

屈鹏飞 编著
雨人科技 审校

JavaScript 网页编程 案例教程



清华大学出版社
<http://www.tup.tsinghua.edu.cn>

网络工程师

JavaScript 网页编程 案例教程

屈鹏飞 编著

雨人科技 审校

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书通过一系列目前流行的应用实例,从基础起步,深入地介绍了应用 JavaScript 进行网络编程的技术。书中不仅给出了一些程序,而且对关键的程序、函数和对象都进行了细致的分析,使读者能够迅速掌握应用 JavaScript 进行网络编程的高级技巧。

全书共分 11 章和两个附录,第 1 章介绍了 JavaScript 的发展历程、特点以及 JavaScript 的变量设置、流程控制、模式匹配等基础知识。第 2 章到第 10 章是本书的核心,以 JavaScript 在实际的应用进行分类,详细地阐述了 JavaScript 在时间、状态栏和标题栏、文字、表单、图像、鼠标效果和对层的控制方法等方面的应用。第 11 章介绍了 JavaScript 的高级应用,包括制作导航菜单和在线游戏以及用于进行 Flash 动画编程的 ActionScript 脚本语言的一些相关知识和高级命令等。最后的两个附录分别给出了客户端和服务端 JavaScript 对象层次、JavaScript 保留字与特殊字符。

本书是从基础起步,难度逐渐增加的,因此适合有一定编程基础的大学生、科研人员和电脑爱好者学习;由于内含丰富的应用实例和语言细节,也适用于网页设计和网络编程的专业人员参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: JavaScript 网页编程案例教程

作 者: 屈鹏飞 编著

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 冯志强

印 刷 者: 北京密云胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 21.25 字数: 518 千字

版 次: 2002 年 6 月第 1 版 2002 年 6 月第 1 次印刷

书 号: ISBN 7-302-05514-9/TP·3242

印 数: 0001~5000

定 价: 30.00 元

前 言

伴随着计算机技术日新月异的发展和 Internet 的迅速普及,越来越多的人要使用网络,同时也有很多的个人和企业要在 Internet 上创建自己的网站。网络是一种新兴的传播方式,影响也将会越来越深入和广泛,很多企业希望能够利用好这种媒介,拓展新的发展空间,进入电子商务领域。这样对网站的服务也提出了新的要求。其中重要的一点就是开发动态的具有交互性的网页,这个要求对传统的 HTML 语言来说,是很难实现的。而 JavaScript 就可以做到这一点。将 JavaScript 和传统的 HTML 语言结合,就可以在目前比较流行的两种浏览器——Microsoft Internet Explorer 和 Netscape Navigator 上实现动态实时的效果,其独特的对表单进行检验的功能也将大大降低服务器的负荷,提高效率。

JavaScript 是一种相对简单易学的编程语言。因为 JavaScript 是镶嵌在网页里并由浏览器执行的,所以想编好 JavaScript 脚本应在学习之前应该对网页制作语言 HTML 有个初步的了解,尤其注意 HTML 中各个标签的含义和使用方法。本书中也使用了一些 HTML 中所特有的符号,如“ ”就代表半角状态下的一个空格。详细了解这些标签和符号将方便你理解本书中的例子和在实际中的应用程序。由于 JavaScript 的风格类似于自然语言,对读者来说可以轻松地在较短的时间内学会并掌握它,甚至不需要学过相关的计算机语言作基础。当然如果先学过 C/C++、Java 等语言,学习本书将会节省更多的时间。

本书是面向初、中级读者的,尤其是一线的网页制作者。本书包含的大量高质量的范例程序具有高度的实用性,其中大多数程序可以立即应用在网页设计中。为了方便读者学习,本书本着由浅入深的方针,首先介绍了 JavaScript 的基础概念,如 JavaScript 的由来、发展方向、JavaScript 语言的特点。之后介绍了 JavaScript 的编程基础如变量设置,流程控制,模式匹配等知识,也展示了一些 JavaScript 的编程技巧。本书的重点是一些核心对象及其实际应用,在编写的过程中相关的内容被归在同一章,包括获取并显示系统时间,控制浏览器的状态栏,改善文字效果,校验表单,图像显示,光标特效,对层的控制等等,使读者对这种语言的应用有一个总体的把握。由于 JavaScript 的语言基础和 Flash 的 ActionScript 非常相似,在本书的最后一个部分介绍了 ActionScript 的一些相关知识和高级命令,使本书的读者能够迅速地转入到 Flash 的 ActionScript 动画编程上去。

