下。

```

<script language="javascript">
xmlDoc=board;
//添加留言信息
function addElement() {
var rootElement=xmlDoc.documentElement;
//创建留言元素
var newBoard=xmlDoc.createElement('board');
//添加留言人子元素
var person=xmlDoc.createElement('person');
var
personF=xmlDoc.createTextNode(document.myform.person.value);
person.appendChild(personF);
newBoard.appendChild(person);
//添加表情子元素
var humor=xmlDoc.createElement('humor');
var
humorF=xmlDoc.createTextNode(document.myform.humor.value);

```

```

humor.appendChild(humorF);
newBoard.appendChild(humor);
//添加电话子元素
var tel=xmlDoc.createElement('tel');
var
telF=xmlDoc.createTextNode(document.myform.tel.value);
tel.appendChild(telF);
newBoard.appendChild(tel);
//添加留言信息子元素
varmessage=xmlDoc.createElement('message');
varmessageF=xmlDoc.createTextNode(document.myform.message
.value);
message.appendChild(messageF);
newBoard.appendChild(message);
//向文档中追加一条留言信息
rootElement.appendChild(newBoard);
}
</script>

```

(4) 编写自定义的Javascript 函数deleteFirstElement ()，用于删除第一个节点，代码如下。

```

<script language="javascript">
functiondeleteFirstElement() { //删除第一个节点
var rootElement=xmlDoc.documentElement;
if (rootElement.hasChildNodes())
rootElement.removeChild(rootElement.childNodes.item(0));
//删除第一个节点
}

```





</table>

</form>

举一反三

根据本实例，读者可以：

实现动态添加或删除网上论坛的主题信息。

# 第15章 JavaScript与ASP结 合

弹出窗口/对话框控制

弹出确认对话框

其他

## 15.1 弹出窗口/对话框控制

本节所说的弹出窗口是指通过 window 对象的 open() 方法打开的新窗口。这种弹出窗口在网站中经常会用到，例如弹出式网站公告或广告，打开新窗口显示详细信息内容等。

这里所说的对话框是指通过指定的脚本代码打开一个新的对话框，并且该对话框可以有返回值，网页对话框可以分为网页模式对话框和非网页模式对话框。下面将通过具体实例介绍弹出窗口/对话框在 ASP 中的应用。

### 实例357 关闭弹出窗口时刷新父窗口

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\15\357

实例说明

关闭弹出窗口时刷新父窗口即关闭子窗口时自动刷新父窗口中的信息。用户通过 window 对象的 open() 方法打开一个新窗口（子窗口），当用户在该窗口中进行了数据库操作（如数据添加、修改和删除等）后，关闭子窗口时，系统会自动刷新父窗口来实现实时更新数据信息。运行本例，在新城校友录后台管理中的班级通讯录模块中，单击用于显示通讯录信息列表页面中的“添加班级信息”超级链接，将弹出班级名称录入页面，在该页面中添加班级名称信息（班级名称为“2000级计算机一班”）后，单击“提交”按钮，班级名称信息将被保存在数据表中，添加班级信息的窗口也将被自动关闭，同时系统

会自动刷新父窗口（也就是班级信息列表窗口），这时新添加的班级名称“2000级计算机一班”将在页面中显示，如图15.1所示。



图15.1 关闭弹出窗口时刷新父窗口

### 技术要点

本实例主要应用window对象的open()方法打开新窗口，并在新窗口中应用opener属性，返回打开本窗口的窗口对象。

### 语法格式：

window.opener

window.opener.方法

window.opener.属性

功能：返回的是一个窗口对象。opener属性与打开该窗口的父窗口相联系，当访问子窗口中opener属性时，返回的是父窗口，通过该属性，可以使用父窗口对象中的方法和属性。

### 实现过程

(1) 制作用于显示班级名称列表的班级管理页面，在该页面中加入空的超级链接，并在其单击事件中加入JavaScript脚本，实现打开用于录入班级名称的新窗口。代码如下。

```
<a href="#" class="style4"
onclick="JScript:window.open(' tongxun_add.asp', '', 'width=300,
height=230')">&nbsp;添加班级信息</a>
```

(2) 向页面中添加用于保存班级名称信息的ASP动态代码。

(3) 在保存班级名称信息不出现错误的时候，弹出“班级添加成功”的对话框并关闭当前窗口，同时刷新父窗口。代码如下。

```
<script language="javascript">
alert("班级添加成功!");
window.opener.location.reload();
//刷新父窗口的网页
window.close();
</script>
```

举一反三

根据本实例，读者可以：

在编写商务网站时应用此技术；

在开发办公自动化管理系统时应用此技术；

在编写宠物网站时应用此技术。

## [实例358 打开新窗口显示详细信息](#)

本实例是一个美化界面、简化操作的程序

实例位置：光盘\mingrisoft\15\358

实例说明

在网页设计时，通常需要同时考虑页面的美观程度及其下载速度，这时可以将要显示数据的概要信息显示在主页面中，并提供连接到显示详细信息的细节页面的超级链接，这样不但可以提高页面的下载速度，而且可以简化界面。运行本实例，在宠物网站的最新宠物宝

宝信息管理模块中，宠物信息主页面中只显示了宠物名称、宠物原价、宠物现价的概要信息，单击“详细信息”按钮即可打开显示宠物详细信息的新窗口，在该窗口中可以查看指定宠物的详细信息，如图15.2所示。

### 技术要点

本实例主要应用window对象的open()方法打开新窗口，在打开新窗口时，将宠物ID信息传递到新窗口中。这时的open()方法的语法格式如下。

```
window.open("链接文件名?参数名称=参数值", "", "width=窗口的宽,height=窗口的高")
```

技巧：在进行传递参数超级链接时，多个参数之间用“&”分隔。

### 实现过程

(1) 制作用于显示宠物概要信息的主页面。在该页面中加入空的超级链接，并在其单击事件中加入 JavaScript 脚本，实现打开新窗口显示宠物详细信息，这里需注意的是：在打开新窗口时，需要将宠物id号作为参数进行传递，代码如下。



图15.2 打开新窗口显示详细信息

```
<a href="#" onClick="javascript:window.open('chakan.asp?id=<%=rs_sale("ID")%&type=new','width=415,height=299,toolbar=no,location=no,status=no,menubar=no')">
```

```

```

```
</a>
```

(2) 制作用于显示宠物详细信息的细节页面。在该页面中通过 `request.QueryString("id")` 来获取主页面中传递的宠物 id 号，并根据宠物 id 号建立结果集，用于显示宠物的详细信息，代码如下。

```
<!--#include file="Conn/conn.asp"-->
```

```

<%
if request.QueryString("id")<>" " then
ID=request.QueryString("id")
end if
set rs=server.CreateObject("adodb.recordset")
sql="select * from tb_Goodswhere ID="&ID
rs.open sql,conn,1,3
IFnot rs.eofor not rs.bof then%>
<tablewidth="323"height="194"border="0"align="center"cell
padding="0"cellspacing="0">
<tr>
<tdwidth="157"height="103">
"width="155"height="95"bo
rder="1">
</td>
<tdwidth="156">
<tablewidth="156"height="119"border="0"align="center">
<tr>
<tdwidth="150"height="24"valign="bottom">
<span class="style10"><%=rs("goodsname")%></span>
</td>
</tr>
<tr>
<td height="22" class="STYLE6">原价: <%=rs("price")%>.00
元</td></tr>
<tr>

```

```

<td height="21">
<span class="style11">
<span class="STYLE4">现价: </span></span>
<span class="STYLE4"><%=rs("nowprice")%>.00元</span>
</td></tr>
<tr>
<td height="20" valign="top" class="STYLE6">发布日期:
<%=rs("INTime")%></td>
</tr>
</table>
</td>
</tr>
<tr>
<td height="12" colspan="2"><div align="center"
class="STYLE1">
<font color="#FFFFFF">宠物</font><span class="STYLE2">相
关信息</span></div>
</td>
</tr>
<tr>
<td height="55" colspan="2">
<div align="left">
<textarea name="textarea" cols="35" rows="4"
class="wenbenkuang"><%=rs("introduce")%></textarea>
</div>
</td>
</tr>

```

```
</table>
<%
End if
set rs=nothing
conn.close
set conn=nothing
%>
```

举一反三

根据本实例，读者可以：

开发新闻系统中的即时新闻、热点新闻等；

开发电子商务系统中的商品展示；

开发在线论坛中的主题信息浏览。

## 实例359 弹出网页模式对话框

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\15\359

实例说明

在制作办公自动化系统时，通常都少不了发送内部邮件的功能。为了方便用户快速将邮件信息发送到指定人的手中，防止因收件人用户名写错而发送失败，可以将公司内全部员工的用户名显示在下拉列表中。但是，如果公司内部的人员众多，这样也不是很方便，因为下拉列表框不能分栏显示。为解决该问题本书提供了一个比较不错的方法，那就是通过将用户名分栏显示在网页对话框中实现。运行本实例，单击“选择用户”按钮，即可弹出显示用户名列表的网页对话框，如图 15.3 所示。这时，用户单击选中的用户名后，该网页对话框

框将关闭，与此同时，选择的用户也将被添加到“收件人”文本框中。运行结果如图15.3所示。



图15.3 弹出网页模式对话框

### 技术要点

本实例主要应用window对象的showModalDialog()方法，该方法用于弹出网页（模式）对话框，详细介绍请参见实例010。

### 实现过程

(1) 制作发送邮件的页面，在该页面中添加收件人文本框，名称为“toname”，添加一个“选择用户”按钮，在该按钮的onClick事件中调用自定义JavaScript函数opendialog()。代码如下。

```
<input name="toname" type="text" class="Sytyle_text" id="toname">
```

```
<input name="Submit1" type="button" value="选择用户" class="Style_button_del" onClick="opendialog()">
```

(2) 编写自定义JavaScript函数opendialog(), 用于打开网页对话框, 并将网页对话框返回的值赋给“收件人”文本框, 代码如下。

```
<script language="javascript">
function opendialog()
{
var
someValue=window.showModalDialog("optUser.asp","", "dialogWidth=569px;dialogHeight=426px;status=no;help=no;scrollbars=no")
document.form1.toname.value=someValue;
}
</script>
```

(3) 制作显示用户名列表页面optUser.asp, 在该页面中应用循环将用户名分栏显示, 并提供将选择的用户返回到发送邮件页面和关闭该网页对话框的超级链接。完整代码如下。

```
<!--#include file="Conn/conn.asp"-->
<%
' 查询用户输入的员工姓名是否存在
Set rs_toname = Server.CreateObject("ADODB.Recordset")
sql_toname = "SELECT UserName,branch FROM Tab_User"
rs_toname.open sql_toname,conn,1,3
%>
<html>
<head>
<title>弹出网页模式对话框</title>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<link href="style.css" rel="stylesheet">
```

```
</head>
<body>
  <table width="557" height="401" border="0"
cellpadding="-2" cellspacing="-2"
  background="images/bbs/bbs_add.gif">
  <tr>
    <td width="817" height="349" valign="top"><table
width="100%" height="115"
  border="0" cellpadding="-2" cellspacing="-2">
  <tr>
    <td>&nbsp;</td>
  </tr>
</table>
  <table width="80%" height="77" border="0" align="center"
  cellpadding="-2" cellspacing="-2">
  <tr>
    <td height="13"></td>
  </tr>
  <%
for i=1 to rs_toname.recordcount
if i mod 2=1 then
%>
  <script language="javascript">
function action(user)
{
parent.window.returnValue=user;
window.close();
```

```

}
</script>
<tr>
<td height="25" valign="top">
<a href="#"
onClick="action(' <%=rs_toname("UserName")%>')">
<%=rs_toname("UserName")%> [<%=rs_toname("branch")%>]</a>
</td>
<%else%>
<td height="25" valign="top">
<a href="#"
onClick="action(' <%=rs_toname("UserName")%>')">
<%=rs_toname("UserName")%> [<%=rs_toname("branch")%>]</a>
</td>
</tr>
<%
end if
rs_toname.movenext()
next
%>
<tr>
<td></td>
</tr>
</table></td>
</tr>
</table>
</body>

```

</html>

举一反三

根据本实例，读者可以：

制作在日期选择器中选择日期；

在编写人力资源管理系统时，为员工指定所属部门；

在客房预订系统中，弹出网页对话框选择客户信息。

## 实例360 日期选择器

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\15\360

实例说明

在动态网站的人机交互页面中，经常会遇到要求用户输入日期的情况。如果让用户手动输入，不但操作不方便，而且日期格式不好控制。为了解决此问题，可以在页面中加入日期选择器。运行本实例，单击“时间图片”超级链接，将弹出具有日期选择功能的网页对话框，如图15.4所示。当用户单击需要的日期后，日期选择网页对话框将关闭，并将选择结果填入“外出时间”文本框。



图15.4 日期选择器

### 技术要点

本实例主要应用window 对象的showModalDialog()方法实现，关于showModalDialog()方法的详细介绍请读者参见实例010。

### 实现过程

(1) 创建需要调用日期选择器的页面，在该页面中添加两个图片按钮后，在图片的onClick事件中调用自定义的JavaScript函数loadCalendar()，调用该函数的程序代码如下。

```

```

```

```

(2) 编写自定义的JavaScript函数loadCalendar()，用于打开网页对话框调用日期选择器，代码如下。

```
<script language="javascript">
function loadCalendar(field)
```

```

    {
        var rtn =
window.showModalDialog("calender.asp","", "dialogWidth:280px
;dialogHeight:250px;status:no;help:no;scrolling=
no;scrollbars=no");
    if(rtn!=null)
    field.value=rtn;
    return;
    }
</script>

```

(3) 制作日期选择器页面calender.asp。首先，添加两个下拉列表，用于选择要显示的年份和月份，然后制作一个7行7列的表格，其中最顶行用于标记星期，其他行显示日期，显示日期的部分主要通过两层嵌套的for循环语句实现。代码如下。

```

<bodyonLoad="load(document.calendar)" topmargin="0">
<center>
    <formname="calendar">
        <tableborder="0" cellpadding="0" cellspacing="0">
            <tr>
                <td colspan="3"height="24"></td>
            </tr>
            <tr>
                <tdwidth="205"nowrap="nowrap"align="right">
                    <selectname="yearList"onChange="display_month(this.fo
                    rm)">
                        </select>
                </td>

```

```

        <tdwidth="65"nowrap="nowrap"align="left">
<selectname="monthList"size="1"onChange="display_mont
h(this. form)">
    <%for i=1 to12%>
        <option value="<%=i%>"><%=i%></option>
    <%next%>
</select>
</td>
<tdwidth="10"></td>
</tr>
<tr>
    <td colspan="3"height="6"></td>
</tr>
<tr>
    <td colspan="3">
<tableborder="0"cellpadding="1"cellspacing="0"align="ce
nter">
    <tr>
        <tdbgcolor="#82664F"><tableborder="0"
cellpadding="0"cellspacing="0">
            <trbgcolor="#82664F"height="18">
                <tdwidth="31"align="center"nowrap="nowrap">
                    <font color="#FF0000">日</font></td>
                <tdwidth="31"align="center"nowrap="nowrap">
一</td>
                <tdwidth="31"align="center"nowrap="nowrap">
二</td>

```

```

        <tdwidth="31" align="center" nowrap="nowrap">
三</td>
        <tdwidth="31" align="center" nowrap="nowrap">
四</td>
        <tdwidth="31" align="center" nowrap="nowrap">
五</td>
        <tdwidth="31" align="center" nowrap="nowrap">
六</td>
</tr>
<%
for r=0 to5
%>
<trbgcolor="#FFFFFF" height="18">
    <%for d=0 to6%>
        <td align="center">
            <input type="text" size="2" name="dayBox"
readOnlyOnClick="javascript:CheckDate(this.value)
;"onMouseOver="this.style.background=' #9966FF' "
onmouseout="this.style.background=' white' ">
        </td>
        <%next%>
    </tr>
    <%next%>
</table>
</td>
</tr>
</table></td>

```

```
    </tr>
  </table>
</form>
</center>
</body>
```

(4) 最后，编写实现日期选择功能的JavaScript代码，程序代码如下。

```
<script language=javascript>
<!--
    varmonthNames=newArray ( "",
"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12" );
    var endDay=newArray
(0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 );
    vardayNow= 0;
    varmonthNow=0;
    varyearNow=0;
    function load ( form )
    {
        set_month_year_now ();
        var found= false;
        for (
            varmonth=0;month<form.monthList.length;month++)
            if (
                form.monthList[month].text==monthNames[monthNow] )
            {
                form.monthList[month].selected= true;
            }
        }
    }
</script>
```

```

        found= true;
    }
    if ( !found)
    {
        error ();
        return;
    }
    var found= false;
    for (
        var year=0;year<form.yearList.length;year++)
        if ( form.yearList[year].text== yearNow ) {
            form.yearList[year].selected= true;
            found= true;
        }
    if ( !found)
    {
        error ();
        return;
    }
    display_month ( form );}
functionpreceding_month ( form )
{
    varmonth_selected= form.monthList.selectedIndex;
    var year_selected= form.yearList.selectedIndex;
    if ( !month_selected&& !year_selected ) {
        error ();
        return;
    }
}

```

```

    }
if (month_selected>0 )
    month_selected--;
else {
    month_selected=11;
    year_selected--;
}
form.monthList[month_selected].selected= true;
form.yearList[year_selected].selected= true;
display_month ( form );
}
function following_month ( form )
{
    varmonth_selected= form.monthList.selectedIndex;
    var year_selected= form.yearList.selectedIndex;
    if (month_selected>= ( form.monthList.length-1
)&&year_selected>= ( form.yearList.length-1 ) )
{
    error ();
    return;
}
if (month_selected<11 )
    month_selected++;
else{
    month_selected=0;
    year_selected++;
}
}

```

```

    form.monthList[month_selected].selected= true;
    form.yearList[year_selected].selected= true;
    display_month ( form );
}
function set_month_year_now ()
{
    var form=document.calendar;
    varnow=newDate ();
    monthNow=now.getMonth ()+1;
    yearNow=now.getYear ();
    dayNow=now.getDate ();
    yearNow= ( yearNow<100 ) ?yearNow+1900 :yearNow;
    var count=0
    for (var i=yearNow-103; i< yearNow+50; i++) {
        eval("form.yearList.options[count]=newOption(' "+i+"
', ' "+i+"')");
        count++;
    }
    form.yearList.selectedIndex=103;
    form.yearList.length= count;
}
function error ()
{
    alert ( "超出范围!" );
}
functiondisplay_month ( form )
{

```

```

var month= form.monthList.selectedIndex+1;
var year=parseInt ( form.yearList.options[
form.yearList.selectedIndex].text );
var start_day= start_day_in_month ( year,month );
var count=0;
for (var row=0; row<6; row++)
{
for ( var col=0; col<7; col++)
{
if ( row==0&& col< ( start_day-1 ) )
varday= "";
else if ( count< endDay[month] )
day=++count;
else
day= "";
form.dayBox[(row*7)+col].style.display= "";
form.dayBox[(row*7)+col].style.color= "black";
if (day== "")
{
form.dayBox[(row*7)+col].style.display= "none";
} else{
form.dayBox[(row*7)+col].value=day;
if (col%7==0)
form.dayBox[(row*7)+col].style.color= "red";
if
(yearNow==year&&monthNow==month&&dayNow==day)
form.dayBox[(row*7)+col].style.color= "blue";;
}
}
}
}

```

```

    }
  }
}
}
function start_day_in_month ( year, month )
{
  var  day, daynum, ndays, mnum;
  sday= start_day_in_year (year );
  endDay[2]= ( year%4 ) ?28 :29;
  if (month== 1 )
    daynum= sday;
  else{
    ndays= sday;
    for (mnum=2;mnum<month+1;mnum++)
      ndays=ndays+endDay[mnum-1];
    daynum=ndays%7;
  }
  daynum= (!daynum) ?7 :daynum;
  return (daynum);
}
function start_day_in_year ( year )
{
  var y, m, d;
  varn;
  y=year-1;m=13;d=1;
  n=d+2 *m+(Math.floor ( (0.6+(m+1) ) )+y );

```

```

        n=n+Math.floor ( ((y/4)-Math.floor ( (y/100 )
)+Math.floor ( ( y/400 ) ) ) )+2 ;
        n=Math.floor ( ( (n/7-Math.floor ( (n/7) ) ) *7+0.5 )
);
        return (n+1);
    }
functionCheckDate(strDay)
{
    vardocFrm=document.calendar;
    var choice_daynum=0;
    var current_daynum=0;
    varday_temp;
    if (strDay != "")
    {
        var strY=docFrm.yearList.value;
        var strM=docFrm.monthList.value;
        var curr_y=newString(yearNow);
        var curr_m=newString(monthNow);
        var curr_d=newString(dayNow);
        if (curr_m.length==1) curr_m= "0"+curr_m;
        if (curr_d.length==1) curr_d= "0"+curr_d;
        current_daynum=newNumber(curr_y+curr_m+curr_d) ;
        if (strM.length==1) strM= "0"+strM;
        if (strDay.length==1) strDay= "0"+strDay;
        choice_daynum=newNumber(strY+strM+strDay);
        parent.window.returnValue= strY+"-"+strM+"-
"+strDay; //将选择的日期传递到父窗口中
    }
}

```

```
        parent.window.close();
    }
    return false;
}
-->
</script>
```

举一反三

根据本实例，读者可以：

开发企业进销存管理系统中的数据查询模块；

开发电子商务管理系统中的订单查询模块；

开发用户注册模块中的生日选择功能。

## 实例361 弹出提示对话框并重定向网页

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\15\361

实例说明

由于在ASP中并没有提供弹出提示对话框的函数，而在程序开发时经常需要应用弹出提示对话框，并在确定后重定向网页，为此本章将实现该功能的代码封装在一个自定义的函数中，以便随时调用。运行程序，在登录新城校友录页面中输入用户名、密码后，单击“登录”按钮，将弹出如图15.5所示的提示对话框，单击“确定”按钮后，页面将重定向到新城校友录后台管理主体页面。



图15.5 弹出提示对话框并重定向网页

### 技术要点

本实例主要应用ASP内置对象Response的Write()方法将弹出提示对话框并重定向网页的JavaScript代码输出到浏览器实现。

Response对象的Write()方法的具体语法格式如下。

Response.Write(variant)

参数说明如下。

● variant: 可以包含任何有效的HTML 标记，但不能包括字符组合“%>”，如果确实需要，可以使用转义序列“%\\>”来代替，Web服务器在处理脚本时，将转义这一转义序列。

功能：将变量作为字符串写入当前的HTTP输出。

技巧：在编写脚本时，<%Response.Write (variant) %>可以用“<%=variant%>”代替。

### 实现过程

(1) 编写弹出提示对话框并重定向网页的函数 message()，该函数有两个参数 content（用于指定要显示的提示信息）和url（用于指

定重定向到的页面)，无返回值。该函数的代码如下。

```
<%  
function message(content,url)  
response.Write("<script  
language=' javascript' >alert(' "&content&"');window.location.hr  
ef=' "&url&"';</script>")  
response.End()  
end function  
%>
```

(2) 在需要弹出提示对话框的页面中首先将该函数放到页面的起始位置，然后在需要弹出提示对话框的位置调用 message() 函数，并将要显示的提示信息 and 重定向到的页面传递到message() 函数中，程序代码如下。

```
<%  
function filter_Str(InString)  
NewStr=Replace(InString, " ", " ")  
NewStr=Replace(NewStr, "<", "&lt;")  
NewStr=Replace(NewStr, ">", "&gt;")  
NewStr=Replace(NewStr, "chr(60)", "&lt;")  
NewStr=Replace(NewStr, "chr(37)", "&gt;")  
NewStr=Replace(NewStr, "\"", "&quot;")  
NewStr=Replace(NewStr, ";", ";;")  
NewStr=Replace(NewStr, "--", "-")  
NewStr=Replace(NewStr, "/*", " ")  
NewStr=Replace(NewStr, "%", " ")  
filter_Str=NewStr  
end function
```

```

    ifRequest.Form("admin_name")<>" " and
request.Form("admin_pass")<>" " then
    admin_name=filter_Str(request.Form("admin_name"))
    admin_pass=filter_Str(request.Form("admin_pass"))
    set rs=server.CreateObject("adodb.recordset")
    sql="select * from tb_adminwhere
admin_name=' "&admin_name&"' "
    rs.open sql,conn,1,3
    if rs.eof and rs.bof then
    %>
    <script language="javascript">
    alert("您输入的用户名称错误, 请重新输入!");
    history.back(-1);
    </script>
    <%
    else
    if rs("admin_pass")=admin_pass then
    session("name")=admin_name
    message "欢迎登录新城校友录后台管理系
统!", "schoolbook_index.asp"
    %>
    <%
    else%>
    <script language="javascript">
    alert("您输入的用户密码错误, 请重新输入!");
    history.back(-1);
    </script>

```

```
<%  
end if  
end if  
end if  
>%
```

举一反三

根据本实例，读者可以：

将弹出提示对话框的代码应用VBScript编写；

将重定向页面改为刷新指定页面或是返回指定页面等。

## 实例362 打开指定大小的新窗口并居中显示

本实例是一个定义窗口的程序

实例位置：光盘\mingrisoft\15\362

实例说明

虽然利用window对象的Open()方法可以很方便地实现弹出指定大小的新窗口，但是该方法弹出的窗口并不能居中显示在整个窗口中。为方便操作，本书将应用window对象的Open()方法打开指定大小的新窗口，并将设置该窗口居中的代码封装在一个单独的函数中，以后每当需要打开居中显示的新窗口时，就可以调用该函数，该函数也可以打开不控制窗口尺寸的新窗口。在本实例中单击新闻标题超级链接，即可打开居中显示的指定大小的新窗口显示新闻详细内容，如图15.6所示。



图15.6 打开指定大小的新窗口并居中显示

### 技术要点

本实例首先应用JavaScript中的window对象的open()方法，打开指定大小的新窗口，然后通过screen对象，获取屏幕的分辨率，再根据获取的值通过Width、Height属性将新窗口移动到屏幕居中位置。

注意：在客户端输出自定义的JavaScript函数时，一定要动态设置函数名，防止在同一页面中出现两个同名的自定义JavaScript函数，如果出现这种情况，页面中将显示脚本错误。

### 实现过程

(1) 编写函数openwin()，该函数有4个参数：url（打开窗口的URL地址）、w（新窗口的宽度，当w等于0时，不控制窗口的尺寸）、h（新窗口高度）和no（新窗口的编号，该编号在应用循环显示打开新窗口的多个超级链接时非常有用，否则程序将出现错误），无返回值。将其保存在function.asp文件中。代码如下。

```
<%  
function openwin(url,w,h)
```

```

    response.Write("<script language=' javascript' >function
openwin"&no&" () {")
    response.Write("if ("&w&"==' 0') {var
winhdc=window.open(' "&url&"');")
    response.Write( "var width=0;var height=0;}else{"")
    response.Write("var
winhdc=window.open(' "&url&"',',', ' width="&w&", height="&h&"');")
)
    response.Write("var width=(screen.width-"&w&")/2;")
    response.Write("var height=(screen.height-"&h&")/2;}")
    response.Write("winhdc.moveTo(width,height);")
    response.Write("}")
    response.Write("</script>")
end function
%>

```

(2) 在需要打开指定大小的新窗口并居中显示的页面中首先包含function.asp文件，然后在需要打开指定大小的新窗口并居中显示的位置调用openwin()函数，并将函数所需的4个参数传递到openwin()函数中，代码如下。

```

<!--#include file="function.asp"-->
<%
call openwin("news_detail.asp", 400, 260, 0)
%>
<td width="90%">
<a href="#" onClick="openwin0()">消费购物新感觉</a>
</td>

```

说明：这里需要说明的是，在调用openwin()函数时，会在页面中生成一个以字符串“openwin+no”命名的自定义的 JavaScript 函数。只有在页面中调用该函数时，才能打开新的窗口，这样做可以控制窗口的打开，防止每次页面运行时都会弹出一个新窗口。

举一反三

根据本实例，读者可以：

开发企业门户网站中的查看公司动态详细内容页面；

开发菜单使用频率的程序。把用户使用菜单的数据信息保存到数据库中，然后统计用户使用菜单的频率，并根据此频率调整菜单的显示顺序。

## 15.2 弹出确认对话框

这里所说的弹出确认对话框是指通过confirm()方法弹出一个确认对话框，并且该对话框可以有返回值，单击“确定”按钮时返回True，否则返回False。下面将通过具体实例介绍confirm()方法在ASP中的应用。

### 实例363 删除数据前弹出确认对话框

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\15\363

实例说明

在编写数据删除程序时，对要删除的数据给予相关提示，这一点非常重要，可以防止用户在错选的情况下删除重要的数据信息。因此，在进行数据删除前需要给予提示信息。运行本实例，单击每条记录信息后面的“删除”按钮时都会弹出一个提示框，提示用户是否确认删除。单击“确定”按钮时，将删除所选中的数据信息。单击“取消”按钮时，将取消删除操作。运行结果如图15.7所示。



图15.7 删除数据前弹出确认对话框

### 技术要点

本实例主要应用到JavaScript脚本中的window对象的confirm()方法实现弹出一个确认框，在确认框内显示提示字符串，当用户单击“确定”按钮时该方法返回True，单击“取消”按钮时该方法返回False。同时，利用到复选框进行多项选择。下面对window对象的confirm()方法和复选框进行介绍。

window对象的confirm()方法的语法如下。

window.confirm(提示字符串)

复选框对象的相关属性如下。

- Checked: 表示复选框是否选中。
- DefaultChecked: 表示默认时被选中。
- Form: 当前窗体。
- Name: 对象属性访问。
- Type: 对象类型。
- Value: 对象值。

复选框对象的相关方法如下。

- Blur: 失去焦点。
- Click: 鼠标单击。

● Focus: 获取焦点。

实现过程

(1) 创建显示记录信息的记录集, 代码如下。

```
<!--#include file="conn/conn.asp"-->  
<%  
set rs=server.CreateObject("adodb.recordset")  
sql="select * from Tab_bm order by time1 desc"  
rs.open sql,conn,1,3  
%>
```

(2) 实现记录信息的分页显示的代码如下。

```
<%  
If(rs.Eof)Then  
Response.Write("暂无部门计划信息!")  
Else  
%>  
<tablewidth="728"height="55"border="0"align="center"cellp  
adding="0"cellspacing="0">  
<tr>  
<tdheight="23"valign="top"><div  
align="center"class="STYLE2">部门名称</div></td>  
<td><div align="center"class="STYLE2">计划主题</div></td>  
<td><div align="center"class="STYLE2">计划内容</div></td>  
<td><div align="center"class="STYLE2">发表日期</div></td>  
<td><div align="center"class="STYLE2">删除</div></td>  
</tr>  
<%  
' 分页
```

```

rs.pagesize=8
page1=cLng(request("page1"))
ifpage1<1 thenpage1=1
rs.absolutePage=page1
for i=1 to rs.pagesize
kk=rs("title")
k=rs("content")
%>
<tr>
<tdheight="23" class="STYLE2"><div align="center">
<%=rs("name1")%>&nbsp;  </div></td>
<tdheight="23" class="STYLE2"><div align="center">
<%if len(kk)>8 then%>
<a href="#" onClick="Javascript:window.open(' bm_xianxi.asp?
ID=<%=rs("ID")%>',',', width=456,height=459')">
<%=replace(left(kk,8)&"....",chr(13),"<br>")%></a>
<%else%>
<a href="#"
onClick="Javascript:window.open(' bm_xianxi.asp?ID=
<%=rs("ID")%>',',', width=456,height=459')"><%= kk%></a>
<%end if%>&nbsp;  
</div>
</td>
<tdheight="23" class="STYLE2"><div align="center">
<%if len(k)>10 then%>
<%=replace(left(k,10)&"....",chr(13),"<br>")%>
<%else%>

```

```

<%=k%>&nbsp;
<%end if%>
</div></td>
<tdheight="23" class="STYLE2">
<div align="center"><%=rs("time1")%>&nbsp;</div>
</td>
<tdheight="23">
<div align="center"><a href="bm_del.asp?id=
<%=rs("id")%>" onclick="returnconfirm(' 是否确认删除?')">
<imgsrc="Images/del.gif"width="16"height="16"border="0"></a>
</div>
</td>
</tr>
<%
rs.movenext
if rs.eof then exit for
next
%>

```

(3) 单击记录后面的“删除”按钮，弹出确认对话框，代码如下。

```

<a href="bm_del.asp?id=<%=rs("id")%>" onclick="return
confirm(' 是否确认删除?')"></a>

```

### 举一反三

根据本实例，读者可以：

在商品信息管理程序中维护数据；

在采购信息管理程序中维护数据。

## 实例364 安全退出前弹出确认对话框

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\15\364

实例说明

在编写用户登录程序时，需要为用户提供一个退出系统的功能模块，这样可以防止其他用户以非法的方式进行系统登录操作。因此，在用户进行安全退出前需要给予提示信息。运行本实例，单击“宠物已购完退出”按钮时将会弹出一个提示框，提示用户是否确定退出。单击“确定”按钮时，将退出用户登录状态。单击“取消”按钮时，将取消确认操作。运行结果如图15.8所示。

技术要点

本实例主要应用JavaScript脚本中的window对象的confirm()方法实现弹出一个确认框。该方法的相关介绍请参见实例363。

实现过程

(1) 在ASP的代码中使用session对象，Abandon()方法可以使session变量立即失效，并激活session\_OnEnd事件，代码如下。

```
<%  
session.Abandon()  
response.Redirect("index.asp")  
%>
```



图15.8 安全退出前弹出确认对话框

(2) 通过以下代码设置“宠物已购完退出”按钮的 Onclick 事件调用 JavaScript 脚本中的 window 对象 confirm() 方法实现弹出一个确认框，代码如下。

```
<input name="Submit4" type="button" value="宠物已购完退出" onClicK="if( confirm('是否真的退出?'))  
{window.location.href=' exit.asp' ;}">
```

举一反三

根据本实例，读者可以：

在进行网站后台登录后退出；

在用户成功登录系统进行数据操作后退出管理。

## 15.3 其他

在以上的几节中分别讲解了弹出窗口/对话框控制和弹出确认对话框等一些经典的实例。本节再通过几个典型实例对JavaScript脚本与ASP结合方面做一个补充。

### 实例365 远程获取其他网页情报

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\15\365

实例说明

在开发某些网站或者系统时，可能需要获取其他网页的信息。运行本实例，将自动获取http://192.168.1.233:8099/cpjs/cpbj.asp网页上的信息。运行结果如图15.9所示。

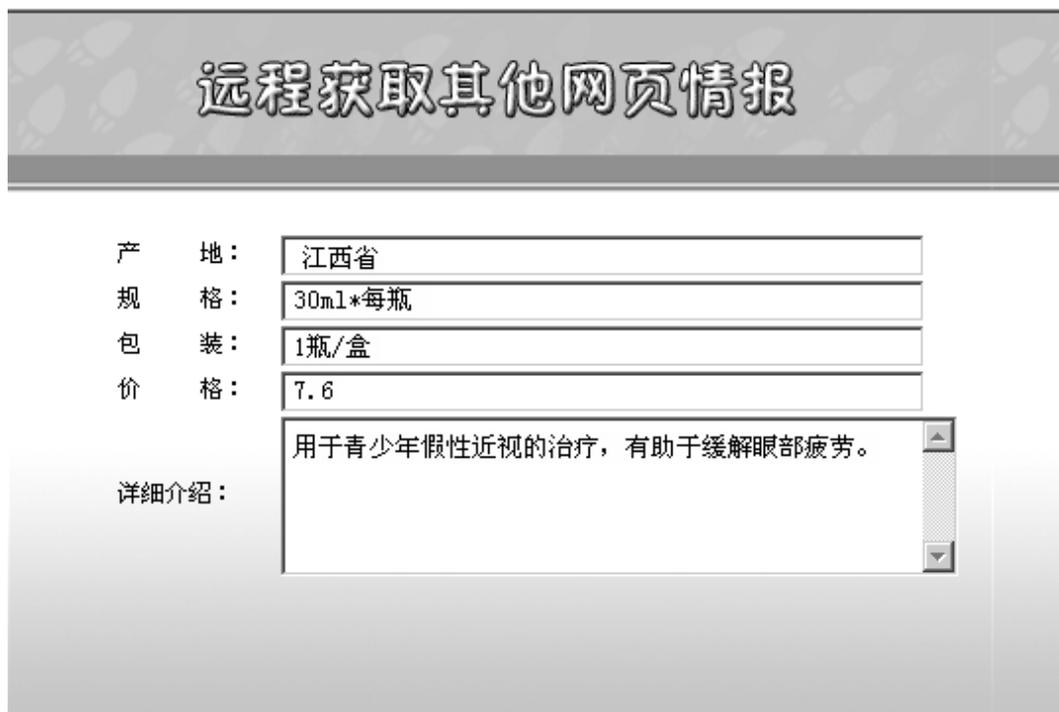


图15.9 远程获取其他网页情报

### 技术要点

本实例主要应用到了JavaScript中的indexOf()方法。

indexOf()方法返回某字符串在另一字符串中第一次出现的位置。

### 语法:

```
strVariable.indexOf(substring, startindex)
```

参数说明如下。

- substring: 必选。接受搜索的字符串表达式。
- startindex: 必选。要搜索的字符串表达式。

### 实现过程

(1) 网页加载的时候调用获取情报的自定义函数,代码如下。

```
<body onLoad="aa()">
```

(2) 定义获取情报的自定义函数,代码如下。

```
<object classid="clsid:48E59293-9880-11CF-9754-00AA00C00908" name="inte">
</object>
<script language="javascript">
function aa()
{
html=inte.openurl("http://192.168.1.233:8099/cpjs/cpbj.asp");
productname="珍视明滴眼液";
//想要查询的字符串
productlocation=html.indexOf(productname);
//indexOf是指向string对象内第一次出现子字符串的字符位置
areal="产地: ";
```

```

//想要查询的字符串
arealocation=html.indexOf(areal,productlocation);
ending1="</font></td>";
//查询字段是</font></td>
productlocation1=html.indexOf(ending1,arealocation);
//以arealocation作为开始位置，查询字段为ending的位置
productarea=html.slice(arealocation+3,productlocation1);
//返回字符串片断
form1.productarea.value=productarea;
standard="规格： ";
standardlocation=html.indexOf(standard,productlocation);
ending11="</font></td>";
standardlocation1=html.indexOf(ending11,standardlocation)
;
standards=html.slice(standardlocation+3,standardlocation1
);
form1.standard.value=standards;
standardlocation=html.indexOf(standard);
casing="包装： ";
casinglocation=html.indexOf(casing,productlocation);
ending12="</font></td>";
casinglocation1=html.indexOf(ending12,casinglocation);
casings=html.slice(casinglocation+3,casinglocation1);
form1.casing.value=casings;
price="单价： ";
pricelocation=html.indexOf(price,productlocation);
ending13="（元）</font></td>";

```

```

    pricelocation1=html.indexOf(ending13,pricelocation) ;
    pricel=html.slice(pricelocation+3,pricelocation1) ;
    form1.price.value=pricel;
    html1=inte.openurl("http://192.168.1.233:8099/cpjs/cpjs.a
sp");
    product="珍视明滴眼液";
    productlocation=html1.indexOf(product);
    introduce="详细介绍： ";
    locations=html1.indexOf(introduce,productlocation);
    ending="</font></td>";
    location1=html1.indexOf(ending,locations);
    introduction=html1.slice(locations+5,location1);
    form1.introduce.value=introduction;
}
</script>

```

(3) 设置相应的表单元素用于显示获取的信息，代码如下。

```

<tablewidth="500"height="363"border="0"align="center"back
ground="images/houtai.gif">
<tr>
<td align="center"height="79"valign="bottom">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<tdwidth="755"height="274"align="center">
<form name="form1"method="post"action="">
<tablewidth="540"height="241"border="0"align="center"ce
llpadding="0" cellspacing="0">
<tr>

