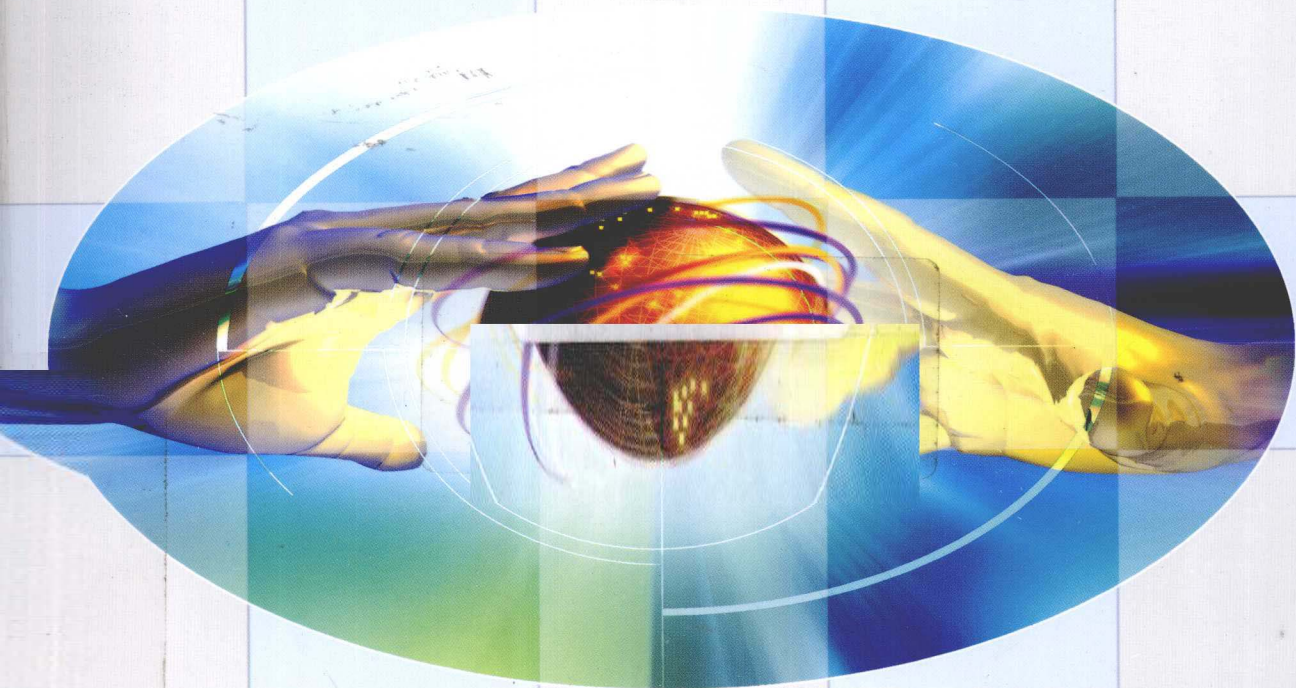


JavaScript网页交互 特效范例与技巧

JAVASCRIPT WANGYE JIAOHU TEXIAO FANLI YU JIQIAO

李云程 著

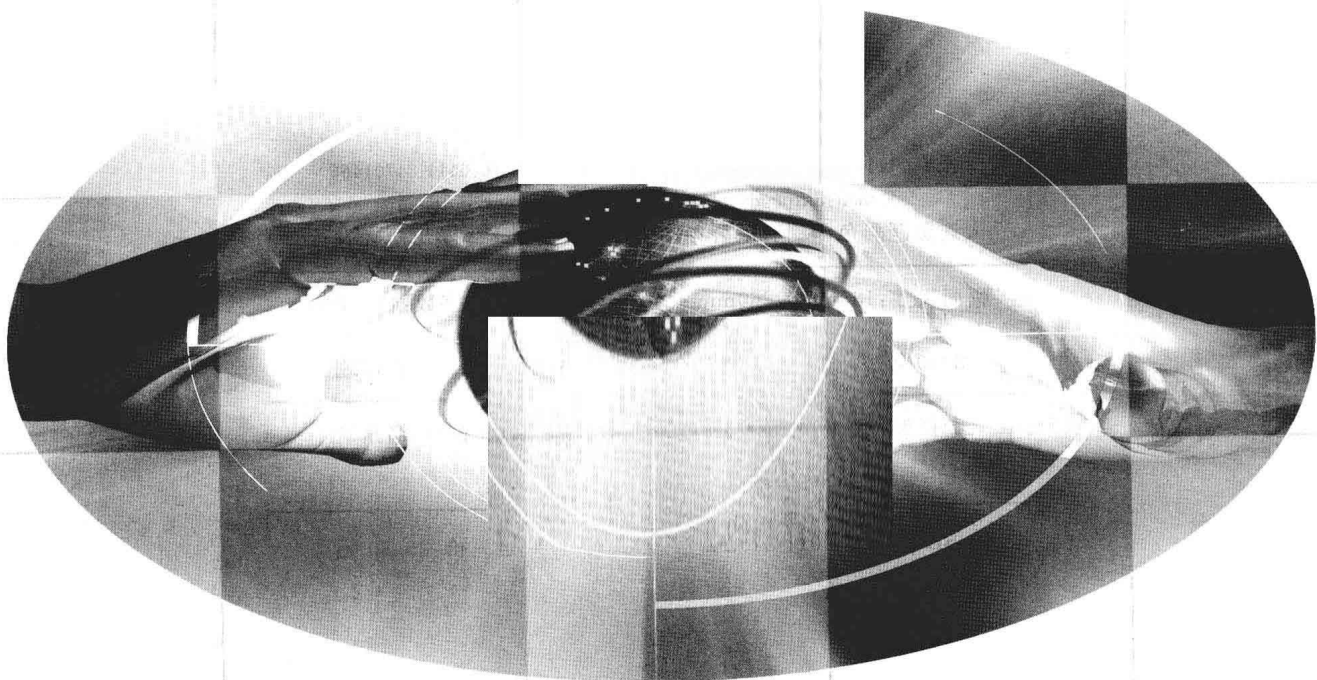


大连理工大学出版社
DALIAN UNIVERSITY OF TECHNOLOGY PRESS

JavaScript网页交互 特效范例与技巧

JAVASCRIPT WANGYE JIAOHU TEXIAO FANLI YU JIQIAO

李云程 著



大连理工大学出版社
DALIAN UNIVERSITY OF TECHNOLOGY PRESS

图书在版编目(CIP)数据

JavaScript 网页交互特效范例与技巧 / 李云程著.

— 大连 : 大连理工大学出版社, 2010. 4

ISBN 978-7-5611-5492-2

I. ①J… II. ①李… III. ①JAVA 语言—主页制作—程序设计 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2010)第 060845 号

大连理工大学出版社出版

地址:大连市软件园路 80 号 邮政编码:116023

电话:0411-84708842 邮购:0411-84703636 传真:0411-84701466

