

高等职业院校教学改革创新示范教材

软件开发系列

JavaScript

程序设计基础与范例教程

◎ 曹维明 主 编

◎ 刘 燕 吴剑文 赵 卉 副主编

- ✓ 涵盖JavaScript主流应用，注重实用性及前瞻性，注重实践教学
- ✓ 知识讲解循序渐进，符合学习规律，源自作者多年软件开发和教学经验
- ✓ 实例丰富，160个课堂案例，包含案例学习目标、知识要点、代码和案例分析
- ✓ 辅以100个有针对性的编程练习题，给出解题提示，注重启发学习者的编程思维

登录华信教育资源网，获取PPT课件、源代码等教学资源：

<http://www.hxedu.com.cn>



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

高等职业院校教学改革创新示范教材·软件开发系列

JavaScript 程序设计 基础与范例教程

曹维明 主 编

刘 燕 吴剑文 赵 卉 副主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书介绍了 JavaScript 脚本语言的基础知识和实用技术。全书共 11 章, 内容包括 JavaScript 概述、JavaScript 编程基础、流程控制语句、对象编程、本地对象、浏览器对象 (BOM)、文档对象 (DOM)、事件处理、函数特性、数据交换格式与数据持久化等技术, 并应用本书所讨论的技术内容制作一个服饰设计网站。

本书配有大量的示例和练习, 介绍详略得当, 所介绍的技术具有很强的实用性、前瞻性, 符合市场就业需求。读者通过本书的学习可以快速掌握 JavaScript 编程基本功。本书可作为高职院校计算机相关专业的教材, 也适合 JavaScript 初学者及程序开发人员学习参考。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

JavaScript 程序设计基础与范例教程/曹维明主编. —北京: 电子工业出版社, 2014.6
高等职业院校教学改革创新示范教材·软件开发系列
ISBN 978-7-121-23403-3

I. ①J… II. ①曹… III. ①JAVA 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 116743 号

策划编辑: 左 雅

责任编辑: 左 雅

印 刷: 涿州市京南印刷厂

装 订: 涿州市京南印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 19.25 字数: 493 千字

版 次: 2014 年 6 月第 1 版

印 次: 2014 年 6 月第 1 次印刷

印 数: 3 000 册 定价: 39.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zits@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

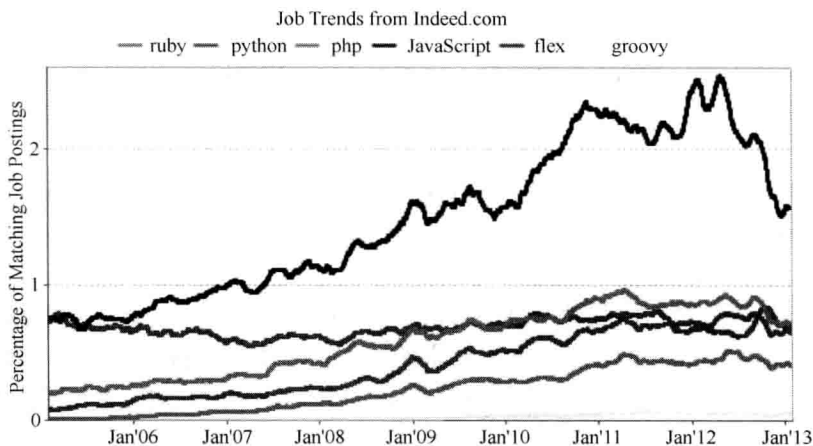
前 言

技术背景

JavaScript 是一种广泛用于客户端网页开发的脚本语言，早期主要用来给 HTML 网页添加动态功能。随着 Web 开发技术领域的迅速发展，JavaScript 以其跨平台、容易上手等优势大行其道。有些特殊功能（如 AJAX）必须依赖 JavaScript 在客户端进行支持。随着 Google-V8 引擎和 Node.js 框架的发展，JavaScript 逐渐被用来编写服务器端程序。HTML5 和移动设备的广泛应用，也会使 JavaScript 的前景更加绚丽。

就业需要

JavaScript 脚本语言是开发 Web 应用程序所必备的技术。随着 HTML5 的广泛应用，就业岗位对于 JavaScript 的需求会越来越强烈。JavaScript 骤然变成了聚光灯下的明星语言，越来越多的开发者加入到 JavaScript 阵营。下图是 Indeed.com 对 Web 开发技术岗位需求的统计。