```





本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\15\366

实例说明

对于一个导航文字很多并且可以对导航内容进行分类的网站来说，可以将页面中的导航文字以树视图的形式显示，这样不仅可以有效节约页面，而且也可以方便用户查看。运行本实例，如图15.10所示，在函数分类查看的页面中，就加入了树状导航菜单，在页面的左侧“函数分类列表”中以树状导航菜单显示对于函数的分类。在默认情况下，第1个节点是展开的，单击该节点名称前的减号“-”可以将该节点折叠，单击节点名称前的加号“+”可以展开指定节点。如果某个节点没有相关子节点，则直接显示减号“-”，同时该节点名称无超级链接。



图15.10 树状导航菜单

技术要点

本实例通过JavaScript控制表格的行

说明：网页中的<table>、<tr>、<td>、<img>、<form>、<input>、<button>等 HTML 标记都具有display属性。

### 实现过程

(1) 利用 JavaScript 定义一个函数 ShowTR(), 用于显示或隐藏表格中指定行的内容。ShowTR()函数的代码如下。

```
<script language="javascript">
ShowTR(img1,OpenRep1)
//设置第1个节点为展开状态
functionShowTR(objImg,objTr)
{
    if(objTr.style.display=="block")
    {
        objTr.style.display="none";
        objImg.src="Images/jia.gif";
        objImg.alt="展开";
    }
    else
    {
        objTr.style.display="block";
        objImg.src="Images/jian.gif";
        objImg.alt="折叠";
    }
}
</script>
```

注意：为了实现在默认情况下第 1 个节点为展开状态，可以在该函数前加入一句调用该函数的代码。

(2) 从数据表tb\_Type中查询函数的分类信息，代码如下。

```

<%
set rs_Type=server.CreateObject("ADODB.RecordSet")
sql="select * from tb_Type where sort='函数应用'"
rs_Type.open sql,conn,1,3
%>

```

(3) 循环显示函数分类信息，并查询各分类中所包括的函数名称，如果该分类有相对应的函数，则在节点前显示加号“+”，否则显示减号“-”。注意：需要为每个包含子节点的节点设置超级链接，该超级链接执行的操作是调用自定义的 JavaScript 函数，实现节点的展开与折叠，代码如下。

```

<tablewidth="100%"height="25" border="0"cellpadding="0"
cellspacing="0">

```

```

<%
rs_Type.movefirst()
if rs_type.eof and rs_type.bof then
%>
<tr>
<td align="center">暂无函数分类信息!</td>
</tr>

```

```

<%
else
m=1
for i=1 to rs_type.recordcount
set rs_L=server.CreateObject("ADODB.RecordSet")
sql="selectP.*,T.name asTypename from tb_functionP inner
join

```

```
tb_TypeTonP.Type=T.IDwhereP.Type="&rs_Type("ID")&"orderbyP.hi
tdesc"
```

```
rs_L.open sql, conn, 1, 3
```

```
%>
```

```
<tr>
```

```
<td>
```

```
<%
```

```
    If(rs_L.eof and rs_L.bof)Then
```

```
%>
```

```
    &nbsp;  
```

```
    <%=server.HTMLEncode(rs_Type("name"))%>
```

```
<%
```

```
    Else
```

```
%>
```

```
    &nbsp;  
```

```
    <ahref="Javascript:ShowTR(img<%=m%>,OpenRep<%=m%>)">">
```

```
</a>
```

```
    <ahref="Javascript:ShowTR(img<%=m%>,OpenRep<%=m%>)">
```

```
    <%=server.HTMLEncode(rs_Type("name"))%></a>
```

```
<%
```

```
End If
```

```
%>
```

```
</td>
```

```
<%
```

```

if rs_L.recordcount>0 then
%>
  <tr id="OpenRep<%=m%>" style="display:none;">
    <td colspan="6">
  <%
for j=1 to rs_L.recordcount
%>
  <tablewidth="100%" border="0" cellspacing="-2"
cellpadding="-2">
  <tr
onMouseOver="this.style.background=' #EEEEEE' "onMouseOut="this
.style.background=' ' ">
    <tdheight="25"align="center">
      <tablewidth="90%" border="0" cellspacing="0"
cellpadding="0">
        <tr>
          <tdwidth="12%"align="center">
            
          </td>
          <tdwidth="88%"><%=rs_L("name")%></td>
        </tr>
      </table>
    </td>
  </tr>
</table>
<%

```

```
m=m+1
'注意，该条语句一定不能少
rs_L.movenext
next
%>
    </td>
    <%end if%>
</tr>
<%
rs_Type.movenext
If rs_Type.EofThen exitFor
next
end if
%>
</table>
```

举一反三

根据本实例，读者可以：

在在线论坛中的论坛管理页面中应用目录树结构显示主题及回复信息；

实现企业进销存管理系统中的导航菜单；

在企业人事管理系统中，进行企业内部邮件发送时，根据部门选择收件人。

## [实例367 使用XML实现不刷新页面查询数据](#)

本实例是一个提高网站安全性的程序

实例位置：光盘\mingrisoft\15\367

## 实例说明

本实例主要使用 XML 技术实现不刷新页面来查询数据库。例如，在文本框中输入所要查询的条件后，单击“查询”按钮，即可将满足条件的查询结果以下拉列表的形式显示出来，运行结果如图15.11所示。

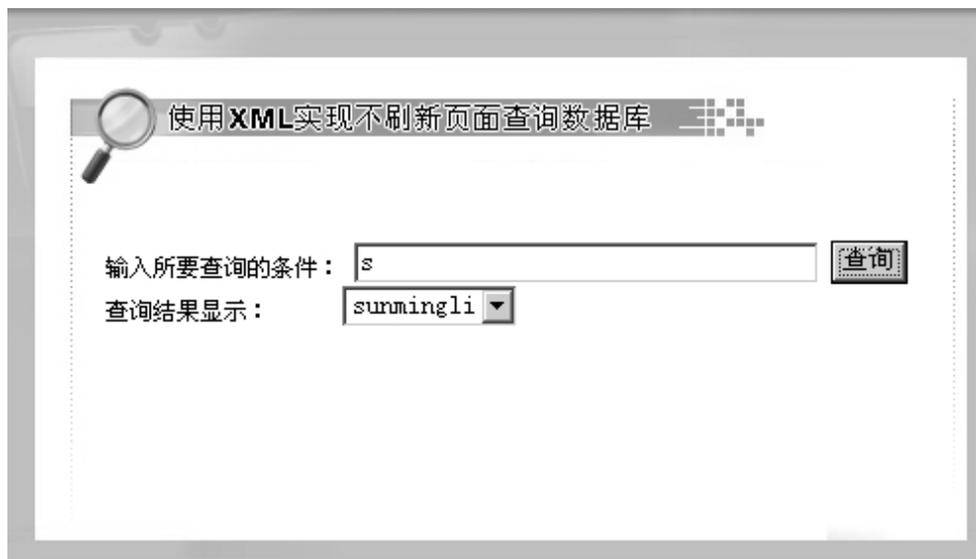


图15.11 使用XML实现不刷新页面查询数据

## 技术要点

本实例主要应用 XML 实现不刷新页面查询数据，首先将需要放入列表框中的数据信息从数据库查询出来保存到XML数据岛中，然后使用JavaScript脚本将输入的查询条件与XML数据岛中的记录信息进行比较，并把符合条件的记录信息放入到下拉列表框中，从而实现不刷新页面查询数据库中的数据。

## 实现过程

(1) 通过JavaScript脚本中的Function语句自定义Mycheck()函数实现信息的初始化，并将数据库中的数据放入到下拉列表框中，代码如下。

```
<ScriptLanguage="JavaScript">  
functionMycheck()
```

```

{
varEmid, Emname;
//用于返回第一个元素
sml=document.all.xmlmpy.childNodes.item(0);
for(i=0;i<sml.childNodes.length;i++)
{
    prevailnode=sml.childNodes(i);
    Emid=sml.childNodes(i).getAttribute("emid");
    for(j=0;j<prevailnode.childNodes.length;j++)
    {
DD=prevailnode.childNodes(j).nodeName;
if(DD=="User_Name")
    {
Emname=prevailnode.childNodes(j).text;
    }
    }
//将符合条件的信息写进select中
if(Emid!=""&&Emname!="")
    {
op=document.createElement("option");
op.text=Emname;
op.value=Emid;
menulist.add(op);
    }
    }
}
</script>

```

(2) 应用JavaScript脚本创建Mycheck1()函数, 该函数主要用于存放从数据库中查询到的数据信息, 代码如下。

```
<ScriptLanguage="JavaScript">
//从数据岛中检索数据, 并将其装入到列表框中
functionMycheck1()
{
varEmname, Emid;
Emname="";
Emid="";
text0=window.condition.value;
//清除列表框
count0=menulist.length
for(i=count0-1;i>=0;i--)
{
menulist.remove(i);
}
sml=window.xmlempty.childNodes(0);
for(i=0;i<sml.childNodes.length;i++)
{
prevailitem=sml.childNodes(i);
Emid=sml.childNodes(i).getAttribute("emid");
for(j=0;j<prevailitem.childNodes.length;j++)
{
if(prevailitem.childNodes(j).nodeName=="User_Name")
{
ep=prevailitem.childNodes(j).text;
}
}
}
}
```

```

//查找相匹配的项
if(ep.indexOf(text0)!=-1)
{
Emname=ep;
}
if(Emid!=""&&Emname!="")
{
op=document.createElement("option");
op.value=Emid;
op.text=Emname;
window.menulist.add(op);
Emid="";
Emname="";
}
}
}
}
</script>

```

(3) 在页面中的适当位置插入用于获取查询条件的表单及表单元  
素，代码如下。

```

<bodybgcolor="#FFFFFF"
text="#000000"onLoad="javascript:Mycheck()">
<tablewidth="498"height="283"border="0"cellpadding="0"
cellspacing="0">
<tr>
<tdwidth="498"height="283"background="0.jpg">
<tablewidth="90%"border="0"align="right"cellpadding="0"ce

```

```

llspacing="0">
    <tr>
        <td><span class="STYLE2">输入所要查询的条件:
    </span>
        <inputname="condition" type="text"
    id="condition" size="32">
        <input type="button" name="Submit" value="查
    询"onClick="javascript:Mycheck1()">
        </td>
    </tr>
    <tr>
        <td class="STYLE2"> 查询结果显示:
        <selectname="menulist" size="1">
            </select>
        </td>
    </tr>
</table>

```

(4) 连接数据库，并从数据库中筛选指定的数据信息，代码如下。

```

<%
    SetConn=Server.CreateObject("ADODB.Connection")
    sql="Driver={MicrosoftAccessDriver (*.mdb)};DBQ=
"&Server.mappath("Database/db_Catecity.mdb")
    conn.open(sql)
    set rs=server.CreateObject("adodb.recordset")
    sql="SelectUser_id,User_Name from tb_User"
    rs.Open sql, conn, 1, 3

```

%>

(5) 实现将数据库中的信息存放到数据岛中，代码如下。

```
<xml id="xmlempy">
<catecity>
<%
do while not rs.eof
%>
<User_id emid="<%=rs("User_id")%>">
<User_Name> <%=rs("User_Name")%> </User_Name >
</User_id>
<%
rs.movenext
%>
<%
loop
%>
</catecity>
</xml>
<%
rs.close
set rs=nothing
%>
</td>
</tr>
</table>
</body>
```

举一反三

根据本实例，读者可以：

在开发搜索网站时，对数据信息进行查询；

在开发办公自动化管理系统时，对员工信息进行查询；

在开发宠物网站时，对最新宠物信息进行查询。

# 第16章 JavaScript与JSP结 合

窗口与对话框

无刷新技术

调用与控制

其他

## 16.1 窗口与对话框

在网站应用程序中，用户在进行相关操作时，系统应及时地给用户以提示或打开新的窗口，以帮助用户正确地操作和使用网站。本节将介绍如何通过 JavaScript与JSP结合控制窗口与对话框。

### 实例368 弹出提示对话框并重定向网页

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\16\368

实例说明

由于在JSP中并没有提供弹出提示对话框的函数，而且在程序开发时经常需要应用弹出提示对话框，并在确定后重定向网页。下面将通过一个具体实例介绍在JSP中如何实现弹出提示对话框并重定向网页。运行本实例，在添加商品信息页面的各文本框中输入相应商品信息，然后单击“保存”按钮，将弹出图16.1所示的提示对话框，单击“确定”按钮后，页面将重定向到添加商品信息页面，即本实例的首页。



图16.1 弹出提示对话框并重定向网页

### 技术要点

本实例主要通过JavaBean将弹出提示对话框并重定向网页的JavaScript代码组合成一个字符串，并返回该字符串，在页面中通过输出该JavaBean返回的字符串实现弹出提示对话框并重定向网页。

输出JavaBean返回的字符串主要应用JSP内置对象out的println()方法实现，该方法的具体语法格式如下。

```
out.println(variant)
```

参数说明如下。

- variant: 可以包含任何有效的HTML 标记，但不能包括字符组合“%>”，如果确实需要，可以使用转义序列“%\\>”来代替。Web服务器在处理脚本时，将转义这一转义序列。

功能：将变量的值输出到客户端浏览器。

技巧：在编写脚本时，`<%out.println(variant)%>`可以用“`<%=variant%>`”代替。

### 实现过程

(1) 编写组成弹出提示对话框并重定向网页的字符串的JavaBean “PopDialog”，将其保存到“com”包中，代码如下。

```
package com;
public class PopDialog {
}
```

(2) 在JavaBean “PopDialog” 中创建一个方法messageBox()，该方法有两个string型参数，分别是message（用于指定要显示的提示信息）和url（用于指定重定向到的页面），返回值为连接后的字符串，代码如下。

```
public String messageBox(String message,String url) {
    String str="<script
language=' javascript' >alert(' "+message+"');";
    str=str+"window.location.href=' "+url+"';</script>";
    return str;
}
```

(3) 在需要弹出提示对话框的页面中首先通过`<jsp:useBean>`指令获取 PopDialog 对象的实例，然后在需要弹出提示对话框的位置调用 PopDialog对象的 messageBox() 方法，并将要显示的提示信息和重定向到页面的 URL 地址传递到 messageBox() 方法中，最后应用 out.println() 方法输出messageBox() 方法的返回值，关键代码如下。

```
<jsp:useBean id="popDialog"
scope="page"class="com.PopDialog"/>
<%
```

```

String goodsName=chStr.chStr(goods.getGoodsName());
ResultSet rs=conn.executeQuery("select * from
tb_goods where goodsname='"+goodsName+"'");
try{
    if (rs.next()) {
        out.println(popDialog.showMessageDialog("该商品信息已经添
加! ", "index.jsp"));
    }else{
        int ret=0;
        ret=goodsDao.insert(goods);
        if (ret!=0) {
            out.println(popDialog.showMessageDialog("商品信息添加成
功! ", "index.jsp"));
        }else{
            out.println(popDialog.showMessageDialog("商品信息添加失
败! ", "index.jsp"));
        }
    }
} catch(Exception e) {
    System.out.println(e.getMessage());
}
%>

```

举一反三

根据本实例，读者可以：

将重定向页面改为刷新指定页面或是返回指定页面等。

## [实例369 打开新窗口显示详细信息](#)

本实例是一个方便、实用的程序

实例位置：光盘\mingrisoft\16\369

实例说明

在网页设计时，考虑到页面的美观程度及其下载速度，可以将要显示数据的概要信息显示在主页面中，并提供连接到显示详细信息的细节页面的超级链接，这样不但可以提高页面的下载速度，而且可以简化界面。运行本实例，在网上图书超市后台管理的图书信息管理模块中，图书信息主页面中只显示了书号、图书名称等概要信息，单击图书名称超级链接即可打开新窗口，显示图书详细信息，如图16.2所示。

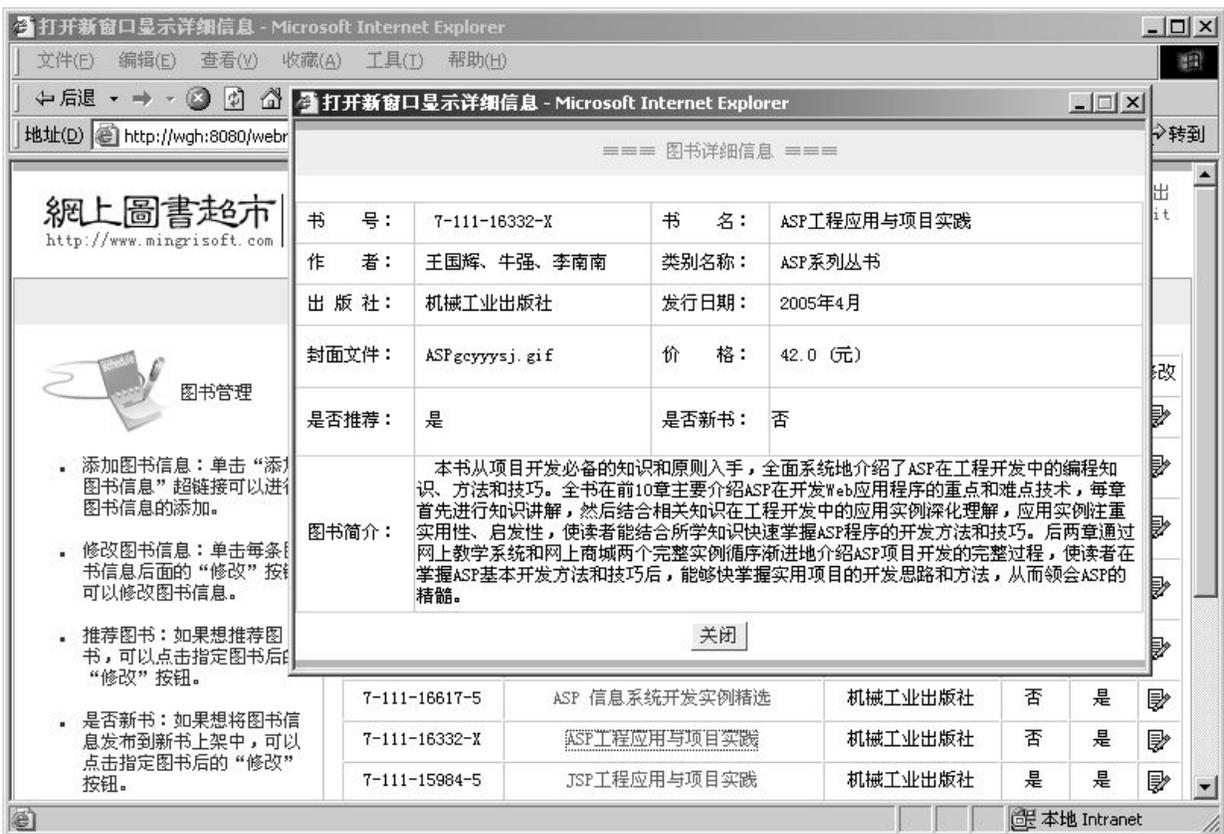


图16.2 打开新窗口显示详细信息

技术要点

本实例主要应用window对象的open()方法打开新窗口，在打开新窗口时，将商品id信息传递到新窗口中。这时的open()方法的语法格

式如下。

```
window.open("链接文件名?参数名称=参数值","","width=窗口的宽,height=窗口的高")
```

技巧：在进行传递参数超级链接时，多个参数之间用“&”分隔。

实现过程

(1) 制作用于显示图书概要信息的主页面。在该页面中加入空的超级链接，并在其单击事件中加入 JavaScript 脚本，实现打开新窗口显示图书详细信息。这里需注意的是，在打开新窗口时，需要将书号作为参数进行传递。关键代码如下：

```
<a href="#"
onclick="window.open('book_detail.jsp?ISBN=
<%=ISBN%>','','width=560,height=361')">
<%=BookName%></a>
```

(2) 制作用于显示图书详细信息的细节页面。在该页面中通过 `request.getParameter("ISBN")` 来获取主页面中传递的书号，并根据书号建立结果集，用于显示图书的详细信息，关键代码如下。

```
<jsp:useBean id="conn" scope="page" class="com.ConnDB"/>
<jsp:useBean id="chStr" scope="page" class="com.ChStr"/>
<%if (request.getParameter("ISBN")!=null) {
    String ISBN=request.getParameter("ISBN");
    ResultSet rs=conn.executeQuery("select * from
tb_bookinfo where ISBN=' "+ISBN+" '");
    Stringbookname="";
    StringType="";
    StringTPI="";
    String author="";
    String intro="";
```

```

StringpDate="";
String cover="";
floatprice=0;
String commend="";
Stringnewbook="";
    try{
        if(!rs.next()){
            out.println("<script
lanuage=' javascript' >alert(' 您的操作有
误!');window.location.href=' inde. jsp' ;</script>");
        }else{
            bookname=rs.getString("bookname");
            Type=rs.getString("Type");
            TPI=rs.getString("publisher");
            author=rs.getString("writer");
            intro=rs.getString("introduce");
            price=rs.getFloat("Price");
            pDate=rs.getString("pDate");
            cover=rs.getString("cover");
            newbook=rs.getInt("newbook")==0 ? "否":"是";
            commend=rs.getInt("commend")==0 ? "否":"是";
        }
    }catch(Exception e){
        System.out.println(e.getMessage());
    }
%>
..... //此处省略了显示图书信息的HTML代码

```

```
<%
    conn.close();
}else{
    out.println("<script language=' javascript' >alert(' 您的操作有误!');window.location.href=' inde. jsp' ;</script>");
}%>
```

### 举一反三

根据本实例，读者可以：

开发企业门户型网站中的系统公告或热点新闻等；

开发电子商务系统中的商品展示；

实现单击博客中的文章名称显示文章详细信息。

## 实例370 打开指定大小的新窗口并居中显示

本实例是一个定义窗口的程序

实例位置：光盘\mingrisoft\16\370

### 实例说明

虽然利用window对象的open()方法可以很方便地实现弹出指定大小的新窗口，但是该方法弹出的窗口并不能居中显示在整个窗口中，为方便操作，这里将应用window对象的open()方法打开指定大小的新窗口，并将设置该窗口居中的代码封装在一个JavaBean中，以后每当需要打开居中显示的新窗口时，就可以调用该JavaBean。运行本实例，单击新品上架栏目中的“详细信息”按钮，即可打开居中显示的指定大小的新窗口显示商品的详细信息，如图16.3所示。



图16.3 打开指定大小的新窗口并居中显示

### 技术要点

本实例主要通过JavaBean将打开指定大小的新窗口并居中显示的JavaScript代码组合成一个字符串，并返回该字符串，在页面中通过输出该JavaBean返回的字符串实现打开指定大小的新窗口并居中显示。

注意：在客户端输出自定义的JavaScript函数时，一定要动态设置函数名，防止在同一页面中出现两个同名的自定义JavaScript函数。如果出现同名的自定义JavaScript函数，页面中将显示脚本错误。

### 实现过程

(1) 编写组成打开指定大小的新窗口并居中显示的字符串的 JavaBean “OpenCenterWindow”，将其保存到“com”包中，代码如下。

```
package com;
public class OpenCenterWindow {
}
```

(2) 在 JavaBean “OpenCenterWindow” 中创建一个方法 open()，该方法有 4 个 string 型的参数，分别是 url（打开窗口的 URL 地址）、w（新窗口的宽度，当 w 等于 0 时，不控制窗口的尺寸）、h（新窗口高度）和 no（新窗口的编号，该编号在应用循环显示打开新窗口的多个超级链接时，非常有用，否则程序将出现错误），返回值为连接后的字符串，代码如下。

```
public String open(String url, String w, String h, String no) {
    String str = "<script
language=' javascript' >function openwin"+no+" () {";
    str = str + "if (" + w + " == ' 0' )
{var winhdc = window.open(' "+url+" ');";
    str = str + "var width = 0; var height = 0; } else {";
    str = str + "var winhdc = window.open(' "+url+" ', ' ', ' width = " + w +
", height = " + h + " ');";
    str = str + "var width = (screen.width - " + w + ") / 2;";
    str = str + "var height = (screen.height - " + h + ") / 2; }";
    str = str + "winhdc.moveTo(width, height);";
    str = str + " } </script >";
    return str;
}
```

(3) 在需要打开居中显示指定大小的新窗口的页面中，首先通过<jsp:useBean>指令获取OpenCenterWindow 对象的实例，然后在需要打开居中显示指定大小的新窗口的位置调用OpenCenterWindow 对象的open()方法，再应用 out.println 输出 open()方法的返回值，最后在打开新窗口的超级链接或按钮上调用生成的自定义JavaScript函数openwinX()，关键代码如下。

```
<jsp:useBean id="openwin" scope="page"
class="com.OpenCenterWindow"/>
<%strs=String.valueOf(s);
out.println(openwin.open("goods_detail.jsp?
ID="+new_ID,"587","230",strs)); %>
<input name="see" type="button" class="btn_grey"
onClick="openwin<%=strs%>()" value="详细信息">
```

说明：这里需要说明的是，在调用 openwinX()函数时，会在页面中生成一个以字符串“openwin+no”命名的自定义的JavaScript函数。只有在页面中调用该函数时，才能打开新的窗口，这样做可以控制窗口的打开，防止每次页面运行时都会弹出一个新窗口。

举一反三

根据本实例，读者可以：

开发企业门户网站中的查看公司动态的详细内容页面；

开发新闻网中的热点新闻、即时新闻的详细信息页面。

## 实例371 弹出网页模式对话框

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\16\371

实例说明

在开发电子商城后台的商品信息添加模块时，为了方便操作，可以将商品类别显示在下拉列表框中，但是当类别信息过多时，放在下拉列表框中不仅会影响页面运行速度，而且不易查找。为解决该问题本书提供了一个比较不错的方法，那就是通过将商品类别分栏显示在网页对话框中实现。运行本实例，单击“选择商品类别”按钮，即可弹出显示商品类别列表的网页对话框，如图16.4所示。单击选中的类别名称后，该网页对话框将关闭，与此同时，选择的类别名称也将被添加到“所属类别”文本框中。

### 技术要点

本实例主要应用window对象的showModalDialog()方法，该方法用于弹出网页（模式）对话框，详细介绍请参见实例010。



图16.4 弹出网页模式对话框

### 实现过程

(1) 制作商品信息添加的页面，在该页面中添加一个名称为 `typename` 的“所属类别”文本框和一个“选择商品类别”按钮，在该按钮的 `onClick` 事件中调用自定义 JavaScript 函数 `opendialog()`。关键代码如下。

```
<input name="typename" type="text" class="Sytle_text"
id="goodsName" size="30">
<input name="Submit1" type="button" value="选择商品类别"
class="btn_grey"
onClick="opendialog()">
```

(2) 编写自定义 JavaScript 函数 `opendialog()`，用于打开网页对话框，并将网页对话框返回的值赋给“所属类别”文本框，代码如下。

```
<script language="javascript">
function opendialog() {
var
typename=window.showModalDialog("optType.jsp","", "dialogWid
th=369px;dialogHeight=257px;status=no;help=
no;scrollbars=no")
document.form1.typename.value=typename;
}
</script>
```

(3) 制作显示类别名称列表的页面 `optType.jsp`，在该页面中应用循环将类别名称分栏显示，并将选择的类别名称返回到商品信息添加页面，关闭该网页对话框的超级链接。完整代码如下。

```

    <%@ page contentType="text/html; charset=gb2312"
language="java" import="java.sql.*" errorPage="" %>
    <jsp:useBean id="chStr" scope="page" class="com.ChStr"/>
    <jsp:useBean id="conn" scope="page" class="com.ConnDB"/>
    <%ResultSet rs=conn.executeQuery("select * from tb_type
order by typeName Desc");%>
    <html>
    <head>
    <metahttp-equiv="Content-Type" content="text/html;
charset=gb2312"/>
    <title>弹出网页模式对话框</title>
    <linkhref="css/style.css" rel="stylesheet"/>
    </head>
    <body>
    <tablewidth="363"height="73"border="0"align="center"cellp
adding="0"
cellspacing="0"background="images/son_bg.jpg">
    <tr><td>&nbsp;</td></tr></table>
    <tablewidth="363"height="35"border="0"align="center"cellp
adding="0"
cellspacing="0">
    <%try {
    while(rs.next()) {
    if(rs.getRow()%2==1) {
    %>
    <script language="javascript">
    function action(typename) {

```

```

        parent.window.returnValue=typename;
        window.close();
    }
</script>
<tr>
    <tdheight="25"align="center"valign="top"><a
href="#"onClick="action(' <%=rs.getString("TypeName")%>')"
><%=rs.getString("TypeName")%></a></td>
    <%} else {%>
        <tdheight="25"align="center"valign="top"><a
href="#"onClick="action(' <%=rs.getString("TypeName")%>
)'><%=rs.getString("TypeName")%></a></td>
    </tr>
<%
}
}
} catch(Exception e) {
    System.out.println(e.getMessage());
} %>
</table>
</body>
</html>

```

### 举一反三

根据本实例，读者可以：

在编写办公自动化系统时，为员工指定所属部门；

在编写校园管理系统时，弹出网页对话框选择班级或专业；

在客房预订系统中，弹出网页对话框选择客户信息。

## 实例372 关闭弹出窗口时刷新父窗口

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\16\372

实例说明

关闭弹出窗口时刷新父窗口也可以说是关闭子窗口时自动刷新父窗口中的信息，即用户通过window对象的open()方法打开一个新窗口（子窗口），当用户在该窗口中进行数据库操作（如数据添加、修改和删除等）后，关闭子窗口时，系统会自动刷新父窗口来实现更新信息。运行本实例，在新奥家电连锁后台管理系统的商品信息管理模块中，单击“添加商品信息”超级链接，将弹出添加商品信息页面，在该页面中添加商品信息（商品名称为“组合音响”）后，单击“保存”按钮，商品信息将被保存在数据表中，添加商品信息的窗口也将自动关闭，同时系统会自动刷新父窗口（也就是商品信息管理页面）。这时，新添加的商品名称“组合音响”将显示在页面中，如图16.5所示。

商品名称	所属类别	简介	是否新品	是否特价
组合音响	音响类	音质高	是	否
纯平彩电	电视类	高清晰度	是	否
等离子电视	电视类	外观精美!	否	是

本站请使用IE6.0或以上版本 1024\*768为最佳显示效果

CopyRight © 2007 吉林省明日科技有限公司

## 图16.5 关闭弹出窗口时刷新父窗口

### 技术要点

本实例主要应用window对象的open()方法打开新窗口，并在新窗口中应用opener属性，该属性返回一个引用，用于指定打开新窗口的窗口对象。

### 语法格式：

window.opener

window.opener.方法

window.opener.属性

功能：返回的是一个窗口对象。opener属性与打开该窗口的父窗口相联系。当访问子窗口中opener属性时，返回的是父窗口，通过该属性，可以使用父窗口对象中的方法和属性。

reload()方法相当于单击浏览器上的“刷新”（IE浏览器）或“Reload”（Netscape浏览器）按钮。

### 实现过程

(1) 制作用于显示商品信息列表的商品管理页面，在该页面中的文字“添加商品信息”上加入空的超级链接，并在其单击事件中加入JavaScript脚本，实现打开新窗口添加商品信息，关键代码如下。

```
<a href="#"  
onClick="window.open('goods_add.jsp','','width=560,height=360')">添加商品信息</a>
```

(2) 制作用于添加商品信息的页面。

(3) 制作将商品信息保存到商品信息表的页面。在该页面中需要根据保存操作的返回值执行不同的操作。当返回值不为0时，将弹出“商品信息添加成功！”的对话框并刷新父窗口，同时关闭当前窗口。关键代码如下。

```
<script language="javascript">
```

```
alert("商品信息添加成功!");  
window.opener.location.reload();//刷新父窗口  
window.close(); //关闭当前窗口  
</script>
```

举一反三

根据本实例，读者可以：

实现在办公自动化系统中，打开新的页面添加员工信息后，关闭该页面时刷新父窗口；

实现在网站后台管理中，添加信息、修改信息和删除信息时刷新父窗口。

## 16.2 无刷新技术

随着AJAX的日益盛行，无刷新技术的应用越来越受到广大程序员的青睐。本节将介绍如何采用AJAX技术在JSP中实现在不刷新页面的情况下检测用户名和级联下拉列表，及如何使用XML实现不刷新页面查询数据。

### 实例373 无刷新的用户名检测

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\16\373

实例说明

在用户注册或商品信息添加页面，经常需要在提交表单前检测注册的用户名或添加的商品名称是否唯一。传统的做法是采用window对象的open()方法或showModalDialog()方法打开新窗口检测，这样会影响网页的运行速度。但是采用AJAX后，就可以实现采用异步方式直接将参数提交到服务器，再通过 alert()方法将服务器返回的校验信息显示出来，大大提高了网页的运行速度。运行本实例，在“用户名”文本框中输入“wgh”，单击“[检测用户]”超级链接，即可在不刷新页面的情况下弹出“很抱歉!用户名[wgh]已经被注册!”的提示对话框，如图16.6所示。



图16.6 无刷新的用户名检测

### 技术要点

本实例主要通过JavaScript操作AJAX技术中的XMLHttpRequest对象实现。XMLHttpRequest对象是XMLHTTP组件的对象，通过该对象，AJAX可以像桌面应用程序一样只同服务器进行数据层面的交换，而不用每次都刷新页面，也不用每次都把数据处理的工作交给服务器来做，这样既减轻了服务器负担又加快了响应速度，缩短了用户等待的时间。下面将以列表的形式给出XMLHttpRequest对象的常用方法和属性。

XMLHttpRequest对象的常用方法如表16.1所示。

表16.1 XMLHttpRequest对象的常用方法

方 法	描 述
abort()	停止当前异步请求
getAllResponseHeaders()	以字符串形式返回完整的 HTTP 头信息
getResponseHeader("headerLabel")	以字符串形式返回指定的 HTTP 头信息
open("method", "URL"[, asyncFlag[, "userName"[, "password"]]])	设置进行异步请求目标的 URL、请求方法以及其他参数信息
send(content)	发送请求
setRequestHeader("label", "value")	设置 HTTP 头信息并和请求一起发送

XMLHttpRequest对象的常用属性如表16.2所示。

表16.2 XMLHttpRequest对象的常用属性

方 法	描 述
onreadystatechange	状态改变的事件触发器
readyState	对象状态 0: 未初始化 1: 读取中 2: 已读取 3: 交互中 4: 完成
responseText	返回服务器进程的文本信息
responseXML	服务器进程返回的兼容 DOM 的 XML 文档对象
status	返回服务器的状态码 200: “成功” 202: “请求被接受, 但尚未成功” 400: “错误的请求” 404: “文件未找到” 500: “内部服务器错误”
statusText	返回服务器的状态文本信息

### 实现过程

(1) 搭建AJAX开发框架, 具体代码如下。

```
<script language="javascript">
var http_request= false;
function createRequest(url) {
    //初始化对象并发出XMLHttpRequest请求
    http_request= false;
```

```

if (window.XMLHttpRequest) { //Mozilla.....
    http_request=newXMLHttpRequest();
    if (http_request.overrideMimeType) {
        http_request.overrideMimeType("text/xml");
    }
} else if (window.ActiveXObject) { //IE浏览器
    try{
        http_request=newActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try{
            http_request=newActiveXObject("Microsoft.XMLHTTP"
);
        } catch (e) {}
    }
}
if (!http_request) {
    alert("不能创建XMLHTTP实例!");
    return false;
}
http_request.onreadystatechange= alertContents; //指
定响应方法
//发出HTTP请求
http_request.open("GET", url, true);
http_request.send(null);
}
function alertContents() { //处理服务器返回的信息
    if (http_request.readyState==4) {

```

```

        if (http_request.status==200) {
            alert(http_request.responseText);
        } else {
            alert('您请求的页面发现错误');
        }
    }
}
</script>

```

(2) 编写JavaScript的自定义函数checkName(), 用于检测用户名是否为空。当用户名不为空时, 调用createRequest()函数发送请求检测用户名是否存在。代码如下。

```

<script language="javascript">
function checkName() {
    var username= form1.username.value;
    if(username=="") {
        window.alert("请添写用户名!");
        form1.username.focus();
        return false;
    }
    else {
        createRequest('checkName.jsp?username='+username);
    }
}
</script>

```

(3) 在页面的适当位置添加“[检测用户]”超级链接, 在该超级链接的onClick事件中调用checkName()方法, 弹出显示检测结果的对话框。关键代码如下。

```
<a href="#" onClick="checkName()">[检测用户]</a>
```

(4) 编写检测用户名是否唯一的JSP处理页checkName.jsp, 在该页面中应用JSP的内置对象out的println()方法输出检测结果, 完整代码如下。

```
<%@page contentType="text/html; charset=gb2312"
language="java" import="java.sql.*" errorPage=""%>
<jsp:useBean class="com.ConnDB" scope="page" id="conn">
</jsp:useBean>
<%
request.setCharacterEncoding("GB2312");
Stringusername=request.getParameter("username");
ResultSet rs=conn.executeQuery("select * from
tb_userwhereusername=' "+username+" '");
if (rs.next()) {
    out.println("很抱歉!用户名["+username+"]已经被注册!");
} else {
    out.println("祝贺您!用户名["+username+"]没有被注册!");
}
%>
```

举一反三

根据本实例, 读者可以:

实现无刷新检测商品信息是否添加;

实现无刷新判断用户密码是否正确。

## [实例374 无刷新的级联下拉列表](#)

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\16\374

## 实例说明

在制作动态网站时，经常会遇到通过选择的父类，自动获取子类信息并显示在下拉列表中。传统的实现方法是当用户改变父类信息时，重新请求当前页面获取子类信息并显示，这样会影响网页的运行速度。但是采用AJAX后，就可以实现采用异步方式直接将参数提交到服务器，再通过innerHTML属性将服务器返回的子类信息显示出来，大大提高了网页的运行速度。运行本实例，在“商品名称”文本框中输入“爱音乐\*\*\*MP3”，在“所属大类”下拉列表框中选择“IT数码”列表项，在“所属小类”下拉列表框中将显示属于“IT数码”的全部子类，如图16.7所示。



图16.7 无刷新的级联下拉列表

## 技术要点

本实例主要通过JavaScript操作AJAX技术中的XMLHttpRequest对象和单元格元素TD的 innerHTML 属性实现。XMLHttpRequest 对象的详细介绍请参见实例 373。下面对innerHTML属性进行介绍。

应用innerHTML属性可以直接给元素分配HTML字符串，而不需考虑使用DOM方法来创建元素。

说明：innerHTML 属性会自动将<、>、“”和&符号进行 HTML 编码，而不需要再进行特殊处理。

## 实现过程

(1) 搭建AJAX开发框架，具体代码如下。

```
<script language="javascript">
var http_request= false;
function createRequest(url) {
    //初始化对象并发出XMLHttpRequest请求
    http_request= false;
    if (window.XMLHttpRequest) { //Mozilla.....
        http_request=newXMLHttpRequest();
        if (http_request.overrideMimeType) {
            http_request.overrideMimeType("text/xml");
        }
    } else if (window.ActiveXObject) { //IE浏览器
        try{
            http_request=newActiveXObject("Msxml2.XMLHTTP");
        } catch (e) {
            try{
                http_request=newActiveXObject("Microsoft.XMLHTTP");
            }
        }
    }
}
```

```

        } catch (e) {}
    }
}
if (!http_request) {
    alert("不能创建XMLHTTP实例!");
    return false;
}
http_request.onreadystatechange= alertContents;    //指
定响应方法
//发出HTTP请求
http_request.open("GET", url, true);
http_request.send(null);
}
function alertContents() {    //处理服务器返回的信息
    if (http_request.readyState==4) {
        if (http_request.status==200) {
            subType.innerHTML=http_request.responseText;
        } else {
            alert('您请求的页面发现错误');
        }
    }
}
}
</script>

```

(2) 编写JavaScript的自定义函数F\_super(), 用于调用createRequest()函数发送请求获取子类信息, 代码如下。

```

<script language="javascript">
functionF_super(val) {