在我们的网站 <http://www.yurennet.com> 中有本书的实例光盘,其中包含了本书中的

4357/10

范例程序。除了范例程序之外，本书还对范例进行了扩充，对书中的范例给出了大量的范例扩充程序。这些范例和范例扩充程序一定会让你在 JavaScript 网络编程中感到游刃有余。

本书的顺利出版得到了雨人网络公司姜云峰等朋友的大力支持，在此表示真诚的感谢。在本书的编写过程中，也得到了很多来自同事们和朋友们宝贵的建议，在此一并表示感谢。

在本书的编写过程中，虽然作者力求准确无误，但限于学识和经验的不足，对于书中出现的欠妥的地方还恳请读者批评指正。

编 者

2002年1月

目 录

第 1 章 JavaScript 专业效果设计概述.....	1
1.1 JavaScript 特点.....	2
1.1.1 JavaScript 语言的特点.....	2
1.1.2 JavaScript 的应用范围.....	8
1.1.3 JavaScript 前景展望.....	11
1.2 JavaScript 的编程内核.....	13
1.2.1 JavaScript 的程序结构.....	13
1.2.2 JavaScript 的语法.....	14
1.2.3 JavaScript 的数据类型和变量.....	19
1.2.4 JavaScript 编程的流程控制.....	22
1.2.5 JavaScript 的函数的使用.....	28
1.2.6 常见的几种对象简述.....	34
1.3 本章小结.....	44
第 2 章 JavaScript 的时间效果.....	45
2.1 Chinaren 主页中的 JavaScript 运用.....	46
2.2 时间的获取和显示.....	48
2.2.1 时间的获取.....	48
2.2.2 动态显示当前的时间.....	50
2.3 倒计时程序和复杂的时间效果.....	53
2.3.1 静态倒计时程序的创建.....	53
2.3.2 创建动态显示的倒计时时间.....	55
2.3.3 其他的时间显示效果.....	57
2.4 JavaScript 时间在网络上的实际应用.....	64
2.4.1 设计一个给朋友的生日贺卡.....	64
2.4.2 定时弹出广告条或分时弹出广告条.....	68
2.5 Date 对象的属性和方法.....	72
2.5.1 与 Date 对象相关的参数简介.....	72
2.5.2 get 方法.....	73
2.5.3 set 方法.....	75
2.5.4 其他属性和方法.....	75
2.6 本章小结.....	81

第 3 章 控制状态栏和标题栏的显示	82
3.1 修改状态栏的默认属性	83
3.1.1 初识状态栏	83
3.1.2 与状态栏有关的 JavaScript 对象	83
3.1.3 设定状态栏的默认信息	84
3.1.4 修改超链接在状态栏的显示信息	84
3.2 状态栏的动态显示效果	88
3.2.1 状态栏里动态显示时间	88
3.2.2 状态栏里的跑马灯效果	89
3.2.3 在状态栏里创建一个闪烁文本	91
3.2.4 状态栏里的打字效果	92
3.3 复杂的运动效果	93
3.3.1 文字挤压效果	93
3.3.2 复合的运动效果	96
3.3.3 可以控制的文字从右边弹入的效果	97
3.4 标题栏效果	100
3.4.1 标题栏效果和状态栏效果的区别	100
3.4.2 标题栏里的闪烁的效果	101
3.4.3 标题栏里创建动画效果	103
3.5 本章小结	108
第 4 章 改善文字效果	109
4.1 用 JavaScript 改变文字属性	110
4.1.1 文字属性简介	110
4.1.2 JavaScript 关于文本格式方面的对象、方法和属性	113
4.2 使用滤镜	117
4.2.1 滤镜简介	117
4.2.2 变色文本的阴影运动效果	119
4.2.3 使用 glow (闪烁) 滤镜	121
4.3 交互显示的文字效果	123
4.4 动态文本的创建	125
4.4.1 文字的旋转与变色	125
4.4.2 文字从远处拉近效果的实现	127
4.4.3 基于文本框的跑马灯设计	129
4.5 垂直运动的字幕设计	131
4.5.1 marquee 简介	131
4.5.2 垂直运动字幕设计	132
4.6 本章小结	134