E-mail:dutp@dutp.cn URL:http://www.dutp.cn

大连美跃彩色印刷有限公司印刷 大连理工大学出版社发行

幅面尺寸:185mm×260mm 印张:19 字数:435 千字

印数:1~2000

2010 年 4 月第 1 版

2010 年 4 月第 1 次印刷

责任编辑:潘弘喆

责任校对:潘素君

封面设计:张莹

ISBN 978-7-5611-5492-2

定 价:33.80 元

随着互联网应用的迅速发展,Web 应用开发已不再是那么高难的技术,能够设计具有良好交互特效功能的网页,就成了网站开发人员的必备能力。而 JavaScript 正是为 Web 页面添加各种交互效果的首选语言,如今 JavaScript 已从一种脚本语言,发展成为功能强大的程序设计语言,备受广大网站开发人员推崇和喜爱。

《JavaScript 网页交互特效范例与技巧》在内容上,针对网页交互特效设计技术发展,分析初学者和专业设计人员的各自需求,在内容选取方面对二者进行了统筹兼顾,选择和编写了 65 个优秀范例。内容既体现了 JavaScript 技术的最新发展,也关注其在网站开发项目中的应用特色和新颖程度。

本书的目的是帮助初学者快速理解 JavaScript 语言结构等基础知识,学会开发高级的页面交互特效设计,并不断积累开发经验;针对有一定经验的中高级开发人员,则侧重介绍交互特效高级范例应用的设计思路和程序设计技巧,使其学到最新技术,产生创新灵感,为他们提供相互学习、相互交流、分享编程经验和体会的平台。