```

```
        createRequest("selSubType.jsp?superID="+val); //实现级  
        联下拉列表  
    }  
</script>
```

(3) 在页面的适当位置添加显示父类信息的下拉列表，在该下拉列表的 onChange 事件中调用F\_super()方法，获取子类信息并显示。关键代码如下。

```
<select name="supertype" class="textarea" id="supertype"  
onChange="F_super(this.value)">  
    <%rs_super.first();  
    do{  
        superID=rs_super.getInt(1);  
        superName=rs_super.getString(2);  
    %>  
    <option value="<%=superID%>"<%if (superID==P_superID)  
{out.println(" selected");/*注意此处的分号*/}%>><%= super  
    Name%></option>  
    <%}while(rs_super.next());%>  
</select>
```

(4) 将显示子类信息的单元格的id属性设置为“subType”，关键代码如下。

```
<td width="42%" height="27" id="subType">
```

(5) 编写获取子类信息的JSP处理页selSubType.jsp，在该页面中首先从数据表中获取子类信息，然后显示在下拉列表中，完整代码如下。

```
<%@page contentType="text/html; charset=gb2312"  
language="java" import="java.sql.*" errorPage=""%>
```

```

<%@page import="com.core.ConnDB"%>
<%
ConnDB conn=newConnDB();
intP_superID=-1;
int superID=-1;
String superName="";
ResultSet rs_sub=null;
//查询所属小类的默认选项
if(request.getParameter("superID")=="" ||
request.getParameter("superID")==null) {
    P_superID=superID;
} else {
    P_superID=Integer.parseInt(request.getParameter("superI
D"));
}
rs_sub=conn.executeQuery("select * from tb_subTypewhere
superType="+P_superID);
int subID=-1;
String subName="";
%>
<selectname="typeID" class="textarea" id="typeID">
<%while(rs_sub.next()) {
    subID=rs_sub.getInt(1);
    subName=rs_sub.getString(3);
%>
    <optionvalue="<%=subID%>"><%=subName%></option>
<%}%>

```

</select>

举一反三

根据本实例，读者可以：

实现无刷新的多级级联下拉列表；

实现无刷新级联菜单。

## 实例375 使用XML实现不刷新页面查询数据

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\16\375

实例说明

在实例373和实例374中已经介绍了如何通过AJAX实现无刷新技术。本实例将介绍如何通过XML实现无刷新查询数据。运行本实例，在搜索结果下拉列表框中将显示全部商品信息，在“请输入查询条件”文本框中输入文字“新新”，单击“搜索”按钮，在搜索结果下拉列表框中将只显示包含文字“新新”的商品信息，如图16.8所示。



图16.8 使用XML实现不刷新页面查询数据

技术要点

本实例的实现思路如下：首先从数据表中查询出需要显示的全部数据并保存到 XML 数据岛中，然后使用JavaScript脚本将输入的查询

条件与XML数据岛中的记录信息进行比较，并把符合条件的记录信息放入到下拉列表框中，从而实现不刷新页面筛选数据库中的数据。

### 实现过程

(1) 从数据表中查询全部数据并存放到数据岛中，关键代码如下。

```
<%@page import="com.ConnDB"%>
<%
ConnDB conn=newConnDB();
String sql="";
sql="select * from tb_goods1";
ResultSet rs_search=conn.executeQuery(sql);
int goodsID=-1;
StringgoodsName="";
String introduce="";
%>
<xml id="xmldao">
<goods>
<%
try{
while(rs_search.next()){
    goodsID=rs_search.getInt("ID");
    goodsName=rs_search.getString("goodsName");
    %>
<goodsID emid="<%=goodsID%>">
<goodsName><%=goodsName%></goodsName>
</goodsID>
<%}
}
```

```

    }catch(Exception e)
{System.out.println(e.getMessage());}%>
</goods>
</xml>
<%conn.close();%>

```

(2) 在页面中需要显示查询结果的位置添加一对<select></select>标记, 名称为“result”, 关键代码如下。

```

<select name="result" size="1" style="width:400pt">
</select>

```

(3) 编写JavaScript的自定义函数Mycheck(), 该函数用于将数据库中的全部数据显示到下拉列表框中, 具体代码如下。

```

<ScriptLanguage="JavaScript">
functionMycheck() {
    varEmid, Emname;
    firstElement=xmldao.childNodes.item(0);//用于返回第一个
元素
    for(i=0;i<firstElement.childNodes.length;i++) {
        prevailnode=firstElement.childNodes(i);
        Emid=firstElement.childNodes(i).getAttribute("emid");
        for(j=0;j<prevailnode.childNodes.length;j++) {
            DD=prevailnode.childNodes(j).nodeName;
            if(DD=="goodsName") {
                Emname=prevailnode.childNodes(j).text;
            }
        }
    }
    //将符合条件的信息写进select中
    if(Emid!=""&&Emname!="") {

```

```

        op=document.createElement("option");
        op.text=Emname;
        op.value=Emid;
        result.add(op);
    }
}
}
</script>

```

(4) 在<body>标记的onLoad事件中调用自定义函数Mycheck()初始化下拉列表框，关键代码如下。

```
<body onLoad="javascript:Mycheck()">
```

(5) 编写JavaScript的自定义函数Mycheck1(), 用于从数据岛中检索数据，并将其装入到下拉列表框中，代码如下。

```

<script language="javascript">
functionMycheck1() {
    varEmname,Emid;
    Emname="";
    Emid="";
    text0=form2.key.value;
    //清除列表框
    count0=result.length
    for(i=count0-1;i>=0;i--) {
        result.remove(i);
    }
    firstElement=xmldao.childNodes(0);
    for(i=0;i<firstElement.childNodes.length;i++) {
        prevailitem=firstElement.childNodes(i);
    }
}

```

```

Emid=firstElement.childNodes(i).getAttribute("emid");
for(j=0;j<prevailitem.childNodes.length;j++){
    if(prevailitem.childNodes(j).nodeName=="goodsName")
    {
        ep=prevailitem.childNodes(j).text;
    }
    //查找相匹配的项
    if(ep.indexOf(text0)!=-1){
        Emname=ep;
    }
    if(Emid!=""&&Emname!=""){
        op=document.createElement("option");
        op.value=Emid;
        op.text=Emname;
        result.add(op);
        Emid="";
        Emname="";
    }
}
}
}
}
</script>

```

(6) 在页面的适当位置添加查询条件表单，在该表单中包括一个名称为key的文本框和名称为search的“搜索”按钮两个表单元素。在“搜索”按钮的onClick事件中调用自定义函数Mycheck1()从数据岛中检索数据。关键代码如下。

```
<form name="form2"method="post"action="">
```

```
<tablewidth="100%"height="23" border="0"cellpadding="0"
cellspacing="0">
<tr>
<td colspan="2"height="7"></td>
</tr>
<tr>
<tdwidth="85%">请输入查询条件:
<inputname="key" type="text"class="txt_grey"
size="33"></td>
<tdwidth="15%"><inputname="search"
type="button"class="btn_grey"value="搜索"
onClick="javascript:Mycheck1()"></td>
</tr>
</table>
</form>
```

### 举一反三

根据本实例，读者可以：

实现无刷新的数据检索，并将检索结果显示在表格中；

实现无刷新的多级级联下拉列表。

## 16.3 调用与控制

在应用JSP开发动态网站时，经常需要对Word、Excel和PowerPoint进行操作，本节将通过具体实例介绍如何将查询结果导出到Word、如何调用Excel和PowerPoint。

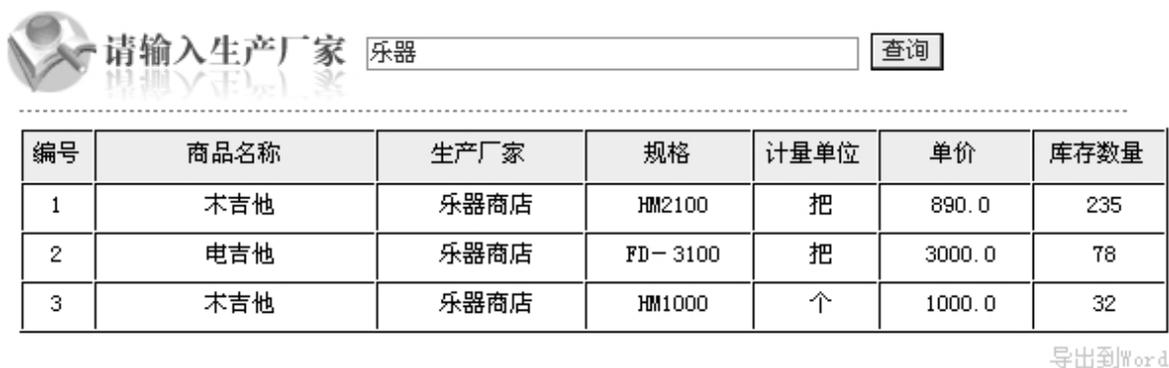
### 实例376 将查询结果导出到Word

本实例是一个非常实用的程序

实例位置：光盘\mingrisoft\16\376

实例说明

在开发Web应用程序时，有时需要将系统中的部分数据导出到Word中进行处理，这就需要在程序中加入查询功能，并将查询结果导出到Word中。运行本实例，在“请输入生产厂家”文本框中输入“乐器”，单击“查询”按钮，如图16.9所示，再单击“导出到Word”超级链接，即可将符合条件的数据导出到Word，如图16.10所示。



The screenshot shows a web interface with a search form and a table of results. The search form has a label '请输入生产厂家' (Please enter manufacturer), a text input field containing '乐器' (Musical instrument), and a '查询' (Search) button. Below the form is a table with 7 columns: 编号 (ID), 商品名称 (Product Name), 生产厂家 (Manufacturer), 规格 (Specification), 计量单位 (Unit), 单价 (Unit Price), and 库存数量 (Inventory Quantity). The table contains 3 rows of data.

编号	商品名称	生产厂家	规格	计量单位	单价	库存数量
1	木吉他	乐器商店	HM2100	把	890.0	235
2	电吉他	乐器商店	FD-3100	把	3000.0	78
3	木吉他	乐器商店	HM1000	个	1000.0	32

导出到Word

图16.9 将查询结果导出到Word



图16.10 导出到Word中的效果

### 技术要点

实现本实例，首先进行条件查询，并将查询结果显示在页面中，然后通过 JavaScript 将表格中的数据导出到Word中。

### 实现过程

(1) 连接数据库，并根据查询条件查询所需数据，关键代码如下。

```

<jsp:useBean id="conn"
scope="page" class="com.core.ConnDB"/>
<%
request.setCharacterEncoding("GBK");
String producer=request.getParameter("str");
String sql="";
String title="";
if(producer==null){
producer="";
title="全部库存信息";
sql="SELECT*FROM tb_storageORDERBYkcsldesc";

```

```

}else{
    title="生产厂家中包含[ "+producer+" ]的商品的库存信息";
    sql="SELECT*FROM tb_storagewhereproducer like
'%" +producer+"%' ORDERBYkcsldesc";
}
ResultSet rs=conn.executeQuery(sql);
%>

```

(2) 将查询结果显示在指定的表格中，并将该表格的id属性设置为storage。关键代码如下。

```

<table id="storage" width="98%" border="1"
cellpadding="0" cellspacing="0"
bgcolor="#000000" bordercolor="#000000"
bordercolordark="#000000"
bordercolorlight="#FFFFFF">

```

(3) 添加“打印”超级链接，应用打印指定框架的方法实现打印指定条件的库存报表。关键代码如下。

```

<script language="javascript">
function outDoc() {
    var table=document.all.storage;
    row=table.rows.length;
    column=table.rows(1).cells.length;
    var wdapp=newActiveXObject("Word.Application");
    wdapp.visible=true;
    wddoc=wdapp.Documents.Add(); //添加新的文档
    thearray=newArray();
//将页面中表格的内容存放在数组中
for(i=0;i<row;i++){

```

```

    thearray[i]=newArray();
    for(j=0;j<column;j++) {
        thearray[i][j]=table.rows(i).cells(j).innerHTML;
    }
}
var range=wddoc.Range(0,0);
range.Text="<%=title%>"+"\n";
wdapp.Application.ActiveDocument.Paragraphs.Add(range);
wdapp.Application.ActiveDocument.Paragraphs.Add();
rngcurrent=wdapp.Application.ActiveDocument.Paragraphs(3)
.Range;
var
objTable=wddoc.Tables.Add(rngcurrent,row,column)    //插入表格
for(i=0;i<row;i++) {
    for(j=0;j<column;j++) {
        objTable.Cell(i+1,j+1).Range.Text= thearray[i]
[j].replace("&nbsp;","");
    }
}
wdapp=null;
}
</script>

```

(4) 添加“导出到Word”超级链接，在该超级链接的onClick事件中调用outDoc()方法，将查询结果导出到Word。关键代码如下。

```
<a href="#" onClick="outDoc()">导出到Word</a>
```

举一反三

根据本实例，读者可以：  
将复杂条件查询的结果导出到Word：  
实现将导出到Word中的数据进行格式化。

## 实例377 调用Excel

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\16\377

实例说明

Excel 表格处理软件具有强大的表格处理功能，被许多人用于保存及处理数据。在开发Web应用程序时，为了方便用户操作，可以直接在页面中调用 Excel 工作簿，显示指定数据。运行本实例，单击图 16.11所示页面中的“查看更多销售排行”按钮，将打开图 16.12 所示的 Excel 工作簿，显示销售排行信息。



图16.11 将查询结果导出到Excel

技术要点

在JSP中实现打开Excel工作簿的思路如下。

(1) 应用JavaScript的ActiveXObject()构造函数创建一个Excel.Application对象的实例。

	A	B	C	D
1	图书销售排行榜			
2	排行	出版日期	ISBN	书名
3	1	200612	711515387	visual basic函数参考大全(附光盘)
4	2	200612	711515355	Visual Basic应用开发完全手册(附光盘程序开发从技术到实践)
5	3	200612	711515410	visual basic控件参考大全(附光盘)
6	4	200612	711515374	Delphi组件参考大全(附光盘软件工程师典藏)
7	5	200612	711515409	DELPHI函数参考大全/附光盘
8	6	200612	711515402	C#数据库系统开发完全手册/附光盘
9	7	200612	711515450	ASP.NET数据库系统开发案例精选/附光盘
10	8	200612	711515341	ASP数据库系统开发完全手册(附光盘)

图16.12 打开的Excel工作簿

(2) 打开指定的Excel文档，这里需要应用JSP的内置对象request的getRequestURL()方法获取Excel工作簿所在路径。

(3) 应用Excel实例的ActiveSheet属性激活工作表。

#### 实现过程

(1) 获取要打开的Excel工作簿所在的路径，关键代码如下。

```
<%
//获取要打开的Excel工作簿所在的路径
int at=request.getRequestURL().lastIndexOf("/");
String
path_part=request.getRequestURL().substring(0,at+1)+"booksort
.xls";
%>
```

(2) 编写自定义JavaScript函数outExcel(),用于打开指定的Excel工作簿，代码如下。

```
<script language="javascript">
function outExcel() {
var excelapp=newActiveXObject("Excel.Application");
```

```
    excelapp.visible=true;
    objBook=excelapp.Workbooks.Open("<%=path_part%>");//打
开工作簿
    varobjSheet=objBook.ActiveSheet;
    objSheet=null;
    excelapp=null;
}
</script>
```

(3) 通过单击“查看更多销售排行”按钮调用自定义JavaScript函数outExcel()。关键代码如下。

```
<input name="Submit" type="button" class="btn_grey"
onClick="outExcel()" value="查看更多销售排行">
```

举一反三

根据本实例，读者可以：

实现打开Excel工作簿后，对工作簿进行操作；

实现批量打开Excel工作簿。

## 实例378 调用PowerPoint

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\16\378

实例说明

PowerPoint是Microsoft的Office系列软件之一，应用PowerPoint可以制作电子文档演示幻灯片或课件。在开发网络应用程序时，有时会调用PowerPoint制作的幻灯片或课件等。运行本实例，单击“调用PowerPoint演示”按钮，即可打开指定的PowerPoint文档，该文件格式是.ppt的，结果如图16.13所示。

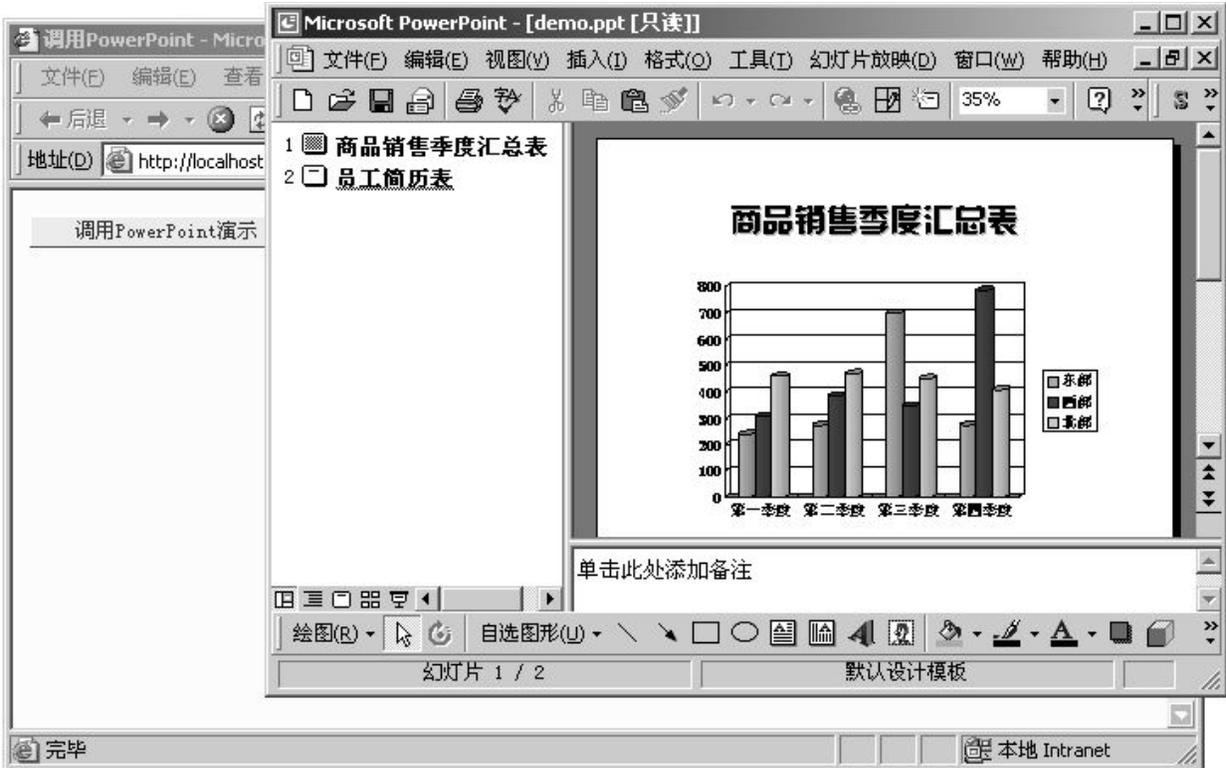


图16.13 调用PowerPoint

说明：在测试该实例时，需要保证客户端机器上安装有 PowerPoint。

### 技术要点

在JSP中实现打开PowerPoint文档的思路如下。

(1) 应用 JavaScript 的 ActiveXObject() 构造函数创建一个 PowerPoint.Application 对象的实例。

(2) 打开指定的PowerPoint文档，这里需要应用JSP的内置对象 request的getRequestURL() 方法获取PowerPoint文档所在路径。

### 实现过程

(1) 获取要打开的PowerPoint文档所在的路径，关键代码如下。

```
<%  
//获取PowerPoint文档所在的路径  
int at=request.getRequestURL().lastIndexOf("/");
```

```
String
path_part=request.getRequestURL().substring(0,at+1)+"demo.ppt
";
%>
```

(2) 在网页的适当位置添加“调用 PowerPoint 演示”按钮，用于打开指定的 PowerPoint 文档，代码如下。

```
<input name="Submit" type="button" class="btn_grey"
value="调用PowerPoint演示"
onClick="outPpt()">
```

(3) 编写打开指定的 PowerPoint 文档的 JavaScript 自定义函数 outPpt()，代码如下。

```
<script language="javascript">
function outPpt() {
varpptapp=newActiveXObject("PowerPoint.Application");
pptapp.visible=true;
pptapp.Presentations.Open("<%=path_part%>");//打开文件
objSheet=null;
excelapp=null;
}
```

<!--在运行该程序时，应该将浏览器中的安全设置——Internet选项的安全值调至最低-->

```
</script>
```

举一反三

根据本实例，读者可以：

批量打开PowerPoint文件；

自动打开指定文件夹下的PowerPoint文件。

## 16.4 其他

在JSP中有时需要实现树状导航菜单，本节将通过具体实例进行介绍。

### 实例379 树状导航菜单

本实例是一个美化界面、人性化的程序

实例位置：光盘\mingrisoft\16\379

实例说明

对于一个导航文字很多，并且可以对导航内容进行分类的网站来说，可以将页面中的导航文字以树视图的形式显示，这样不仅可以有效节约页面，而且也可以方便用户查看。运行本实例，在“商品分类”中以树状导航菜单显示商品的分类信息。在默认情况下，第一个节点是展开的，单击该节点名称前的“-”可以将该节点折叠；单击节点名称前的“+”可以展开指定节点，如图16.14所示。如果某个节点没有相关子节点，则直接显示“-”，同时该节点名称无超级链接。



图16.14 树状导航菜单

### 技术要点

本例主要通过JavaScript控制表格的行

说明：网页中的<table>、<tr>、<td>、<img>、<form>、<input>、<button>等 HTML 标记都具有display属性。

### 实现过程

(1) 利用 JavaScript 定义一个函数 ShowTR()，用于显示或隐藏表格中指定行的内容。ShowTR()函数的代码如下。

```
<script language="javascript">
ShowTR(img1,OpenRep1) //设置第一个节点为展开状态
functionShowTR(objImg,objTr){
    if(objTr.style.display==""){
        objTr.style.display="none";
        objImg.src="images/jia.gif";
        objImg.alt="展开";
    }
}
```

```

    }else{
        objTr.style.display= "";
        objImg.src= "images/jian.gif";
        objImg.alt= "折叠";
    }
}
</script>

```

注意：为了实现在默认情况下第一个节点为展开状态，可以在该函数前加入一句调用该函数的代码。

(2) 从视图V\_Type中查询商品的分类信息，代码如下。

```

<%@ page import="com.core.ConnDB" %>
<%
ConnDB conn=new ConnDB();
ResultSet rs_superType=conn.executeQuery("select
ID,superType from V_Type group by ID,superType order by
superType desc");
%>

```

(3) 循环显示商品分类信息，并查询各分类中所包括的子类名称。如果该分类有相对应的子类，则在节点前显示“+”，否则显示“-”。值得注意的是，需要为每个包含子节点的节点设置超级链接，该超级链接执行的操作是调用自定义的 JavaScript 函数，实现节点的展开与折叠，代码如下。

```

<tablewidth="100%"height="40" border="0"cellpadding="0"
cellspacing="0">
<%
int ID_superType=0;
String superType="";

```



```

    <%if (subType_RC>0) {%>
<tr id="OpenRep<%=m%>" style="display:none;">
    <td colspan="6"><%do{
        subType=rs_subType.getString(3);
        subID=rs_subType.getInt(4);
        %>
        <tablewidth="100%" border="0" cellspacing="-2"
cellpadding="-2">
            <tr>
                <tdheight="25" align="center"><tablewidth="90%"
border="0" cellspacing="0"
cellpadding="0">
                    <tr onMouseOver="this.style.background=' #EEEEEE' "
onMouseOut="this.style.background=' ' ">
                        <tdwidth="15%">&nbsp;</td>
                        <tdwidth="10%" align="center"></td>
                        <tdwidth="75%"><a href="type.jsp?type=
<%=subID%>"><%=subType%></a></td>
                    </tr>
                </table></td>
            </tr>
        </table></td>
    </tr>
</table>
<% m++; //注意，该条语句一定不能少
}while(rs_subType.next());%></td>
<%}%>

```

</tr>

<%}%>

</table>

举一反三

根据本实例，读者可以：

开发在线论坛中的论坛管理页面中的应用目录树结构，显示主题及回复信息；

开发网站后台的导航菜单；

实现办公自动化系统的企业内部邮件的发送模块，并根据部门选择收件人功能。

# 第17章 JavaScript与 ASP.NET结合

窗口与对话框

其他

## 17.1 窗口与对话框

在网站应用程序中，用户在进行相关操作时，系统应及时地给用户以提示或打开新的窗口，以帮助用户正确地操作和使用网站。本节将介绍如何通过JavaScript与ASP.NET结合控制窗口与对话框。

### 实例380 使用JavaScript脚本弹出对话框

本实例可以美化界面、简化操作

实例位置：光盘\mingrisoft\17\380

实例说明

在信息输入窗口中，可以通过对话框的形式，提醒用户输入信息出错的位置及其原因。例如，在本实例中，当用户输入非法的会员名和密码时，将会弹出对话框“您的输入有误，请核对后，重新输入！”提醒用户，示例运行结果如图17.1所示。



图17.1 使用JavaScript脚本弹出对话框

## 技术要点

Response.Write方法主要是用来在页面上输出信息。在本实例中，通过使用Response.Write方法结合JavaScript脚本语言，弹出提示对话框。其语法格式如下。

```
Response.Write("<script>alert('在此处输入你要提示的信息')</script>");
```

## 实现过程

(1) 新建一个网站，将其命名为Ex01\_01，默认主页为Default.aspx。

(2) 在Default.aspx页添加两个TextBox控件，分别用于输入会员名和密码。然后，再添加两个Button控件，分别用于执行会员登录和取消功能。

(3) 在“登录”按钮的Click事件下，实现判断用户输入的信息是否合法。如果输入的信息不合法，将会弹出对话框，提醒用户正确输入信息。其代码如下。

```
protectedvoidButton1_Click(object sender,EventArgs e)
{
    if (this.TextBox1.Text != ""&& this.TextBox2.Text !=
    "")
    {
        if (this.TextBox1.Text.Trim() == "mr"&&
        this.TextBox2.Text.Trim() == "mrsoft")
        {
            Response.Redirect("Default2.aspx");
        }
        else
        {
```

```
        Response.Write("<script>alert('您的输入有误,请核  
对后,重新输入!')</script>");  
    }  
}  
else  
{  
    Response.Write("<script>alert('请输入必要的信息!')  
</script>");  
}  
}  
}
```

举一反三

根据本实例,读者可以:

实现在开发商品信息管理模块时,使用弹出窗口,提醒用户正确输入商品信息;

实现在开发论坛时,使用弹出窗口,提醒用户输入正确的注册信息。

## [实例381 打开新窗口显示详细信息](#)

本实例是一个提高效率、人性化的程序

实例位置: 光盘\mingrisoft\17\381

实例说明

在网页设计时,通常需要考虑页面的美观程度及其浏览速度,这时可以将要显示数据的概要信息显示在主页面中,并提供连接到显示详细信息的细节页面的超级链接。这样不但可以提高页面的浏览速度,而且可以简化界面。运行本实例,在显示商品信息表格中,只显

示商品号和商品名，双击表格中的单行数据即可打开显示商品详细信息的窗口，在该窗口中，用户可以查看商品的详细信息，并可以通过单击“关闭”按钮，关闭该窗口，如图17.2所示。



图17.2 打开新窗口显示详细信息

### 技术要点

本实例主要应用 window对象的 open() 方法打开新窗口，同时在打开新窗口时，将商品 id信息传递到新窗口中。本实例中的open() 方法的语法格式如下。

window.open(“链接文件名?参数名称=参数值”, “”, “width=窗口的宽,height=窗口的高”)

注意：在进行传递参数时，多个参数之间用“&”分隔。

### 实现过程

(1) 新建一个网站，将其命名为Ex01\_03，默认主页为Default.aspx。

(2) 在Default.aspx页面中添加一个GridView控件，显示数据信息。

(3) 主要程序代码。

为了实现当用户双击GridView控件中的某行数据信息时，打开一个新页面，同时显示该行数据的详细信息，需要在GridView控件的RowDataBound事件下编写如下代码。

```
protectedvoidGridView1_RowDataBound(object sender,GridViewRowEventArgs e)
{
    if (e.Row.RowType==DataControlRowType.DataRow)
    {
        e.Row.Attributes.Add("onclick",
        "window.open('Default2.aspx?
        id="+e.Row.Cells[0].Text+"','', 'width=300,height=230')")
        );//双击行打开新页
    }
}
```

在解决方案中添加一个新页，默认名为Default2.aspx。在该页添加4个TextBox控件，分别用于显示商品名、商品单位、商品价格和商品描述。然后，再添加一个Button按钮控件，用于执行关闭操作。页面加载时显示商品信息，其代码如下。

```
protectedvoidPage_Load(object sender,EventArgs e)
{
    if (!IsPostBack)
    {
        XmlDocumentdoc=newXmlDocument();
        doc.Load(Server.MapPath("~/GoodsInfo.xml"));
        XmlNodeListnodes;
        XmlElement root=doc.DocumentElement;
```

```

nodes=
root.SelectNodes("descendant::Item[GoodsID='"+Request["
id"]+"' ]");
foreach (XmlNode node in nodes )
{
    this.TextBox1.Text=node.ChildNodes[2].InnerText;
    this.TextBox2.Text=node.ChildNodes[3].InnerText;
    this.TextBox3.Text=node.ChildNodes[4].InnerText;
    this.TextBox4.Text=node.ChildNodes[5].InnerText;
}
}
}

```

注意：编写代码前，要引用命名空间“using System.Xml”。

在“关闭”按钮的Click事件下，实现用户单击该按钮关闭“显示商品详细信息窗口”。其代码如下。

```

protected void Button1_Click(object sender, EventArgs e)
{
    Response.Write("<script>>window.close();</script>");
}

```

举一反三

根据本实例，读者可以：

实现新闻系统中的即时新闻、热点新闻等；

实现电子商务系统中的商品展示；

实现在线论坛中的主题信息浏览。

## [实例382 关闭弹出窗口刷新父窗口](#)

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\17\382

实例说明

关闭弹出窗口刷新父窗口也可说是关闭子窗口时自动刷新父窗口中的信息，即用户通过window对象的open()方法打开一个新窗口（子窗口），当用户在该窗口中进行了数据库操作（如数据添加、修改和删除等）后，关闭子窗口，此时系统会自动刷新父窗口来实现更新信息。

运行本实例，如图17.3所示，单击商品列表页面

(Default.aspx)中的“添加数据”超级链接按钮，程序弹出一个新窗口，如图17.4所示，在该窗口中添加了商品信息后，单击“添加”按钮，添置的商品信息将被保存在XML文件中，同时该窗口会自动关闭，并刷新父窗口（商品列表页面），将刚添加的信息显示在页面表格中，如图17.5所示。



图17.3 商品信息显示窗口（添加前）



图17.4 添加商品信息窗口



图17.5 商品信息显示窗口（添加后）

### 技术要点

本实例主要应用window对象的open()方法打开新窗口，并在新窗口中应用opener属性，该属性返回一个引用，用于指定打开本窗口的窗口对象。

语法格式：

window.opener

window.opener.方法

window.opener. 属性

功能：返回的是一个窗口对象。opener属性与打开该窗口的父窗口相联系，当访问子窗口中opener属性时，返回的是父窗口，通过该属性，可以使用父窗口对象中的方法和属性。

本实例中应用的关键语句如下所示。

```
Response.Write("<script language=javascript>alert(' 添加成功! ');opener.location.reload();window.close();</script>");
```

reload()方法相当于单击浏览器上的“刷新”(IE)或“Reload”(Netscape)按钮。

实现过程

(1) 新建一个网站，将其命名为Ex01\_04，默认主页为Default.aspx。

(2) 在该页添加一个GridView控件，用于显示数据。同时，在该页的HTML源码中添加一个超级链接，在该超级链接的onClick事件中添加打开新窗口的代码，具体代码如下。

```
<a href="#"
onClick="window.open(' Default2.aspx','', 'width=300,height=250
');"><strong>
```

```
<span style="font-size: 9pt;text-decoration:none" >添加
数据</span></strong></a>
```

母版页中的占位符

(3) 在解决方案中添加一个新页，默认名为 Default2.aspx，用于添加数据并刷新父窗口，代码如下。

```
protectedvoidButton1_Click(object sender,EventArgs e)
{
    XmlDocumentdoc1=newXmlDocument();
    doc1.Load(Server.MapPath("~/GoodsInfo.xml"));
```

```

XmlNode newNode1;
XmlNode newNode2;
newNode1=doc1.CreateElement("Item");
newNode2=doc1.CreateElement("GoodsClass");
newNode2.InnerText=TextBox1.Text;
newNode1.AppendChild(newNode2);
newNode2=doc1.CreateElement("GoodsName");
newNode2.InnerText=TextBox2.Text;
newNode1.AppendChild(newNode2);
newNode2=doc1.CreateElement("Unit");
newNode2.InnerText=TextBox3.Text;
newNode1.AppendChild(newNode2);
newNode2=doc1.CreateElement("Price");
newNode2.InnerText=TextBox4.Text;
newNode1.AppendChild(newNode2);
newNode2=doc1.CreateElement("Description");
newNode2.InnerText=TextBox5.Text;
doc1.DocumentElement.AppendChild(newNode1);
doc1.Save(Server.MapPath("~/GoodsInfo.xml"));
Response.Write("<script language=javascript>alert('添加成功!');opener.location.reload();window.close();</script>");
添加成功后，关闭窗口并刷新父窗口
}

```

注意：添加数据信息页面（Default2.aspx）的JavaScript介绍是在后台cs代码中实现的。编写代码前，要引用命名空间“using System.Xml”。

举一反三

根据本实例，读者可以：

实现在办公自动化系统中，打开新的页面添加员工信息页面后，关闭该页面时刷新父窗口；

实现在电子商务网站中的后台管理中进行添加商品信息、修改商品信息和删除商品信息时，都需要刷新父窗口。

## 实例383 弹出网页模式对话框

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\17\383

实例说明

在制作管理系统时，通常需要录入一些数据信息。为了方便用户快速录入数据信息，可以将一些数据量比较大、使用频繁的数据在对话框中以分栏的形式显示出来，用户只需在对话框中选择需要的信息即可，这样可以提高录入数据信息的速度，而且也可以降低录入数据信息的出错率。

运行本实例，在图17.6所示窗口上，单击“类别名”文本框，即可弹出显示商品类别的网页对话框，如图17.7所示。这时用户单击选中的商品类别名（如电脑软件）后，该网页对话框将自动关闭，同时，选择的商品类别名被添加到“类别名”文本框中，如图17.8所示。



图17.6 录入商品信息（添加商品类别名前）



图17.7 弹出网页模式对话框



图17.8 录入商品信息（添加商品类别名后）

#### 技术要点

本实例主要应用window对象的showModalDialog()方法，该方法用于弹出网页（模式）对话框，其语法格式如下。

```
variant = object.showModalDialog(sURL [, vArguments [, sFeatures]])
```

参数说明如下。

- sURL：指定URL 文件地址。
- vArguments：用于向网页对话框传递参数，传递参数的类型不限制，对于字符串类型，最大为4096个字符，但也可以传递对象，例如index.htm。
- sFeatures：可选项，窗口对话框的设置参数，主要参数如下。

dialogWidth:number，可选，用于设置对话框的宽度。

dialogHeight:number，可选，用于设置对话框的高度。

dialogTop:number, 可选, 用于设置对话框相对于桌面左上角的top位置。

dialogLeft:number, 可选, 用于设置对话框相对于桌面左上角的left位置。

center:{yes | no | 1 | 0 }, 可选, 用于指定是否将对话框在桌面上居中, yes|1 为居中显示; no|0为不居中显示, 默认值是yes。

Help: {yes|no 1|0}, 可选, 用于指定对话框中是否显示上下文敏感的帮助图标, 默认值是yes。

scroll: {yes|no 1|0}, 可选, 用于指定对话框是否出现滚动栏。

resizable: {yes|no |1|0 }, 可选, 用于指定对话框大小是否可变, 默认值是no。

status: {yes|no 1|0}, 可选, 用于指定对话框是否出现状态栏。

### 实现过程

(1) 新建一个网站, 将其命名为Ex01\_05, 默认主页为Default.aspx。

(2) 向Default.aspx页添加4个TextBox控件、1个Label控件和1个Button按钮控件, 分别进行用户录入商品信息、显示商品信息和执行显示操作, 在“类别名”文本框的HTML源码中设置其onClick事件, 并调用自定义JavaScript函数opendialog(), 其代码如下所示。

```
<asp:TextBox ID="TextBox1" runat="server"
onClick="opendialog()" Width="148px">单击这里, 选择类别名
</asp:TextBox>
```

(3) 编写自定义JavaScript方法opendialog(), 用于打开网页对话框并将网页对话框返回的值赋给“类别名”文本框。其代码如下。

```
<script language="javascript">
```

```

function opendialog()
{
    varsomeValue=window.showModalDialog("Default2.aspx","",
    "dialogWidth=300px;dialogHeight=250px;status=no;help=no;scr
ollbars=no")
    document.form1.TextBox1.value=someValue;
}
</script>

```

(4) 在解决方案中添加1个新页，默认名为Default2.aspx，在该页上添加1个DataList控件，用于显示用户名列表，后台数据绑定，其代码如下。

```

using System.Data.SqlClient;
public partial class Default2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            DataSet ds = new DataSet();
            ds.ReadXml(Server.MapPath("~/goodsClass.xml"));
            this.DataList1.DataSource = ds;
            this.DataList1.DataBind();
        }
    }
}

```

(5) 在Default2.aspx页的HTML源码中编写代码，用于将选择的商品类别返回到商品添加页 (Default1.aspx)，并关闭该网页对话框

的超级链接，完整代码如下。

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="Default2.aspx.cs" Inherits="Default2" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<htmlxmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">
<title>弹出网页模式对话框</title>
<script language="javascript">
function a(goodsClass)
{
parent.window.returnValue=goodsClass;
window.close();
}
</script>
定义函数a()，用于实现将选择的商品类别返回到父窗口中
</head>
<body>
<form id="form1" runat="server">
<div style="text-align: center">
<table>
<tr>
<td style="width:249px;height:35px;">
<spanstyle="font-size:16pt;font-family:楷体
_GB2312"><strongstyle="color:red">选择商品类别
</strong></span></td>
```

```

    </tr>
    <tr>
        <td style="width:249px">
            <asp:DataList ID="DataList1" runat="server" Font-
            Size="9pt" RepeatColumns="2" RepeatDirection="Horizontal"
            Width="232px">
                <ItemTemplate>
                    <table style="width:104px;border="2">
                        <tr>
                            <td
                                style="width:100px;height:17px;"align="center">
                                    <ahref="#" style="text-decoration:none;"
                                        onclick="a ('<#Eval("goodsClass")%>')">
                                        <#Eval("goodsClass")%></a>

```

调用函数a()，用于实现将选择的商品类别返回到父窗口中

```

                                </td>
                            </tr>
                        </table>
                    </ItemTemplate>
                </asp:DataList></td>
            </tr>
        </table>
        <br/>
        &nbsp;</div>
    </form>
</body>
</html>

```

举一反三

根据本实例，读者可以：

实现在编写人力资源管理系统时，为员工指定所属部门；

实现在编写校园管理系统时，弹出网页对话框选择班级或专业；

实现在客房预订系统中，弹出网页对话框选择客户信息。

## 实例384 弹出全屏显示网页

这是一个美化网站的实例

实例位置：光盘\mingrisoft\17\384

实例说明

在设计网页时，对于一些数据信息量比较大的网页，可以将其以全屏显示的网页（模式）对话框展示给用户，这样可以保证用户看后能及时将其关闭。运行本实例，单击页面中的“全屏显示”超级链接按钮，这时的网站首页是以全屏网页对话框模式进行显示的，如图17.9所示，用户只有关闭该页面后，才可以查看网站的其他内容。



图17.9 弹出全屏显示网页

### 技术要点

本实例主要通过window对象的open()方法打开新窗口并创建新的window对象。其语法格式如下。