第 5 章 JavaScript 在表单元素上的应用	135
5.1 表单总览	136
5.1.1 查询、搜索、登录使用的表单	136
5.1.2 用户申请和免费邮件申请	136
5.1.3 用于在线购物的表单设计	137
5.2 检验表单元素的方法	138
5.2.1 alert 方法和 JavaScript 警告框的设计	138
5.2.2 button	140
5.2.3 checkbox	141
5.2.4 text 文本框	144
5.2.5 radio 检查框	146
5.2.6 textarea 文本域	151
5.2.7 select 与 option 创建选择框	152
5.2.8 reset、submit、hidden	155
5.3 表单元素对事件的触发和处理	158
5.3.1 焦点的获得与失去	158
5.3.2 单击的模拟和处理	160
5.3.3 对 onChange 事件的处理	162
5.3.4 select 和 onSelect	163
5.4 利用 JavaScript 对用户填写表单进行指导	165
5.4.1 原始 HTML 页面的创建	165
5.4.2 JavaScript 效验程序的设计	165
5.5 表单的其他应用	168
5.5.1 在线购物的计算功能——对表单的数学处理	168
5.5.2 浏览器功能的模拟	171
5.6 本章小结	175
第 6 章 JavaScript 的图像操作	176
6.1 如何使用 JavaScript 来显示图片	177
6.1.1 HTML 显示图片的方法	177
6.1.2 使用 JavaScript 来显示图片	180
6.2 深入使用 Image 对象	182
6.2.1 使用 JavaScript 来控制图片的格式	182
6.2.2 Image 对象的方法和事件	189
6.3 翻转图片的制作	193
6.3.1 基本原理和程序	193
6.3.2 翻转图片在实际中的应用	194
6.3.3 控制更多的图片和页面元素	198

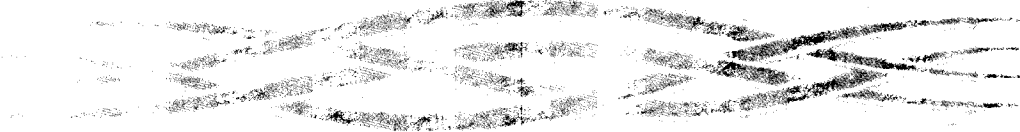
6.4	使用 JavaScript 和滤镜来创建页面特效	201
6.4.1	图片透明度的周期变化	201
6.4.2	使图片的透明度响应鼠标事件	202
6.5	使用 JavaScript 创建动画	204
6.5.1	单个图片的运动	204
6.5.2	改变图片的 src 属性实现动画	206
6.6	本章小结	210
第 7 章	JavaScript 鼠标效果	211
7.1	扩展鼠标效果	212
7.1.1	鼠标效果总览	212
7.1.2	在程序中使用鼠标效果	215
7.2	设计图片跟随鼠标的效果	216
7.2.1	动态获得鼠标的位置	217
7.2.2	图片跟随鼠标效果的创建	218
7.2.3	文字跟随鼠标效果	220
7.3	常见鼠标动态效果	222
7.3.1	三色鼠标效果	223
7.3.2	文字随鼠标跳跃	226
7.3.3	彩色环环绕鼠标的 3D 运动	228
7.4	本章小结	230
第 8 章	JavaScript 的窗口操作	231
8.1	window 对象方法总览	232
8.1.1	window 对象方法与属性总览	232
8.1.2	窗口焦点的取得与移出	234
8.1.3	back、forward、home 方法和 history 属性的应用	235
8.1.4	激活“打印”对话框	237
8.1.5	确认对话框的创建	239
8.1.6	获得当前窗口的详细信息	241
8.2	控制窗口的大小和位置	242
8.2.1	窗口的定位	242
8.2.2	窗口大小的控制	245
8.2.3	窗口大小和定位的综合应用	248
8.3	新建窗口与 E-mail 预览功能的实现	250
8.3.1	新建窗口	250
8.3.2	E-mail 预览的实现	255
8.4	弹出式窗口的设计	258