本书在撰写思路和选材上体现如下创新:摒弃传统知识讲授型书籍的套路,做出大胆探索,避免长篇大论地单纯讲解,而是将知识学习与技能训练融入范例制作过程中,让学习者一直参与实践,实现“在学中做、在做中学”;其次,通过与传统面向对象编程语言比较,给出了 JavaScript 面向对象程序设计的独特表达方式,简练地对其进行了讲解并提供有代表性的应用范例。本书注重技能训练过程,既是学习又是实践。每个范例都包括:实例效果、任务要求、程序设计思路、技术要点、程序代码编写、重点代码分析以及任务拓展或技术拓展等。学习是从任务目标开始的,有针对性地引导学习者完成任务,轻轻松松地学会如何将技术用于实践,同时掌握针对项目要求进行制作的方法。拓展部分也是一大特色,利用任务拓展或技术拓展让学习者对技术有更深入的理解,并体现出技术的灵活运用,十分有利于积累开发经验。通过这样一系列范例的制作学习与训练,最终掌握网页交互设计的实用技能和最新技术。



全书分为 10 章,包括:JavaScript 基础,简洁而精炼地介绍语言结构和基础知识,通过一些短小代码示例清楚地说明其用法,易于初学者学习;对象应用,提供了 JavaScript 应用基础范例,学习如何在程序中应用 Date、String、Image 和 Style 对象等,并为后面复杂程序代码编写打下基础;动态栏效果,给出了在浏览器标题栏和状态栏区域内显示各种动态文字效果,同时带给用户以情趣;页面动态文字效果,学习如何让页面中文字鲜活地动起来,包括文字在单行文本框、多行文本框以滤镜方式动态变化等;时间应用,提供在页面上呈现各种时间和计时特殊效果;JavaScript 面向对象编程应用,是在基本掌握 JavaScript 基础上,重点讲解其独特的大部分人尚未知晓的面向对象程序设计;动态广告,给出新颖的展示信息变化或切换的效果;网页导航菜单,提供了具有各种灵活形式的网页导航菜单效果;动态位置变化效果,包括图片、图形或文字的位置随着鼠标操作而发生变化;jQuery 应用,jQuery 是目前非常流行的 JavaScript 库技术,本部分应用全面而简练地介绍了其实际应用。

在书稿撰写过程中,参考了一些书籍和网上资料,笔者对这些书籍和资料的作者表示最诚挚的谢意。

由于时间仓促、水平有限,在编程算法和技巧、JavaScript 面向对象编程等方面还需不断探索和总结,所以书中难免存在错误和不妥之处,恳请各位专家和同行批评指正。

所有意见和建议请发往:gzjckfb@163.com

欢迎访问我们的网站:<http://www.dutpgz.cn>

联系电话:0411-84707492 84706104

编 者

2010 年 4 月



第 1 章 JavaScript 基础	1
1.1 JavaScript 概述	1
1.1.1 JavaScript 的组成	1
1.1.2 JavaScript 的特点	2
1.2 JavaScript 基本语法	5
1.2.1 程序结构	5
1.2.2 JavaScript 的数据结构	7
1.3 JavaScript 程序基本构成	17
1.3.1 JavaScript 程序设计	18
1.3.2 函数	21
1.3.3 对象	22
1.4 JavaScript 面向对象编程	28
1.4.1 函数与对象	29
1.4.2 JavaScript 中函数的深入认识	32
1.5 CCS 样式语法基础	34
1.5.1 基本语法	34
1.5.2 选择符	34
第 2 章 对象应用	36
2.1 日期时间对象	36
2.1.1 显示当前星期	36
2.1.2 显示当前日期	38
2.2 字符串和图片对象	40
2.2.1 应用 String 对象截取特定文字	40
2.2.2 应用 Image 对象实现动画	42
2.2.3 Style 对象应用	44
第 3 章 动态栏效果	48
3.1 修改标题栏和状态栏的默认属性	48
3.1.1 利用 JavaScript 更改标题栏和状态栏显示内容	48
3.1.2 修改超链接在状态栏上的显示信息	49
3.2 在状态栏显示动态效果	52
3.2.1 在状态栏显示当前时间	52