```
window.open(url, name, [options])
```

通过应用 screen 对象的 width 和 height 属性，使打开的对话框全屏显示。screen 对象是window对象的属性，提供给用户屏幕尺寸和颜色深度信息。Navigator 和Internet Explorer 支持的screen属性包括以下几种。

- Height: 是用户屏幕当前高度（单位为：像素数）。
- Width: 是用户屏幕当前宽度（单位为：像素数）。
- ColorDepth: 是用户屏幕/显示卡当前支持的每种颜色位数。

例如，8 位颜色深度对应于256色。

### 实现过程

(1) 新建一个网站，将其命名为Ex01\_06，默认主页为Default.aspx。

(2) 在Default.aspx页的HTML源码中添加空的超级链接，在该超级链接的onClick事件中调用自定义JavaScript函数winopen()。关键代码如下。

```
<a href="#" onClick="winopen()"><span style="font-size: 9pt; text-decoration: none; color: white;"><strong>&nbsp;  </strong>全屏显示</span></a>
```

(3) 在主页 HTML 源码<head runat="server">与<head>之间编写自定义 JavaScript 函数winopen()，用于打开全屏显示的网页对话框，代码如下。

```
<script>
<!--
functionwinopen() {
var targeturl="default.aspx"
newwin=window.open("", "", "scrollbars")
if (document.all) {
newwin.moveTo(0,0)
newwin.resizeTo(screen.width, screen.height)
}
newwin.location=targeturl
```

```
}  
-->  
</script>
```

举一反三

根据本实例，读者可以：

实现弹出网站后台管理员登录窗口；

实现当用户执行非法操作后，弹出一个警告框。

## 17.2 其他

### 实例385 自动隐藏式菜单

本实例可以美化界面、简化操作

实例位置：光盘\mingrisoft\17\385

实例说明

在一些个性化的网站中，经常会看到自动隐藏式的菜单。该类菜单不仅使页面美观别致，而且节省页面空间。当鼠标移到菜单标签上时，该菜单将自动展开；当鼠标离开菜单标签时，该菜单将自动隐藏。本实例制作的自动隐藏式菜单运行效果如图17.10、图17.11所示。



图17.10 菜单自动隐藏



图17.11 菜单展开

### 技术要点

本实例主要应用window对象的setTimeout()方法和clearTimeout()方法实现。

(1) setTimeout()方法：设置在指定的若干毫秒后要调用一次的函数。

语法：

setTimeout (code, delay)

说明：在延迟delay参数指定的毫秒数后执行code字符串指定的代码。在Netscape 4 和IE 5及以后的版本中，code参数可以用函数引用来代替字符串，此时，延迟delay毫秒后执行函数。该方法返回一个内部id值，该值可以传递给clearTimeout()方法来取消延迟的代码执行。

注意：该方法立即返回，并在返回后等待delay指定的毫秒数。

本实例应用该方法的代码如下。

```
c = setTimeout("MenuHide()", 1)
```

(2) clearTimeout()方法：该方法用于取消在指定的若干毫秒后要调用一次的函数。

本实例应用该方法的代码如下。

clearTimeout(d)

实现过程

(1) 新建一个网站，将其命名为Ex01\_02，默认主页为Default.aspx。

(2) 将代码置于主页 HTML 源码中的<head>与</head>标记之间，实现菜单的自动展开和隐藏功能，其代码如下。

```
<style>#dviID {
    BACKGROUND-COLOR:#66cdaa;
    BORDER-BOTTOM:white2px outset;
    BORDER-LEFT:white2pxoutset;
    BORDER-RIGHT:white2px outset;
    BORDER-TOP:white2px outset;
    LEFT:0px;POSITION: absolute;
    TOP:0px;VISIBILITY:hidden;
    WIDTH:200px;
    layer-background-color:#e6e6fa
}
a{font-size:9pt}
a:link{text-decoration:none}
a:hover{text-decoration:none;color:#FFFFFF}
a:visited{text-decoration:none}
.40pt{font-size:40pt;color:#e6e6fa;font-family:文鼎琥珀
繁}
</style>
<script language="javascript">
a=null
b=null
```

```
c=null
d=null
Netscape4= (document.layers)?1:0
IE4= (document.all)?1:0;
functionMenuHide()//菜单隐藏
{
    if(Netscape4)
    {
        clearTimeout(d)
        if(menu.left>menuW*-1+20+10 )
        {
            menu.left-=10
            c= setTimeout("MenuHide()", 1)
        }
        else if(menu.left>menuW*-1+20 )
        {
            menu.left--
            c= setTimeout("MenuHide()", 1)
        }
    }
}
else
{
    clearTimeout(d)
    if(menu.pixelLeft>menuW*-1+20+10 )
    {
        menu.pixelLeft-= 10
        c= setTimeout("MenuHide()", 1)
    }
}
```

```

    }
    else if(menu.pixelLeft>menuW*-1+20 )
    {
        menu.pixelLeft--
        c= setTimeout("MenuHide()", 1)
    }
}
}
functionMenuOut ()//菜单显示
{
    if(Netscape4)
    {
        clearTimeout(c)
        if(menu.left<-10)
        {
            menu.left+=4
            d= setTimeout("MenuOut()", 1)
        }
        else if(menu.left<0)
        {
            menu.left++
            d= setTimeout("MenuOut()", 1)
        }
    }
}
else
{
    clearTimeout(c)

```

```

    if(menu.pixelLeft<-10)
    {
        menu.pixelLeft+=2
        d= setTimeout("MenuOut()", 1)
    }
    else if(menu.pixelLeft< 0 )
    {
        menu.pixelLeft++
        d= setTimeout("MenuOut()", 1)
    }
}
}
function fireOver()
{
    clearTimeout(b)
    a= setTimeout("MenuOut()", 10)
}
function fireOut()
{
    clearTimeout(a)
    b= setTimeout("MenuHide()", 10)
}
function Init()
{
    if(Netscape4)
    {
        menu=document.D1
    }
}

```

```

    menuW=menu. document. Width
    menu. left=menu. document. Width*-1+20
    menu. onmouseover=menuOut
    menu. onmouseout=menuIn
    menu. visibility= "visible"
}
else if(IE4)
{
    menu=dviID. style
    menuW=dviID. offsetWidth
    dviID. style. pixelLeft=dviID. offsetWidth*-1+20
    dviID. onmouseover= fireOver
    dviID. onmouseout = fireOut
    dviID. style. visibility = "visible"
}
}
</script>

```

(3) 设置<body>标记的 onload 事件, 调用函数 Init(), 实现菜单的自动展开和隐藏功能, 同时将代码置于<body>标记和</body>标记之间, 为菜单栏添加显示的内容。其代码如下。

```

<bodyonload="Init()">
<div id="dviID" style="left:0px; top:0px">
<tableborder="0"width="150"height="500">
<TBODY>
<tr>
<td valign="middle" bgColor="#2e8b57" rowspan="2"
style="width:13px"><font color="#00ff99">

```

菜<br>

单

栏</font></td>

<td colspan="2" valign="middle" rowspan="2"><ul>

<br/><br/>

<li><a href="javascript:void(0)">.NET范例宝典</a>

</li>

<li><a href="javascript:void(0)">C#技术大全</a></li>

<li><a href="javascript:void(0)">C#案例精选</a></li>

<li><a href="javascript:void(0)">C#范例宝典</a></li>

<li><a href="javascript:void(0)">C#完全手册</a></li>

<li><a href="javascript:void(0)">.NET2.0图书</a>

</li>

</ul>

</td>

</tr>

<tr>

</tr>

</tbody>

</table>

</div>

</body>

举一反三

根据本实例，读者可以：

实现图片资源网站导航栏；

实现任意网站页面导航模块。

# 第18章 JavaScript与PHP结 合

窗口与对话框

其他

## 18.1 窗口与对话框

在网站应用程序中，用户在进行相关操作时，系统应及时地给用户以提示或打开新的窗口，从而帮助用户正确地操作和使用网站。本节将介绍如何通过JavaScript与PHP结合控制窗口与对话框。

### 实例386 弹出提示对话框并重定向网页

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\18\386

实例说明

在设计网页时，通常需要考虑页面的美观程度及其下载速度，这时可以将要显示的数据的概要信息显示在主页面中，并提供连接到显示详细信息的细节页面的超级链接。运行本实例，在博客网的浏览博客文章页面中，显示了博客的文章主题信息，单击博客文章主题即可打开显示博客文章详细信息的新窗口，在该窗口中可以查看该博客文章的详细信息，如图18.1所示。



图18.1 打开新窗口显示详细信息

### 技术要点

本实例主要应用 window 对象的 open() 方法打开新窗口。在打开新窗口时, 将博客文章 id 信息传递到新窗口中。

open() 方法的语法格式如下。

window.open("链接文件名?参数名称=参数值", "", "width=窗口的宽,height=窗口的高")

技巧: 在进行传递参数超级链接时, 多个参数之间用 "&" 分隔。

### 实现过程

(1) 建立数据源连接, 从数据表中检索博客文章的主题信息。

```
<?php include "Conn/conn.php"; ?>
```

```
<?php
```

```
$sql=mysql_query("select id,title from tb_articleorderby iddesc");
```

```
$info=mysql_fetch_array($sql);
```

```

do{
?>
<tr>
    <tdwidth="504"align="left"valign="top">&nbsp;• &nbsp;
    ..... //此处省略了博客文章主题信息的链接，详见步骤
(2)
</td>
</tr>
    <?php
        }while($info=mysql_fetch_array($sql));
?>

```

(2) 开发用于显示博客文章主题信息的主页面。在该页面中加入空的超级链接，并在其单击事件中加入JavaScript脚本，实现打开新窗口显示博客文章的详细信息。这里需要注意的是：在打开新窗口时，需要将博客文章id作为参数进行传递。关键代码如下。

```

<a href="#" onClick="JScript:window.open('article.php?
file_id=<?php echo $info[id];?>',',', 'width=600,height=260')">
<?php echo $info[title];?></a>

```

(3) 开发用于显示博客文章详细信息的页面。在该页面中通过GET方法来获取主页面中传递的博客文章 id 号，并根据博客文章 id 号创建记录集，输出显示博客文章的详细信息，关键代码如下。

```

<?php include "Conn/conn.php";$file_id1=$_GET[file_id]; ?
>
<?php
    $sql=mysql_query("select * from tb_articlewhere id=
". $file_id1);
    $result=mysql_fetch_array($sql);

```



## 实例387 关闭弹出窗口时自动刷新父窗口

这是一个PHP与JavaScript结合的实例

实例位置：光盘\mingrisoft\18\387

实例说明

在网站后台制作过程中，经常需要在弹出窗口中添加商品信息，但添加完商品后，如果不对父窗口进行刷新是看不到新商品信息的，制作本实例的目的是为了解决上述问题。运行本实例，首先显示图 18.2所示的商品大类信息窗口，单击“添加”按钮将弹出图 18.3所示的商品大类添加窗口，在该窗口中输入商品大类名称，单击“添加”按钮后，新类别被添加到数据库中，同时关闭该弹出窗口并自动刷新父窗口，新类别显示在图 18.2 所示的父窗口中。

商品大类管理	
商品类别	类别添加时间
家用电器	2007-02-13 01:25:47
家居日用	2007-02-13 01:25:32
服装类	2007-02-13 01:24:57

图18.2 父窗口



图18.3 弹出窗口

技术要点

本实例主要应用window对象的open()方法弹出一个新窗口，open()方法的语法格式如下。

```
open("url", "name", "features")
```

参数说明如下。

● url: 指弹出新窗口的 URL 地址, 如果省略这个参数或该参数的值是一个空字符串, 那么新窗口就不显示任何文档。

● name: 新窗口的名字, 如果这个名字是一个已经存在的窗口, 那么open()方法就不再创建一个新窗口, 而只是返回这个窗口的引用, 在这种情况下参数features将被忽略。

● features: 该参数主要是显示窗口的一些特征, 该参数如果不设定, 则显示这个窗口的所有特征。features的常用特征如表18.1所示。

表18.1 参数features的常用特征

名 称	用 法
channelmode	指定窗口是否应该以频道的方式显示, 取值有(yes/no 或 1/0)
fullscreen	指定窗口是否全屏显示, 取值有(yes/no 或 1/0)
menubar	指定窗口是否有菜单栏, 取值有(yes/no 或 1/0)
scrollbars	指定窗口是否有水平及垂直滚动条, 取值有(yes/no 或 1/0)
status	指定窗口是否有状态栏, 取值有(yes/no 或 1/0)
toolbar	指定窗口是否有工具栏, 取值有(yes/no 或 1/0)
location	指定窗口是否有地址栏, 取值有(yes/no 或 1/0)
directions	指定新建窗口是否有标准目录按钮, 取值有(yes/no 或 1/0)
resizable	指定新窗口大小是否可以调整, 取值有(yes/no 或 1/0)
top	以像素为单位指定新窗口上沿与屏幕上沿的距离
left	以像素为单位指定新窗口左沿与屏幕左沿的距离
width	以像素为单位指定新窗口的宽度
height	以像素为单位指定新窗口的高度

父窗口刷新主要通过window的opener属性实现, 该属性的语法格式如下。

window.opener.方法

window.opener.属性

功能: 通过该属性可以调用父窗口的属性和方法来操纵父窗口。

实现过程

(1) 制作父窗口，并为该窗口的“添加”按钮添加如下代码，调用自定义的 open\_add() 函数实现子窗口弹出功能。

```
<input type="button" value="添加" class="buttoncss"
onclick="javascript:open_add()">
```

(2) 编写 open\_add() 方法实现子窗口的弹出，代码如下。

```
<script language="javascript">
function open_add()
{
    window.open("addtype.php", "newframe", "width=200,height=50
, left=200 , top=200 , location=no,menubar=no, toolbar=no,
scrollbars=no, status=no");
}
</script>
```

(3) 保存商品信息，实现父窗口的刷新。

```
<?php
if($_POST[submit]!="")
{
    include_once("conn.php");
    $typename=$_POST[typename];
    if(trim($typename)=="")
    {
        echo "<script>alert('请输入商品类别名称!');
</script>";
    }
    $createtime=date("Y-m-jH:i:s");
    $sql=mysql_query("select * from tb_typewhere
typename=' ".$typename." ", $conn);
```

```

        $info=mysql_fetch_array($sql);           //判断该商品
是否已经被添加
        if($info!=false)
        {
            echo "<script>alert(' 该类别已经添加!');
</script>";
            exit;
        }
        else
        {
            if(mysql_query("insert into
tb_type(typename, createtime) values (' $typename', '$cr
eatetime')", $conn))
            {
                echo "<script>alert(' 类别添加成
功!');window.opener.location.reload();window.cl
ose();</script>";
                //关闭弹出窗口，自动刷新父窗口
            }
        }
    }
?>

```

### 举一反三

根据本实例，读者可以：

在电子商务管理系统中编辑商品信息时，刷新父窗口；

在企业信息网中添加解决方案信息时，刷新父窗口。

## 实例388 在弹出的网页模式对话框中选择个性头像

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\18\388

实例说明

在论坛、博客等注册页面添加头像选择功能，不仅可以使网页更具有活力，而且便于网站经营者了解注册用户的性格及区别不同的用户。运行本实例，首先单击图18.4所示的“头像选择”超级链接，在弹出窗口中选择自己喜爱的头像后，弹出窗口自动关闭，注册页面中的头像改变成用户所选择的头像。



图18.4 头像选择

技术要点

本实例通过调用window对象的showModalDialog()方法弹出一个窗口，当用户单击某个头像后，将该头像对应的数值传递给父窗口，通

过传递过来的数值指定用户所选择的头像。window对象的showModalDialog()方法说明如下。

```
window.showModalDialog(‘弹出窗口地址’, ‘弹出窗口名称’, ‘弹出窗口所具有的属性’)
```

弹出窗口所具有的属性指窗口的高度、宽度，以及窗口是否有地址栏、状态栏、滚动条等特性。

### 实现过程

(1) 首先为图18.4所示的窗口中的“[头像选择]”添加超级链接以便调用函数select\_face()实现窗口的弹出，代码如下。

```
[<a href="#" onclick="select_face()" class="a1">头像选择</a>]
```

(2) 利用JavaScript编写函数select\_face()，实现头像选择窗口的弹出以及返回用户所选择头像对应的数值。select\_face()函数代码如下。

```
<script language="javascript">
function select_face()
{
var somevalue;
somevalue=window.showModalDialog('browse.php', '', 'dialogWidth=600px dialogHeight=450px status=no help=no scrollbars=no');
document.form_regist.user_src.src="images/head/"+somevalue+".gif";
}
</script>
```

(3) 在弹出窗口中通过函数selectface()返回用户所选择头像对应的数值，代码如下。

```
<script language="javascript">
function selectface(ICQ)
{
window.returnValue=ICQ;
window.close();
}
</script>
```

举一反三

根据本实例，读者可以：

在留言板中实现头像选择功能；

在聊天室中实现头像选择功能。

## 实例389 时间选择器

本实例是一个人性化的程序

实例位置：光盘\mingrisoft\18\389

实例说明

在录入时间信息时，可以直接在文本框中输入。但由于日期型数据格式有多种，书写起来相对繁琐，而且采用这种方式也不利于日期格式的统一，所以可以在信息录入页面中加入一个简单的时间选择器来解决上述问题。运行本实例，首先将显示图18.5所示的信息录入页面，单击该页面“时间”文本框后的图标将弹出图18.6所示的对话框，在该对话框中选好时间后单击“确定”按钮，会发现用户所选择的时间信息显示在对应的的时间文本框中。

工作计划		
主题：	<input type="text"/>	
时间：	从 <input type="text"/>	到 <input type="text"/>
内容：	<div style="border: 1px solid gray; height: 100px; width: 100%;"></div>	
保存		

图18.5 信息录入页面

日期选择器 -- 网页对话框

2012 年 3 月 5 日

确定

图18.6 时间选择器

### 技术要点

由于时间、日期都为连续的数字，开发网页时，这些信息一个接一个地写在<option>与</option>标记之间固然可行，但这会降低程序开发时间，在开发本实例时本书将在下拉列表框中显示的连续数据通过PHP的循环结构显示出来，这样更灵活方便，实现该过程的代码如下。

```

<?php
    for($i=2005;$i<=2050;$i++)
    {
    ?>
        <optionvalue=<?php echo$i;?><?php echo$i;?></option>
    <?php
        }
    ?>

```

```

</select>
年</td>
<td width="64"><select name="y1">
<?php
for($i=1;$i<=12;$i++)
    {
    ?>
        <option value=<?php echo $i;?>><?php echo $i;?>
</option>
<?php
    }
    ?>
</select>
月</td>
<td width="77"><select name="r1">
<?php
for($i=1;$i<=31;$i++)
    {
    ?>
        <option value=<?php echo $i;?>><?php echo $i;?>
</option>
<?php
    }
    ?>
</select>

```

实现窗口之间的信息传递主要通过window对象的 showModalDialog() 方法实现，该方法说明如下。

window.showModalDialog(‘弹出窗口地址’, ‘弹出窗口名称’, ‘弹出窗口所具有的属性’)

弹出窗口所具有的属性指窗口的高度、宽度，以及窗口是否有地址栏、状态栏、滚动条等特性。

### 实现过程

(1) 在父窗口中创建函数open\_day\_from()和函数open\_day\_to(), 实现子窗口的弹出和父窗口与子窗口之间的信息传递，代码如下。

```
<script language="javascript">
function open_day_from()
{
    varday;
    day=window.showModalDialog(' day. php', '', ' dialogWidth=300px; dialogHeight=50px; scrollbars=no; status=no');
    if(day==undefined)
    {
        day="";
    }
    document.form1.timefrom.value=day;
}
function open_day_to()
{
    varday;
    day=window.showModalDialog(' day. php', '', ' dialogWidth=300px; dialogHeight=50px; scrollbars=no; status=no');
    if(day==undefined)
    {
```

```
        day="";
    }
    document.form1.timeto.value=day;
}
</script>
```

(2) 在弹出窗口中创建函数closeday( )，实现时间的传值，代码如下。

```
<script language="javascript">
function close_day()
{
    window.returnValue=document.form2.n1.value+"-
"+document.form2.y1.value+"-"+document.form2.r1.value;
    window.close();
}
</script>
```

举一反三

根据本实例，读者可以：

在开发论坛网站的用户注册页面中添加日期选择器；

在开发博客网的发表文章页面中添加日期选择器。

## 实例390 弹出提示对话框并重定向网页

这是一个可以提高分析能力的实例

实例位置：光盘\mingrisoft\18\390

实例说明

在开发某些程序的内部管理系统时，需要保证这个内部管理系统的安全性，允许什么人进入内部管理系统是一项非常重要的技术。可

可以在每个页面中设置一个访问的权限，判断是否可以访问该网页。如果用户名和密码不正确，就不可以访问该网页。这个技术通过JavaScript 脚本和PHP语句结合实现，即“弹出提示对话框并重定向网页”，如图18.7所示。



图18.7 弹出提示对话框并重定向网页

### 技术要点

本实例主要应用PHP与JavaScript相结合。通过JavaScript脚本来弹出对话框信息，应用window对象中的alert()方法和窗口属性location 实现此功能，语法的格式如下。

```
alert ('弹出对话框的内容');  
window.location.href='重新定向的页面';
```

参数说明如下。

- alert(): 弹出一个警告对话框。
- location: 定位区，也叫地址栏，是可以输入URL 的浏览器文本区。

### 实现过程

弹出提示对话框并重定向网页文件的代码如下。

```
<?php session_start();?>
<title>弹出提示对话框并重定向网页</title>
<?php
if($gly==true&&$mima==true) {
if($ljid==true) {
$ljid=$_GET['ljid'];
echo "这就是弹出提示对话框并重定向网页!";}
?>
<?php }else{
echo "<script>alert ('这就是弹出提示对话框并重定向网页!');window.location.href='admin.php';</script>";
}
?>
```

举一反三

根据本实例，读者可以：

在网站的后台管理系统中应用弹出提示对话框并重定向网页；

在涉及到用户访问权限的网页中应用弹出提示对话框并重定向网页。

## [实例391 删除数据前弹出确认对话框](#)

本实例是一个提高网站安全的实例

实例位置：光盘\mingrisoft\18\391

实例说明

在论坛系统的开发过程中，论坛的后台管理系统中，经常要删除一些数据，为了确保删除的数据是没有用处的，在执行删除数据之前

有一个确认删除数据的提示对话框是非常实用的。它可以提醒管理者对要删除的数据进行最后的确认，从而能够避免重要的数据被删除，造成不必要的损失，如图18.8所示。



图18.8 删除数据前弹出确认对话框

### 技术要点

本实例应用JavaScript脚本实现在删除数据前弹出确认对话框，主要应用window对象中的confirm( )方法和窗口属性设置中的location 参数实现此功能。该方法的语法如下。

window.confirm('对话框中输出的文字')

window.location.href='确定时跳转的页面';

参数说明如下。

- confirm(): 在确认对话框中显示指定的字符串。

### 实现过程

(1) 连接数据库文件的代码如下。

```
<?php
```

```

    $id=mysql_connect("localhost","root","root") or die('连接
失败:' . mysql_error());
    if(mysql_select_db("db_database01",$id))
        echo "";
    else
        echo ('数据库选择失败:' . mysql_error());
    mysql_query("set names gb2312");
?>

```

(2) 首先应用include()连接数据库，然后创建一个JavaScript脚本代码，使用select语句从数据库中查询出数据，最后应用onClick事件执行删除数据的操作，代码如下。

```

<?php session_start();include("Conn/conn.php");?>
<script language="javascript">
functiondelconfirm(x){
if(window.confirm('您确定删除该项吗?')==true)
    {window.location.href='gly_ok.php?ljid='+x;}}
</script>
<tablewidth="300"border="1"
cellpadding="0"cellspacing="2"background="image/left.jpg">
    <tr>
        <tdwidth="111"align="center"><span class="style1">主
题</span></td>
        <tdwidth="111"align="center"class="style1">内容</td>
        <tdwidth="70">&nbsp;&nbsp;&nbsp;</td>
    </tr>
    <?php
    $query="select * from tb_javascriptwhere id";

```

```

$result=mysql_query($query);
while (@$myrow=mysql_fetch_array($result)) {
?>
<tr>
    <td align="center" valign="middle" class="style2"><?
phpecho$myrow[id];?>&nbsp;&nbsp;&nbsp;<?phpecho$myrow[lmlb];?>
</td>
    <td align="center" class="style2"><p><?php
echo$myrow[lmnr];?></p> </td>
    <td align="center">
    <input
type="submit" name="Submit3" onClick="delconfirm(<?php
echo$myrow[id];?>)" value="删除">
    </td>
</tr>
<?php }?>
</table>

```

### 举一反三

根据本实例，读者可以：

在博客的后台管理系统中，在删除数据前弹出确认对话框；

在学校内部管理系统中，在删除学生信息前弹出确认对话框。

## 18.2 其他

### 实例392 树状导航菜单

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft\18\392

实例说明

在企业内部管理系统中，通过树状导航菜单来显示内部管理系统中的栏目是很棒的选择，它可以更加直观地显示出内部管理系统中的内容，如图18.9所示。



图18.9 树状导航菜单

技术要点

本实例主要应用JavaScript脚本和PHP语言的结合来实现，应用JavaScript脚本语言来实现树状菜单的效果，通过PHP语句从数据库中



```

        </tr>
    </table>
</div>
<div id='KB1Child' >
    <tablewidth="85%"border="0"cellspacing="0"cellpadding="0">
        <?php$query="select * from tb_javascriptwhere
lmlb=' 企业管理' ";
        $result=mysql_query($query);
        while($myrow=mysql_fetch_array($result)) {
        ?>
        <tr>
            <tdwidth="42%"height="23"align="right">-|
        </td>
            <tdwidth="58%"height="23"><a
href="index.php?lmb=<?php echo$myrow[lmnr];?
>" ><?php echo
        $myrow[lmnr];?></a></td>
        </tr>
        <?php }?>
    </table>
</div>
<div id='KB2Parent' class='parent' >
    <tablewidth="85%"border="0"cellspacing="0"cellpadding="0">
        <tr>

```

```

        <tdwidth="84%" height="25"><a href="#"
class="white01" onClick="expandIt('KB2');
return false">&nbsp;
        业务管理</a></td>
    </tr>
</table>
</div>
<div id='KB2Child' class='child' >
    <tablewidth="85%"border="0" cellspacing="0"
cellpadding="0">
        <?php$query=mysql_query("select * from
tb_javascriptwhere lmlb='业务管理' ");
        while($myrows=mysql_fetch_array($query)) {
            ?>
            <tr>
                <td align="right">-|</td>
                <tdheight="23"><a href="index.php?lmb=<?
php echo $myrows[lmnr];?>"><?php echo $myrows
[lmnr];?></a></td>
            </tr>
            <?php }?>
        </table>
    </div>
    <div id='KB3Parent' class='parent' >
        <tablewidth="85%"border="0"cellspacing="0"cellp
adding="0">
            <tr>

```

```

        <tdwidth="83%" height="25"><a href="#"
class="white01" onClick="expandIt('KB3');
return false">&nbsp;
        公司产品</a></td>
    </tr>
</table>
</div>
<div id='KB3Child' class='child' >
    <tablewidth="85%"border="0"
cellspacing="0"cellpadding="0">
        <?php$queryys=mysql_query("select * from
tb_javascriptwhere lmlb=' 公司产品' ");
        while($myrows=mysql_fetch_array($queryys)) {
            ?>
            <tr>
                <tdwidth="42%"align="right"class="white01">-|
            </td>
                <tdwidth="58%"height="23"><a href="index.php?
lmb=<?php echo$myrows[lmnr];?>"><?php echo
                $myrows[lmnr];?></a></td>
            </tr>
            <?php }?>
        </table>
    </div>
<SCRIPT>numTotal=3;scores[1]='KB1';scores[2]='KB1';scores
[3]='KB2' </SCRIPT></td>
</tr>

```

```
</table>
```

(3) 控制网页输出的switch语句的部分代码如下。

```
<?php
switch ($lmb) {
    case "经验技巧":
        include "jyjq.php";
        break;
    ..... //省略部分链接, 详细代码请读者参见光盘
    case "任务管理":
        include "ygrwgl.php";
        break;}
?>
```

(4) 处理树状菜单的JavaScript脚本的代码如下。

```
<script language="JavaScript" type="text/JavaScript">
functionMM_preloadImages() {
    var d=document; if(d.images) {
if(!d.MM_p)d.MM_p=newArray();
        var i,j=d.MM_p.length,a=MM_preloadImages.arguments;
        for(i=0; i<a.length; i++)
            if (a[i].indexOf("#")!=0) { d.MM_p[j]=new Image;
d.MM_p[j++].src=a[i];}}
    }
functionMM_findObj(n, d) {
    var p, i, x; if(!d) d=document;
if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document;n=n.substrin
g(0, p);}
}
```

```

        if (!(x=d[n])&& d.all) x=d.all[n]; for
(i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
        for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findOb
j(n, d.layers[i].document);
        if(!x&&d.getElementById) x=d.getElementById(n);
return x;
    }
    function MM_swapImgRestore() {
        var i,x,a=document.MM_sr; for(i=0;a&&i<a.length&&
(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
    }
    function MM_swapImage() {
        var
i, j=0, x, a=MM_swapImage.arguments; document.MM_sr=newArray;
for(i=0;i<(a.length-2);i+=3)
    if ((x=MM_findObj(a[i]))!=null) {document.MM_sr[j++]=x;
if(!x.oSrc)x.oSrc=x.src;x.src=a[i+2];}
    }
</script>

```

(5) 在本实例中调用script文件夹中的一个脚本文件  
spgl\_lmkz.js, 代码如下。

```

scores=newArray(20);varnumTotal=0;NS4= (document.layers)
?1 :0;
IE4= (document.all) ?1 :0;
ver4= (NS4 || IE4) ?1 :0;
if (ver4) {with (document){
    write("<STYLETYPE='text/css'>");

```

```

    if (NS4) {
        write(".parent
{position:absolute;visibility:visible}");
        write(".child
{position:absolute;visibility:visible}");
        write(".regular
{position:absolute;visibility:visible}")
    }else { write(".child {display:none}") }
write("</STYLE>"); }}
function arrange() {
    nextY=document.layers[firstInd].pageY+document.layers[f
irstInd].document.height;
    for (i=firstInd+1; i<document.layers.length; i++) {
        whichEl=document.layers[i];
        if (whichEl.visibility != "hide") {
            whichEl.pageY=nextY;
            nextY+=whichEl.document.height; } }}
function initIt() {
    if (!ver4) return;
    if (NS4) {
        for (i=0; i<document.layers.length; i++) {
            whichEl=document.layers[i];
            if (whichEl.id.indexOf("Child")
!=-1)whichEl.visibility= "hide";}
            arrange();
        }else{
            divColl=document.all.tags("DIV");

```

```

        for (i=0; i<divColl.length; i++) {
            whichEl=divColl(i);
            if (whichEl.className==
                "child")whichEl.style.display= "none";} }}
function expandIt(e1) {
    if (!ver4) return;
    if (IE4) {
        whichEl1= eval(e1+"Child");
        for(i=1;i<=numTotal;i++){
            whichEl= eval(scores[i]+"Child");
            if(whichEl!=whichEl1) {
                whichEl.style.display= "none";}}
        whichEl1= eval(e1+"Child");
        if (whichEl1.style.display== "none") {
            whichEl1.style.display= "block";
        }else{whichEl1.style.display= "none";}
    }else{
        whichEl= eval("document."+e1+"Child");
        for(i=1;i<=numTotal;i++){whichEl=
eval("document."+scores[i]+"Child");
            if(whichEl!=whichEl1) {whichEl.visibility=
"hide";}}
        if (whichEl.visibility== "hide") {
            whichEl.visibility= "show";
        }else{whichEl.visibility= "hide"; }
        arrange(); }}
onload= initIt;

```

举一反三

根据本实例，读者可以：

在开发博客管理系统中应用树状导航菜单；

在开发论坛的管理系统中应用树状导航菜单。

# 第19章 综合应用

计算器

日历

购物车

## 19.1 计算器

本节将通过JavaScript制作一个计算器，使读者对JavaScript的计算、赋值及自定义函数有更深入的了解。

### 实例393 简单计算器

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\19\393

实例说明

本实例是通过JavaScript脚本在IE浏览器中制作了一个计算器，它可以对数据进行加、减、乘、除的运算，也可以通过“Backspace”按钮对文本框中的数据进行撤销，还可以对数据进行正负值的切换。运行结果如图19.1所示。

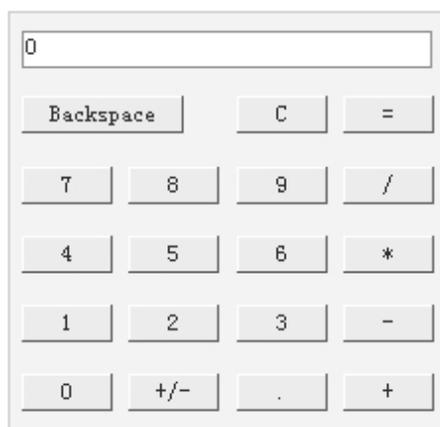


图19.1 简单计算器

技术要点

本实例的技术要点主要有以下几个方面。

(1) 在对文本框中的数据进行计算前，要用parseFloat()方法将字符串转换成单精度型。

(2) 用Math对象的floor()方法来获取当前文本框中数据的最小整数，当获取后的数据与文本框中的数据不相等时，则表示当前数据为小数。

(3) 当对数据进行正负值切换时，可以用Math对象的abs()方法，该方法用于获取数据的绝对值。

(4) 利用String对象的substr()方法来截取指定字符串中的子字符串，用于实现“Backspace”按钮的操作。

下面对substr()方法进行详细说明。

substr()方法的语法格式：

```
strObj. substr(startIndex, endIndex)
```

参数说明如下。

- strObj: 必选项，String 对象或文字。
- startIndex: 必选项，截取字符串的起始位置。
- endIndex: 必选项，截取字符串的结束位置。

功能：返回以startIndex开始，以endIndex结束的子字符串。

注意：本实例在进行加、减、乘、除运算时，结果只保留4位小数位数。

实现过程

(1) 在页面中用表格制作一个计算器，代码如下。

```
<tablewidth="213"height="214"border="1"cellspacing="5"
style="
background-color:#CCFFFF">
<tr>
<td id="td1"colspan="6"height="20">
```

```

        <inputname="zhi" type="text" value="0" size="33"
style="background-color:
        #FFFFFF;font-size:9pt;color:#333333;border:1px
solid#999999; cursor:text">
    </td>
</tr>
<tr>
    <td align="right"height="30"colspan="2">
        <div align="left">
            <input
type="button" name="back" value="Backspace" onClick="backs
pace()">
        </div></td>
    <td align="right">
        <div align="left">
            <input align="right" type="button" name="C" value="
C "onClick="Aclose()">
        </div></td>
    <td>
        <input type="button" name="Button" value="
= "onClick="js('=')">
    </td>
</tr>
<tr>
    <tdwidth="47"height="28">
        <input type="button" name="A7" value="
7 "onClick="num(7)">

```

```

</td>
<tdwidth="50">
  <input type="button" name="A8" value="
8  "onClick="num(8)">
</td>
<tdwidth="47">
  <input type="button" name="A9" value="
9  "onClick="num(9)">
</td>
<tdwidth="42">
  <input
type="button" name="chu2" value=" / "onClick="js(' /')">
  </td>
</tr>
<tr>
  <tdwidth="47" height="28">
    <input type="button" name="A4" value="
4  "onClick="num(4)">
  </td>
  <tdwidth="50">
    <input type="button" name="A5" value="
5  "onClick="num(5)">
  </td>
  <tdwidth="47">
    <input type="button" name="A6" value="
6  "onClick="num(6)">
  </td>

```

```

    <td>
        <input type="button" name="chu" value="
*  "onClick="js('*')">
    </td>
</tr>
<tr>
    <tdwidth="47" height="28">
        <input type="button" name="A1" value="
1  "onClick="num(1)">
    </td>
    <tdwidth="50">
        <input type="button" name="A2" value="
2  "onClick="num(2)">
    </td>
    <tdwidth="47">
        <input type="button" name="A3" value="
3  "onClick="num(3)">
    </td>
    <td>
        <input
type="button" name="chu" value=" - "onClick="js('-')">
    </td>
</tr>
<tr>
    <tdwidth="47" height="28">
        <input type="button" name="A0" value="
0  "onClick="num(0)">

```

```

</td>
<tdwidth="50">
    <input type="button" name="jgj" value="+/-"
"onClick="zf()">
</td>
<tdwidth="47">
    <input
type="button" name="dian" value=" ." "onClick="dian()">
</td>
<td>
    <input type="button" name="jia" value="
+ "onClick="js('+')">
</td>
</tr>
</table>

```

(2) 编写用于实现图片总置于工作区右上角的JavaScript代码。自定义全局变量。

```

<script language="Javascript">
var isnum=false;
var n1=0;
var fu="";
var zong=0;
var isdian=false;

```

自定义函数num(), 在单击按键时, 将数字和小数点显示在文本框中。

```

functionnum(n) {
    if (isnum==true) {

```

```

    if (zhi.value==='0.') {zhi.value=zhi.value+n;}
    else {zhi.value=n;}
    isnum=false;
}
else{
    if (zhi.value==='0') {zhi.value=n;}
    else{
        zhi.value=zhi.value+n;
    }
}
nl=parseFloat(zhi.value);
}

```

自定义函数dian(), 用于判断小数点是否显示。

```

functiondian() {
    var isfirst=isfloor();
    if ((isnum== true) || (zhi.value==='0')) {zhi.value='0.'}
    if ((isdian==false)&&(isfirst==true)) {
        zhi.value='0.';
    }
    else if (isdian==false) {
        if (parseFloat(zhi.value)===0) {zhi.value='0.';}
        else if (isfirst==true) {zhi.value=zhi.value;}
        else {zhi.value=zhi.value+'.';}
        isdian=true;
    }
}
}

```

自定义函数isfloor(), 用于判断文本框中的数据是否为小数。

```
function isfloor() {
    var int1=parseFloat(zhi.value);
    var int2=Math.floor(int1);
    if (int1>int2){return true;}
    else{return false;}
}
```

自定义函数Aclose(), 将计算器设为初始状态, 相当于一般计算器中的“C”键功能。

```
function Aclose() {
    isdian=isnum=false;
    zong= parseFloat(zhi.value);
    zhi.value="0";
    isnum=true;
    fu=base="";
    dd=jsq=zong=n1=0;
}
```

自定义函数isxiao(), 用于判断计算的结果是否为小数。

```
function isxiao(n) {
    var int1=parseFloat(n);
    var int2=Math.floor(int1);
    if (int1>int2){return true;}
    else{return false;}
}
```

自定义函数 js(), 用于对数据进行加、减、乘、除的计算, 可以直接用运算符键进行计算, 不需要使用“=”号键。

```
function js(s) {
    var lin= parseFloat(n1);
```

```

if ((fu=="")&&(s=="")) {
    if (isxiao(zong)==true) {zhi.value= zong.toFixed(4);}
    else {zhi.value= zong}
}
else if ((fu=="")&&(s=="*" || s=="/" || s=="+" || s=="-")) {
    if (isxiao(zong)==true) {zhi.value= zong.toFixed(4);}
    else {zhi.value= zong}
    fu= s;
}
else {
    if ( '+' == fu )
        zong+= lin;
    else if ( '-' == fu )
        zong-= lin;
    else if ( '/' == fu )
        zong/= lin;
    else if ( '*' == fu )
        zong*= lin;
    else
        zong= lin;
    if (isxiao(zong)==true) {zhi.value= zong.toFixed(4);}
    else {zhi.value= zong}
    fu= s;
}
isnum= true;
}

```

自定义函数zf(), 使数据进行正负值的切换。