由趋势分析图可以看出，JavaScript 的就业岗位需求量远远超过其他语言。

本书编写目的

很多高职高专院校已将 JavaScript 作为一门重要的程序设计课程。本书在编写过程中力求突出高职教育的特点，以“应用”为主旨和特征，以“实践教学”为主要目的，以“培养学生的专业技术应用能力和职业操作技能”为教材编写重点，注重对学生编程思想的启发和培养，提高他们的分析问题、解决问题的能力，将教学内容与职业培养目标相结合。通过本书的学习，读者可以掌握如下技能：

- JavaScript 语法基础；
- 处理 JavaScript 内置对象，如 String、Array、Number、Math、RegExp 等；
- 使用 BOM、DOM 控制文档元素；
- 处理网页事件；
- 使用不同的方式持久化数据；

- 访问常见元数据，如 XML、JSON。

本书主要内容

本书所讨论的内容包括 JavaScript 基本知识和实用技术，以及不断发展、与时俱进的新技术、新应用。内容讲解由浅入深、循序渐进，将理论知识与实践相结合，案例选取实用性强、针对性强。

第 1 章介绍了 JavaScript 脚本语言的作用和特点，如何学习 JavaScript 语言，如何选择适合自己的开发工具，如何添加注释，如何选择浏览器，如何将 JavaScript 嵌入到 HTML 页面并运行调试，以及如何养成良好的代码书写规范。

第 2 章介绍了 JavaScript 函数的基本用法，包括函数的定义及调用、参数的用法、返回值的用法、函数编写规则。函数是 JavaScript 的灵魂，将函数放在本书第 2 章讲述，以便让后面章节的课堂案例及练习使用函数实现，使读者切实掌握函数的用法。

第 3 章介绍了 JavaScript 编程的基本元素，包括基本数据类型、常量、变量、运算符、表达式以及类型转换。

第 4 章介绍了 JavaScript 流程控制语句的使用，掌握选择语句、循环语句、循环控制语句是任何程序设计语言的基本功。

第 5 章介绍了如何使用 JavaScript 自定义对象，如何私有化数据成员，使用 this 关键字访问自身数据，使用 with 关键字简化对象访问，使用 for...in 循环访问对象属性，以及 ECMAScript5 的对象保护功能等。

第 6 章介绍了 JavaScript 本地对象，包括 String、Number、Boolean、Array、Date、RegExp、Error、Global、Math 对象的用法。JavaScript 内置了很多对象，这使得开发工作轻松了许多。

第 7 章介绍了浏览器对象模型 (BOM)，包括 window、history、screen、location、navigator 对象的用法。通过本章的学习，读者可以掌握创建新的浏览器窗口、获取客户机屏幕信息、获取浏览器相关信息。

第 8 章介绍了文档对象模型 (DOM)，包括 DOM 核心接口的定义、DOM HTML 接口的定义、常见页面元素的操作、表单元素的操作、DOM 节点操作方法。通过本章的学习，读者可以通过 DOM 对象控制页面上的任何内容，包括修改元素属性、验证表单、使用画布元素等。

第 9 章介绍了 JavaScript 事件处理，包括事件触发、事件对象。通过本章的学习，读者可以处理常见事件，了解 DOM Event 接口，访问事件对象。

第 10 章介绍了几种元数据存储格式以及数据持久化技术，包括 XML 格式、JSON 格式、cookie、localStorage、sessionStorage 对象。通过本章的学习，读者可以使用 JavaScript 访问 XML 文档、访问 JSON 数据，使用 cookie、localStorage 和 sessionStorage 对象存储数据。

第 11 章综合运用 Web 开发技术来制作一个完整的服饰设计网站，使用 JavaScript 为网站添加动态、交互效果，验证表单数据，通过实践提高读者编程技法。

本书特色

1. 符合初学者学习特点

本书实例丰富、技术实用、结构安排合理，知识讲解循序渐进，对实例的分析清晰到位，让读者快速步入 JavaScript 开发殿堂。

2. 适合学校组织教学的需要

根据教学需要,设计有针对性的练习题,注重启发读者编程思维。课堂案例和练习准确地覆盖学习目标和知识要点,方便教师设计教学过程,方便学生了解学习目的。

3. 内容新颖,具有技术前瞻性

本书内容新颖多样、概念清晰、实用性强,涵盖当前主流 JavaScript 应用开发技术。许多其他同类教材所介绍的技术内容比较基础,技术更新比市场需求慢半拍。而本书更注重技术的实用性以及前瞻性,介绍了许多当前主流,或即将流行的新技术。