3.2.2 状态栏文字由左端弹出显示	57
3.3 文字循环滚动效果	61
3.3.1 文字首尾相接循环滚动显示	61
3.3.2 状态栏文字在右端与左端之间循环滚动	63
第4章 页面动态文字效果	69
4.1 单行文本框中的文字特效	69
4.1.1 单行文本框文字动态移动	69
4.1.2 任务拓展1	71
4.1.3 任务拓展2	71
4.1.4 任务拓展3	73
4.1.5 任务拓展4	76
4.2 多行文本框动态效果	79
4.2.1 多行文本框的跳动小人	79
4.2.2 任务拓展1	84
4.2.3 任务拓展2	86
4.2.4 任务拓展3	86
4.3 文本框中的动态公告	88
4.3.1 多条公告显示	88
4.3.2 任务拓展:带图片的公告栏	94
4.4 页面中动态文字滤镜效果	100
4.4.1 文字光晕滤镜效果	100
4.4.2 任务拓展1	104
4.4.3 任务拓展2	105
4.4.4 任务拓展3	108
第5章 时间应用	112
5.1 日期时间显示	112
5.1.1 日期与数字时钟	112
5.1.2 任务拓展1	114
5.1.3 任务拓展2	118
5.2 网页中时钟动态效果	120
5.2.1 网页中图像时钟动态效果	120
5.2.2 网页中带有倒影时钟动态效果	122
5.2.3 指针时钟动态效果	125
5.3 特定日期计时	130
5.3.1 进入网页时间计时	130
5.3.2 倒计时天数	132
5.3.3 倒计时秒数	134
5.3.4 生日信息提示	136

第 6 章 JavaScript 面向对象编程应用	139
6.1 创建类和对象	139
6.1.1 使用 Object() 对象定义的形式	139
6.1.2 采用对象初始器定义的形式 Tson	140
6.1.3 采用构造器函数定义的形式	141
6.1.4 关于 JavaScript 中的函数与对象	142
6.2 对象数据封装及实例	142
6.2.1 JavaScript 中 OOP 的封装	142
6.2.2 通过匿名函数创建带有私有静态成员的类	145
6.2.3 带有公有静态成员的类	150
6.2.4 静态类	151
6.3 继 承	152
6.3.1 原型继承法	152
6.3.2 调用继承法	155
6.4 多 态	157
6.4.1 重载及其实现	157
6.4.2 覆盖的实现	158
6.5 JavaScript 的两种类型系统	159
6.5.1 基础类型系统	159
6.5.2 对象类型系统	161
6.5.3 函数在 JavaScript 面向对象机制中的五重身份	162
6.6 继承与多态	170
6.6.1 定义和赋值之间程序执行过程	170
6.6.2 this 和执行上下文	172
第 7 章 动态广告	175
7.1 动态文字消息	175
7.1.1 两个消息框同时滚动显示	175
7.1.2 消息框中渐变交替显示文字信息	180
7.1.3 消息框文字自下而上不停地滚动	184
7.2 图片广告效果	187
7.2.1 利用 CSS 技术弹出图像浏览器	187
7.2.2 控制图片左右滚动	190
7.3 图片渐变交替显示	197
7.3.1 图片渐变交替显示 1	197
7.3.2 图片渐变交替显示 2	202
7.3.3 图片渐变交替显示 3	206

第 8 章 网页导航菜单	211
8.1 树形目录导航设计	211
8.1.1 使用层对象设计树形目录	211
8.1.2 技术扩展:使用表格设计多级树形目录	219
8.2 利用 CSS 和 JavaScript 技术设计动态菜单	223
8.2.1 伸缩菜单	223
8.2.2 设计弹出菜单	231
8.3 页面移动菜单	237
8.3.1 浮在页面可移动导航菜单	237
8.3.2 浮在页面可移动和显示/隐藏导航菜单	240
8.4 推拉式导航菜单	244
8.4.1 单击推拉式导航菜单	244
8.4.2 指向推拉式浮动导航菜单	248
第 9 章 动态位置变化效果	257
9.1 动态对联广告	257
9.1.1 随滚动条移动的对联广告	257
9.1.2 QQ 在线咨询链接上下浮动型代码	262
9.1.3 任务拓展:位于页面带有 Flash 浮动广告的代码编写	265
9.2 鼠标控制的变化	269
9.2.1 跟随鼠标移动的蛇形文字	269
9.2.2 围绕鼠标旋转的尾巴	272
第 10 章 jQuery 应用	277
10.1 常用 jQuery 完成的任务	277
10.2 jQuery 简单示例	281
10.3 jQuery 应用	285
10.3.1 图片切换显示效果	285
10.3.2 利用 jQuery 呈现图片滚动效果	289