```
function zf() {  
    var pp=parseFloat(zhi.value);  
    if (pp>0){zhi.value="-"+pp}  
    if (pp<0){zhi.value=Math.abs(pp)}  
}
```

自定义函数backspace(), 用于将文本框中数据从低位向高位递减。当文本框中数据的绝对值为个位数时, 该函数将文本框清零, 相当于一般计算器中的删除键Backspace。

```
functionbackspace() {  
    varbstr=zhi.value;  
    if (bstr!="0") {  
        var isabs=String(Math.abs(bstr));  
        if ((bstr.length==1) || (isabs.length==1)) {  
            zhi.value="0";  
            isdian=false;  
        }  
        else{zhi.value=bstr.substr(0,bstr.length-1)}  
    }  
}  
</script>
```

(3) 在页面中添加CSS样式, 用于改变计算器的颜色, 代码如下。

```
<style>  
table td{  
    border:#407D2A0px solid;  
}
```

```
input {
    font-family: "宋体"; font-size:9pt;color:#333333;
    background-
color:#e6e6e6;cursor:hand;padding:1px;height:19px;
    border-top:1px solid#FFFFFF;border-right:1px
solid#666666;
    border-bottom:1px solid#666666;border-left:1px
solid#FFFFFF;
}
</style>
```

举一反三

根据本实例，读者可以：  
制作一个标准型计算器。

## 实例394 复杂计算器

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\19\394

实例说明

在浏览一些网络数据时，经常要对数据进行计算，如果每次计算都打开Windows系统中的计算器，将会非常繁琐，可以在页面中调用自制的计算器。本实例使用JavaScript脚本在IE浏览器中制作了一个计算器，它与Windows系统中的标准型计算器实现的功能相同。运行结果如图19.2所示。

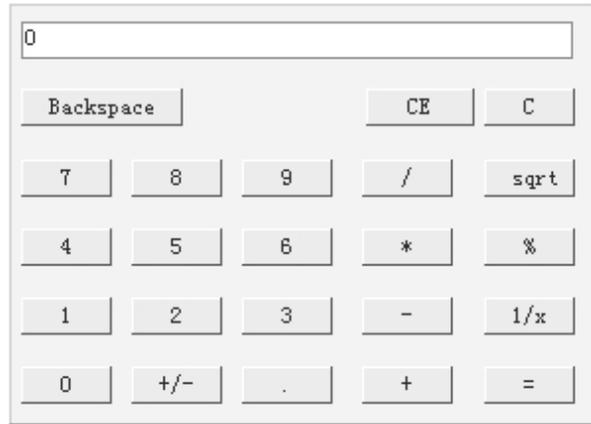


图19.2 复杂计算器

### 技术要点

本实例主要应用了赋值运算符和 Math 对象中的一些方法对计算器进行相应的操作。下面对赋值运算符和Math对象进行说明。

#### 1. Math对象

Math对象的相关方法及说明如表19.1所示。

表19.1 Math对象的方法

编 号	方 法	描 述
1	abs(x)	返回 x 的绝对值
2	acos(x)	返回 x 弧度的反余弦
3	asin(x)	返回 x 弧度的反正弦
4	atan(x)	返回 x 弧度的反正切
5	atan2(x,y)	返回坐标 (x,y) 对应的极坐标角度
6	ceil(x)	返回大于或等于 x 的最小整数
7	cos(x)	返回 x 的余弦
8	exp(x)	返回 e 的 x 乘方
9	floor(x)	返回小于或等于 x 的最大整数
10	log(x)	返回 x 的自然对数
11	max(x,y)	返回 x 和 y 中的最大数
12	min(x,y)	返回 x 和 y 中的最小数
13	pow(x,y)	返回 x 对 y 的次方
14	random()	返回 0 和 1 之间的随机数
15	round(x)	返回最接近 x 的整数
16	sin(x)	返回 x 的正弦值
17	sqrt(x)	返回 x 的平方根
18	tan(x)	返回 x 的正切值

## 2. 赋值运算符

最基本的赋值运算符是“=”，用于对变量进行赋值，而其他运算符可以和赋值运算符“=”联合使用，构成组合赋值运算符。

JavaScript支持的常用赋值运算符如表19.2所示。

表19.2 赋值运算符

运算符	描述
=	将右边表达式的值赋给左边的变量。例如：username="mr"
+=	将运算符左边的变量加上右边表达式的值赋给左边的变量。例如，a+=b，相当于a=a+b
-=	将运算符左边的变量减去右边表达式的值赋给左边的变量。例如，a-=b，相当于a=a-b
*=	将运算符左边的变量乘以右边表达式的值赋给左边的变量。例如，a*=b，相当于a=a*b
/=	将运算符左边的变量除以右边表达式的值赋给左边的变量。例如，a/=b，相当于a=a/b
%=	将运算符左边的变量用右边表达式的值求模，并将结果赋给左边的变量。例如，a%=b，相当于a=a%b

其他的技术要点请参考实例393。

注意：本实例在进行加、减、乘、除运算时，结果只保留4位小数。

### 实现过程

(1) 在页面中用表格制作一个计算器，代码如下。

```
<table width="300" height="214" border="1"
cellpadding="0" cellspacing="5"
style="background-color: #CCFFFF">
  <tr>
    <td id="td1" colspan="6" height="20">
      <input name="zhi" type="text" value="0" size="45"
style="background-color:
      #FFFFFF;font-size: 9pt;color: #333333;border: 1px
solid #999999; cursor:text">
    </td>
  </tr>
</tr>
<tr>
```

```
<td align="left"height="30"colspan="2">
  <input
type="button"name="back"value="Backspace"onClick="backs
pace()">
</td>
<td align="right"colspan="2">
  <input align="right" type="button"name="CE"value="
CE "onClick="ce()">
</td>
  <tdwidth="53">
    <input type="button"name="C"value="
C "onClick="Aclose()">
  </td>
</tr>
<tr>
  <tdwidth="51"height="28">
    <input type="button"name="A7"value="
7 "onClick="num(7)">
  </td>
  <tdwidth="51">
    <input type="button"name="A8"value="
8 "onClick="num(8)">
  </td>
  <tdwidth="56">
    <input type="button"name="A9"value="
9 "onClick="num(9)">
  </td>
```

```

    <tdwidth="57">
        <input
type="button" name="chu" value=" / " onClick="js(' / ') ">
    </td>
    <td>
        <input type="button" name="sqrt" value="
sqrt" onClick="kfang()" ">
    </td>
</tr>
<tr>
    <tdwidth="51" height="28">
        <input type="button" name="A4" value="
4 " onClick="num(4)" ">
    </td>
    <tdwidth="51">
        <input type="button" name="A5" value="
5 " onClick="num(5)" ">
    </td>
    <tdwidth="56">
        <input type="button" name="A6" value="
6 " onClick="num(6)" ">
    </td>
    <td>
        <input type="button" name="chu" value="
* " onClick="js('*') ">
    </td>
    <td>

```

```

        <input type="button" name="bai" value="
%  "onClick="bai()" ">
    </td>
</tr>
<tr>
    <tdwidth="51" height="28">
        <input type="button" name="A1" value="
1  "onClick="num(1)" ">
    </td>
    <tdwidth="51">
        <input type="button" name="A2" value="
2  "onClick="num(2)" ">
    </td>
    <tdwidth="56">
        <input type="button" name="A3" value="
3  "onClick="num(3)" ">
    </td>
    <td>
        <input
type="button" name="chu" value="  -  "onClick="js('-')">
    </td>
    <td>
        <input type="button" name="xz1" value="1/x
"onClick="ji()" ">
    </td>
</tr>
<tr>

```

```

        <tdwidth="51"height="28">
            <input type="button" name="A0" value="
0 "onClick="num(0)">
        </td>
        <tdwidth="51">
            <input type="button" name="jgj" value="+/-
"onClick="zf()">
        </td>
        <tdwidth="56">
            <input
type="button" name="dian" value=" . "onClick="dian()">
        </td>
        <td>
            <input type="button" name="jia" value="
+ "onClick="js('+')">
        </td>
        <td>
            <input type="button" name="deng" value="
= "onClick="js('=')">
        </td>
    </tr>
</table>

```

(2) 编写用于实现计算操作的JavaScript代码。  
定义全局变量。

```

<script language="Javascript">
var isnum=false;
var n1=0;

```

```
var fu="";
var zong=0;
var isdian=false;
```

自定义函数num(), 在单击按键时, 将数字和小数点显示在文本框中。

```
functionnum(n) {
    if (isnum==true) {
        if (zhi.value=='0.') {zhi.value=zhi.value+n;}
        else {zhi.value=n;}
        isnum=false;
    }
    else {
        if (zhi.value=='0') {zhi.value=n;}
        else {
            zhi.value=zhi.value+n;
        }
    }
    nl=parseFloat(zhi.value);
}
```

自定义函数dian(), 用于判断小数点是否显示。

```
functiondian() {
    var isfirst=isfloor();
    if ((isnum== true) || (zhi.value=='0')) {zhi.value='0.'}
    if ((isdian==false)&&(isfirst==true)) {
        zhi.value='0.';
    }
    else if (isdian==false) {
```

```

        if (parseFloat(zhi.value)=='0') {zhi.value='0.';}
        else if (isfirst==true) {zhi.value=zhi.value;}
        else {zhi.value=zhi.value+'.';}
        isdian=true;
    }
}

```

自定义函数isfloor(), 用于判断文本框中的数据是否为小数。

```

function isfloor() {
    var int1=parseFloat(zhi.value);
    var int2=Math.floor(int1);
    if (int1>int2) {return true;}
    else {return false;}
}

```

自定义函数ce(), 用于将文本框中的数据清0, 相当于计算器中的“CE”。

```

function ce() {
    zong= parseFloat(zhi.value);
    zhi.value="0";
    isnum=true;
    isdian=false;
}

```

自定义函数Aclose(), 用于将计算器设为初始状态, 相当于计算器中的“C”。

```

function Aclose() {
    isdian=isnum=false;
    ce();
    fu=base="";
}

```

```
    dd=jsq=zong=n1=0;
}
```

自定义函数isxiao ()，用于判断计算的结果是否为小数。

```
function isxiao(n) {
    var int1=parseFloat(n);
    var int2=Math.floor(int1);
    if (int1>int2){return true;}
    else{return false;}
}
```

自定义函数js ()，用于对数据进行加、减、乘、除的计算，可以直接用运算符进行计算，不需要使用“=”号，也可以用“=”号对同一运算进行重复操作，与计算器中的运算符相同。

```
var base;
function js(s) {
    var lin= parseFloat(n1);
    if ((s=="=")&&(fu=="=")) {
        if ((base=="+" || (base=="-") || (base=="*") ||
(base==" /")) {
            zong=eval(zong+base+lin);
            if (isxiao(zong)==true) {zhi.value=
zong.toFixed(4);}
            else {zhi.value= zong}
        }
    }
    else if ((fu=="=")&&(s==("*") || s==("/") || s==("+") || s==
("-")))) {
        if (isxiao(zong)==true) {zhi.value= zong.toFixed(4);}
    }
}
```

```

else{zhi.value= zong}
base=fu;
fu= s;
}
else{
if (isnum&& fu != "=") {
if ( '+' == fu )
zong=eval(zong+fu+lin);
else if ( '-' == fu )
zong=eval(zong+fu+lin);
else if ( '/' == fu )
zong=eval(zong+fu+lin);
else if ( '*' == fu )
zong=eval(zong+fu+lin);
else if ( '.' == fu)
zong= lin;
if (isxiao(zong)==true) {zhi.value=
zong.toFixed(4);}
else{zhi.value= zong}
base=fu;
fu= s;
}
else{
if ( '+' == fu )
zong+= lin;
else if ( '-' == fu )
zong=zong-lin;

```

```

else if ( '/' == fu )
    zong/= lin;
else if ( '*' == fu )
    zong*= lin;
else
    zong= lin;
if ( isxiao(zong)==true) {zhi.value=
zong.toFixed(4);}
else {zhi.value= zong}
base=fu;
fu= s;
}
}
isnum= true;
}

```

自定义函数bai(), 用于计算数据的百分数, 相当于一般计算器中的“%”功能。

```

function bai() {
    zhi.value = (parseFloat(zhi.value)/100) *
parseFloat(zong);
    isdian=false;
}

```

自定义函数kfang(), 对文本框中的数据进行开方, 相当于一般计算器中的sqrt功能。

```

function kfang() {
    if (zhi.value!="0" || zhi.value!="") {
        zhi.value=Math.sqrt(parseFloat(zhi.value));
    }
}

```

```

        isnum=true;
        isdian=false;
    }
}

```

自定义函数zf(), 使数据进行正负值的切换。

```

function zf() {
    var pp=parseFloat(zhi.value);
    if (pp>0) {zhi.value="-"+pp}
    if (pp<0) {zhi.value=Math.abs(pp)}
}

```

自定义函数 ji(), 将当前数据设为分母, 取其中的一份, 相当于计算器中一般的“1/x”功能。

```

function ji() {
    var pp=parseFloat(zhi.value);
    zhi.value=parseFloat(1/pp);
    isnum=true;
    isdian=false;
}

```

自定义函数backspace(), 用于将文本框中数据从低位向高位递减。当文本框中数据的绝对值为个位数时, 该函数将文本框清零, 相当于一般计算器中的Backspace功能。

```

functionbackspace() {
    varbstr=zhi.value;
    if (bstr!="0") {
        var isabs=String(Math.abs(bstr));
        if ((bstr.length==1) || (isabs.length==1)) {
            zhi.value="0";
        }
    }
}

```

```
        isdian=false;
    }
    else{zhi.value=bstr.substr(0,bstr.length-1)}
}
}
</script>
```

(3) 在页面中添加CSS样式，用于改变计算器的颜色，代码如下。

```
<style>
table td{
    border: #407D2A 0px solid;
}
input{
    font-family: "宋体"; font-size: 9pt;color: #333333;
    background-color: #eeeeee;cursor:
hand;padding:1px;height:19px;
    border-top: 1px solid #FFFFFF;border-right:1px solid
#666666;
    border-bottom: 1px solid #666666;border-left: 1px solid
#FFFFFF;
}
</style>
```

举一反三

根据本实例，读者可以：

制作一个科学型计算器。

## 19.2 日历

在网站中加入日历，不但可以美化页面，而且可以方便浏览者。本节将介绍如何实现精美日历及带农历的日历。

### 实例395 精美日历

这是一个实用的实例

实例位置：光盘\mingrisoft\19\395

实例说明

日历常常会被应用在各个中小型网站中，日历在网页中的主要用处是显示日期，但我们还可以借助日历实现填写日期或者日期查询等功能。本例使用 JavaScript 制作了一个精美日历，主要显示了1904年到2056年间的所有日期，其中每周的星期日都用红色显示，当天日期用蓝色显示，如图19.3所示。



图19.3 精美日历

### 技术要点

本实例主要通过Menu组件的options数组列表来判断下拉列表中所选择的年、月，并通过自定义函数来计算当月日期在表格中的相应位置。

options数组是select对象的一个属性，即选择框中的所有选项（<Option>）的一个列表。

options数组的相关属性如表19.3所示。

表19.3 options数组的相关属性

属 性	描 述
defaultSelected	选项列表中的缺省选项
index	选项列表中某选项的索引位置
length	选项列表中的选项数（<OPTIONS>）
name	选项列表的名字（NAME 特性）
selected	表示选项列表中某选项<OPTION>是否被选中的一个布尔类型值
selectedIndex	选项列表中已选中的<OPTION>的索引（位置）
text	选项列表中<OPTION>标记后的文本
value	选项列表中的 VALUE（特性）

## 实现过程

(1) 在页面中添加CSS样式，代码如下。

```
<style>
  body,td { font-size:9pt; }
  input {
    color:333333;
    font-size:9pt;
    border-width:0;
    padding:2;
    width:20;
  }
</style>
```

(2) 编写用于实现精美日历的JavaScript代码。

```
<script language=javascript>
  varmonthNames=newArray ( "",
"1","2","3","4","5","6","7", "8", "9", "10", "11", "12" );
  var endDay=newArray (0,31,28,31,30,31,
30,31,31,30,31,30,31 );
  vardayNow= 0;
  varmonthNow=0;
  varyearNow=0;
  function load ( form ) {
    set_month_year_now ();
    var found= false;
    for
(varmonth=0;month<form.monthList.length;month++)
```

```

        if (
form.monthList[month].text==monthNames[monthNow] ) {
            form.monthList[month].selected= true;
            found= true;
        }
if ( !found) {
    error ();
    return;
}
var found= false;
for (var year=0;year<form.yearList.length;year++)
    if ( form.yearList[year].text== yearNow ) {
        form.yearList[year].selected= true;
        found= true;
    }
if ( !found) {
    error ();
    return;
}
display_month ( form );
}

functionpreceding_month ( form ) {
    varmonth_selected= form.monthList.selectedIndex;
    var year_selected= form.yearList.selectedIndex;
    if ( !month_selected&& !year_selected ) {
        error ();
        return;
    }
}

```

```

}
if (month_selected>0 )
    month_selected--;
else{
    month_selected=11;
    year_selected--;
}
form.monthList[month_selected].selected= true;
form.yearList[year_selected].selected= true;
display_month ( form );
}

function following_month ( form ) {
    varmonth_selected= form.monthList.selectedIndex;
    var year_selected= form.yearList.selectedIndex;
    if (month_selected>= ( form.monthList.length-1
)&&year_selected>= ( form.yearList.length-1 ) ) {
        error ();
        return;
    }
    if (month_selected<11 )
        month_selected++;
    else{
        month_selected=0;
        year_selected++;
    }
    form.monthList[month_selected].selected= true;
    form.yearList[year_selected].selected= true;
}

```

```

    display_month ( form );
}
function set_month_year_now () {
    var form=document.calendar;
    varnow=newDate ();
    monthNow=now.getMonth ()+1;
    yearNow=now.getFullYear ();
    dayNow=now.getDate ();
    yearNow= ( yearNow<100 ) ?yearNow+1900 :yearNow;
    var count=0
    for (var i=yearNow-103; i< yearNow+50; i++) {
        eval ("form.yearList.options[count]=newOption(' "+i+"
', ' "+i+"')");
        count++;
    }
    form.yearList.selectedIndex=103;
    form.yearList.length= count;
}
function error () {
    alert ( "超出范围!" );
}
functiondisplay_month ( form ) {
    varmonth= form.monthList.selectedIndex+1;
    var year=parseInt ( form.yearList.options[
form.yearList.selectedIndex].text );
    var start_day= start_day_in_month ( year,month );
    var count=0;

```

```

for (var row=0; row<6; row++){
    for ( var col=0; col<7; col++){
        if ( row==0&& col< ( start_day-1 ) )
            varday= "";
        else if ( count< endDay[month] )
            day=++count;
        else
            day= "";
        form.dayBox[(row*7)+col].style.display= "";
        form.dayBox[(row*7)+col].style.color= "black";
        if (day== "") {
            form.dayBox[(row*7)+col].style.display= "none";
        } else {
            form.dayBox[(row*7)+col].value=day;
            if (col%7==0)
                form.dayBox[(row*7)+col].style.color= "red";
            if
                (yearNow==year&&monthNow==month&&dayNow==day)
                form.dayBox[(row*7)+col].style.color= "blue";;
        }
    }
}
}

function start_day_in_month ( year, month ) {
    var day, daynum, ndays, mnum;
    sday= start_day_in_year ( year );
    endDay[2]= ( year%4 ) ?28 :29;
}

```

```

    if (month==1 )
        daynum= sday;
    else {
        ndays= sday;
        for (mnum=2;mnum<month+1;mnum++)
            ndays=ndays+endDay[mnum-1];
        daynum=ndays%7;
    }
    daynum= (!daynum) ?7 :daynum;
    return (daynum);
}

function start_day_in_year ( year ) {
    vary, m, d;
    varn;
    y= year-1;m=13;d=1;
    n=d+2*m+(Math.floor ( (0.6+(m+1) ) )+y );
    n=n+Math.floor ( ((y/4)-Math.floor ( (y/100 )
)+Math.floor ( ( y/400 ) ) ) )+2 ;
    n=Math.floor ( ( (n/7-Math.floor ( (n/7) ) ) *7+0.5 )
);
    return (n+1);
}

functionCheckDate(strDay) {
    vardocFrm=document.calendar;
    var choice_daynum=0;
    var current_daynum=0;
    varday_temp;

```

```

if (strDay != "") {
    var strY=docFrm.yearList.value;
    var strM=docFrm.monthList.value;
    var curr_y=newString(yearNow);
    var curr_m=newString(monthNow);
    var curr_d=newString(dayNow);
    if (curr_m.length==1) curr_m= "0"+curr_m;
    if (curr_d.length==1) curr_d= "0"+curr_d;
    current_daynum=newNumber(curr_y+curr_m+curr_d) ;
    if (strM.length==1) strM= "0"+strM;
    if (strDay.length==1) strDay= "0"+strDay;
    choice_daynum=newNumber(strY+strM+strDay);
    parent.window.returnValue= strY+"-"+strM+"-
"+strDay;//将选择的日期传递到父窗口中
    //parent.window.close();
}
return false;
}
</script>

```

(3) 添加页面设计代码，当页面加载时调用load()函数。

```

<bodyonLoad="load(document.calendar)" topmargin="0">
<center>
    <formname="calendar">
        <tableborder="0" cellpadding="0" cellspacing="0">
            <tr>
                <td colspan="3"height="24"></td>
            </tr>

```

```

<tr>
  <tdwidth="205" align="right" VALIGN="MIDDLE" nowrap=
"nowrap">
    <fontSIZE="6pt" COLOR="#FF9900" FACE="黑体">
      精美日历
    </font>
    <selectname="yearList" onChange="display_month
(this.form)">
    </select>    </td>
  <tdwidth="65" nowrap="nowrap" align="left">
  <selectname="monthList" size="1" onChange="display_mo
nth(this.form)">
    <optionvalue="1">1</option>
    <optionvalue="2">2</option>
    <optionvalue="3">3</option>
    <optionvalue="4">4</option>
    <optionvalue="5">5</option>
    <optionvalue="6">6</option>
    <optionvalue="7">7</option>
    <optionvalue="8">8</option>
    <optionvalue="9">9</option>
    <optionvalue="10">10</option>
    <optionvalue="11">11</option>
    <optionvalue="12">12</option>
  </select>
</td>
<tdwidth="10"></td>

```

```

</tr>
<tr>
  <td colspan="3"height="6"></td>
</tr>
<tr>
  <td colspan="3">
<tableborder="0"cellpadding="1"cellspacing="0"align="
center">
  <tr>
    <tdbgcolor="#82664F">
<tableborder="0"cellpadding="0" cellspacing="0">
  <trbgcolor="#82664F"height="18">
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF"><fontcolor="#FF0000">日</font>
</td>
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF">一</td>
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF">二</td>
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF">三</td>
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF">四</td>
    <tdwidth="31"align="center"nowrap="nowrap"BG
COLOR="#0099FF">五</td>
    <td width="31" align="center"
nowrap="nowrap" BGCOLOR="#0099FF">六</td>

```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</tr>
<tr bgcolor="#FFFFFF" height="18">
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript:CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
```

```

onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
</tr>
<tr bgcolor="#FFFFFF" height="18">
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"

```

```
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
</tr>
<tr bgcolor="#FFFFFF" height="18">
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate(this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```

        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly onClick
="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
</tr>
<tr bgcolor="#FFFFFF" height="18">

```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
```

```

onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
</tr>
<tr bgcolor=" #FFFFFF" height="18">
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"

```

```
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
    <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background='#EEEEEE'"
onmouseout="this.style.background='white'">
    </td>
```

```
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
</tr>
<tr bgcolor="#FFFFFF" height="18">
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
<td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
</td>
```

```

        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE
' " onmouseout="this.style.background=' white' ">
        </td>
        <td align="center"><input type="text"
size="2" name="dayBox" readOnly
onClick="javascript: CheckDate (this.value);"
onMouseOver="this.style.background=' #EEEEEE' "
onmouseout="this.style.background=' white' ">
        </td>
    </tr>
</table></td>
</tr>

```

```
        </table></td>
    </tr>
</table>
</form>
</center>
</body>
```

举一反三

根据本实例，读者可以：

制作一个带备忘录的日历；

在制作的日历中添加节日提示信息。

## 实例396 带农历的日历

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\19\396

实例说明

在人们的日常生活中，对日历的查看是不可缺少的。本实例制作了一个既可以查看公历，又可以查看农历的复杂日历，并且在日期的下面显示了公历与农历的各个节日及农历的节气。运行结果如图19.4所示。

公历 2006 年 12 月 <span style="float: right;">【狗】</span>						
日	一	二	三	四	五	六
					<b>1</b> 十一	<b>2</b> 十二
<b>3</b> 十三	<b>4</b> 十四	<b>5</b> 十五	<b>6</b> 十六	<b>7</b> 大雪	<b>8</b> 十八	<b>9</b> 十九
<b>10</b> 二十	<b>11</b> 廿一	<b>12</b> 廿二	<b>13</b> 廿三	<b>14</b> 廿四	<b>15</b> 廿五	<b>16</b> 廿六
<b>17</b> 廿七	<b>18</b> 廿八	<b>19</b> 廿九	<b>20</b> 11月大	<b>21</b> 初二	<b>22</b> 冬至	<b>23</b> 初四
<b>24</b> 平安夜	<b>25</b> 圣诞节	<b>26</b> 初七	<b>27</b> 初八	<b>28</b> 初九	<b>29</b> 初十	<b>30</b> 十一
<b>31</b> 十二						

图19.4 带农历的日历

### 技术要点

本实例主要利用Date对象来获取指定日期的相关信息，并用Lunar对象将指定日期转换成相应的农历日期。下面对Date和Lunar对象进行详细的介绍。

Date 对象是一个有关日期和时间的对象。它具有动态性，即必须使用 new 运算符创建一个实例。

例如：

```
dateObj=newDate(); //获取的是当前日期
```

```
dateObj=newDate(y, m, d); //获取的是指定年月的日期
```

Date对象没有提供直接访问的属性，只具有获取和设置日期和时间的方法。Date对象的方法如表19.4所示。

表19.4 Date对象的方法

编 号	方 法	描 述
1	getFullYear()	返回客户端的系统时间中的年数
2	getMonth()	返回客户端的系统时间中的月数 (0~11)
3	getDate()	返回客户端的系统时间中的日数 (1~31)
4	getDay()	返回客户端的系统时间中的星期 (1~6)
5	getHours()	返回客户端的系统时间的小时数 (0~23)
6	getMinutes()	返回客户端的系统时间的分钟数 (0~59)
7	getSeconds()	返回客户端的系统时间的秒数 (0~59)
8	getTime()	返回客户端的系统时间的毫秒数
9	setYear()	设置年
10	setMonth()	设置月
11	setDate()	设置日
12	setHours()	设置小时
13	setMinutes()	设置分钟
14	setSeconds()	设置秒数
15	setTime()	设置毫秒数

下面对比较重要的方法进行详细介绍。

(1) `getFullYear`，返回Date对象中的年份值。

语法：

`dateObj.getFullYear()`

`dateObj`：必选项，`dateObj`参数为Date对象。

(2) `getMonth`，返回Date对象中用本地时间表示的月份值。

语法：

`dateObj.getMonth()`

`dateObj`：必选项，`dateObj`参数为Date对象。

(3) `getDate`，返回Date对象中用本地时间表示的一个月中的日期值。

语法：

`dateObj.getDate()`

`dateObj`：必选项，`dateObj`参数为Date对象。

(4) `getTime`，返回Date对象中的时间值。

语法:

`dateObj.getTime()`

`dateObj`: 必选项, `dateObj`参数为Date对象。

(5) `setYear`, 设置Date对象中的年份值。

语法:

`dateObj.setYear(numYear)`

`dateObj`: 必选项, 任意Date对象。

`numYear`: 必选项, 其数值等于年份减去1900。

实现过程

(1) 在页面中添加一个表格, 代码如下。

```
<CENTER>
```

```
<FORM name=CLD>
```

```
<TABLE>
```

```
<TR>
```

```
<TD align=middle>
```

```
<TABLE border=1 cellpadding="0" cellspacing="0"
```

```
bordercolordark="#FFFFFF"
```

```
bordercolor="#ffffff" bordercolorlight="#EEEEEE">
```

```
<TR bgcolor="#006600">
```

```
<TD colSpan=7><FONT color=#ffffff style="FONT-  
SIZE: 9pt">公历
```

```
<SELECT name=SY onchange=changeCld()
```

```
style="FONT-SIZE: 9pt">
```

JavaScript代码在Menu组件中动态添加下拉菜单(年)。

```
<SCRIPT language="JavaScript">
```

```
for(i=1900;i<2050;i++) document.write('<option>'+i);
```

```
</SCRIPT>
```

```
</SELECT> 年 <SELECT name=SM onchange=changeCld()  
style="FONT-SIZE: 9pt">
```

JavaScript代码在Menu组件中动态添加下拉菜单（月）。

```
<SCRIPT language="JavaScript">  
for(i=1;i<13;i++) document.write('<option>' +i);  
</SCRIPT>
```

在表格中添加单元格并设置单元格文本大小的HTML代码如下。

```
</SELECT> 月 </FONT> <FONT color=#ffffff face=宋体  
id=GZ  
style="FONT-SIZE: 12pt"></FONT><BR></TD>  
</TR>  
<TR align=middle bgColor=#e0e0e0>  
<TD width=54 style="font-size:9pt; padding:5pt;">日  
</TD>  
<TD width=54 style="font-size:9pt ">一</TD>  
<TD width=54 style="font-size:9pt ">二</TD>  
<TD width=54 style="font-size:9pt ">三</TD>  
<TD width=54 style="font-size:9pt ">四</TD>  
<TD width=54 style="font-size:9pt ">五</TD>  
<TD width=54 style="font-size:9pt ">六</TD></TR>
```

JavaScript代码在表格中添加6行7列的单元格，代码如下。

```
<SCRIPT language="JavaScript">  
var gNum;  
for(i=0;i<6;i++) {  
    document.write('<tr align=center>');  
    for(j=0;j<7;j++) {  
        gNum= i*7+j;
```

```

        document.write(' <td id="GD'+gNum+' "><font
id="SD'+gNum+' "
        size=2 face="ArialBlack">);
        if(j==0)document.write(' color=red');
        if(j==6)document.write(' color=#000080');
        document.write(' TITLE=""></font><br><font
id="LD'+gNum+' "
        size=2 style="font-size:9pt"></font></td>');
    }
    document.write(' </tr>');
}
</SCRIPT>

```

HTML代码的相关结束标记，代码如下。

```

        </TABLE>
        </TD>
    </TR>
</TABLE>
</FORM>
</CENTER>

```

(2) 编写用于实现公历日历与农历日历的JavaScript代码。  
用数组记录日历中的相关信息。

```

<SCRIPT language="JavaScript">
var lunarInfo=new Array(
    0x04bd8, 0x04ae0, 0x0a570, 0x054d5, 0x0d260, 0x0d950, 0x16554, 0
x056a0, 0x09ad0, 0x055d2,
    0x04ae0, 0x0a5b6, 0x0a4d0, 0x0d250, 0x1d255, 0x0b540, 0x0d6a0, 0
x0ada2, 0x095b0, 0x14977,

```

0x04970, 0x0a4b0, 0x0b4b5, 0x06a50, 0x06d40, 0x1ab54, 0x02b60, 0x09570, 0x052f2, 0x04970,  
0x06566, 0x0d4a0, 0x0ea50, 0x06e95, 0x05ad0, 0x02b60, 0x186e3, 0x092e0, 0x1c8d7, 0x0c950,  
0x0d4a0, 0x1d8a6, 0x0b550, 0x056a0, 0x1a5b4, 0x025d0, 0x092d0, 0x0d2b2, 0x0a950, 0x0b557,  
0x06ca0, 0x0b550, 0x15355, 0x04da0, 0x0a5d0, 0x14573, 0x052d0, 0x0a9a8, 0x0e950, 0x06aa0,  
0x0aea6, 0x0ab50, 0x04b60, 0x0aae4, 0x0a570, 0x05260, 0x0f263, 0x0d950, 0x05b57, 0x056a0,  
0x096d0, 0x04dd5, 0x04ad0, 0x0a4d0, 0x0d4d4, 0x0d250, 0x0d558, 0x0b540, 0x0b5a0, 0x195a6,  
0x095b0, 0x049b0, 0x0a974, 0x0a4b0, 0x0b27a, 0x06a50, 0x06d40, 0x0af46, 0x0ab60, 0x09570,  
0x04af5, 0x04970, 0x064b0, 0x074a3, 0x0ea50, 0x06b58, 0x055c0, 0x0ab60, 0x096d5, 0x092e0,  
0x0c960, 0x0d954, 0x0d4a0, 0x0da50, 0x07552, 0x056a0, 0x0abb7, 0x025d0, 0x092d0, 0x0cab5,  
0x0a950, 0x0b4a0, 0x0baa4, 0x0ad50, 0x055d9, 0x04ba0, 0x0a5b0, 0x15176, 0x052b0, 0x0a930,  
0x07954, 0x06aa0, 0x0ad50, 0x05b52, 0x04b60, 0x0a6e6, 0x0a4e0, 0x0d260, 0x0ea65, 0x0d530,  
0x05aa0, 0x076a3, 0x096d0, 0x04bd7, 0x04ad0, 0x0a4d0, 0x1d0b6, 0x0d250, 0x0d520, 0x0dd45,  
0x0b5a0, 0x056d0, 0x055b2, 0x049b0, 0x0a577, 0x0a4b0, 0x0aa50, 0x1b255, 0x06d20, 0x0ada0)

```

var
solarMonth=newArray(31,28,31,30,31,30,31,31,30,31,30,31);
    varAnimals=newArray("鼠","牛","虎","兔","龙","蛇","马","
羊","猴","鸡","狗","猪");
    var solarTerm=newArray("小寒","大寒","立春","雨水","惊
蛰","春分","清明","谷雨","立夏"
    ,"小满","芒种","夏至","小暑","大暑","立秋","处暑","白
露","秋分","寒露","霜降","立冬","小雪"
    ,"大雪","冬至");    //农历的节气
var
sTermInfo=newArray(0,21208,42467,63836,85337,107014,128867,15
0921,173149,
    195551,218072,240693,263343,285989,308563,331033,353350,3
75494,397447,419210,
    440795,462224,483532,504758);
var
nStr1=newArray('日','一','二','三','四','五','六','七','八','
九','十');
    var nStr2=newArray('初','十','廿','卅');
    用数组保存公历的节日。
    var sFtv = new Array(
    "0101 元旦",
    "0214 情人节",
    "0308 妇女节",
    "0312 植树节",
    "0315 消费者权益日",
    "0401 愚人节",

```

```
"0501 劳动节",  
"0504 青年节",  
"0512 护士节",  
"0601 儿童节",  
"0701 建党节",  
"0801 建军节",  
"0910 教师节",  
"0928 孔子诞辰",  
"1001 国庆节",  
"1006 老人节",  
"1024 联合国日",  
"1224 平安夜",  
"1225 圣诞节")
```

用数组保存农历的节日。

```
var lFtv = new Array(  
"0101 春节",  
"0115 元宵节",  
"0505 端午节",  
"0707 七夕情人节",  
"0715 中元节",  
"0815 中秋节",  
"0909 重阳节",  
"1208 腊八节",  
"1223 小年")
```

自定义函数lYearDays(y)，用于返回农历y年的总天数。

```
function lYearDays(y) {  
    var i, sum = 348;
```

```

        for(i=0x8000; i>0x8; i>>=1) sum+=(lunarInfo[y-1900]&i)?
1:0;
        return(sum+leapDays(y));
    }

```

自定义函数leapDays(y)，用于返回农历y年闰月的天数。

```

function leapDays(y) {
    if(leapMonth(y)) return((lunarInfo[y-1900]&0x10000)?
30:29);
    else return(0);
}

```

自定义函数leapMonth(y)，用于判断y年的农历中哪个月是闰月，不是闰月返回0。

```

function leapMonth(y) {
    return(lunarInfo[y-1900]&0xf);
}

```

自定义函数monthDays(y, m)，用于返回农历y年m月的总天数。

```

function monthDays(y, m) {
    return((lunarInfo[y-1900]&(0x10000>>m))?30:29);
}

```

自定义函数 Dianaday()，用于算出当前月第一天的农历日期和当前农历日期下一个月农历的第一天日期。

```

functionDianaday(objDate) {
    var i, leap=0, temp=0;
    varbaseDate=newDate(1900, 0, 31);
    var offset = (objDate-baseDate)/86400000;
    this.dayCyl= offset+40;
    this.monCyl= 14;
}

```

```

for(i=1900; i<2050&& offset>0; i++){
    temp= 1YearDays(i)
    offset-= temp;
    this.monCyl+=12;
}
if(offset<0) {
    offset+= temp;
    i--;
    this.monCyl-= 12;
}
this.year= i;
this.yearCyl=i-1864;
leap= leapMonth(i); //闰哪个月
this.isLeap= false;
for(i=1; i<13&& offset>0; i++) {
    if(leap>0&& i==(leap+1)&& this.isLeap==false) { //闰
月
        --i; this.isLeap= true; temp= leapDays(this.year);}
    else{
        temp=monthDays(this.year, i);}
    if(this.isLeap==true&& i==(leap+1)) this.isLeap=
false; //解除闰月
    offset-= temp;
    if(this.isLeap== false) this.monCyl++;
}
if(offset==0&& leap>0&& i==leap+1)
    if(this.isLeap) { this.isLeap= false;}

```

```

        else {this.isLeap=true;--i;--this.monCyl;}
    if(offset<0) {offset+=temp;--i;--this.monCyl;}
    this.month=i;
    this.day=offset+1;
}

```

自定义函数solarDays(y,m)，用于返回公历y年m+1月的天数。

```

function solarDays(y,m) {
    if(m==1)
        return(((y%4==0)&&(y%100!=0)|| (y%400==0))?29:28);
    else
        return(solarMonth[m]);
}

```

自定义函数calElement()用于记录公历和农历某天的日期。

```

function
calElement(sYear, sMonth, sDay, week, lYear, lMonth, lDay, isLeap) {
    this.isToday= false;
    //公历
    this.sYear= sYear;
    this.sMonth= sMonth;
    this.sDay= sDay;
    this.week=week;
    //农历
    this.lYear= lYear;
    this.lMonth= lMonth;
    this.lDay= lDay;
    this.isLeap= isLeap;
    //节日记录
}

```

```

    this.lunarFestival = ''; //农历节日
    this.solarFestival = ''; //公历节日
    this.solarTerms = ''; //节气
}

```

自定义函数sTerm(y, n)用于返回某年的第n个节气为几日（从小寒算起）。

```

function sTerm(y, n) {
    var offDate = new Date((31556925974.7*(y-
1900)+sTermInfo[n]*60000)+Date.UTC
    (1900, 0, 6, 2, 5));
    return(offDate.getUTCDate())
}

```

自定义函数calendar(y, m)用于保存y年m+1月的相关信息。

```

var fat=mat=9;
var eve=0;
function calendar(y, m) {
    fat=mat=0;
    var sDObj, lDObj, lY, lM, lD=1, lL, lX=0, tmp1, tmp2;
    var lDPOS=newArray(3);
    varn=0;
    var firstLM=0;
    sDObj=newDate(y, m, 1); //当月第一天的
信息
    this.length= solarDays(y, m); //公历当
月天数
    this.firstWeek= sDObj.getDay(); //公历当月
1日星期几

```