8.4.1	使用 JavaScript 导入已有页面	259
8.4.2	使用外嵌式.js 文件来实现	260
8.5	窗口的关闭	260
8.5.1	关闭当前窗口	261
8.5.2	关闭当前页面的子页面	266
8.6	其他关于 window 对象的应用技巧	267
8.6.1	获取访客的姓名	267
8.6.2	加入收藏夹、设为主页的设计方法	269
8.7	本章小结	270
第 9 章	JavaScript 中的密码	271
9.1	客户端密码的特点和 Password 对象	272
9.1.1	客户端效验密码的方式	272
9.1.2	Password 对象简介	272
9.2	面向个人主页的 JavaScript 密码设计	276
9.2.1	最简单的密码设计	276
9.2.2	使用密码表	279
9.2.3	使用外嵌式文档	280
9.3	常用的构建密码的方法	282
9.3.1	使用特殊字符作为密码	282
9.3.2	使用链接作为密码	284
9.4	密码和源代码的保护	285
9.4.1	禁止使用右键	285
9.4.2	使用 prompt() 的方法	287
9.5	本章小结	288
第 10 章	对框架、层、Cookies 的操作	289
10.1	从 freemail.263.net 学习框架网页的使用	290
10.1.1	创建最简单的框架页面	290
10.1.2	框架网页的嵌套	292
10.2	使用 JavaScript 对框架进行操作	293
10.2.1	Frame 对象	293
10.2.2	使用 JavaScript 来管理跨框架的通信	297
10.3	用 JavaScript 来控制层	301
10.3.1	层的显示和隐藏的控制	302
10.3.2	层的运动的控制	303
10.4	使用 Cookies	304
10.4.1	Cookie 的属性	305

10.4.2	读写 cookies 的方法	305
10.4.3	应用 cookie 的范例	306
10.5	本章小结	308
第 11 章	JavaScript 的高级应用与 ActionScript	309
11.1	制作 JavaScript 菜单	310
11.1.1	简单的菜单设计	310
11.1.2	树形菜单的设计	311
11.2	在线游戏的设计	316
11.2.1	JavaScript 游戏: 石头、剪子、布	316
11.2.2	射击游戏的设计	318
11.3	ActionScript 命令简介	322
11.3.1	ActionScript 的语法基础	322
11.3.2	使用属性	324
11.4	ActionScript 动作简介	325
11.4.1	常见事件说明	325
11.4.2	Flash 5 自带动作	326
11.5	本章小结	327
附录 1	客户端和服务端 JavaScript 对象层次	328
附录 2	JavaScript 保留字与特殊字符	329

第 I 章

JavaScript 专业效果设计概述



JavaScript 语言的特点

JavaScript 的应用范围

JavaScript 的前景展望

JavaScript 的编程内核

几种常见的 JavaScript 对象简述



本章要建立起关于 JavaScript 的明确的概念,包括它的由来、特点、应用情况和发展前景等。本章的内容将有助于理解 JavaScript 的深刻内涵,并对如何应用这种语言给出最客观、有效的指导,其中涉及到的内容——JavaScript 的应用方向和编程内核,将是本章要讨论的核心。