第 1 章

学习 JavaScript 之前应该具备条件：

- 对 Internet 和万维网 (WWW) 有基本了解；
- 对超文本标记语言 (HTML) 有良好掌握。

本章基本要求：

- 了解 JavaScript 语言基本组成；
- 通过与已经学过的程序语言对比, 理解 JavaScript 语言语法；
- 理解 JavaScript 面向对象编程。

1.1 JavaScript 概述

JavaScript 是 Web 项目开发中, 使用最为广泛的脚本编程语言之一, 能够处理相当多的任务。它既可以应用于 HTML 页面实现动态效果, 也可以应用在服务器端完成数据库访问和文件读取等工作。但多情况下用于动态网页中信息的控制、对表单数据的确认、创建复杂用户界面等 Web 页面交互设计及其页面特效。

1.1.1 JavaScript 的组成

JavaScript 是一种跨平台、具有面向对象编程特性的脚本语言。虽然它被更多地使用在浏览器上, 但同样能够用于服务器端。JavaScript 语言可以分为三个部分: JavaScript 核心语言、JavaScript 客户端扩展、JavaScript 服务器端扩展。

1. JavaScript 核心语言

它的核心部分包括 JavaScript 基本语法: 操作符、语句、函数和内置对象; JavaScript 内置对象: Array 对象、Date 对象和 Math 对象等。

2. JavaScript 客户端扩展

客户端运行的 JavaScript, 在核心语言基础上扩展了控制浏览器对象和文档对象模型 DOM (Document Object Model)。客户端运行的 JavaScript 程序, 将 JavaScript 核心语言部分和 JavaScript 客户端扩展结合起来, 可以对页面上的对象进行控制, 完成各种功能。

3. JavaScript 服务器端扩展

服务器端运行的 JavaScript, 是在核心语言基础上扩展了在服务器上运行需要的对象。这些对象可以与关系数据库互连、可以对服务器上文件进行控制、可以在应用程序之间交换信息。服务器端运行的 JavaScript 应用程序, 必须将 JavaScript 核心语言部分和 JavaScript 服务器端扩展结合起来。在服务器端使用 JavaScript, 可以分为两个方面:

- Netscape 服务器端 JavaScript;
- 活动服务器页面 ASP 中 JavaScript 编写的脚本功能。

服务器端 JavaScript 的核心语法包括:变量、数据类型、表达式、控制流程语句等,同客户端完全相同。但是,在运行于客户端和运行于服务器端的代码之间,会有很多差异。在应用于服务器端情况下,JavaScript 将在服务器上被解释甚至编译,其结果将作为一个标准的 Web 文档(HTML)传送到 Web 浏览器。

1.1.2 JavaScript 的特点

JavaScript 是一种用来提高开发 Web 页面效果的脚本语言,它能够使 Web 页面具有更好的交互性,给网页信息添加了各种动态的效果。它也是一种面向对象和事件驱动并具有安全性能的脚本语言。使用时将其嵌入 HTML 超文本标记语言内,实现网页上面向客户的各种增强效果。既可以对客户端数据进行操作,也可以对服务器端数据进行控制和调用。

虽然 Microsoft 启用了自己的服务器端脚本解决方案,即 ASP。但在实际面向 Web 开发中,仍然要使用 JavaScript 或 JScript 作为编写 ASP 应用程序的语言。在编写 JavaScript 的 ASP 代码的时候,用户可以用许多方式来告诉服务器运行一段脚本:

- 用 JavaScript 作为 ASP 语言,使用<% %>标签;
- 使用服务器端的 include;
- 带 runat="server"的属性的<script>标签来包含一段脚本。