使用方法及课时分配

本书中有大量精彩的范例、练习题,这些素材来自作者多年来的软件开发、教育教学经验。本书内容在实际教学过程中运用多年,效果良好。计划学时为 108 学时,建议将课时分为讲解与实训两部分。

著作者分工

本书由曹维明担任主编,编写本书第 6 章至第 11 章内容。刘燕、吴剑文、赵卉担任副主编,编写本书第 1 章至第 4 章的内容,白建华参与编写了第 5 章的内容。

在本书编写过程中,我们本着科学、严谨的态度,力求精益求精,但错误、疏漏之处在所难免,请广大读者批评指正。

编 者

目 录

CONTENTS

第 1 章 JavaScript 程序概述	/1
1.1 学习 JavaScript 程序设计	/1
1.2 JavaScript 简介	/2
1.3 开发 JavaScript 应用程序	/5
【课堂案例 1-1】: 第 1 个 JavaScript 程序	/7
第 2 章 程序的构成——函数	/10
2.1 函数的使用	/10
【课堂案例 2-1】: 在网页上输出消息	/10
【课堂案例 2-2】: 在网页上显示图片	/11
【课堂案例 2-3】: 在网页中播放视频, 显示视频信息	/12
2.2 函数的参数	/13
【课堂案例 2-4】: 使用参数传递姓氏和名字, 在网页上输出姓名	/14
【课堂案例 2-5】: 定义函数 area(), 用于计算矩形面积	/15
【课堂案例 2-6】: 按照参数显示图片	/16
2.3 函数的返回值	/17
【课堂案例 2-7】: 计算任意 3 个数的平均值	/17
【课堂案例 2-8】: 测试 return 语句功能	/18
【课堂案例 2-9】: 制作简易杏仁巧克力	/19
【课堂案例 2-10】: 计算实发工资	/21
2.4 函数的嵌套定义	/21
【课堂案例 2-11】: 计算两个圆的面积之和	/22
*2.5 高级函数特性	/22
【课堂案例 2-12】: 调用匿名函数	/22
【课堂案例 2-13】: 使用匿名函数限制变量的作用域	/24
【课堂案例 2-14】: 使用闭包 (closure) 特性调整页面的字号	/25
【课堂案例 2-15】: 数组作参数, 计算购物总金额	/28
【课堂案例 2-16】: 使用参数对象 Arguments 来计算任意 n 个数的和	/29
【课堂案例 2-17】: 使用函数对象编写函数测试页面	/30
【课堂案例 2-18】: 使用函数递归, 输出递增的数字序列	/32
2.6 本章练习	/33

第3章 JavaScript 语言基础	/37
3.1 基本数据类型	/37
【课堂案例 3-1】: 数据类型测试及转换	/40
3.2 变量和常量	/41
【课堂案例 3-2】: 变量的定义和使用	/41
【课堂案例 3-3】: 变量的赋值	/42
【课堂案例 3-4】: 变量的作用域	/43
【课堂案例 3-5】: 使用常量 PI, 转换角度与弧度	/45
3.3 运算符和表达式	/46
【课堂案例 3-6】: 算术运算符使用示例	/46
【课堂案例 3-7】: 赋值运算符使用示例	/48
【课堂案例 3-8】: 使用关系运算符、条件运算符判断用户输入的年龄	/50
【课堂案例 3-9】: 逻辑运算符使用示例	/51
【课堂案例 3-10】: 字符串运算符示例	/52
【课堂案例 3-11】: 位运算符示例	/53
【课堂案例 3-12】: 使用 typeof 运算符检测数据类型	/55
3.4 本章练习	/55
第4章 JavaScript 语句	/59
4.1 JavaScript 语句和基本程序结构	/59
4.2 选择语句	/60
【课堂案例 4-1】: 使用单分支 if 语句判断两个数字中的较大数	/60
【课堂案例 4-2】: 使用 if 语句将两个数字按从小到大的顺序输出	/61
【课堂案例 4-3】: 使用 if 语句检查参数值的有效性	/62
【课堂案例 4-4】: 使用 if...else 语句计算数字的绝对值	/62
【课堂案例 4-5】: 使用 if...else 语句判断成绩是否及格	/63
【课堂案例 4-6】: 使用 if 嵌套删除文件(伪代码)	/65
【课堂案例 4-7】: 使用 switch 语句查询简单的日程表	/66
4.3 循环语句	/68
【课堂案例 4-8】: 使用 while 语句输出递增的数字序列	/68
【课堂案例 4-9】: 使用 while 语句在网页上显示一组图片	/69
【课堂案例 4-10】: 使用 do...while 语句计算 100 到 500 之间所有整数的和	/70
【课堂案例 4-11】: 使用 do...while 语句计算 m 到 n 之间所有偶数的和	/71
【课堂案例 4-12】: 使用 for 语句输出 1 到 n 之间所有的整数	/72
4.4 循环控制语句	/73
【课堂案例 4-13】: 使用 break 语句计算最小公倍数	/73
【课堂案例 4-14】: continue 语句演示	/74
4.5 循环嵌套	/74