1.1 JavaScript 特点

JavaScript 是一种面向对象、跨平台、结构化、多用途的语言,可以根据不同的要求,不同的平台,设计出各种类型的解决方案。其主要特点如下。

1.1.1 JavaScript 语言的特点

1. JavaScript 的由来

1995 年 12 月 4 日, Netscape 公司与 SUN 公司合作,推出了 JavaScript 1.0,当时称为 LiveScript。与已经存在的 Perl 等基于服务器的计算机语言相比,它能够在当时的 Netscape Navigator 2.0 浏览器中进行编译操作,也就意味着可以在客户端运行,具有巨大的优势,于是就飞速发展起来。通过 JavaScript,网站的制作者可以预先设计合乎自己要求的程序,并使其在客户端运行,这很受欢迎。不但如此, Netscape 公司和 Microsoft 公司还分别实现了服务器端的 JavaScript。这使开发者可以在服务器端使用与浏览器页面上相同的程序语言。

Netscape 通过把数据库连接功能加入到语言中去实现了 Liveware。Liveware 是一个基于 JavaScript 的服务器端的网站管理和应用程序开发环境。利用 JavaScript 的优良的性能, Liveware 能简化对 CGI 编程的要求,实现更紧凑的“客户端-服务器端”的应用程序开发。可以说,是 JavaScript 在简单的 HTML 和服务器端复杂的 CGI 程序之间架设起一座桥梁。

Microsoft 公司在其 Internet Explorer 3.0 浏览器中实现了 JScript。与 Netscape 公司一样, Microsoft 公司也是在服务器端执行这种语言。JScript 也是一种解释性的语言,可以看成是 JavaScript 的分支。

随着 JavaScript 被广泛的接受, JavaScript 技术本身也逐渐形成了一种标准。1997 年,欧洲计算机制造商联合会 (ECMA) 将其标准化,于 1997 年 6 月正式发行了 ECMAScript 1.0。在 1998 年 4 月国际标准组织与国际电子技术协会通过了这个标准。ECMAScript 正在被越来越多的组织承认。从 ECMAScript 的角度看, JavaScript 和 JScript 就是 Netscape 公司和 Microsoft 公司分别对 ECMAScript 的实现的不同的手段。

2. 客户端 JavaScript 和服务端 JavaScript

JavaScript 既可以用于客户端,也可以用于服务器端,但二者的使用有所区别,本小节主要展现这些区别并简单介绍服务器端 JavaScript。

(1) 编程区别

首先在程序开头的地方二者有所区别。客户端使用 `<script> </script>` 标签,服务器端使用 `<server> </server>` 标签。在一个页面里可以既有客户端代码,也可以有服务器端代码,在服务器上,客户端的代码不会被服务器解释,在客户端服务器端的代码也不会被浏览器所解释。

其次,服务器端有扩展的对象、方法、属性和事件。另外有些客户端存在的对象、方法、属性和事件在服务器端不存在,二者拥有的功能不一致,编程时要注意,否则可能出错。

(2) 编译脚本区别

服务器端 JavaScript 经过编译以后生成二进制代码组成的“.web”文件。这个文件包括了关于网络应用的所有文件,包括“.html”和“.js”文件。而客户端则简单地由浏览器解释后就可以执行了。

(3) 服务器端 JavaScript 的应用

服务器端 JavaScript 有着较强的同数据库连接的能力。它可以连接 DB2、Informix、Oracle、Sybase 等数据库,通过 ODBC 它也可以同目前最流行的 Microsoft 的 SQL 的数据库连接。服务器端 JavaScript 的数据库连接是在线进行的,连接成功以后就可以使用 SQL 语句进行数据库操作。

可以使用服务器端 JavaScript 创建发送电子邮件的系统,很多著名的网络公司都是这样做的。服务器端有一个 SendMail 对象来实现发送电子邮件。

3. JavaScript 面向对象的特性

在学习 JavaScript 面向对象的特性之前,先来认识一下“对象”的概念。

现在流行的编程语言如 C++, Java, Perl 等可以大致分成两类,其中类似于 Java/C++ 的属于面向对象(OO)的语言;其他的则称为非面向对象的语言。面向对象的语言编程(OOP)需要建立起对象和类的概念。通过预先创建和定义一些对象或类,依靠这些对象和类就可以生成各种新的子对象,这些子对象与预定义的对象具有相同的特征。这种现象被称为对象的继承。