```

if ((m+1)==5) {fat=sDObj.getDay()}
if ((m+1)==6) {mat=sDObj.getDay()}
for(var i=0;i<this.length;i++){
    if(1D>1X) {
        sDObj=newDate(y, m, i+1);           //当月第一天的信息
        1DObj=newDianaday(sDObj);         //农历
        1Y= 1DObj.year; //农历年
        1M= 1DObj.month; //农历月
        1D= 1DObj.day; //农历日
        1L= 1DObj.isLeap; //农历是否闰月
        1X= 1L? leapDays(1Y):monthDays(1Y, 1M); //农历当月最后一天
        if (1M==12) {eve=1X}
        if(n==0) firstLM= 1M;
        1DPOS[n++]= i-1D+1;
    }
    this[i]=new
calElement(y, m+1, i+1, nStr1[(i+this.firstWeek)%7], 1Y, 1M, 1D
++, 1L);
    if((i+this.firstWeek)%7==0) {
        this[i].color= 'red'; //周日颜色
    }
}
//节气
tmp1=sTerm(y, m*2)-1;
tmp2=sTerm(y, m*2+1)-1;

```

```

this[tmp1].solarTerms= solarTerm[m*2];
this[tmp2].solarTerms= solarTerm[m*2+1];
if((this.firstWeek+12)%7==5) //黑色星期五
    this[12].solarFestival+= '黑色星期五';
if(y==tY&& m==tM) this[tD-1].isToday= true; //今日
}

```

自定义函数cDay( )，用中文显示农历的日期。

```

function cDay(d) {
    var s;
    switch (d) {
        case 10:
            s = '初十'; break;
        case 20:
            s = '二十'; break;
            break;
        case 30:
            s = '三十'; break;
            break;
        default :
            s = nStr2[Math.floor(d/10)];
            s+= nStr1[d%10];
    }
    return(s);
}

```

自定义函数drawCld( )，在表格中显示公历和农历的日期以及相关节日。

```
var cld;
```



```

        else{lObj.innerHTML= cDay(cld[sD].lDay);} //显示
农历日
        varSlfw=Ssfw=null;
        s=cld[sD].solarFestival;
        for (var ipp=0;ipp<lFtv.length;ipp++){ //农历节日
            if (parseInt(lFtv[ipp].substr(0,2))==
(cld[sD].lMonth)){
                if (parseInt(lFtv[ipp].substr(2,4))==
(cld[sD].lDay)){
                    lObj.innerHTML=lFtv[ipp].substr(5);
                    Slfw=lFtv[ipp].substr(5);
                }
            }
            if (12==(cld[sD].lMonth)){ //判断是否为除夕
                if (eve==(cld[sD].lDay)){lObj.innerHTML="除
夕";Slfw="除夕";}
            }
        }
        for (var ipp=0;ipp<sFtv.length;ipp++){ //公历节日
            if (parseInt(sFtv[ipp].substr(0,2))==(SM+1)){
                if (parseInt(sFtv[ipp].substr(2,4))==(sD+1)){
                    lObj.innerHTML=sFtv[ipp].substr(5);
                    Ssfw=sFtv[ipp].substr(5);
                }
            }
        }
        if ((SM+1)==5){ //母亲节

```

```

    if (fat==0) {
        if ((sD+1)==7) {Ssfw="母亲节";lObj.innerHTML="母
亲节"}
    }
    else if (fat<9) {
        if ((sD+1)==((7-fat)+8)) {Ssfw="母亲
节";lObj.innerHTML="母亲节"}
    }
}
if ((SM+1)==6) { //父亲节
    if (mat==0) {
        if ((sD+1)==14) {Ssfw="父亲
节";lObj.innerHTML="父亲节"}
    }
    else if (mat<9) {
        if ((sD+1)==((7-mat)+15)) {Ssfw="父亲
节";lObj.innerHTML="父亲节"}
    }
}
if (s.length<=0) { //设置节气的颜色
    s=cld[sD].solarTerms;
    if(s.length>0) s= s.fontcolor('limegreen');
}
if(s.length>0) {lObj.innerHTML=s;S1fw=s;} //节气
if ((S1fw!=null)&&(Ssfw!=null)) {
    lObj.innerHTML=S1fw+ "/" +Ssfw;
}
}

```

```

    }
    else{//非日期
        sObj.innerHTML= '';
        lObj.innerHTML= '';
    }
}
}
}

```

自定义函数 `changeCld()`，在下拉列表中选择年或月时，调用自定义函数 `drawCld()` 显示公历和农历的相关信息。

```

function changeCld() {
    var y,m;
    y=CLD.SY.selectedIndex+1900;
    m=CLD.SM.selectedIndex;
    drawCld(y,m);
}

```

用自定义变量保存当前系统中的年、月、日。

```

var Today = new Date();
var tY = Today.getFullYear();
var tM = Today.getMonth();
var tD = Today.getDate();

```

自定义函数 `initial()`，打开网页时，在下拉列表中显示当前年月，并调用自定义函数 `drawCld()`，显示公历和农历的相关信息。

```

function initial() {
    CLD.SY.selectedIndex=tY-1900;
    CLD.SM.selectedIndex=tM;
    drawCld(tY,tM);
}

```

</SCRIPT>

(3) 在<body>标记的窗体载入 (onLoad) 事件中调用自定义函数 initial()。

<BODY onload=initial()>

举一反三

根据本实例，读者可以：

实现显示带农历的当天日期；

实现带备忘录及农历的日历。

## 19.3 购物车

电子商务系统中的购物车同实际生活中的购物车一样，都用于暂时保存挑选的商品。购物车主要包括所选商品的添加、查看购物车、单件商品购买数量的修改、从购物车移去指定商品、清空购物车这5部分。用户单击商品展台中的“购买”按钮，可以将对应的商品添加至购物车，购物车中将保存商品编号、商品名称、单价、购买数量、金额以及购物车内全部商品的合计金额。在查看购物车页面中，单击“修改订单”按钮，可以从购物车中移去指定商品或修改指定商品的购买数量，操作完成后，需要单击“更新订单”按钮保存所做的操作，单击“清空购物车”按钮，将退回购物车中的全部商品。本节将以宠物网站中的购物车为例介绍如何通过JavaScript实现购物车。

### 实例397 添加至购物车

本实例是一个实用性很强的程序

实例位置：光盘\mingrisoft\19\397

实例说明

在电子商务网站中，“添加至购物车”是客户端程序中非常关键的一个功能，主要用来帮助用户将选择的商品信息暂时保存到购物车中。运行本实例，如图19.5所示，在想要购买的宠物信息下方的“数量”文本框中输入购买数量，单击“购买”按钮，即可将选择的宠物信息添加到购物车中。



图19.5 添加至购物车

### 技术要点

本实例主要通过 JavaScript 在浏览器本地读取和设置 Cookie 实现。下面将介绍如何通过JavaScript读取和设置Cookie的方法。

- 读取Cookie，通过document 对象的cookie 属性可以读取Cookie 值，具体代码如下。

```
var cookieString=document.cookie;
```

- 设置 Cookie，一个完整的 Cookie 通常由“名字=值”、“expires=日期”、“path=路径”、“domain=域名”和“secure”5 组参数组成，其语法格式如下。

```
document.cookie="名字=值; expires=日期; path=路径;  
domain=域名; secure";
```

参数说明如下。

● 名字=值：“名字=值”是一个 Cookie 的最小量信息，这里的名字和值可以是任何字符串，但不能包含分号“;”，为了避免该问题，可以使用escape()函数清除字符串中的分号。

● expires=日期：用于指定浏览器Cookie保留的时间。如果没有指定有效期，那么Cookie将保持到浏览器被关闭。

● path=路径：用于指定一个 Cookie 可以被其他目录中的页面可用。默认情况下，Cookie可以被与创建该Cookie的文件在同一目录下的文件所使用。

● domain=域名：用于指定一个Cookie 可以被同一站点的其他 Web 服务可用。

● secure：用于告诉浏览器只有在与 Web 服务器的安全连接下此 Cookie 才能改变，这意味着服务器和浏览器必须支持 HTTPS（HTTPS 是指在 Internet 上传输加密信息的一个协议）。

实现过程

(1) 编写自定义函数 cart\_read ()，用于从 Cookie 中读取商品信息（这里为宠物信息）。readCookie ()函数是一个用于指定 Cookie 名称的参数，返回值为 Cookie 中的商品信息，代码如下。

```
<script language="javascript">  
function cart_read(cookieName) {  
    var search= cookieName+"=";  
    if (document.cookie.length>0) {  
        offset=document.cookie.indexOf(search);  
        if (offset !=-1) {  
            offset+= search.length;
```

```

        end=document.cookie.indexOf(";", offset);
        if (end==-1) {
            end=document.cookie.length;
            returnunescape(document.cookie.substring(offset,
end));
        }
    }else{
        return false;
    }
}else{
    return false;
}
}
</script>

```

(2) 编写自定义函数`cart_add ()`，用于将指定的商品信息（这里指宠物信息）保存到Cookie中，也就是购物车中，Cookie的有效期为30天。`writeCookie ()`函数包括4个参数，分别是`item_no`（用于指定商品编号）、`item_name`（用于指定商品名称）、`item_amount`（用于指定购买数量）和`item_price`（用于指定单价），该函数无返回值，代码如下。

```

<script language="javascript">
function
cart_add(item_no,item_name,item_amount,item_price) {
    var cookieString=document.cookie;
    if (cookieString.length>=2000) {
        alert("您的订单已满\n请结束此次订单操作后添加新订
单!");
    }
}

```

```

    }else
if(isNaN(item_amount) || item_amount<1 || item_amount.indexOf(
.')!==-1) {
    alert("数量输入错误! ");
}else{
    varmer_list=cart_read('petInfo');
    var today=newDate();
    var expires=newDate();
    expires.setTime(today.getTime()+1000*60*60*24*30);
    //有效期为30天
    var
item_detail="|" +item_no+"&" +item_name+"&" +item_amount+"&"
+item_price;
    if(mer_list==false) {
        document.cookie="petInfo="+escape(item_detail)+" ;ex
pires="+expires.toGMTString();
        alert(" “”+item_name+”” \n”+”已经加入您的订单! ");
    }else{
        if (mer_list.indexOf(item_no)!==-1) {
            alert(' 此商品您已添加\n请修改购买数量! ');
        }else{
            document.cookie="petInfo="+mer_list+escape(item
_detail)+" ;expires="+expires.toGMTString();
            alert(" “”+item_name+”” \n”+”已经加入您的订
单! ");
        }
    }
}

```

```
        window.location.href="cart.htm";
    }
}
</script>
```

(3) 在页面中添加单个商品信息（这里为宠物信息），并在代表“购买”按钮图片的onClick事件中调用cart\_add()函数，代码如下。

```
<tablewidth="75%"
border="0"cellpadding="0"cellspacing="0">
    <tr>
        <tdheight="95"colspan="2" align="center">
            
        </td>
    </tr>
    <tr>
        <tdheight="18"colspan="2" align="center">旺旺星</td>
    </tr>
    <tr>
        <tdheight="18"colspan="2" align="center"
style="text-decoration:line-through;color:#FF0000">原
价：750</td>
    </tr>
    <tr>
        <tdheight="18"colspan="2" align="center">现价：
600</td>
    </tr>
</table>
```

```

        <tdwidth="48%"height="31"align="center">
        数量: <input type="text" name="sl0001"
size="4" value="1" maxlength="5"></td>
        <tdwidth="52%" align="center">
        </td>
    </tr>
</table>

```

(4) 根据步骤(2)的方法依次添加其他商品。

举一反三

根据本实例，读者可以：

利用购物车原理实现批量添加商品信息；

利用购物车原理实现批量商品销售。

## [实例398 查看购物车](#)

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\19\398

实例说明

所谓查看购物车是指将保存在Cookie中的商品信息显示在页面中。在本节介绍的购物车模块中有两种方法可以进入到查看购物车页面，一种是通过单击“购买”按钮，在将商品信息添加到购物车后，进入到查看购物车页面；另一种是通过导航条中的“查看购物车”超级链接进入到查看购物车页面。查看购物车页面的运行结果如图19.6所示。



图19.6 查看购物车

### 技术要点

在制作本实例时，首先需要通过自定义的JavaScript函数 `cart_read()` 读取Cookie中保存的商品信息，然后再通过document对象的 `write()` 方法将获取的商品信息输出到页面中。其中，自定义函数 `cart_read()` 的代码请参见实例397的实现过程。

### 实现过程

(1) 在页面的合适位置添加用于保存购物车中商品信息的表单及表单元素，关键代码如下。

```
<form name="cartForm" method="post" action="">
<input type="hidden" name="cart" id="cart" size="100">
</form>
```

(2) 在步骤(1)创建的表单中加入以下代码，用于获取Cookie中的商品信息（这里为宠物信息），并通过document对象的 `write()` 方法在页面中以表格的形式输出。

```
<script language="JavaScript">
```

```

var the_list=unescape(cart_read('petInfo'));
if (the_list=="false"||the_list.indexOf("|")==-1) {
    window.location.href="cart_empty.htm";
}else{
    cartForm.cart.value=the_list;
    var totals;
    varbroken_list= the_list.split("|");
    for (i=1;i<broken_list.length;i++) {
        var single_list=broken_list[i];
        varbroken_single_list= single_list.split("&");
        var subtotals=broken_single_list[2]*
(broken_single_list[3]*100)/100;
        if (totals==null) {
            totals=subtotals;
        }else{
            totals=totals*100+subtotals*100;
            totals=totals/100;
        }
        document.write("<table
id=' Table"+broken_single_list[0]+"'' width=' 85%' border=' 1' c
ellspacing=' 0' cellpadding=' 0' bordercolor=' #FFFFFF' borderc
olordark=' #FFFFFF' bordercolorlight=' #FB9914' class=' td_car
t'>")
        document.write("<tr>");
        document.write("<tdwidth=' 46%'>" +broken_single_list[1]+"</td>");

```

```

        document.write("<tdwidth=' 14%'>" + broken_single_list[3] + "</td>");
        document.write("<tdwidth=' 12%'>" + "<input
id=' aid'" + broken_single_list[0] + "'
type=' text' name=' textfield' value='" + broken_
single_list[2] + "' size=' 5' maxlength=' 5' disabled
onKeyUp=modifySubTotal
(' '" + broken_single_list[0] + "'", this.value, ' '" + broken _single
_list [3] + "' )
onchange=modifyTotal(' '" + broken_single_list[0] + "'", this.val
ue)>" + "</td>");
        document.write("<td width=' 20%'><input type=' text'
size=' 10'
id=' id'" + broken_single_list[0] + "' style=' border:medium
none; ' readonly value='" + subtotals + "'></td>");
        document.write("<td width=' 8%'><input type=' button'
name=' Submit' class=' btn_grey' value=' 删除' disabled
onClick=Delete(Table" + broken_single_list[0] + ", ' '" + broken_s
ingle_list[0] + "' )></td>");
        document.write("</tr></table>");
    }
}
</script>

```

(3) 在页面的合适位置加入“合计金额”文本框，并通过JavaScript 为其赋值，关键代码如下。

```

<input type="text" id="total" name="total" size="10"
readonly style="border: medium none;" >

```

```
script language="JavaScript">  
    cartForm.total.value=totals;  
</script>
```

举一反三

根据本实例，读者可以：

以其他格式显示Cookie中的商品信息；

显示保存到Cookie中的任何信息。

## 实例399 修改商品购买数量

本实例是一个提高数据准确性的程序

实例位置：光盘\mingrisoft\19\399

实例说明

为了满足不同用户的需求，购物车中还需要加入修改指定商品购买数量的功能。运行本实例，单击“修改订单”按钮后，页面中的“数量”文本框将变为可用状态，同时“修改订单”按钮将变为“更新订单”按钮，如图19.7所示。在想要修改购买数量的商品的“数量”文本框中输入相应的数量，单击“更新订单”按钮即可修改该商品的购买数量。

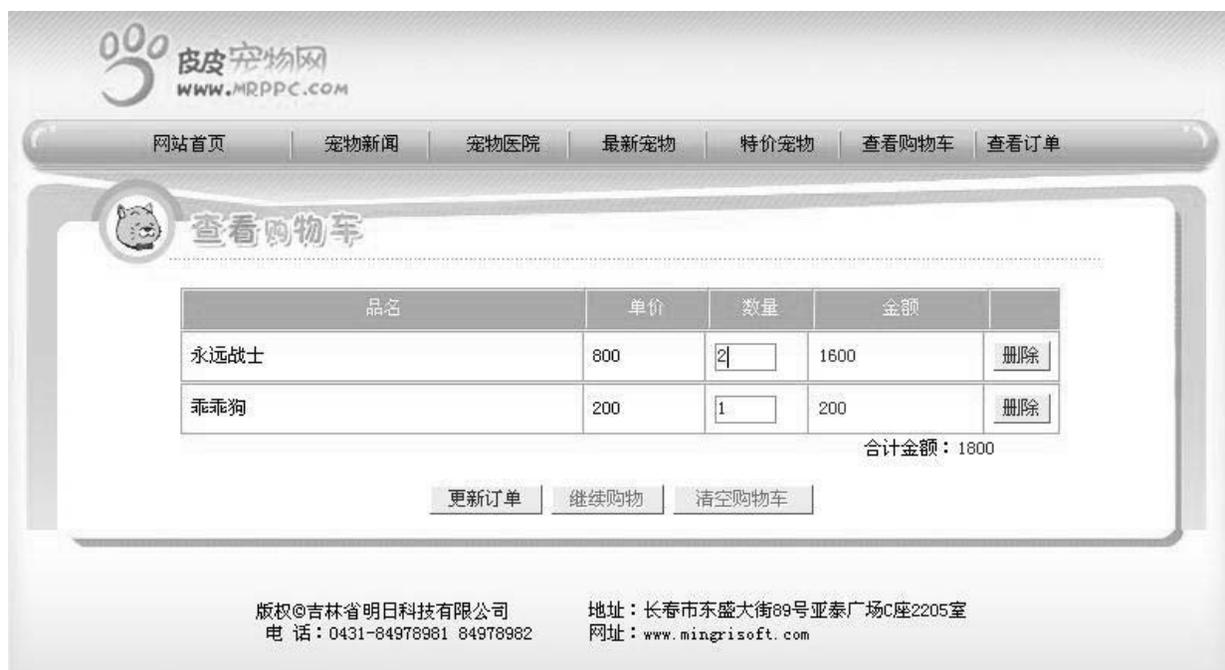


图19.7 修改商品购买数量

### 技术要点

在实现本实例时，首先需要将“修改订单”按钮的值修改为“更新订单”或是将“更新订单”按钮的值修改为“修改订单”，然后再通过表单元素的disabled属性控制相应表单元素是否可用，最后重新保存购物车中的商品信息。

### 实现过程

(1) 编写自定义的JavaScript函数modifyCart(), 用于修改订单信息，关键代码如下。

```
<script language="JavaScript">
functionmodifyCart() {
    var limit=cartForm.elements.length-5;
    if (cartForm.update.value=="修改订单") {
        cartForm.update.value="更新订单";
        for (i=1;i<limit;i++) {
            cartForm.elements[i].disabled=false;
        }
    }
}
```

```

        i=i+2;
        cartForm.elements[i].disabled=false;
    }
    cartForm.continueBuy.disabled=true;
    cartForm.clearCart.disabled=true;
}else{
    cartForm.update.value="修改订单"
    for (i=1;i<limit;i++){
        cartForm.elements[i].disabled=true;
        i=i+2;
        cartForm.elements[i].disabled=true;
    }
    cartForm.continueBuy.disabled=false;
    cartForm.clearCart.disabled=false;
    window.location.reload();
var today=newDate();
var expires=newDate();
    expires.setTime(today.getTime()+1000*60*60*24*30);    /
/有效期为30天
    varvalues=cartForm.cart.value
    document.cookie="petInfo="+escape(values)+";expires="+e
xpires.toGMTString();
    }
}
</script>

```

(2) 在实例398中介绍的查看购物车页面的适当位置加入“修改订单”按钮，并在其onClick事件中调用modifyCart()函数修改订单信

息，关键代码如下。

```
<input name="Button" type="button" class="btn_grey"
id="update" onClick=" modifyCart ()" value="修改订单">
```

举一反三

根据本实例，读者可以：

修改销售单中的销售数量；

修改采购单中的采购数量。

## 实例400 从购物车中移去指定商品

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft19\400

实例说明

在购物过程中，经常会遇到将已经选择的商品退回到货物架上的情况，因此在开发购物车模块时，必须考虑到这一点。下面将介绍如何从购物车中移去指定商品。运行本实例，单击“修改订单”按钮后，页面中的“删除”按钮将变为可用状态，同时“修改订单”按钮将变为“更新订单”按钮，如图19.8所示。单击指定商品后面的“删除”按钮，即可将该商品退回到货物架上，但是此时并没有真正从购物车中移去，还需要单击“更新订单”按钮保存所做的修改。

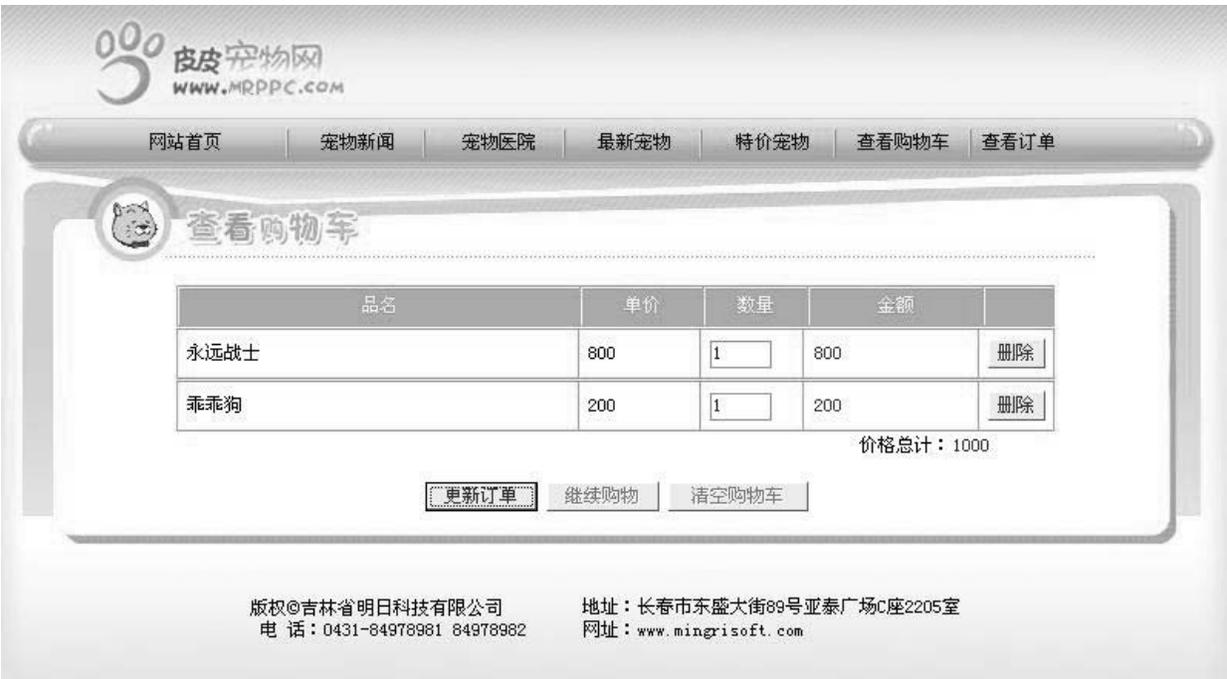


图19.8 从购物车中移去指定商品

### 技术要点

实现本实例的基本思路如下：由于在实现将指定商品添加到购物车时，已经将各个商品信息组合成一个字符串并保存在隐藏的表单元素cart中，并且各个商品信息之间通过“|”进行分隔，所以在实现删除指定的商品信息时，需要通过String对象的split()方法将各个商品信息分析并保存到数组中，再从该数组中删除指定的商品信息，同时删除保存该商品信息的表格。

### 实现过程

(1) 编写自定义的JavaScript函数Delete()，用于删除指定的商品信息，该函数包括两个参数，分别是TableID（用于指定要删除的商品所在的表格的ID）和id（用于指定商品编号），无返回值，关键代码如下。

```
<script language="JavaScript">
functionDelete(TableID, id) {
    var confirm_delete=window.confirm("确定要删除吗？")
```

```

if (confirm_delete) {
    vardeletedList=cartForm.cart.value;
    var split_deletedList=deletedList.split("|");
    for (i=1;i<split_deletedList.length;i++) {
        if(split_deletedList[i].indexOf(id)!=-1) delete
split_deletedList[i];
    }
    varnew_deletedList="";
    for (i=1;i<split_deletedList.length;i++) {
        if (split_deletedList[i]!=undefined) {
            new_deletedList=new_deletedList+"|"+split_deleted
List[i];
        }
    }
    cartForm.cart.value=new_deletedList;
    TableID.deleteRow();
}
}
</script>

```

(2) 用于删除指定商品信息的“删除”按钮，在实例398中自动生成，在生成的“删除”按钮的onClick 事件中调用Delete ()函数，用于删除指定的商品。

举一反三

根据本实例，读者可以：

将销售单中的商品信息删除；

将退货单中的商品信息删除。

## 实例401 清空购物车

本实例是一个提高效率的程序

实例位置：光盘\mingrisoft\19\401

实例说明

在前面的实例中已经介绍了如何从购物车中移去指定商品，下面将介绍如何将购物车中的全部商品一次性退回，即清空购物车。运行本实例，单击“清空购物车”按钮，即可清空购物车中的全部商品，并重定向到购物车为空页面，如图19.9所示。



图19.9 清空购物车

技术要点

在制作本实例时，首先需要通过自定义的JavaScript函数 `cart_read()` 读取Cookie中保存的商品信息，然后将Cookie的有效期设置为当前时间-1，这样可以让Cookie立即过期，即删除了指定的Cookie，从而达到清空购物车的目的。

说明：自定义函数`cart_read()`的代码请参见实例397的实现过程。

### 实现过程

(1) 编写自定义的 JavaScript 函数 `cart_clear()`，用于删除购物车中全部商品，关键代码如下。

```
<script language="JavaScript">
function cart_clear() {
var mer_list=cart_read('petInfo');
var expires = new Date();
expires.setTime(expires.getTime()-1);
document.cookie="petInfo="+mer_list+";expires="+expires.toGMTString();
window.location.href="cart_empty.htm";
}
</script>
```

(2) 在实例 398 所介绍的查看购物车页面的适当位置加入“清空购物车”按钮，并在其`onClick`事件中调用`cart_clear()`函数清空购物车，关键代码如下。

```
<input name="clearCart" type="button" class="btn_grey"
id="clearCart" onClick="cart_clear()" value="清空购物车">
```

### 举一反三

根据本实例，读者可以：

删除保存在Cookie中的采购单信息；

删除保存在Cookie中的销售单信息。

## 第20章 jQuery应用开发

jQuery基础开发

jQuery表格应用

jQuery在实际中的应用

## 20.1 jQuery基础开发

### 实例402 获取文本输入框中输入的值

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\402

实例说明

在页面中添加一个 ID 属性值为 testInput 的文本输入框和一个按钮，通过单击按钮来获取在文本输入框中输入的值。在IE浏览器中运行本实例，在文本框中输入“JavaScript”，如图20.1所示，单击“输入的值”按钮，将弹出提示对话框显示输入的文字，如图20.2所示。

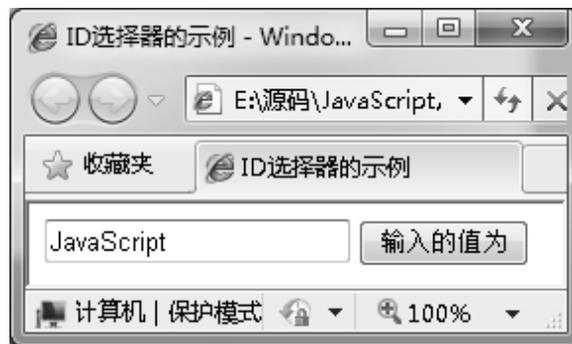


图20.1 在文本框中输入文字



图20.2 弹出的提示对话框

### 技术要点

本实例用到了jQuery中的ID选择器。

ID选择器#id顾名思义就是利用DOM元素的id属性值来筛选匹配的元素，并以jQuery包装集的形式返回给对象。这就像一个学校中每个学生都有自己的学号一样，学生的姓名是可以重复的，但学号却是不可以的，因此，可以根据学生的学号获取指定学生的信息。

ID选择器的使用方法如下。

```
$("#id");
```

其中，id为要查询元素的ID属性值。例如，要查询ID属性值为user的元素，可以使用下面的jQuery代码：

```
$("#user");
```

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个 ID属性值为 testInput 的文本输入框和一个按钮，代码如下。

```
<input type="text" id="testInput" name="test" value="" />  
<input type="button" value="输入的值为" />
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现单击按钮来获取在文本输入框中输入的值，代码如下。

```
<script type="text/javascript">  
    $(document).ready(function() {  
        $("input[type='button']").click(function()  
        {  
            //为按钮绑定单击事件
```

```
        var inputValue=$("#testInput").val();           //获  
取文本输入框的值  
        alert(inputValue);  
    });  
});  
</script>
```

举一反三

根据本实例，读者可以：

获取按钮上的文本；

获取Image中的图片链接地址。

### 实例403 动态修改<div>标记内容

这是一个可以启发思维的实例

实例位置：光盘\mingrisoft\20\403

实例说明

在页面中添加两个<div>标记和一个按钮，通过单击按钮来获取这两个<div>，并修改它们的内容。在IE浏览器中运行本实例，首先显示图20.3所示的页面，单击“若干年后”按钮，将显示图20.4所示的页面。



图20.3 单击按钮前



图20.4 单击按钮后

### 技术要点

本实例用到了jQuery中的元素选择器。

元素选择器是根据元素名称匹配相应的元素。通俗地讲，元素选择器指向的是DOM元素的标记名，也就是说元素选择器是根据元素的标记名选择的。可以把元素的标记名理解成学生的姓名，在一个学校中可能有多个姓名为“刘伟”的学生，但是姓名为“吴语”的学生也

许只有一个，所以通过元素选择器匹配到的元素可能有多个，也可能是一个。多数情况下，元素选择器匹配的是一组元素。

元素选择器的使用方法如下。

```
$("#element");
```

其中，element为要查询元素的标记名。例如，要查询全部div元素，可以使用下面的jQuery代码。

```
$("#div");
```

实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加两个<div>标记和一个按钮，代码如下。

```
<div>这里种植了一棵草莓</div>
```

```
<div>这里养殖了一条鱼</div>
```

```
<input type="button" value="若干年后"/>
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现单击按钮获取全部<div>元素，并修改它们的内容，具体代码如下。

```
<script type="text/javascript">
```

```
$(document).ready(function() {
```

```
    $("#input[type='button']").click(function()
```

```
    { //为按钮绑定单击事件
```

```
        $("#div").eq(0).html("<img
```

```
src='images/strawberry1.jpg'/>这里长出了一片草
```

```
莓"); //获取第一个div元素
```

```
    $("div").get(1).innerHTML="<img  
src='images/fish1.jpg' />这里的鱼没有了"; //获取第二个div  
元素  
    });  
});  
</script>
```

举一反三

根据本实例，读者可以：

动态修改按钮的文本值；

动态改变<div>标记input值。

## 实例404 通过类名选择器选择元素并更改样式

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\404

实例说明

在页面中，首先添加两个<div>标记，并为其中的一个设置CSS类，然后通过jQuery的类名选择器选取设置了CSS类的<div>标记，并设置其CSS样式。在IE浏览器中运行本示例，将显示图20.5所示的页面。其中，左面的<div>为更改样式后的效果，右面的<div>为默认的风格。由于使用了\$(document).ready()方法，所以选择元素并更改样式在DOM元素加载就绪时就已经自动执行完毕。



图20.5 通过类名选择器选择元素并更改样式

### 技术要点

本实例用到了jQuery中的类名选择器。

类名选择器是通过元素拥有的CSS类的名称查找匹配的DOM元素。在一个页面中，一个元素可以有多个CSS类，一个CSS类又可以匹配多个元素，如果有元素中有一个匹配的类的名称就可以被类名选择器选取到。

类名选择器很好理解，在大学的时候大部分人一定都选过课，可以把CSS类名理解为课程名称，元素理解成学生，学生可以选择多门课程，而一门课程又可以被多名学生所选择。CSS类与元素的关系既可以是多对多的关系，也可以是一对多或多对一的关系。简单地说，类名选择器就是以元素具有的CSS类名称查找匹配的元素。

类名选择器的使用方法如下。

```
$(".class");
```

其中，class 为要查询元素所用的CSS类名。例如，要查询使用CSS类名为 word\_orange 的元素，可以使用下面的jQuery代码：

```
$("#word_orange");
```

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加两个<div>标记，一个使用CSS类 myClass，另一个不设置CSS类，代码如下。

```
<div class="myClass">注意观察我的样式</div>
<div>我的样式是默认的</div>
```

注意：这里添加了两个<div>标记是为了对比效果，默认的背景颜色都是蓝色的，文字颜色都是黑色的。

(3) 在引入JQuery库的代码下方编写JQuery代码，实现按CSS类名选取DOM元素，并更改其样式（这里更改了背景颜色的文字颜色），具体代码如下。

```
<script type="text/javascript">
    $(document).ready(function() {
        varmyClass=$( ".myClass" );           //选取DOM元素
        myClass.css("background-color", "#C50210"); //为选取的DOM元素设置背景颜色
        myClass.css("color", "#FFF");         //为选取的DOM元素设置文字颜色
    });
</script>
```

举一反三

根据本实例，读者可以：

选中指定的<div>标记；

为HTML页面中的不同<div>设置不同的颜色。

## [实例405 隐藏超级链接地址](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\405

实例说明

通常情况下，在网页中，如果将鼠标移动到一个超级链接上时，在浏览器的地址栏中将显示该超级链接所指向的链接地址。在某种程度上，上述情况将威胁到网站的安全。为了网站的安全，可以将超级链接的链接地址隐藏。下面将通过一个具体的例子来介绍如何应用jQuery实现隐藏超级链接地址。运行本实例，将鼠标移动到“登录”、“注册”和“找回密码”超级链接上时，状态栏将显示http://www.mrbccd.com，如图20.6所示。将鼠标移出超级链接，状态栏中将显示文字“完成”。将鼠标移动到“首页”超级链接上，状态栏将显示该超链接的目标地址，如图20.7所示。将鼠标移出该超级链接，状态栏中将显示文字“完成”。



图20.6 鼠标移到“登录”超级链接上的效果



图20.7 鼠标移到“首页”超级链接上的效果

(1) 创建一个名称为index.html的文件，并在该文件中应用DIV+CSS样式进行页面布局，技术要点

本实例主要用到了jQuery中的元素选择器和类名选择器，关于元素选择器的讲解请参见实例403，关于类名选择器的讲解请参见实例404。

实现过程

并添加4个超级链接，关键代码如下。

```
<div style="float:right;text-align: right;">
  <a href="index.html">首页</a>
  &nbsp; | &nbsp; &nbsp;<a href="index.html" class="main">登录
</a>
  &nbsp; | &nbsp; &nbsp;<a href="index.html" class="main">注册
</a>
  &nbsp; | &nbsp; &nbsp;<a href="index.html" class="main">找回密码
</a>
```

</div>

布局后的效果如图20.8所示。



图20.8 布局后的效果

(2) 在index.html文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(3) 在引入 jQuery库的代码下方编写 jQuery代码，实现绑定鼠标移到或移出超级链接事件，并为状态栏设置不同的显示文档，具体代码如下。

```
<script type="text/javascript">
$(document).ready(function() {
    $("a.main").mouseover(function() { //
绑定鼠标移到超级链接事件
        window.status="http://www.mrbccd.com";return
true; //设定状态栏文本
    }).mouseout(function() { //绑
定鼠标移出超级链接事件
```

```
        window.status="完成";return true;           //设  
定状态栏文本  
    });  
});  
</script>
```

举一反三

根据本实例，读者可以：

在购物网站中隐藏商品的超链接地址；

隐藏带有用户信息的超链接地址。

## 实例406 为版权列表设置样式

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\406

实例说明

本实例通过 jQuery 为版权列表设置样式，运行本示例将显示图 20.9 所示的效果，其中上面的版权信息是通过 jQuery 添加样式的效果，下面的版权信息为默认的效果。

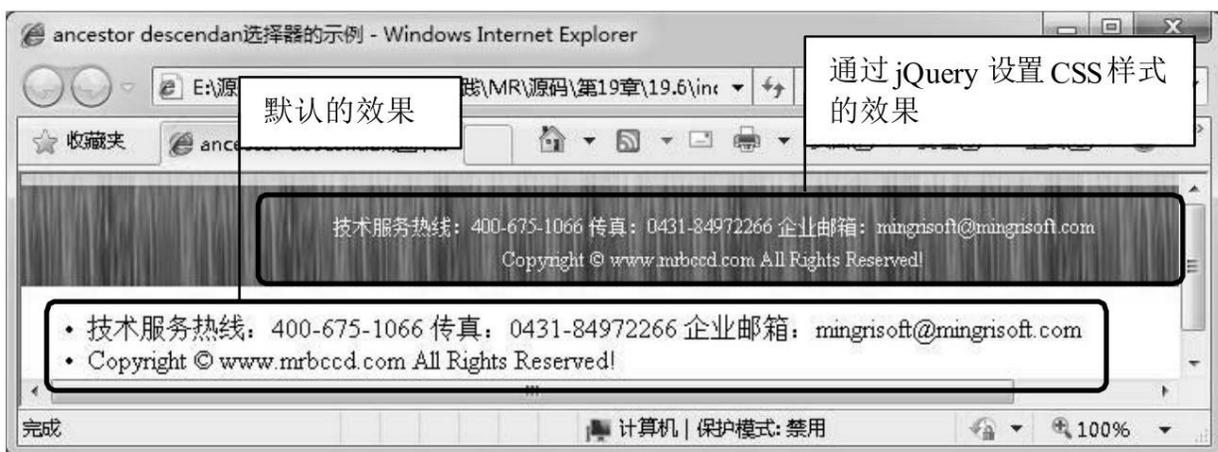


图20.9 通过jQuery为版权列表设置样式

技术要点

本实例用到了jQuery 中的ancestor descendant 选择器。

ancestor descendant 选择器中的 ancestor 代表祖先，descendant 代表子孙，用于在给定的祖先元素下匹配所有的后代元素。ancestor descendant 选择器的使用方法如下。

```
$("#ancestor descendant");
```

- ancestor: 指任何有效的选择器。

- descendant: 用以匹配元素的选择器，并且它是ancestor 所指定元素的后代元素。

例如，要匹配ul元素下的全部li元素，可以使用下面的jQuery代码。

```
$("#ul li");
```

实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，首先添加一个<div>标记，并在该<div>标记内添加一个<ul>标记及其子标记<li>，然后在<div>标记的后面再添加一个<ul>标记及其子标记<li>，代码如下。

```
<div id="bottom">
<ul>
<li>技术服务热线：400-675-1066 传真：0431-84972266 企业
邮箱：mingrisoft@mingrisoft.com</li>
<li>Copyright&copy;www.mrbccd.comAllRightsReserved!
</li>
</ul>
</div>
```

```
<ul>
  <li>技术服务热线：400-675-1066 传真：0431-84972266 企业
  邮箱：mingrisoft@mingrisoft.com</li>
  <li>Copyright&copy;www.mrbccd.comAllRightsReserved!
</li>
</ul>
```

(3) 编写CSS样式，通过ID选择符设置<div>标记的样式，并且编写一个类选择符Copyright，用于设置<div>标记内的版权列表的样式，关键代码如下。

```
<style type="text/css">
#bottom{
  background-image:url(images/bg_bottom.jpg);      /*设置背景*/
  width:800px;                                     /*设置宽度*/
  height:58px;                                     /*设置高度*/
  clear:both;                                     /*设置左右两侧无浮动内容*/
  text-align:center;                               /*设置居中*/
  padding-top:10px;                                /*设置顶边距*/
  font-size:9pt;                                  /*设置字体大小*/
}
.copyright{
```