前两种方法是传统用来在 HTML 页面内包含服务器端脚本的技术,最后一种是没有用的方法,它的格式与客户端脚本一样。

1. JavaScript 是脚本语言

它采用小程序段方式与 HTML 结合起来实现编程,并且是一种脚本语言。它的基本程序结构形式与其他编程语言十分相似,所不同的是它不需编译,而是在 Web 浏览器内由解释器逐行解释执行。每次运行程序的时候,解释器都会把程序代码翻译为可执行的格式。所谓解释器就是一个脚本引擎,它是浏览器的一部分。

JavaScript 语言,满足欧洲计算机制造商联合会(即 ECMA)制定的一个国际通用标准化版本 ECMAScript,目前是 ECMAScript2.6。

2. JavaScript 面向对象的特性

在 Java 和 JavaScript 中,同样都要使用到对象,但 Java 是一种基于类的语言,编程时需要先定义类,定义类的属性和方法,使用时必须创建类的实例。而 JavaScript 是一种基于原型的语言,编程时并不区分类和实例,在使用对象时也不需要关心对象的所有属性和方法。因此,常把 JavaScript 称为基于对象的语言。

3. JavaScript 与 HTML

人们使用 JavaScript 编程,是为了控制 HTML 网页上所显示的信息或对象,所以 JavaScript 代码必须与 HTML 结合。在将 JavaScript 嵌入 HTML 网页时,必须使用<script>标签。

使用<script>标签的一般格式为:

```
<script>
    JavaScript 程序代码
</script>
```

其中<script>是 HTML 中的一种扩展标签,JavaScript 代码写在标签内。浏览器通过标签才能够识别并解释其中的 JavaScript 代码。例如

```
<script>
    window. document. write("hello");// JavaScript 代码
</script>
```

程序运行结果是将 hello 写入(显示)在页面上。这里用到了 window 对象和 document 对象。window 对象是浏览器窗口的对象,window 对象之下有 HTML 的页面对象 document,document 对象有一个方法 write(),这个方法的功能就是将字符串写入当前页面中。

4. JavaScript 程序设计

在进行 JavaScript 程序设计时,JavaScript 程序可以嵌入在网页代码中的任何位置。通常是如下三种情况:

(1)将 JavaScript 程序放在网页中的<body> </body>主体部分。

示例 1-1 在页面上显示当前的日期:年、月、日。

```
<html>
<head>
<title>显示当前的日期</title>
</head>
<body>
今天是:
<script language="JavaScript">
    var today=new Date();           //定义生成一个日期对象实例;
    date=today. getDate();          //利用日期对象的日、月、年方法获得数据;
    month=today. getMonth();
    month=month+1;                  //实际月是所取得数据+1;
    if(month<=9)month="0"+month;    //为所显示月设定两位数显示格式;
    year=today. getYear();
    document. write(year,"-",month,"-",date); //利用文件对象的写方法显示数据;
</script>
</body>
</html>
```

(2) 将 JavaScript 程序放在网页中的 `<head>` `</head>` 之间, 使用一个 `showdate()` 函数。然后在 `<body>` `</body>` 主体部分调用。

示例 1-2 在页面上显示当前的日期: 年、月、日。

```
<html>
<head>
<title>显示当前的日期</title>
<script language="JavaScript">
    function showdate(){
        var today=new Date();
        date=today.getDate();
        month=today.getMonth();
        month=month+1;
        if(month<=9)month="0"+month;
        year=today.getFullYear();
        document.write(year,"-",month,"-",date);
    }
</script>
</head>
<body>
今天:
<script language="JavaScript">
    showdate();
</script>
</body>
</html>
```

(3) 将 `showdate.js` 文件保存在与该 HTML 文件在同一文件夹里, 然后在 `<body>` `</body>` 主体部分进行调用。利用了外嵌式文件, 使得网页程序更加简化了。

示例 1-3 在页面上显示当前的日期: 年、月、日。

```
<html>
<head>
<title>显示当前的日期</title>
</head>
<body>
今天:
<script language="JavaScript" src="showdate.js"></script>
</body>
</html>
```

使用一个外嵌式“.js”文件,即使用记事本编辑如下内容并保存为 showdate.js 文件。

```
var today=new Date();
date=today.getDate();
month=today.getMonth();
month=month+1;
if(month<=9)month="0"+month;
year=today.getFullYear();
document.write(year,"-",month,"-",date);
```