【课堂案例 4-15】: 使用二重循环嵌套在页面上显示五子棋棋盘	/75
4.6 异常处理	/76
【课堂案例 4-16】: 使用 throw 语句抛出异常	/76
【课堂案例 4-17】: 使用 try...catch()...finally 处理异常	/77
4.7 本章练习	/79
第 5 章 基于原型的面向对象编程	/86
5.1 对象编程概述	/86
5.2 自定义对象的创建和使用	/87
【课堂案例 5-1】: 使用 Object 创建自定义对象 book, 用于描述图书信息	/87
【课堂案例 5-2】: 使用 Object 创建自定义对象 calc, 用于简单数学计算	/88
【课堂案例 5-3】: 使用构造函数创建自定义对象 phone, 用于描述电话信息	/89
【课堂案例 5-4】: 创建图片对象, 使用 this 访问对象自身的属性和方法	/90
【课堂案例 5-5】: 使用 with 简化对象操作	/91
【课堂案例 5-6】: 使用 instanceof 运算符判断对象类型	/93
【课堂案例 5-7】: 使用 instanceof 运算符检查参数的类型	/94
【课堂案例 5-8】: 使用 for...in 循环遍历对象成员	/95
【课堂案例 5-9】: 使用私有对象属性实现数据隐藏	/96
【课堂案例 5-10】: 为属性添加赋值方法 (Setter) 和取值方法 (Getter)	/97
【课堂案例 5-11】: 使用原型 (prototype) 扩展对象类型	/99
5.3 对象继承	/100
【课堂案例 5-12】: 使用 call() 方法实现对象继承	/100
【课堂案例 5-13】: 使用原型链 (Prototype Chain) 实现对象继承	/101
5.4 定义对象的不同方式	/103
【课堂案例 5-14】: 使用工厂函数方式创建对象	/103
【课堂案例 5-15】: 使用混合的构造函数/原型方式创建对象	/104
【课堂案例 5-16】: 使用动态原型方式创建对象	/106
5.5 本章练习	/107
第 6 章 本地对象	/109
6.1 本地对象概述	/109
6.2 Boolean 对象	/109
【课堂案例 6-1】: 比较布尔值与布尔对象的区别	/110
【课堂案例 6-2】: 复制布尔对象	/111
6.3 Number 对象	/112
【课堂案例 6-3】: 使用 Number 对象获取数值极限	/113
【课堂案例 6-4】: 将数字转换成字符串	/113
【课堂案例 6-5】: 设置数值精确度	/114
【课堂案例 6-6】: 数值进制转换	/115
6.4 String 对象	/116