```

        color:#FFFFFF;                /*设置文字
颜色*/
        list-style:none;              /*不显示
项目符号*/
        line-height:20px;            /*设置行高
*/
    }
</style>

```

(4) 在引入jQuery库的代码下方编写jQuery代码，匹配div元素的子元素u1，并为其添加CSS样式，具体代码如下。

```

<script type="text/javascript">
$(document).ready(function() {
    $("divu1").addClass("copyright"); //为div
元素的子元素u1添加样式
});
</script>

```

举一反三

根据本实例，读者可以：

为商品列表设置指定的样式；

为同一个<div>中的所有元素设置统一的样式。

## [实例407 为表单的直接子元素input换肤](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\407

实例说明

本实例主要实现为表单的直接子元素input换肤的功能。运行本实例将显示图20.10所示的效果。单击“换肤”按钮，将显示图 20.11所示的效果。单击“恢复默认”按钮，将再次显示图20.10所示的效果。



图20.10 默认的效果



图20.11 单击“换肤”按钮之后的效果

在图20.11中，虽然“换肤”按钮也是form元素的子元素input，但由于该元素不是form元素的直接子元素，所以在执行换肤操作时，

该按钮的样式并没有改变。如果将步骤（4）中的第4行和第7行的代码中的\$("form > input")修改为\$("form input")，那么单击“换肤”按钮后，将显示图20.12所示的效果，即“换肤”按钮也将被添加CSS类。这也就是parent > child选择器和ancestor descendant选择器的区别。



图20.12 为“换肤”按钮添加CSS类的效果

### 技术要点

本实例用到了jQuery 中的parent > child 选择器。

parent > child 选择器中的parent 代表父元素，child 代表子元素，用于在给定的父元素下匹配所有的子元素。使用该选择器只能选择父元素的直接子元素。parent > child 选择器的使用方法如下。

```
$("parent > child");
```

- parent: 指任何有效的选择器。
- child: 用以匹配元素的选择器，它既是匹配元素的选择器，也是parent 元素的子元素。

例如，要匹配表单中所有的子元素input，可以使用下面的jQuery 代码。

```
$("#form > input");
```

## 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个表单，并在该表单中添加6个input元素，并且将“换肤”按钮用<span>标记括起来，关键代码如下。

```
<form id="form1" name="form1" method="post" action="">
  姓       名: <input type="text" name="name"
id="name"/>
  <br/>
  籍       贯: <input name="native" type="text"
id="native"/>
  <br/>
  生       日: <input type="text" name="birthday"
id="birthday"/>
  <br/>
  E-mail: <input type="text" name="email" id="email"/>
  <br/>
  <span>
    <input type="button" name="change" id="change" value="换
肤"/>
  </span>
  <input type="button" name="default"
id="default" value="恢复默认"/>
```

```
<br/>
```

```
</form>
```

(3) 编写CSS样式，用于指定input元素的默认样式，并且添加一个用于改变input元素样式的CSS类，具体代码如下。

```
<style type="text/css">
input {
    margin:5px;                /*设置input元素的外边
距为5像素*/
}
.input {
    font-size:12pt;           /*设置文字大小*/
    color:#333333;           /*设置文字颜色*/
    background-color:#cef;    /*设置背景颜色*/
    border:1px solid#000000;   /*设置边框*/
}
</style>
```

(4) 在引入 jQuery库的代码下方编写 jQuery代码，实现匹配表单元素的直接子元素并为其添加和移除CSS样式，具体代码如下。

```
<script type="text/javascript">
$(document).ready(function() {
    $("#change").click(function() {                //绑定“换
肤”按钮的单击事件
        $("form> input").addClass("input");        //为表单元素
的直接子元素input添加样式
    });
    $("#default").click(function() {                //绑定“恢
复默认”按钮的单击事件
```

```
    $("form> input").removeClass("input"); //移除为表单
    元素的直接子元素input添加的样式
  });
});
</script>
```

举一反三

根据本实例，读者可以：

为<div>中的所有input换肤；

为<div>中的所有button按钮换肤。

## 实例408 改变指定元素的背景颜色为淡蓝色

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\408

实例说明

本实例主要筛选紧跟在<label>标记后的<p>标记，并改变匹配元素的背景颜色为淡蓝色。运行本实例将显示图20.13所示的效果。在图中可以看到“第一个p”和“第二个p”的段落被添加了背景，而“div外面的p”由于不是label元素的同级元素，所以没有被添加背景。

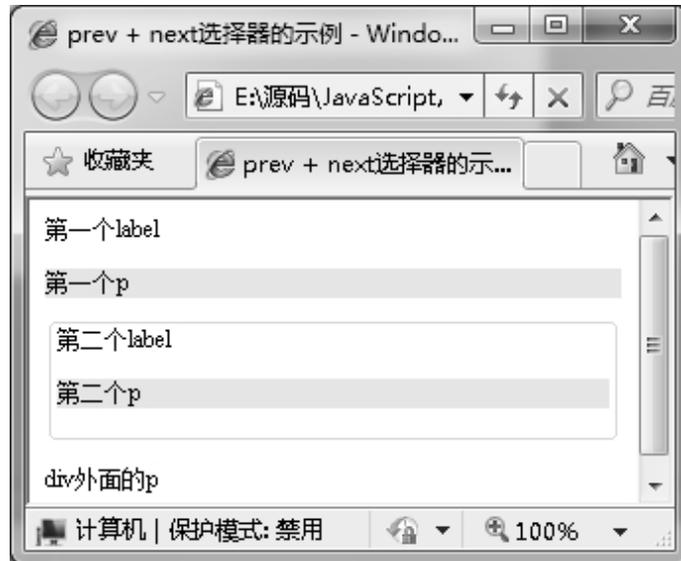


图20.13 将label元素的同级元素p的背景设置为淡蓝色

### 技术要点

本实例用到了jQuery中的prev+next 选择器。

prev+next 选择器用于匹配所有紧接在prev 元素后的next 元素。其中，prev 和next 是两个相同级别的元素。prev+next 选择器的使用方法如下。

```
$("#prev+next");
```

- prev: 指任何有效的选择器。
- next: 一个有效选择器并紧接着prev 选择器。

例如，要匹配<div>标记后的<img>标记，可以使用下面的jQuery 代码。

```
$("#div+img");
```

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，首先添加一个<div>标记，并在该<div>标记中添加两个<label>标记和<p>标记，其中第二对<label>标记和<p>标记用<fieldset>括起来，然后在<div>标记的下方再添加一个<p>标记，关键代码如下。

```
<div>
  <label>第一个label</label>
  <p>第一个p</p>
  <fieldset>
    <label>第二个label</label>
    <p>第二个p</p>
  </fieldset>
</div>
<p>div外面的p</p>
```

(3) 编写CSS样式，用于设置body元素的字体大小，并且添加一个用于设置背景的CSS类，具体代码如下。

```
<style type="text/css">
  .background {background:#cef}
  body {font-size:12px;}
</style>
```

(4) 在引入jQuery库的代码下方编写jQuery代码，实现匹配label元素的同级元素p，并为其添加CSS类，具体代码如下。

```
<script type="text/javascript" charset="GBK">
  $(document).ready(function() {
    $("label+p").addClass("background"); //为匹配的元
    素添加CSS类
  });
</script>
```

举一反三

根据本实例，读者可以：

设置button元素的背景颜色为橙红色；

设置密码文本框的背景颜色为浅绿色。

## 实例409 筛选页面中div元素的同辈元素

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\3409

实例说明

本实例主要实现筛选页面中 div 元素的同辈元素的功能。运行本实例将显示图 20.14 所示的效果。在图中可以看到“div外面的p”被添加了背景，而“第一个p”和“第二个p”的段落由于不是div元素的同辈元素，所以没有被添加背景。

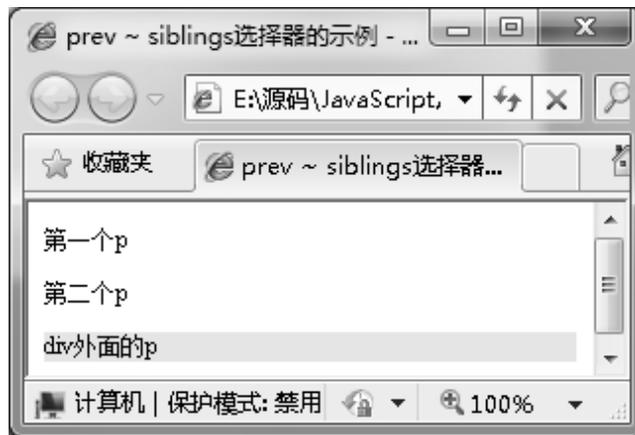


图20.14 为div元素的同辈元素设置背景

技术要点

本实例用到了jQuery中的prev~siblings 选择器。

prev~siblings 选择器用于匹配prev 元素之后的所有siblings 元素。其中，prev 和siblings是两个同辈元素。prev~siblings 选择器的使用方法如下。

```
$("#prev~siblings");
```

- prev: 指任何有效的选择器。
- siblings: 一个有效选择器并紧接着prev 选择器。

例如，要匹配div元素的同辈元素ul，可以使用下面的jQuery代码。

```
$("#div~ul");
```

实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，首先添加一个<div>标记，并在该<div>标记中添加两个<p>标记，然后在<div>标记的下方再添加一个<p>标记，关键代码如下。

```
<div>
  <p>第一个p</p>
  <p>第二个p</p>
</div>
<p>div外面的p</p>
```

(3) 编写CSS样式，用于设置body元素的字体大小，并且添加一个用于设置背景的CSS类，具体代码如下。

```
<style type="text/css">
  .background{background:#cef}
  body{font-size:12px;}
</style>
```

(4) 在引入jQuery库的代码下方编写jQuery代码，实现匹配div元素的同辈元素p，并为其添加CSS类，具体代码如下。

```
<script type="text/javascript" charset="GBK">
    $(document).ready(function() {
        $("div~p").addClass("background");           //为匹配
        的元素添加CSS类
    });
</script>
```

举一反三

根据本实例，读者可以：

获取密码文本框的所有同级元素；

获取登录列表的所有同级<div>标记。

## 实例410 获取页面上隐藏和显示的input元素值

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\410

实例说明

本实例主要实现获取页面上隐藏和显示的input元素值的功能。运行本实例，将显示图20.15所示的效果。



图20.15 弹出隐藏和显示的input元素的值

技术要点

本实例用到了jQuery中的可见性过滤器。

元素的可见状态有两种，分别是隐藏状态和显示状态。可见性过滤器就是利用元素的可见状态匹配元素的。因此，可见性过滤器也有两种，一种是匹配所有可见元素的:visible过滤器，另一种是匹配所有不可见元素的:hidden过滤器。

注意：在应用:hidden过滤器时，display属性是none以及input元素的Type属性为hidden的元素都会被匹配到。

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加3个input元素，其中第一个为显示的文本框，第二个为不显示的文本框，第3个为隐藏域，关键代码如下。

```
<input type="text" value="显示的input元素">
<input type="text" value="我是不显示的input元素"
style="display:none">
<input type="hidden" value="我是隐藏域">
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现匹配div元素的同辈元素p，并为其添加CSS类，具体代码如下。

```
<script type="text/javascript">
$(document).ready(function() {
    var
    visibleVal=$( "input:visible" ).val();           //取得
    显示的input的值
    var
    hiddenVal1=$( "input:hidden:eq(0)" ).val();     //取得隐
```

藏的input的值

```
var  
hiddenVal2=$( "input:hidden:eq(1)").val(); //取得隐  
藏的input的值  
    alert(visibleVal+"\n\r"+hiddenVal1+"\n\r"+hiddenVal2)  
; //弹出取得的信息  
});  
</script>
```

举一反三

根据本实例，读者可以：

在考试系统中获取隐藏的单选题答案；

在学生管理系统中获取只读的学生编号值。

## 实例411 获取和设置元素的文本内容与HTML内容

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\411

实例说明

本实例主要使用 jQuery 技术获取和设置元素的文本内容与 HTML 内容。运行本实例将显示图20.16所示的运行结果。从该运行结果中，我们可以看出，在应用 text() 设置文本内容时，即使内容中包含 HTML 代码，也将被认为是普通文本，并不能作为 HTML 代码被浏览器解析，因此，应用html() 设置的HTML内容中包括的HTML代码就可以被浏览器解析。

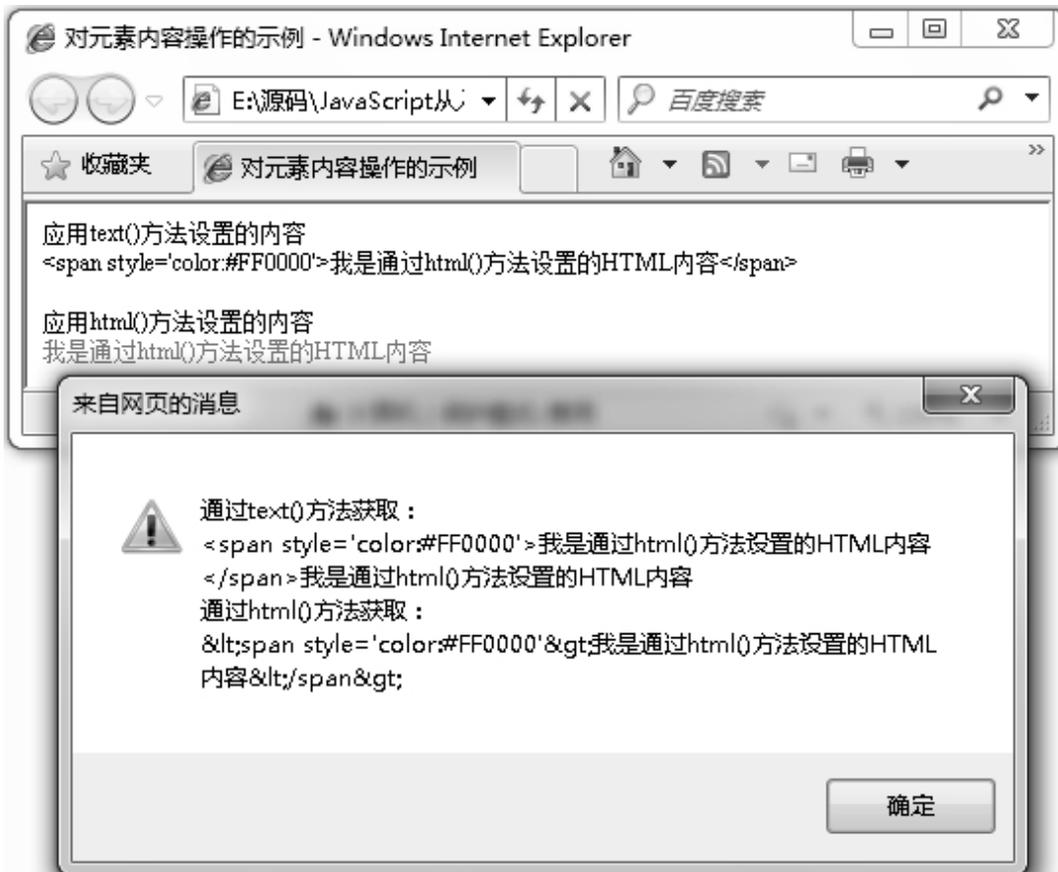


图20.16 获取和设置元素的文本内容与HTML内容

### 技术要点

本实例实现时，主要是使用 jQuery对 HTML的元素内容进行操作。对元素内容的操作可以分为对文本内容操作和对HTML内容进行操作。下面分别进行详细介绍。

#### 1. 对文本内容操作

jQuery提供了text()和text(val)两个方法用于对文本内容操作。其中，text()用于获取全部匹配元素的文本内容，text(val)用于设置全部匹配元素的文本内容。例如，在一个HTML页面中，包括下面3行代码。

```
<div>
```

```
<span id="clock">当前时间：2011-07-06 星期三
```

```
13:20:10</span>
```

```
</div>
```

要获取div元素的文本内容，可以使用下面的代码。

```
$("#div").text();
```

得到的结果为：当前时间：2012-08-12 星期三 13:20:10。

注意：text()方法取得的结果是所有匹配元素包含的文本组合起来的文本内容，这个方法也对XML文档有效，可以用text()方法解析XML文档元素的文本内容。

要重新设置div元素的文本内容，可以使用下面的代码。

```
$("#div").text("我是通过text()方法设置的文本内容");
```

这时，再应用“\$("#div").text();”获取div元素的文本内容时，将得到以下内容。

我是通过text()方法设置的文本内容

注意：使用text()方法重新设置div元素的文本内容后，div元素原来的内容将被新设置的内容替换掉，包括HTML内容。例如，下面的代码：

```
<div><span id="clock">当前时间：2011-07-06 星期三  
13:20:10</span></div>
```

## 2. 对HTML内容操作

jQuery提供了html()和html(val)两个方法用于对HTML内容进行操作。其中，html()用于获取第一个匹配元素的HTML内容，html(val)用于设置全部匹配元素的HTML内容。例如，在一个HTML页面中，包括下面3行代码。

```
<div>
```

```
<span id="clock">当前时间：2011-07-06 星期三
```

```
13:20:10</span>
```

```
</div>
```

要获取div元素的HTML内容，可以使用下面的代码。

```
alert($("#div").html());
```

要重新设置div元素的HTML内容，可以使用下面的代码。

```
$("#div").html("<span style=' color:#FF0000' >我是通过html ()  
方法设置的HTML内容</span>");
```

注意：html()方法与html(val)不能用于XML文档，但是可以用于XHTML文档。

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-  
1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加两个<div>标记，这两个<div>标记除了id属性不同外，其他均相同，关键代码如下。

```
<div id="div1">  
<span id="clock">当前时间：2011-07-06星期三  
13:20:10</span>  
</div>  
<br/>应用html()方法设置的内容  
<div id="div2">  
<span id="clock">当前时间：2011-07-06星期三  
13:20:10</span>  
</div>
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现为<div>标记设置文本内容和HTML内容，并获取设置后的文本内容和HTML内容，具体代码如下。

```
<script type="text/javascript">  
$(document).ready(function() {
```

```
$("#div1").text("<span style=' color:#FF0000' >我是通过  
html ()方法设置的HTML内容</span>");  
$("#div2").html("<span style=' color:#FF0000' >我是通过  
html ()方法设置的HTML内容</span>");  
alert("通过text ()方法获取：  
\r\n"+$("#div").text()+"\r\n通过html ()方法获取：  
\r\n"+$("#div").html());  
});  
</script>
```

举一反三

根据本实例，读者可以：

使用text ()方法设置<div>标记的文本内容；

使用html ()方法设置<div>标记的HTML内容。

## 实例412 为多行列表框设置并获取值

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\412

实例说明

本实例主要实现为多行列表框设置并获取值的功能，运行结果如图20.17所示。

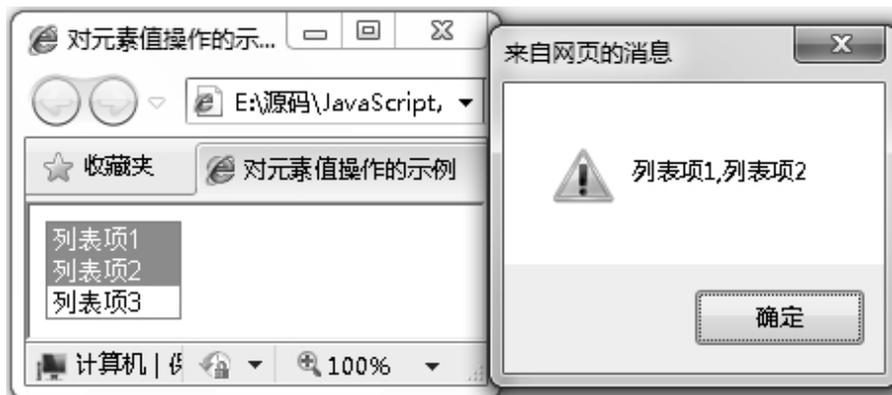


图20.17 获取到的多行列表框的值

技术要点

本实例实现时，主要是使用jQuery对元素值进行操作，jQuery中提供了3种对元素值操作的方法，如表20.1所示。

表20.1 对元素的值进行操作的方法

方法	说明	示例
var()	用于获取第一个匹配元素的当前值，返回值可能是一个字符串，也可能是一个数组。例如当 select 元素有两个选中值时，返回结果就是一个数组	<code>\$("#username").val();</code> //获取 id 为 username 的元素的值
var(val)	用于设置所有匹配元素的值	<code>\$("#input:text").val("新值")</code> //为全部文本框设置值
var(arrVal)	用于为 check、select 和 radio 等元素设置值，参数为字符串数组	<code>\$("#select").val(['列表项 1','列表项 2']);</code> //为下拉列表框设置多选值

实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个包含3个列表项的可多选的多行列表框，默认为后两项被选中，代码如下。

```
<select name="like" size="3" multiple="multiple" id="like">
  <option>列表项1</option>
  <option selected="selected">列表项2</option>
  <option selected="selected">列表项3</option>
</select>
```

(3) 在引入jQuery库的代码下方编写jQuery代码，应用jQuery的val(arrVal)方法将其第一个和第二个列表项设置为选中状态，并应用val()方法获取该多行列表框的值，具体代码如下。

```
<script type="text/javascript">
```

```
$(document).ready(function() {  
    $("select").val(['列表项1', '列表项2']);  
    alert($("select").val());  
});  
</script>
```

举一反三

根据本实例，读者可以：

在线考试系统中动态设置毕业学校下拉列表框的值；

求职类网站中设置期望薪水下拉列表的值。

## 20.2 jQuery表格应用

### 实例413 匹配表单中相应的元素

这是一个实用的实例

实例位置：光盘\mingrisoft\20\413

实例说明

本实例主要利用表单过滤器匹配表单中相应的元素。运行本实例，选中下拉列表框中的列表项3，将弹出提示对话框显示选中列表项的值，如图20.18所示。在该图中，选中的两个复选框的背景为红色，另外的一个复选框没有设置背景颜色，不可用按钮的 value 值被修改为“我是不可用的”。

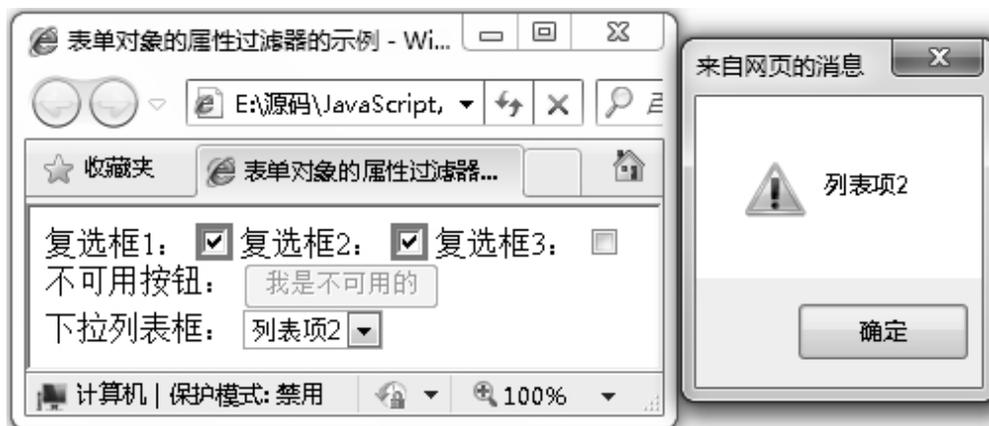


图20.18 利用表单过滤器匹配表单中相应的元素

技术要点

本实例用到了jQuery中的表单对象的属性过滤器。

表单对象的属性过滤器通过表单元素的状态属性（例如选中、不可用等状态）匹配元素，包括:checked过滤器、:disabled过滤器、:enabled过滤器和:selected过滤器 4种，如表20.2所示。

表20.2 JQuery的表单对象的属性过滤器

过滤器	说明	示例
:checked	匹配所有选中的被选中元素	<code>\$("input:checked")</code> //匹配 checked 属性为 checked 的 input 元素
:disabled	匹配所有不可用元素	<code>\$("input:disabled")</code> //匹配 disabled 属性为 disabled 的 input 元素
:enabled	匹配所有可用的元素	<code>\$("input:enabled")</code> //匹配 enabled 属性为 enabled 的 input 元素
:selected	匹配所有选中的 option 元素	<code>\$("select option:selected")</code> //匹配 select 元素中被选中的 option

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个表单，并在该表单中添加3个复选框、一个不可用按钮和一个下拉列表框，其中，前两个复选框为选中状态，关键代码如下。

```
<form>
    复选框1: <input
type="checkbox" checked="checked" value="复选框1"/>
    复选框2: <input
type="checkbox" checked="checked" value="复选框2"/>
    复选框3: <input type="checkbox" value="复选框3"/><br/>
    不可用按钮: <input type="button" value="不可用按钮" disabled><br/>
    下拉列表框:
<select onchange="selectVal()">
    <option value="列表项1">列表项1</option>
    <option value="列表项2">列表项2</option>
    <option value="列表项3">列表项3</option>
</select>
```

```
</form>
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现匹配表单中的被选中的checkbox元素、不可用元素和被选中的option元素的值，具体代码如下。

```
<script type="text/javascript">
    $(document).ready(function() {
        $("input:checked").css("background-
color", "red");        //设置选中的复选框的背景颜色
        $("input:disabled").val("我是不可用
的");                //为灰色不可用按钮赋值
    })

    function selectVal() {                                //
下拉列表框变化时执行的方法
        alert($("#select
option:selected").val());                            //显示选中的值
    }
</script>
```

举一反三

根据本实例，读者可以：

匹配表单中的所有表格；

匹配表单中的所有input元素默认值。

## 实例414 带表头的双色表格

这是一个应用比较广泛的实例

实例位置：光盘\mingrisoft\20\414

实例说明

本实例主要实现一个带表头的双色表格，运行本实例将显示图 20.19 所示的效果。其中，第一行为表头，编号为1和3的行采用的是偶数行样式，编号为2和4的行采用的是奇数行的样式。



编号	祝福对象	祝福者	字条内容	发送时间
1	琦琦	妈妈	愿你健康快乐的成长!	2011-07-05 13:06:06
2	wgh	无语	每天有份好心情!	2011-07-05 13:26:17
3	天净沙小晓	wgh	煮豆燃豆其，豆在釜中泣。 本是同根生，相煎何太急。	2011-07-05 13:36:06
4	明日科技	wgh	明天会更好!	2011-07-05 13:46:06

图20.19 带表头的双色表格

### 技术要点

本实例用到了jQuery中的简单过滤器。

简单过滤器是指以冒号开头，通常用于实现简单过滤效果的过滤器。例如，匹配找到的第一个元素等。jQuery提供的过滤器如表20.3所示。

表20.3 jQuery的简单过滤器

过滤器	说明	示例
:first	匹配找到的第一个元素,它是与选择器结合使用的	\$("#tr:first") //匹配表格的第一行
:last	匹配找到的最后一个元素,它是与选择器结合使用的	\$("#tr:last") //匹配表格的最后一行
:even	匹配所有索引值为偶数的元素,索引值从 0 开始计数	\$("#tr:even") //匹配索引值为偶数的行
:odd	匹配所有索引值为奇数的元素,索引从 0 开始计数	\$("#tr:odd") //匹配索引值为奇数的行
:eq(index)	匹配一个给定索引值的元素	\$("#tr:eq(1)") //匹配第二个 div 元素
:gt(index)	匹配所有大于给定索引值的元素	\$("#tr:gt(0)") //匹配第二个及以上的 div 元素
:lt(index)	匹配所有小于给定索引值的元素	\$("#tr:lt(2)") //匹配第二个及以下的 div 元素
:header	匹配如 h1, h2, h3...之类的标题元素	\$("#:header") //匹配全部的标题元素
:not(selector)	去除所有与给定选择器匹配的元素	\$("#input:not(:checked)") //匹配没有被选中的 input 元素
:animated	匹配所有正在执行动画效果的元素	\$("#:animated ") //匹配所有正在执行的动画

## 实现过程

(1) 创建一个名称为 index.html 的文件,在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中,添加一个5行5列的表格,关键代码如下。

```
<table width="98%" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#3F873B">
<tr>
<tdwidth="11%"height="27">编号</td>
<tdwidth="14%">祝福对象</td>
<tdwidth="12%">祝福者</td>
<tdwidth="33%">字条内容</td>
<tdwidth="30%">发送时间</td>
</tr>
<tr>
```

```

<tdheight="27">1</td>
<td>琦琦</td>
<td>妈妈</td>
<td>愿你健康快乐的成长! </td>
<td>2011-07-0513:06:06</td>
</tr>
,,,      <!--此处省略了其他行的代码-->
</table>

```

(3) 编写CSS 样式，通过元素选择符设置单元格的样式，并且编写th、even 和odd 3 个类选择符，用于控制表格中相应行的样式，具体代码如下。

```

<style type="text/css">
  td{
    font-size:12px;          /*设置单元格的样式*/
    padding:3px;           /*设置内边距*/
  }
  .th{
    background-color:#B6DF48; /*设置背景颜色*/
    font-weight:bold;        /*设置文字加粗显示*/
    text-align:center;      /*文字居中对齐*/
  }
  .even{
    background-color:#E8F3D1; /*设置偶数行的背景颜色
*/
  }
  .odd{

```

```
        background-color:#F9FCEF;        /*设置奇数行的背景颜色
*/
    }
</style>
```

(4) 在引入jQuery库的代码下方编写jQuery代码，实现匹配div元素的同辈元素p，并为其添加CSS类，具体代码如下。

```
<script type="text/javascript">
    $(document).ready(function() {
        $("tr:even").addClass("even");        //设置奇数行
        所用的CSS类
        $("tr:odd").addClass("odd");        //设置偶数行
        所用的CSS类
        $("tr:first").removeClass("even");    //移除even
        类
        $("tr:first").addClass("th");        //添加th类
    });
</script>
```

举一反三

根据本实例，读者可以：

制作一个显示学生各科成绩的双色表格；

制作一个显示本月工作计划的双色表格。

## [实例415 应用内容过滤器匹配指定单元格](#)

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\415

实例说明

本实例主要使用内容过滤器匹配为空的单元格、不为空的单元格和包含指定文本的单元格。运行本实例将显示图20.20所示的效果。其中，内容为wgh的单元格元素被标记为红色，编号为4的行中“字条内容”在设计时空，这里应用jQuery为其添加文本“暂无内容”，除该单元格外的其他单元格的背景颜色均被设置为#E8F3D1色。

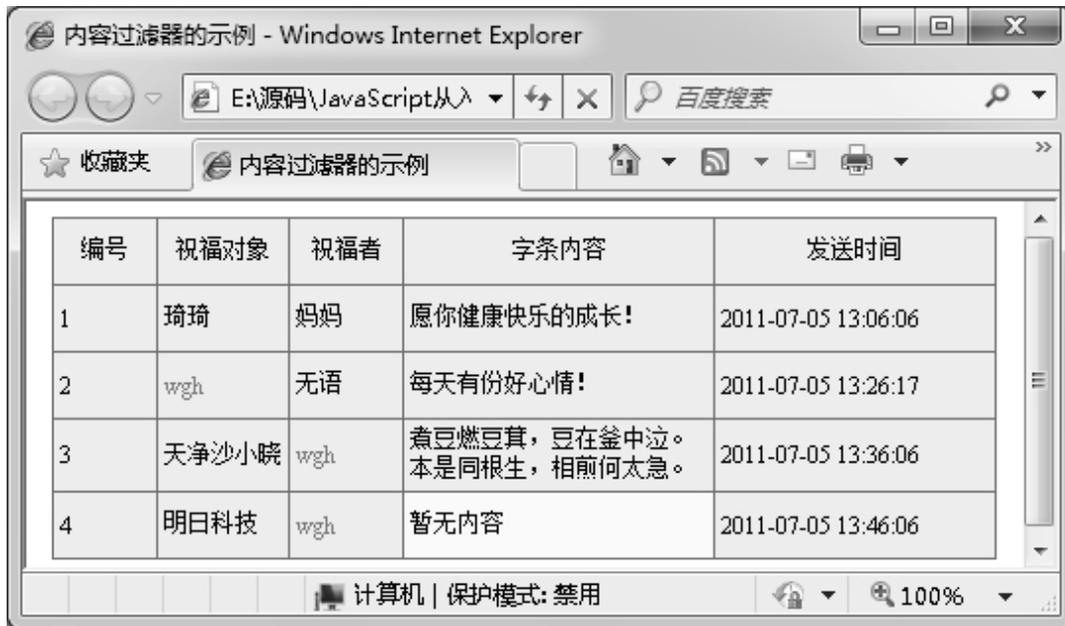


图20.20 应用内容过滤器匹配指定单元格

### 技术要点

本实例用到了jQuery中的内容过滤器。

内容过滤器就是通过 DOM元素包含的文本内容以及是否含有匹配的元素进行筛选。内容过滤器共包括:contains(text)、:empty、:has(selector)和:parent4种，如表20.4所示。

表20.4 jQuery的内容过滤器

过滤器	说明	示例
contains(text)	匹配包含给定文本的元素	<code>\$("li:contains('DOM'))</code> //匹配含有“DOM”文本内容的li元素
:empty	匹配所有不包含子元素或者文本的空元素	<code>\$("td:empty")</code> //匹配不包含子元素或者文本的单元格
:has(selector)	匹配含有选择器所匹配元素的元素	<code>\$("td:has(p)")</code> //匹配表格的单元格中含有<p>标记的单元格
:parent	匹配含有子元素或者文本的元素	<code>\$("td:parent")</code> //匹配不为空的单元格，即在该单元格中还包括子元素或者文本

## 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个5行5列的表格，关键代码如下。

```
<tablewidth="98%"border="0"align="center"cellpadding="0"cellspacing="1"bgcolor="#3F873B">
```

```
,,, <!--此处省略了其他行的代码-->
```

```
<tr>
```

```
<tdheight="27">4</td>
```

```
<td>明日科技</td>
```

```
<td>wgh</td>
```

```
<td></td>
```

```
<td>2011-07-0513:46:06</td>
```

```
</tr>
```

```
</table>
```

(3) 在引入jQuery库的代码下方编写jQuery代码，实现匹配div元素的同辈元素p，并为其添加CSS类，具体代码如下。

```
<script type="text/javascript">
```

```
$(document).ready(function() {
    $("td:parent").css("background-
color", "#E8F3D1");    //为不为空的单元格设置背景颜色
    $("td:empty").html("暂无内容");           //为
空的单元格添加默认内容
    $("td:contains('wgh')").css("color", "red");
    //将含有文本wgh的单元格的文字颜色设置为红色
});
</script>
```

举一反三

根据本实例，读者可以：

使用不同的颜色突出显示不及格的学生信息；

在表格中匹配指定日期之后的所有购物信息。

## 实例416 隔行换色并且鼠标指向行变色的表格

这是一个可以提高基础技能的实例

实例位置：光盘\mingrisoft\20\416

实例说明

对于一些清单型数据，通常是利用表格展示到页面中。如果数据比较多，很容易看串行，这时，可以为表格添加隔行换色并且鼠标指向行变色的功能，即：在表格中显示数据时，相邻的两行显示的颜色不同，而当用户将鼠标移动到某一行时，该行数据再以其他颜色重点显示。运行程序，效果如图20.21所示。

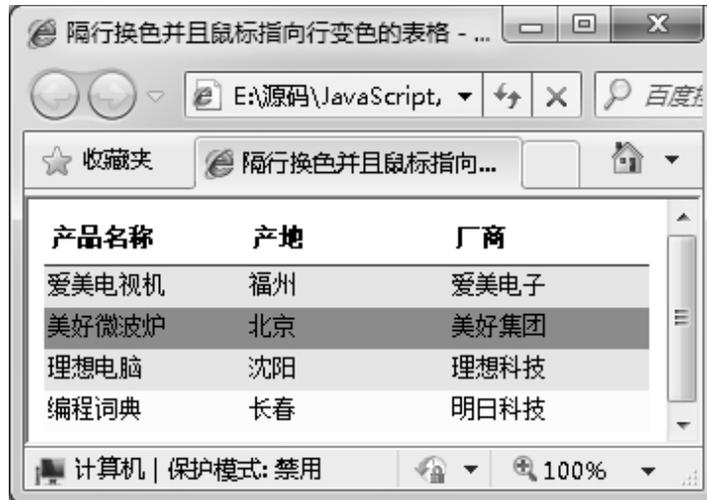


图20.21 隔行换色并且鼠标指向行变色的表格

### 技术要点

本实例实现的关键：如何使用jQuery库中的函数为表格中的行添加不同的样式，这主要用到addClass函数。

addClass函数用来为所有匹配的元素添加指定的CSS类名，语法格式：

```
addClass(class)
```

参数class表示要添加的CSS类名。

例如，为所有的div元素添加blue和line两个CSS类，代码如下。

```
$("div").addClass("blue line");
```

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库，代码如下。

```
<script type="text/javascript" src="JS/jquery-1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，添加一个5行3列的表格，并使用<thead>标记将表格的标题行括起来，再使用<tbody>标记将表格的其他行括起来，关键代码如下。

```
<table>
```

```

<thead>
  <tr>
    <th>产品名称</th>
    <th>产地</th>
    <th>厂商</th>
  </tr>
</thead>
<tbody>
  <tr>
    <td>爱美电视机</td>
    <td>福州</td>
    <td>爱美电子</td>
  </tr>
  ,,, <!--此处省略了其他3行的代码-->
</tbody>
</table>

```

(3) 编写CSS样式，用于控制表格整体样式、表头的样式、表格的单元格的样式，以及奇数行样式、偶数行样式和鼠标移到行的样式，具体代码如下。

```

<style type="text/css">
  table{border:0;border-
collapse:collapse;} /*设置表格整体样式*/
  td{font:normal12px/17pxArial;padding:2px;width:100px;} /*
*设置单元格的样式*/
  th{ /*设置表头的
样式*/
font:bold12px/17pxArial;

```

```

        text-align:left;
        padding:4px;
        border-bottom:1px solid#333;
    }
    .odd{background:#cef;} /*设置
奇数行样式*/
    .even{background:#ffc;} /*设置
偶数行样式*/
    .light{background:#00A1DA;} /*设置鼠
标移到行的样式*/
</style>

```

(4) 在引入 jQuery库的代码下方编写 jQuery代码，实现表格的隔行换色，并且让鼠标移到行变色的功能，具体代码如下。

```

<script type="text/javascript">
$(document).ready(function() {
    $("tbody tr:even").addClass("odd"); //
为偶数行添加样式
    $("tbody tr:odd").addClass("even"); //
为奇数行添加样式
    $("tbody tr").hover( //为
表格主体每行绑定hover方法
        function() {$(this).addClass("light");},
        function() {$(this).removeClass("light");}
    );
});
</script>

```

举一反三

根据本实例，读者可以：  
在购物车中隔行换色显示所有的已选购商品；  
查看学生成绩时，当鼠标选中时突出显示。

## 20.3 jQuery在实际中的应用

### 实例417 自动隐藏式菜单

本实例是一个实用性很强的程序

实例位置：光盘\mingrisoft\20\417

实例说明

在设计网页时，可以在页面中添加自动隐藏式菜单，这种菜单简洁易用，在不使用时能自动隐藏，保持页面的清洁。下面将通过一个具体的例子来说明如何通过 jQuery 实现自动隐藏式菜单。运行本实例将显示图 20.22 所示的效果。将鼠标移到“隐藏菜单”图片上时，将显示图20.23所示的菜单。将鼠标从该菜单上移出后，又将显示为图 20.22所示的效果。



图20.22 鼠标移出隐藏菜单的效果



图20.23 鼠标移入隐藏菜单的效果

### 技术要点

本实例实现自动隐藏式菜单时，主要使用了jQuery中基本动画效果。

基本动画效果指的就是元素的隐藏和显示。在jQuery中提供了两种控制元素隐藏和显示的方法。一种是分别隐藏和显示匹配元素。另一种是切换元素的可见状态，也就是如果元素是可见的，切换为隐藏；如果元素是隐藏的，切换为可见的。

#### 1. 隐藏匹配元素

使用hide()方法可以隐藏匹配的元素。hide()方法相当于将元素CSS样式属性display的值设置为none，它会记住原来的display的值。hide()方法有两种语法格式，一种是不带参数的形式，用于实现不带任何效果的隐藏匹配元素，其语法格式如下。

hide()

例如，要隐藏页面中的全部图片，可以使用下面的代码。