了解这些对象和这些对象彼此之间的关系是非常重要的,只有了解了这些基本的对象在客户端基本的层次,才能够做到正确地引用这些对象和相关的属性。

在服务器端,JavaScript 的核心对象是 DbPool 和 Database。通过这两个对象和从它们派生的对象,就可以实现对数据库的操作,进行运算和存储。

服务器端的对象层次主要如图 1.1 和图 1.2 所示。

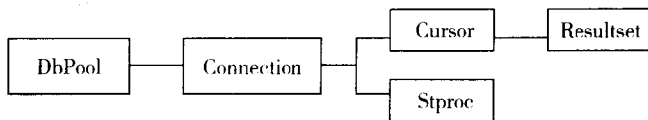


图 1.1 Dbpool 对象层次

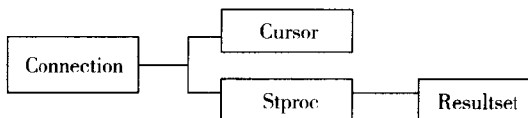


图 1.2 connection 对象层次

了解 JavaScript 的对象层次相当重要，在本书以后的部分会对范例中出现的 JavaScript 对象予以一定的解释和拓展。

4. 结构化程序设计

在 JavaScript 里，可以通过定义自己的函数，使用外嵌式的“.js”或者其他的相关文件使程序更加清晰明了。

比如，可以通过很多种方式获得计算机当前的时间。以下的例子尝试其中等效的三种。

第一种，不使用函数，也不使用外嵌式“.js”文件。

程序清单 1-1: 应用 JavaScript 的三种方式(1)

```

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

第二种, 在<head>与</head>之间使用一个 showdate()函数。需要说明的是, 在JavaScript 的实际应用中要经常设计各种功能的函数。设计并调用 JavaScript 函数是实现各种复杂的 JavaScript 效果的基础。

程序清单 1-2: 应用 JavaScript 的三种方式(2), 使用函数

```
<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.getYear();
document.write(year,"-",month,"-",date);
}
//-->
</script>
</head>
<body>
今天是:
<script language="JavaScript">
<!--
showdate();
</script>
// - - >
</body>
</html>
```

这个例子似乎比上个例子更麻烦一些, 但是使用了函数 showdate()以后, 一旦在程序中或者页面的某一个部位需要显示时间, 就可以用以下的代码段轻易实现:

```
<script language="JavaScript">
<!--
showdate();
//-->
</script>
```

实际上给程序设计提供了极大的方便。

第三种, 使用一个 showdate.js 的 JavaScript 文件。首先要创建这个 JScript 文件。将上面的一部分程序复制下来, 如下所示:

```
var today=new Date();
date=today.getDate();
```



```
month=today.getMonth();  
month=month+1;  
if(month<=9) month="0"+month;  
year=today.getYear();  
document.write(year, "-", month, "-", date);
```

然后将这个文件存成 showdate.js, 所需要的 JavaScript 文件就准备好了。以后的工作就是在需要的地方引用它。使用了 JScript 文件以后的页面源程序如下:

程序清单 1-3: 应用 JavaScript 的三种方式(3), 使用 .js 文件

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

可以看到, 源文件更加简洁了, 更为重要的是, 现在这个 showdate.js 文件可以在很多地方使用了。这样就可以节省大量的空间。使用“.js”文件可以为编写程序带来巨大的方便。具体操作时, 可以将经常使用的小程序编成很多个小的“.js”文件, 然后可以在程序中方便地使用了。修改这些“.js”文件也是相当方便的, 不必在页面里一一去修改源代码。

以上三个例子的效果是完全一致的, 在浏览器中的效果如图 1.3 所示。



图 1.3 获得当前的日期

通过以上的三个例子, 应该建立起结构化程序设计的概念, 并掌握相关的方法。JavaScript 的编程比较随意, 风格容易松散, 但并不是说 JavaScript 就不需要漂亮的风格。一个好的编程风格对任何一种计算机语言来说都是非常重要的, 所以应学好结构化程序设计, 让自己的程序更易懂, 更清晰。

5. JavaScript 的安全性

安全性始终为用户最关心的问题之一。一个应用程序安全水平的高低将决定用户是