其实,脚本程序可以嵌入在 HTML 文件任何位置。例如<head></head>标记的下面。

5. JavaScript 的应用范围

JavaScript 扩展了网页中的 HTML 的功能,从而在网页设计中发挥重要作用。在实现网页设计的动态效果方面,JavaScript 可以用于:

- 页面修饰和特殊效果;
- 表单确认;
- 导航系统;
- 基本数学运算;
- 动态文档生成。

1.2 JavaScript 基本语法

1.2.1 程序结构

1. 基本结构

典型 JavaScript 程序如下:

```
<Script Language ="JavaScript">
    JavaScript 语言代码(语句);
    JavaScript 语言代码;
    ....
</script>
```

说明

每一句 JavaScript 都有类似的格式:每个语句以分号“;”结束。

语句块是用大括号“{ }”括起来的一个或 n 个语句。在大括号里边是几个语句,但在大括号外边,语句块是被当作一个语句的。语句块是可以嵌套的,也就是说,一个语句块里边可以再包含一个或多个语句块。

示例 1-4 在页面上显示文字：“这是用文档对象输出文字”。

```
<html>
<head>
<Script Language ="JavaScript">
    document.write("这是用文档对象输出文字");
    document.close();
</script>
</head>
<body>
</body>
</html>
```

说明

document.write()是文档对象 document 的输出函数,其功能是将括号中的字符或变量值输出到窗口;document.close()是将输出关闭。

可将<script>...</script>标识放入<head>... </head>或<body> ... </body>之间。当 JavaScript 标识放置在<head>... </head>头部之间,会使之在主页和其余部分代码之前装载,从而可使代码的功能更强大;当将 JavaScript 标识放置在<body>... </body>主体之间,以实现某些部分动态地创建文档。

2. JavaScript 中的变量

只要编写程序,就少不了变量和语句。

变量是用来存储可变的量或不变的量。从编程角度讲,变量是用于存储某种数值的存储器。它所储存的值,可以是数字、字符或其他的一些东西。

要同时满足以下变量的命名要求:

- 只包含字母、数字和/或下划线;
- 要以字母开头;
- 不能太长(其实又有什么人喜欢使用又长又臭的名字呢?);
- 不能与 JavaScript 保留字(Javascript 命令的字都是保留字)重复。

 **注意:**1)变量是区分大小写的。

2)命名变量时,最好能用清楚表达该变量在程序中的作用的词语。

变量如果是由多个单词组成的,那么,提倡第一个单词用小写,其他单词的第一个字母用大写。这与 JavaScript 的一些命令的命名是一致的。

程序代码编写时,声明变量的格式如下:

```
var 变量名称 [= 值];
```

var 为保留字,这个关键字用于声明局部变量。最简单的声明方法就是“var 变量;”,这将为变量准备内存,给它赋初始值“null”。如果加上“=值”,则给变量赋予自定的初始值“值”。

3. 注释

在程序代码编写过程中,经常要使用注释给程序员提供解释性信息,用于提高程序

的可读性。像其他编程语言一样,JavaScript 的注释在运行时也是被忽略的。

JavaScript 注释有两种:

- 单行注释。用双反斜杠“//”来标记,后面文字为注释。
- 多行注释。用“/*”和“*/”括起来标记,标记之间可以是一行或多行文字。

提示

如果程序需要草稿,或者需要让别人阅读,注释能帮上大忙。养成及时添加注释的习惯,能节省你和其他程序员的宝贵时间,使他们不用花费额外的时间琢磨你的程序。在程序调试的时候,有时需要把一段代码换成另一段,或者暂时不用一段代码,这时最忌用 Delete 键,如果还需要那段代码怎么办? 最好还是用注释,把暂时不要的代码“隐”去,到确定方法以后再删除也不迟。

4. 隐藏代码

编写 JavaScript 代码时,可以考虑在不兼容的 Web 浏览器中把 JavaScript 代码隐藏起来。如果 HTML 文档包含嵌入 JavaScript 代码而不是调用一个外部 .js 源代码文件,那么不兼容的 Web 浏览器就会把代码当作标准的文本显示出来。因此,为了预防遇到不兼容的浏览器,就应该将嵌入的 JavaScript 代码隐藏。具体做法是:把<Script>与</Script>标签之间的某些代码段,使用 HTML 的注释以“<!--”开始,以“-->”结束,让所有位于注释标签之间的代码都不会被浏览器提交而显示,达到隐藏的目的。