【课堂案例 6-7】: 合成新的字符串	/118
【课堂案例 6-8】: 显示字符串的 Unicode 编码	/119
【课堂案例 6-9】: 截取字符串内容	/119
【课堂案例 6-10】: 在字符串中精确查找指定内容	/121
【课堂案例 6-11】: 在字符串中进行模糊查找	/122
【课堂案例 6-12】: 精确查找替换字符串内容	/123
【课堂案例 6-13】: 将字符串分割成数组, 提取英文句子中前 3 个单词	/124
【课堂案例 6-14】: 转换字母大小写	/125
【课堂案例 6-15】: 为字符串添加样式	/126
6.5 RegExp 对象	/127
【课堂案例 6-16】: 使用正则表达式替换字符串中的文本	/128
【课堂案例 6-17】: 使用正则表达式验证电子邮箱格式	/129
【课堂案例 6-18】: 使用正则表达式交换单词的位置	/130
6.6 Array 对象	/131
【课堂案例 6-19】: 使用 Array 对象创建数组	/132
【课堂案例 6-20】: 使用 for...in 循环遍历数组, 并找到最大值	/133
【课堂案例 6-21】: 对数组进行排序	/134
【课堂案例 6-22】: 使用 Array 提供的方法添加、删除或替换数组元素	/135
【课堂案例 6-23】: 将数组转换成字符串	/136
【课堂案例 6-24】: 使用现有数组元素生成新数组	/137
【课堂案例 6-25】: 使用回调函数处理数组元素	/139
【课堂案例 6-26】: 使用二维数组	/141
6.7 Math 对象	/143
【课堂案例 6-27】: 使用 Math 对象完成数学计算 1	/144
【课堂案例 6-28】: 使用 Math 对象完成数学计算 2	/144
6.8 Date 对象	/146
【课堂案例 6-29】: 创建 Date 对象	/148
【课堂案例 6-30】: 使用 Date 对象计算程序运行时间	/150
【课堂案例 6-31】: 使用 Date 对象的方法设置/获取日期时间信息	/151
6.9 Error 对象	/152
【课堂案例 6-32】: 使用自定义 Error 对象抛出异常	/153
【课堂案例 6-33】: 处理系统抛出的异常	/154
6.10 全局对象	/155
【课堂案例 6-34】: 使用全局方法	/156
6.11 本章练习	/157
第 7 章 浏览器对象模型 (BOM)	/167
7.1 浏览器对象模型概述	/167
7.2 window 对象	/168

【课堂案例 7-1】: 获取浏览器窗口的位置和大小	/169
【课堂案例 7-2】: 控制浏览器窗口的位置和大小	/170
【课堂案例 7-3】: 使用模式对话框	/173
【课堂案例 7-4】: 制作简单的数字时钟	/175
【课堂案例 7-5】: 在网页中实现滚动屏幕功能	/176
7.3 navigator 对象	/178
【课堂案例 7-6】: 获取浏览器及操作系统的相关信息	/179
【课堂案例 7-7】: 获取当前浏览器安装的插件信息	/180
7.4 location 对象	/181
【课堂案例 7-8】: 获取浏览器 URL 的相关信息	/182
【课堂案例 7-9】: 使用 location 对象实现页面跳转和刷新	/182
【课堂案例 7-10】: 创建页面导航	/184
7.5 history 对象	/186
【课堂案例 7-11】: 访问历史记录中的 URL	/186
7.6 screen 对象	/187
【课堂案例 7-12】: 获取用户屏幕信息	/188
【课堂案例 7-13】: 根据用户屏幕信息切换网页显示效果	/189
7.7 本章练习	/190
第 8 章 HTML 文档对象模型 (DOM)	/192
8.1 文档对象模型概述	/192
8.2 DOM 核心接口	/196
【课堂案例 8-1】: 获取 DOM 树中的节点信息	/200
【课堂案例 8-2】: 删除 DOM 树中的节点	/202
【课堂案例 8-3】: 在 DOM 树中添加子节点	/203
【课堂案例 8-4】: 替换 DOM 树中的节点	/205
【课堂案例 8-5】: 复制 DOM 树中的节点	/206
【课堂案例 8-6】: 获取节点的属性	/207
【课堂案例 8-7】: 控制文本节点	/209
【课堂案例 8-8】: 提取网页中的超链接地址	/211
8.3 DOM HTML	/212
【课堂案例 8-9】: 获取文档信息	/215
【课堂案例 8-10】: 修改文档中的链接	/216
【课堂案例 8-11】: 操作文档中的表格	/217
【课堂案例 8-12】: 获取文本框中用户输入的内容	/220
【课堂案例 8-13】: 获取单选框用户选择的内容	/221
【课堂案例 8-14】: 获取复选框用户选择的内容	/222
【课堂案例 8-15】: 控制下拉菜单	/223