```
$("img").hide();
```

另一种是带参数的形式，用于以优雅的动画隐藏所有匹配的元素，并在隐藏完成后可选地触发一个回调函数，其语法格式如下。

```
hide(speed, [callback])
```

● **speed**: 用于指定动画的时长，可以是数字，也就是元素经过多少毫秒（1000ms=1s）后完全隐藏；也可以是默认参数 `slow`（600ms）、`normal`（400ms）和 `fast`（200ms）。

● **callback**: 可选参数，用于指定隐藏完成后要触发的回调函数。

例如，要在300ms内隐藏页面中的id为ad的元素，可以使用下面的代码。

```
$("#ad").hide(300);
```

说明：jQuery的任何动画效果都可以使用默认的3个参数，`slow`（600ms）、`normal`（400ms）和`fast`（200ms）。在使用默认参数时需要加引号，例如`show("fast")`；使用自定义参数时，不需要加引号，例如`show(300)`。

## 2. 显示匹配元素

使用`show()`方法可以显示匹配的元素。`hide()`方法相当于将元素CSS样式属性`display`的值设置为`block`或`inline`或除了`none`以外的值，它会恢复为应用`display: none`之前的可见属性。`show()`方法有两种语法格式，一种是不带参数的形式，用于实现不带任何效果的显示匹配元素，其语法格式如下。

```
show()
```

例如，要隐藏页面中的全部图片，可以使用下面的代码。

```
$("#img").show();
```

另一种是带参数的形式，用于以优雅的动画隐藏所有匹配的元素，并在隐藏完成后可选择地触发一个回调函数，其语法格式如下。

```
show(speed, [callback])
```

● **speed**: 用于指定动画的时长，可以是数字，也就是元素经过多少毫秒（1000ms=1s）后完全显示；也可以是默认参数 `slow`（600ms）、`normal`（400ms）和`fast`（200ms）。

● **callback**: 可选参数，用于指定隐藏完成后要触发的回调函数。

例如，要在300ms内显示页面中的id为ad的元素，可以使用下面的代码。

```
$("#ad").show(300);
```

### 3. 切换元素的可见状态

使用`toggle()`方法可以实现切换元素的可见状态。也就是如果元素是可见的，切换为隐藏；如果元素是隐藏的，切换为可见的。

`toggle()`方法的语法格式如下。

```
toggle()
```

例如，要实现通过单击普通按钮隐藏和显示全部div元素可以使用下面的代码。

```
$(document).ready(function() {  
    $("input[type='button']").click(function() {  
        $("div").toggle();           //切换有所有div元素的显示  
        状态  
    });  
});
```

等效于：

```
$(document).ready(function() {  
    $("input[type='button']").toggle(function() {  
        $("div").hide();           //显示div元素  
    }, function() {  
        $("div").show();           //隐藏div元素  
    });  
});
```

```
});  
});
```

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-  
1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，首先添加一个图片，id属性为flag，用于控制菜单显示；然后，添加一个id为menu的<div>标记，用于显示菜单；最后在<div>标记中添加用于显示菜单项的<ul>和<li>标记，关键代码如下。

```
  
<div id="menu">  
<ul>  
<li><a href="www.mingribook.com">图书介绍</a></li>  
<li><a href="www.mingribook.com">新书预告</a></li>  
,,, <!--省略了其他菜单项的代码-->  
<li><a href="www.mingribook.com">联系我们</a></li>  
</ul>  
</div>
```

(3) 编写CSS样式，用于控制菜单的显示样式，具体代码请参见光盘。

(4) 在引入jQuery库的代码下方编写jQuery代码，应用jQuery的val(arrVal)方法将其第一个和第二个列表项设置为选中状态，并应用val()方法获取该多行列表框的值，具体代码如下。

```
<script type="text/javascript">
```

```
$(document).ready(function() {
    $("#flag").mouseover(function() {
        $("#menu").show(300);           //显示菜单
    });
    $("#menu").hover(null,function() {
        $("#menu").hide(300);          //隐藏菜单
    });
});
</script>
```

举一反三

根据本实例，读者可以：

制作购物网站中的商品导航菜单；

制作企业门户网站中的全站导航菜单。

## 实例418 伸缩式导航菜单

本实例可以方便操作、提高效率

实例位置：光盘\mingrisoft\20\418

实例说明

本实例主要介绍应用jQuery实现滑动效果的具体应用，这里实现一个伸缩式导航菜单。运行本实例将显示图20.24所示的效果，单击某个主菜单时，将展开该主菜单下的子菜单，例如，单击“薪酬管理”主菜单，将显示图20.25所示的子菜单。通常情况下，“退出系统”主菜单没有子菜单，所以单击“退出系统”主菜单将不展开对应的子菜单，而是激活一个超级链接。



图20.24 未展开任何菜单的效果



图20.25 展开“薪酬管理”主菜单的效果

### 技术要点

本实例实现伸缩式导航菜单时，主要使用了jQuery中滑动效果。

在jQuery中，提供了slideDown()方法（用于滑动显示匹配的元素）、slideUp()方法（用于滑动隐藏匹配的元素）和slideToggle()方法（用于通过高度的变化动态切换元素的可见性）来实现滑动效果。下面分别进行介绍。

### 1. 滑动显示匹配的元素

使用slideDown()方法可以向下增加元素高度动态显示匹配的元素。slideDown()方法会逐渐向下增加匹配的隐藏元素的高度，直到元素完全显示为止。slideDown()方法的语法格式如下。

slideDown(speed, [callback])

- speed: 用于指定动画的时长，可以是数字，也就是元素经过多少毫秒（1000ms=1s）后完全显示；也可以是默认参数slow（600ms）、normal（400ms）和fast（200ms）。

- callback: 可选参数，用于指定显示完成后要触发的回调函数。

例如，要在300ms内滑动显示页面中的ID为ad的元素，可以使用下面的代码。

```
$("#ad").slideDown(300);
```

### 2. 滑动隐藏匹配的元素

使用slideUp()方法可以向上减少元素高度动态隐藏匹配的元素。slideUp()方法会逐渐向上减少匹配的显示元素的高度，直到元素完全隐藏为止。slideUp()方法的语法格式如下。

slideUp(speed, [callback])

- speed: 用于指定动画的时长，可以是数字，也就是元素经过多少毫秒（1000ms=1s）后完全隐藏；也可以是默认参数slow（600ms）、normal（400ms）和fast（200ms）。

- callback: 可选参数，用于指定隐藏完成后要触发的回调函数。

例如，要在300毫秒内滑动隐藏页面中的id为ad的元素，可以使用下面的代码：

```
$("#ad").slideDown(300);
```

### 3. 通过高度的变化动态切换元素的可见性

通过 `slideToggle()` 方法可以实现通过高度的变化动态切换元素的可见性。在使用 `slideToggle()` 方法时，如果元素是可见的，就通过减小高度使全部元素隐藏；如果元素是隐藏的，就增加元素的高度使元素最终全部可见。`slideToggle()` 方法的语法格式如下。

```
slideToggle(speed, [callback])
```

● **speed**：用于指定动画的时长，可以是数字，也就是元素经过多少毫秒（1000ms=1s）后完全显示或隐藏；也可以是默认参数 `slow`（600ms）、`normal`（400ms）和 `fast`（200ms）。

● **callback**：可选参数，用于指定动画完成时触发的回调函数。

例如，要实例单击id为flag的图片时，控制菜单的显示或隐藏（默认为不显示，奇数次单击时显示，偶数次单击时隐藏），可以使用下面的代码。

```
$("#flag").click(function() {  
    $("#menu").slideToggle(300); //显示/隐藏菜单  
});
```

### 实现过程

（1）创建一个名称为 `index.html` 的文件，在该文件的 `<head>` 标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-  
1.6.1.min.js"></script>
```

（2）在页面的 `<body>` 标记中，首先添加一个 `<div>` 标记，用于显示导航菜单的标题。然后，添加一个字典列表，用于添加主菜单项及其子菜单项，其中主菜单项由 `<dt>` 标记定义，子菜单项由 `<dd>` 标记定

义，最后再添加一个<div>标记，用于显示导航菜单的结尾，关键代码如下。

```
<div id="top"></div>
<dl>
  <dt>员工管理</dt>
  <dd>
    <div class="item">添加员工信息</div>
    <div class="item">管理员工信息</div>
  </dd>
  <dt>招聘管理</dt>
  <dd>
    <div class="item">浏览应聘信息</div>
    <div class="item">添加应聘信息</div>
    <div class="item">浏览人才库</div>
  </dd>
  <dt>薪酬管理</dt>
  <dd>
    <div class="item">薪酬登记</div>
    <div class="item">薪酬调整</div>
    <div class="item">薪酬查询</div>
  </dd>
  <dt class="title"><a href="#">退出系统</a></dt>
</dl>
<div id="bottom"></div>
```

(3) 编写CSS样式，用于控制导航菜单的显示样式，具体代码请参见光盘。

(4) 在引入 jQuery库的代码下方编写 jQuery代码，首先隐藏全部子菜单，然后再为每个包含子菜单的主菜单项添加模拟鼠标连续单击的事件toggle()，具体代码如下。

```
<script type="text/javascript">
$(document).ready(function() {
    $("dd").hide();
    //隐藏全部子菜单
    $("dt[class!='title']").toggle(
        function() {
            // slideDown:通过高度变化（向下增长）来动态地显示所有
            匹配的元素
            $(this).css("backgroundImage","url(images/title_hide.gif)"); //改变主菜单的背景
            $(this).next().slideDown("slow");
        },
        function() {
            // slideUp:通过高度变化（向上缩小）来动态地隐藏所有
            匹配的元素
            $(this).css("backgroundImage","url(images/title_show.gif)"); //改变主菜单的背景
            $(this).next().slideUp("slow");
        }
    );
});
</script>
```

举一反三

根据本实例，读者可以：

制作企业人事管理系统中的伸缩式导航菜单；  
制作OA办公化系统中的伸缩式导航菜单。

## 实例419 实现图片传送带

本实例是一个提高数据准确性的程序

实例位置：光盘\mingrisoft\20\419

实例说明

所谓图片传送带是指在页面的指定位置固定显示一定张数的图片（其他图片隐藏），单击最左边的图片时，全部图片均向左移动一张图片的位置；单击最右边的图片时，全部图片均向右移动一张图片的位置。这样不仅可以查看到全部图片，还能节省页面空间，比较实用。下面我们就来介绍如何通过jQuery实现图片传送带。运行本实例将显示图20.26所示的效果。将鼠标移动到左边的图片上，将显示图20.27所示的箭头，单击将向左移动一张图片；将鼠标移动到右边的图片上时，将显示向右的箭头，单击将向右移动一张图片；单击中间位置的图片，可以打开新窗口查看该图片的原图。

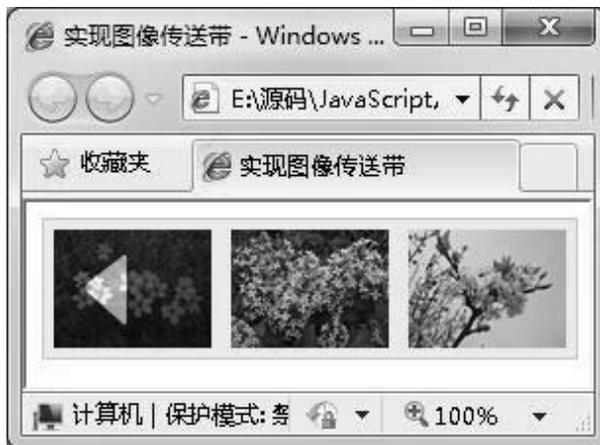


图20.26 鼠标不在任何图片上的效果

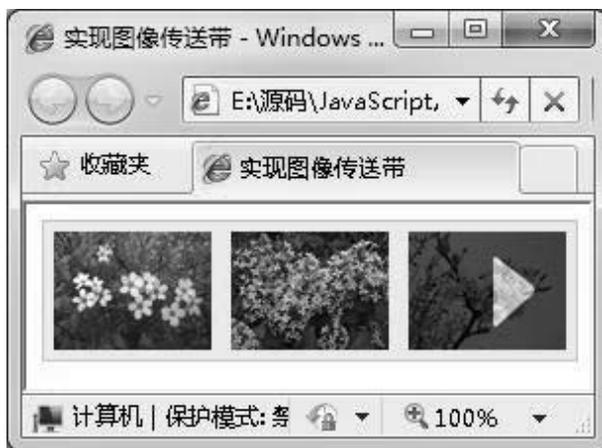


图20.27 将鼠标移动到第一张图片的效果

### 技术要点

本实例实现时，需要在jQuery中自定义动画效果，主要应用`animate()`方法创建自定义动画，应用`stop()`方法停止动画。下面分别进行介绍。

#### 1. 使用`animate()`方法创建自定义动画

`animate()`方法的操作更加自由，可以随意控制元素的属性，实现更加绚丽的动画效果。`animate()`方法的基本语法格式如下。

`animate(params, speed, callback)`

- `params`：表示一个包含属性和值的映射，可以同时包含多个属性，例如`{left:"200px",top:"100px"}`。

- `speed`：表示动画运行的速度，参数规则同其他动画效果的`speed`一致，它是一个可选参数。

- `callback`：表示一个回调函数，当动画效果运行完毕后执行该回调函数，它也是一个可选参数。

注意：在使用`animate()`方法时，必须设置元素的定位属性`position`为`relative`或`absolute`，元素才能动起来。如果没有明确定义元素的定位属性，并试图使用`animate()`方法移动元素时，它们只会静止不动。

例如，要实现将id为fish的元素在页面移动一圈并回到原点，可以使用下面的代码。

```
<script type="text/javascript">
$(document).ready(function() {
    $("#fish").animate({left:300},1000)
    .animate({top:200},1000)
    .animate({left:0},200)
    .animate({top:0},200);
});
</script>
```

在上面的代码中，使用了连缀方式的排队效果，这种排队效果，只对jQuery的动画效果函数有效，对于jQuery其他的功能函数无效。

说明：在animate()方法中可以使用属性opacity来设置元素的透明度。如果在{left:"400px"}中的400px之前加上“+”就表示在当前位置累加，“-”就表示在当前位置累减。

## 2. 使用stop()方法停止动画

stop()方法也属于自定义动画函数，它会停止匹配元素正在运行的动画，并立即执行动画队列中的下一个动画。stop()方法的语法格式如下。

stop(clearQueue, gotoEnd)

● clearQueue：表示是否清空尚未执行完的动画队列（值为true时表示清空动画队列）。

● gotoEnd：表示是否让正在执行的动画直接到达动画结束时的状态（值为true时表示直接到达动画结束时状态）。

例如，需要停止某个正在执行的动画效果，清空动画序列并直接到达动画结束时的状态，只需在\$(document).ready()方法中加入下面这句代码即可。

```
$("#btn_stop").click(function() {  
    $("#fish").stop("true", "true");           //停止动画效果  
});
```

说明：参数gotoEnd设置为true时，只能直接到达正在执行的动画的最终状态，并不能到达动画序列所设置的动画的最终状态。

### 实现过程

(1) 创建一个名称为 index.html 的文件，在该文件的<head>标记中应用下面的语句引入jQuery库。

```
<script type="text/javascript" src="JS/jquery-  
1.6.1.min.js"></script>
```

(2) 在页面的<body>标记中，首先添加一个<div>标记，用于显示导航菜单的标题。然后，添加一个字典列表，用于添加主菜单项及其子菜单项，其中主菜单项由<dt>标记定义，子菜单项由<dd>标记定义。最后再添加一个<div>标记，用于显示导航菜单的结尾，关键代码如下：

```
<div id="container">  
    <div class="box">  
        <a href="images/01.jpg"></a>  
        <a href="images/02.jpg"></a>  
        <a href="images/03.jpg"></a>  
        <a href="images/04.jpg"></a>  
        <a href="images/05.jpg"></a>
```

```
<a href="images/06.jpg"></a>
</div>
</div>
```

(3) 编写CSS样式，用于控制图片传送带容器及图片的样式，具体代码请参见光盘。

(4) 在引入jQuery库的代码下方编写jQuery代码，实现图片传送带效果，具体代码如下。

```
<script type="text/javascript">
$(document).ready(function() {
    var spacing=90; //定义保存间距
    的变量
    function createControl(src) { //定义创
    建控制图片的函数
        return$('<img/>')
            .attr('src', src) //设置图
            片的来源
            .attr("width",80)
            .attr("height",60)
            .addClass('control')
            .css('opacity', 0.6) //设置透
            明度
            .css('display', 'none'); //默认
            为不显示
    }
    var$leftRollover= createControl('images/left.gif'); //
    创建向左移动的控制图片
```

```

    var$rightRollover= createControl(' images/right.gif');//
创建向左移动的控制图片
    $(' #container').css({                                     //改变图像
传送带容器的CSS样式
        'width': spacing*3,
        'height': '70px',
        'overflow': 'hidden'                                 //溢出时
隐藏
    }).find('.box a').css({
        'float': 'none',
        'position': 'absolute',                             //设置为
绝对布局
        'left':1000                                         //将左边距设
置为1000, 目的是不显示
    });
    var setUpbox= function() {
        var$box=$(' #container .box a');
        $box.unbind(' clickmouseentermouseleave');        //移除绑
定的事件
        /*****左边的图片
        *****/
        $box.eq(0)
            .css(' left', 0)
            .click(function(event) {
                $box.eq(0).animate({' left': spacing},
' fast'); //为第一张图片添加动画

```

```

    $box.eq(1).animate({'left' :
spacing*2}, 'fast'); //为第二张图片添加动画
    $box.eq(2).animate({'left' :
spacing*3}, 'fast'); //为第3张图片添加动画
    $box.eq($box.length-1)
        .css('left', -spacing) //设置左
边距
        .animate({'left':0}, 'fast', function() {
            $(this).prependTo('#container .box');
            setUpbox();
        }); //添加动画
    event.preventDefault(); //取消事件
的默认动作
    }).hover(function() { //设置鼠标
的悬停事件
        $leftRollover.appendTo(this).fadeIn(200); //显
示向左移动的控制图片
    }, function() {
        $leftRollover.fadeOut(200); //隐藏向
左移动的控制图片
    });
/*****右边的图片
*****/
    $box.eq(2)
        .css('left', spacing*2) //设置左边
距

```

```

        .click(function(event) { //绑定单击
事件
            $box.eq(0) //获取左边的
            图片，也就是第一张图片
                .animate({'left':-spacing}, 'fast', function()
                {
                    $(this).appendTo('#container .box');
                    setUpbox();
                }); //添加动画
            $box.eq(1).animate({'left':0}, 'fast'); //添
            加动画
            $box.eq(2).animate({'left': spacing},
            'fast'); //添加动画
            $box.eq(3)
                .css('left', spacing*3) //设置左
            边距
                .animate({'left': spacing*2}, 'fast'); //
            添加动画
            event.preventDefault(); //取消事件
            的默认动作
        }).hover(function() { //设置鼠标
            的悬停事件
                $rightRollover.appendTo(this).fadeIn(200); //显示
            向右移动的控制图片
            }, function() {
                $rightRollover.fadeOut(200); //隐藏向
            右移动的控制图片
    
```

```

    });
    /*****中间的图片
    *****/
    $box.eq(1).css('left', spacing);           //设置
    置中间图片的左边距
    };
    setUpbox();
    $("a").attr("target", "_blank");         //查看
    原图时，在新的窗口中打开
    });
</script>

```

举一反三

根据本实例，读者可以：

在购物网站中循环展示商品图片；

在婚恋网站中显示用户头像。

## [实例420 打造自己的开心农场](#)

本实例是一个提高效率、人性化的程序

实例位置：光盘\mingrisoft20\420

实例说明

应用 jQuery 提供的对 DOM 节点进行操作的方法实现自己的开心农场。运行程序将显示图20.28所示的效果。单击“播种”按钮，将显示图20.29所示的效果。单击“生长”按钮，将显示图20.30所示的效果。单击“开花”按钮，将显示图20.31所示的效果。单击“结果”按钮，将显示一棵结满果实的草莓秧。



图20.28 页面的默认运行结果



图20.29 单击“播种”按钮的结果



图20.30 单击“生长”按钮的结果



图20.31 单击“开花”按钮的结果

### 技术要点

本实例主要使用jQuery库中的remove函数、prepend函数、replaceWith函数和replaceAll函数，演示植物播种、生长、开花和结果的全过程。下面分别对这几个函数进行讲解。

#### 1. remove函数

用于从DOM中删除所有匹配的元素，语法格式：

```
remove([expr])
```

参数expr表示要移除的元素，如果为空，则表示移除所有元素。

## 2. prepend函数

为所有匹配的元素的内前置内容，语法格式：

```
prepend(content)
```

参数content表示要前置的内容。

## 3. replaceWith函数

用于将所有匹配的元素替换成指定的HTML或DOM元素，语法格式：

```
replaceWith(content)
```

参数content表示要替换成的HTML或DOM元素。

## 4. replaceAll函数

用于使用匹配的元素替换掉所有selector匹配到的元素，语法格式：

```
replaceAll(selector)
```

参数selector表示要替换的元素。

实现过程

(1) 创建一个名称为index.html的文件，在该文件的<head>标记中应用下面的代码解决PNG图片背景不透明的问题，代码如下。

```
<!--使用jQuery解决PNG图片背景不透明的问题-->
<script src="JS/jquery-1.3.2.min.js"></script>
<script src="JS/jquery.pngFix.js"></script>
<script src="JS/jquery.pngFix.pack.js"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $("#bg").pngFix();
    });
</script>
<!--*****-->
```

注意：在jQuery 的新版本中，由于忽略了IE 8 下PNG 图片背景不透明的情况，因此要实现让PNG 图片在IE 8 下背景透明，需要使用以前的jQuery版本1.3.2。

(2) 在页面的<body>标记中，添加一个显示农场背景的<div>标记，并且在该标记中添加4个<span>标记，用于设置控制按钮，代码如下。

```
<div id="bg">
  <span id="seed"></span>
  <span id="grow"></span>
  <span id="bloom"></span>
  <span id="fruit"></span>
</div>
```

(3) 编写CSS代码，控制农场背景、控制按钮和图片的样式，具体代码参见光盘。

(4) 编写jQuery代码，分别为播种、生长、开花和结果按钮绑定单击事件，并在其单击事件中应用操作DOM节点的方法控制作物的生长，具体代码如下。

```
<script type="text/javascript">
  $(document).ready(function() {
    $("#seed").bind("click", function() {          //绑定播种按钮的单击事件
      $("img").remove();                          //移除img元素
      $("#bg").prepend("<img src='images/seed.png' />");
    });
    $("#grow").bind("click", function() {         //绑定生长按钮的单击事件
      $("img").remove();                          //移除img元素
```

```
    $("#bg").append("<img src=' images/grow.png' />");
});
$("#bloom").bind("click", function() {           //绑定开花
按钮的单击事件
    $("#img").replaceWith("<img
src=' images/bloom.png' />");
});
$("#fruit").bind("click", function() {           //绑定结果
按钮的单击事件
    $("#img
src=' images/fruit.png' />").replaceAll("img");
});
});
</script>
```

举一反三

根据本实例，读者可以：

制作一个画桃花游戏；

开发类似QQ网页平台的游戏。

# 附录

名 称	实 例
window 对象	实例 001*
window 对象的 open()方法	实例 001*、实例 003、实例 005、实例 006、实例 022、实例 023*、实例 025、实例 026、实例 027、实例 041、实例 043、实例 184*、实例 310*、实例 358*、实例 369*、实例 381*、实例 384*、实例 386*、实例 387*
window 对象的 setTimeout()方法	实例 002*、实例 004、实例 035*、实例 049、实例 052*、实例 081*、实例 108、实例 116、实例 124、实例 133、实例 135*、实例 136、实例 138*、实例 193*、实例 143*
window 对象的 close()方法	实例 004、实例 007*、实例 009、实例 032
window 对象的 moveTo()方法	实例 005*、实例 015*、实例 017、实例 019、实例 020、实例 022、实例 029、实例 043
screen 对象	实例 005*、实例 015 ~ 实例 018、实例 020、实例 029、实例 362
Cookie	实例 006*
window 对象的 opener 属性	实例 008*、实例 009、实例 357*、实例 372、实例 382*、实例 387*
window 对象的 reload()方法	实例 008*、实例 028、实例 357、实例 372、实例 382
window 对象的 showModalDialog()方法	实例 010*、实例 011、实例 013、实例 241、实例 263、实例 359、实例 360、实例 371、实例 383*、实例 388*、实例 389
创建数组	实例 012*、实例 064*、实例 113
window 对象的 scroll()方法	实例 014*
window 对象的 resizeTo()方法	实例 015*、实例 016、实例 017、实例 020、实例 029、实例 031、实例 036、实例 043
window 对象的 resizeBy()方法	实例 015*
window 对象的 moveBy()方法	实例 018*、实例 021
document 对象的 focus()方法来	实例 023*
<object>标记	实例 024*、实例 248*
<param>标记	实例 024*、实例 159*
document 对象的 location 属性	实例 026*、实例 065
history 对象	实例 030*
window 对象的 confirm()方法	实例 032*、实例 363*、实例 364、实例 391
浮动框架标记<iframe>	实例 033*、实例 038*、实例 281
body 标记的 style 样式	实例 034*
document 对象的 title 属性	实例 036*
框架集标记<frameset>	实例 037*、实例 040
Frame 框架	实例 039*
<div>标记	实例 042*、实例 054、实例 133、实例 192、实例 238、实例 251*、实例 366、实例 379
<map>标记	实例 044*
Image 对象	实例 045*、实例 049、实例 240

续表

名 称	实 例
CSS 的波浪变形效果滤镜 Wave	实例 046*
switch 语句	实例 047*、实例 186
CSS 样式的透明效果滤镜 alpha	实例 048*、实例 150、实例 151、实例 233
display 属性	实例 050*
rect 属性	实例 050*、实例 165*、实例 172*、实例 256*、实例 258、实例 259
opacity 属性	实例 051*
pixelLeft 属性	实例 051*
clearTimeout()方法	实例 052*、实例 124、实例 385*
document 对象	实例 055*、实例 056、实例 061
window 对象的 alert()方法	实例 055*、实例 061、实例 077、实例 115、实例 390*
readOnly 属性	实例 057
if...else 条件判断语句	实例 058*、实例 075、实例 077、实例 116、实例 223
for 循环语句	实例 058*、实例 062、实例 073、实例 082、实例 100
String 对象的 charCodeAt()方法	实例 058*、实例 096*、实例 175*
String 对象的 substring()方法	实例 058、实例 111*、实例 138*、实例 139、实例 166*、实例 284*、实例 285、实例 286、实例 393*
select()属性	实例 059*
Event 对象	实例 060*
while 循环语句	实例 063
select 元素	实例 063
创建 Option 对象	实例 064*
String 对象的 fromCharCode()方法	实例 067*、实例 175*
innerText 属性	实例 067*、实例 200、实例 204
option 数组的 remove 方法	实例 067*
display 属性	实例 068*、实例 071*、实例 072、实例 080
<optgroup>标记	实例 069*
单选按钮	实例 070*
复选框的 checked 属性	实例 073、实例 074、实例 075
表单的 submit()提交方法	实例 078*、实例 080、实例 083
表单的 reset()重置方法	实例 078
<meta>标记	实例 079*
表单元素的 title 属性	实例 082
表单的 action 属性	实例 083
使用正则表达式对象	实例 084*、实例 087～实例 095、实例 104、实例 313
from 对象	实例 086*
正则表达式的常用字符类	实例 087*
正则表达式的常用重复字符	实例 087*
isNaN()	实例 094*
eval()方法	实例 095*、实例 280*
String 对象的 substr()方法	实例 097*、实例 100

续表

名 称	实 例
String 对象的 length 属性	实例 097*、实例 100
String 对象的 replace()方法	实例 098*、实例 099、实例 104*
String 对象的 indexOf()方法	实例 101*、实例 313、实例 314*、实例 365*
parseInt()方法	实例 102*、实例 152、实例 199
Number 对象的 toString()方法	实例 102*、实例 143、实例 265
Math 对象的 random()方法	实例 105*、实例 152、实例 161*、实例 174、实例 230、实例 238、实例 239、实例 261、实例 269、实例 315
String 对象的 split()方法	实例 106*、实例 123、实例 154*、实例 326*、实例 400
Date 对象	实例 107*、实例 108、实例 109、实例 113 ~ 实例 115、实例 125、实例 126 ~ 实例 133、实例 226、实例 396*
innerHTML 方法	实例 108、实例 114、实例 126、实例 133、实例 135、实例 136、实例 138、实例 190、实例 215、实例 374*
Date 对象的 toLocaleString()方法	实例 110*
window 对象的 status 属性	实例 112、实例 263*、实例 281 ~ 实例 290
document 对象的 lastModified 属性	实例 117*
三目运算符	实例 119*、实例 227
prompt()方法	实例 120*、实例 277*
Math 对象的 abs()方法	实例 121*、实例 145、实例 393
Math 对象的 round()方法	实例 121*、实例 163
Math 对象的 floor()方法	实例 122*、实例 143*、实例 161*、实例 239、实例 263、实例 393
CSS 样式的滤镜	实例 127*、实例 221、实例 222、实例 231
Date 对象的 getTimezoneOffset()方法	实例 132*
window 对象的 setInterval()方法	实例 134*、实例 143、实例 174、实例 219、实例 266
document 对象的 write()方法	实例 137*、实例 226、实例 265、实例 269、实例 398
filter 滤镜的 glow 属性	实例 137*、实例 140、实例 274*
String 对象的 fromCharCode()方法	实例 141*
Math 对象的 sin()方法	实例 145*、实例 154、实例 158、实例 169、实例 235、实例 236*、实例 272、实例 273
eval()函数	实例 147*、实例 163*
options 数组	实例 149*、实例 247*、实例 395*
body 对象中的 scrollTop 属性	实例 153*、实例 154、实例 262*、实例 266、实例 268*、实例 280
body 对象中的 scrollLeft 属性	实例 153*、实例 154、实例 266、实例 268*、实例 278*、实例 280
event 对象 button 属性	实例 155*、实例 182
onmousedown 事件	实例 155
onmousemove 事件	实例 155、实例 290*
列表框的 selectedIndex 属性	实例 156、实例 240
body 对象的 clientHeight 属性	实例 157、实例 161、实例 259、实例 260、实例 276
body 对象的 clientWidth 属性	实例 161、实例 260、实例 278*
String 对象的 charAt()方法	实例 158*、实例 160、实例 164、实例 166*、实例 283*
<object>标记的 scale()方法	实例 159*
<object>标记的 rotate()方法	实例 159*

续表

名 称	实 例
RGB()函数	实例 160*、实例 265
Math 对象的 route()方法	实例 162*
event 对象的 x 属性	实例 168、实例 220、实例 279*
event 对象的 y 属性	实例 168、实例 220、实例 279*
filter 滤镜的 wave 属性	实例 170*、实例 232
filter 滤镜的 revealTrans 属性	实例 171*、实例 242、实例 267、实例 279
filter 滤镜中的 shadow 属性	实例 173*、实例 210*
超级链接样式的 color 属性	实例 177*
超级链接样式的 backgroundColor 属性	实例 178*
建立 E-mail 超级链接	实例 179*
DOM 提供的 getElementsByTagName()方法	实例 180*
超级链接对象的 outerHTML 属性	实例 180*
window.external.AddFavorite 方法	实例 183*
window.location.href 方法	实例 185*、实例 390、实例 391
arguments 对象	实例 187*
<marquee>标记	实例 189、实例 320*
style 对象的 borderColor 属性	实例 193
onMouseOver 事件	实例 195、实例 214
onMouseOut 事件	实例 195、实例 214、实例 290*
window 对象的 lastSelection 属性	实例 196*
event 对象的 srcElement 属性	实例 196*、实例 197*
单元格的 parentElement 属性	实例 201
style 对象的 cursor 属性	实例 202*
removeExpression()方法	实例 203*
setExpression()方法	实例 203*
document 对象的 createElement()方法	实例 205
style 样式的 table-layout 属性	实例 208*
<fieldset>标记	实例 209*
<legend>标记	实例 209*
Event 对象的 wheelDelta 属性	实例 217*
style 属性的 zoom 属性	实例 217*
Math 对象	实例 224*、实例 394*
滤镜 light	实例 234
Math 对象的 cos 方法	实例 235、实例 236*、实例 273
onChange 事件	实例 240*
undefined	实例 241*
图像热区	实例 244*
<bgsound>标记	实例 245*、实例 246
Array 对象	实例 246*

续表

名 称	实 例
<embed>标记	实例 248、实例 249、实例 250*、实例 251
style 样式的 backgroundImage 属性	实例 252*、实例 253
style 样式的 backgroundPosition 属性	实例 252*、实例 253
style 样式的 backgroundRepeat 属性	实例 252*、实例 253
style 样式的 backgroundAttachment 属性	实例 252*、实例 253
document 对象的 bgColor 属性	实例 254*、实例 255、实例 257
body 对象的 scrollHeight 属性	实例 262*
body 对象中的 layersOffsetLeft 属性	实例 268*
body 对象中的 offsetTop 属性	实例 268*
style 样式的 paddingTop 属性	实例 274*
style 样式的 position 属性	实例 275*
window 对象的 scrollBy()方法	实例 278*
event 对象的 clientX 属性	实例 280、实例 289*
event 对象的 clientY 属性	实例 280、实例 289*
window 对象的 print()方法	实例 291*、实例 292
WebBrowser 控件	实例 293*、实例 304、实例 305、实例 306
WshShell 对象的 RegWrite()方法	实例 294*
WshShell 对象	实例 294*
ActiveXObject()构造函数	实例 295*、实例 296～实例 300、实例 376、实例 377、实例 378
JSP 的内置对象 request 的 getRequestURL()方法	实例 297、实例 377、实例 378
ASP 的内置对象 request 的 ServerVariables()方法	实例 298
PrintOut()方法	实例 301*
CSS 样式的 media 类型	实例 302*、实例 304、实例 305、实例 306
thead 标记	实例 303*
tfoot 标记	实例 303*
page-break-after 属性	实例 303*
<body>标记的事件	实例 307*、实例 308
<noscript>标记	实例 309*
keyCode 属性值	实例 311*
Event 对象	实例 312*
parseFloat()方法	实例 313*
String 对象的 lastIndexOf()方法	实例 314*
CSS 样式的字体属性	实例 317*、实例 349
CSS 样式的颜色和背景属性	实例 317*、实例 349
CSS 样式的边框属性	实例 317*、实例 323*、实例 324
CSS 样式的伪类	实例 318*
样式表对象的 href 属性	实例 319*
CSS 样式的 Alpha 滤镜	实例 320*

续表

名 称	实 例
CSS 样式的 border 属性	实例 322*
CSS 样式的 progid 滤镜	实例 325*
CSS 样式的 cursor 属性	实例 327*、实例 328*
CSS 样式的滚动条属性	实例 329*
CSS 样式的字体属性中的 text-decoration 属性	实例 330*
<rt>和<ruby>标记	实例 331*
Style 属性的 list-style-image 属性	实例 332*
CSS 样式的 glow 滤镜	实例 333*、实例 338、实例 339、实例 342、实例 343
CSS 样式的 dropshadow 滤镜	实例 334*、实例 336、实例 340、实例 342
CSS 样式的 Shadow 滤镜	实例 335*、实例 340
CSS 样式的 chroma 滤镜	实例 336*、实例 339、实例 340
CSS 样式的 light 滤镜	实例 337*
CSS 样式的 mask 滤镜	实例 339*、实例 340、实例 341
CSS 样式的 Blur 滤镜	实例 343*、实例 346
CSS 样式的 Wave 滤镜	实例 343*、实例 344、实例 347
CSS 样式的 Alpha 滤镜	实例 345*
CSS 样式的 Gray 滤镜	实例 348*
CSS 样式的定位属性	实例 349*
在 IE 中创建 DOM 并载入 XML	实例 350*
在 Mozilla 中创建 DOM 并载入 XML	实例 350*
XMLHttpRequest 对象	实例 351*
XML 文档中引用外部 CSS 样式表文件	实例 352*
XML 文档中引用外部 XSL 样式表文件	实例 353*
使用 IE XML 数据岛输出 XML 文档	实例 354*
XML 数据岛的 recordset 对象的 absolutePosition 属性	实例 355*
XML 数据岛的 recordset 对象的 recordcount 属性	实例 355*
XML 数据岛的 recordset 对象的 moveprevious()方法	实例 355*
XML 数据岛的 recordset 对象的 movenext()方法	实例 355*
XML DOM 对象的 createElement()方法	实例 356*
XML DOM 对象的 createAttribute()方法	实例 356*
XML DOM 对象的 createNode()方法	实例 356*
XML DOM 对象的 createTextNode()方法	实例 356*
XML DOM 对象的 insertBefore()方法	实例 356*
XML DOM 对象的 removeChild()方法	实例 356*
XML DOM 对象的 appendChild()方法	实例 356*
XML DOM 对象的 hasChildNodes ()方法	实例 356*

续表

名 称	实 例
XML DOM 对象的 lastChild 属性	实例 356*
ASP 内置对象 Response 的 Write()方法	实例 361*
JSP 内置对象 out 的 println()方法	实例 368*
XMLHttpRequest 对象的常用方法	实例 373*、实例 374
ASP.Net 中的 Response.Write 方法	实例 380*
赋值运算符	实例 394*
读取 Cookie	实例 397*
设置 Cookie	实例 397*

## 图书在版编目 (CIP) 数据

JavaScript网页特效范例宝典/张鑫, 薛莉颖, 高茹编著. --北京: 人民邮电出版社, 2015. 1

(软件工程师: 曲藏版)

ISBN 978-7-115-37110-2

I. ①J… II. ①张…②薛…③高… III. ①JAVA语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字 (2014) 第256442号

## 内容提要

本书以网站开发者在开发中遇到的实际问题和应该掌握的技术为中心, 全面介绍了JavaScript在网站开发中各个方面的常用技术和技巧。全书共20章, 分别是窗口/框架与导航条设计、表单及表单元素、实用JavaScript函数、日期和时间、文字特效、超级链接特效、操作表格、图形图像与多媒体、页面特效、状态栏特效、报表与打印、网站安全、HTML/CSS样式、JavaScript与XML、JavaScript与ASP结合、JavaScript与JSP结合、JavaScript与ASP.NET结合、JavaScript与PHP结合、综合应用、jQuery应用开发等。全书共提供了420个实例, 每个实例都突出实用性, 其中大部分是网站开发者梦寐以求的问题解决方案。

本书附有配套光盘。光盘提供了书中所有实例的全部源代码, 所有源代码都经过精心调试, 在Windows XP/Windows 2003/Windows 7环境下测试通过, 保证能够正常运行。

本书适用于广大计算机爱好者和编程人员使用, 也可供大中专院校师生学习参考。

◆编著 张鑫 薛莉颖 高茹

责任编辑 李永涛

责任印制 杨林杰

◆人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

三河市海波印务有限公司印刷

◆开本：787×1092 1/16

印张：46.75

字数：1266千字 2015年1月第1版

印数：1-3000册 2015年1月河北第1次印刷

定价：99.00元（附光盘）

读者服务热线：(010)81055410 印装质量热线：

(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京崇工商广字第0021号

链接: <http://pan.baidu.com/s/li3u00gH> 密码: mr8t