1.2.2 JavaScript 的数据结构

JavaScript 语言同其他语言类似,有它自身的基本数据类型、表达式和算术运算符以及程序的基本结构。

1. 基本数据类型

JavaScript 提供了四种基本的数据类型,用来处理数字和文字。这四种基本数据类型是:

- 数值(整数和实数);
- 字符串型(用“”号或“'”括起来的字符或数值);
- 布尔型(使 true 或 false 表示);
- 空值。

JavaScript 的基本数据类型中,数据可以是常量,也可以是变量。由于 JavaScript 采用弱类型的形式,因而存放数据的变量或常量不必首先作声明,而是在使用或赋值时确定其数据的类型。当然也可以先声明该数据的类型,它是通过在赋值时自动说明其数据类型的。

整型常量。又称字面常量,它是不能改变的数据。该常量可以使用十六进制、八进制和十进制表示其值。

实型常量。是由整数部分加小数部分表示,如 12.32、193.98。可以使用科学计数法或标准方法表示:5E7、4e5 等。

字符型常量。使用单引号(')或双引号(")括起来的一个或多个字符。如 "This is a book of JavaScript"、"3245"、"ewrt234234" 等。

布尔型。常用于判断,只有两个值可选:true(表“真”)和 false(表“假”)。true 和

false 是 JavaScript 的保留字。它们属于“常数”。

2. 操作符

JavaScript 的操作符有赋值、比较、算术、位、逻辑、字符串和特殊操作符。下面描述操作符以及关于操作符优先级的一些信息。

JavaScript 所有操作符如表 1-1 所示。

表 1-1 JavaScript 操作符

操作符分类	操作符	描 述
算术操作符	+	(加法)将两个数相加
	++	(自增)将表示数值的变量加一(可以返回新值或旧值)
	-	(求相反数,减法)作为求相反数操作符时返回参数的相反数。作为二进制操作符时,将两个数相减
	--	(自减)将表示数值的变量减一(可以返回新值或旧值)
	*	(乘法)将两个数相乘
	/	(除法)将两个数相除
	%	(求余)求两个数相除的余数
字符串操作符	+	(字符串加法)连接两个字符串
	+=	连接两个字符串,并将结果赋给第一个字符串
逻辑操作符	&&	(逻辑与)如果两个操作数都是真,则返回真。否则返回假
		(逻辑或)如果两个操作数都是假,则返回假。否则返回真
	!	(逻辑非)如果其单一操作数为真,则返回假。否则返回真
位操作符	&	(按位与)如果两个操作数对应位都是 1,则在该位返回 1
	^	(按位异或)如果两个操作数对应位只有一个 1,则在该位返回 1
		(按位或)如果两个操作数对应位都是 0,则在该位返回 0
	~	(求反)反转操作数的每一位
	<<	(左移)将第一操作数的二进制形式的每一位向左移位,所移位的数目由第二操作数指定。右面的空位补零
	>>	(算术右移)将第一操作数的二进制形式的每一位向右移位,所移位的数目由第二操作数指定。忽略被移出的位
	>>>	(逻辑右移)将第一操作数的二进制形式的每一位向右移位,所移位的数目由第二操作数指定。忽略被移出的位,左面的空位补零
赋值操作符	=	将第二操作数的值赋给第一操作数
	+=	将两个数相加,并将和赋给第一个数
	-=	将两个数相减,并将差赋给第一个数
	*=	将两个数相乘,并将积赋给第一个数
	/=	将两个数相除,并将商赋给第一个数
	%=	计算两个数相除的余数,并将余数赋给第一个数
	&=	执行按位与,并将结果赋给第一个操作数
	^=	执行按位异或,并将结果赋给第一个操作数
	=	执行按位或,并将结果赋给第一个操作数
	<<=	执行左移,并将结果赋给第一个操作数
	>>=	执行算术右移,并将结果赋给第一个操作数
	>>>=	执行逻辑右移,并将结果赋给第一个操作数