【课堂案例 8-16】: 判断用户选取的文件类型	/225
【课堂案例 8-17】: 限制用户使用表单元素	/226
【课堂案例 8-18】: 验证表单数据	/227
【课堂案例 8-19】: 为所有段落加边框	/228
【课堂案例 8-20】: 选项卡效果	/230
【课堂案例 8-21】: Web 相册	/231
【课堂案例 8-22】: 修改网页背景色	/234
【课堂案例 8-23】: 显示/隐藏页面元素	/235
【课堂案例 8-24】: 覆盖显示图片	/236
【课堂案例 8-25】: 在网页中绘图 1	/238
【课堂案例 8-26】: 在网页中绘图 2	/239
8.4 本章练习	/240
第 9 章 事件 (Event) 处理	/245
9.1 事件处理概述	/245
9.2 基于 HTML 属性的事件处理方法	/245
【课堂案例 9-1】: 文档事件	/246
【课堂案例 9-2】: 鼠标事件	/248
【课堂案例 9-3】: 获得/失去焦点事件	/249
【课堂案例 9-4】: 键盘事件	/250
【课堂案例 9-5】: onchange 事件	/250
【课堂案例 9-6】: 使用 this 作参数	/251
【课堂案例 9-7】: 为事件设置响应函数	/252
9.3 DOM EVENT 事件处理	/253
【课堂案例 9-8】: 注册事件监听器, 设置背景图片	/256
【课堂案例 9-9】: 注册多个事件监听器, 实现简易加法计算器	/257
【课堂案例 9-10】: 事件指派	/258
【课堂案例 9-11】: 显示鼠标位置	/259
【课堂案例 9-12】: 创建快捷菜单, 缩放图片	/261
【课堂案例 9-13】: 创建快捷菜单, 缩放图片	/262
9.4 本章练习	/265
第 10 章 常用的数据交换格式和数据存储技术	/267
10.1 XML 文档	/267
【课堂案例 10-1】: 同步访问 XML 文档, 获取图书信息	/269
【课堂案例 10-2】: 异步访问 XML 文档, 设置段落样式	/270
【课堂案例 10-3】: XML 生成树状菜单	/272
10.2 JSON	/274
【课堂案例 10-4】: 使用 JSON 对象	/274

【课堂案例 10-5】 : 访问 JSON 对象中的对象	/276
【课堂案例 10-6】 : 使用 JSON 数组	/276
【课堂案例 10-7】 : 访问 JSON 对象数组	/277
10.3 数据存储	/278
【课堂案例 10-8】 : 使用 cookie 存储用户账户信息	/278
【课堂案例 10-9】 : 使用 localStorage 存储数据	/280
【课堂案例 10-10】 : 使用 sessionStorage 存储数据	/282
10.3 本章练习	/284
第 11 章 综合练习——服饰设计网站	/286
11.1 网站整体说明	/286
11.2 JavaScript 程序说明	/287

JavaScript 程序概述

JavaScript 是一种基于对象和事件驱动并具有相对安全性的客户端脚本语言，同时也是一种广泛应用于客户端 Web 开发的脚本语言，常用来给 HTML 网页添加动态功能，如响应用户的各种操作等。JavaScript 可用于 HTML 和 Web 编程，更可广泛应用于服务器、PC、笔记本电脑、平板电脑和智能手机等设备。

课堂学习目标：

- 了解 JavaScript 语言特色；
- 了解 JavaScript 开发工具；
- 了解浏览器对 JavaScript 的支持情况；
- 能够在页面中嵌入 JavaScript 脚本；
- 能够使用浏览器运行并调试 JavaScript 程序。

1.1 学习 JavaScript 程序设计

JavaScript 是目前互联网很流行的脚本语言，许多网页及其应用都用 JavaScript 来改进设计、验证表单、检测浏览器、创建 Cookies 等。JavaScript 也被公认为一种拥有开发现代应用程序所需的一整套新特性的语言。而随着当下 HTML5 和 Node.js 的流行，JavaScript 也将日益普及，将从过去装饰性的脚本语言转变为主流的编程语言。如果你是 JavaScript 的初学者，建议你在正式开始学习 JavaScript 之前先了解 JavaScript 脚本语言的特色，以及学习它所必备的知识。

1. 脚本语言

脚本语言 (Scripting Language) 是指由 ASCII 字符组成的，可以使用任何一种文本编辑器编写的简单的程序语言。脚本语言按运行环境划分，可以分为以下两类：

- 服务器端脚本语言：在服务器上执行的脚本语言，这类语言有 ASP、JSP、PHP、Perl、Ruby 等。这类脚本语言对服务器的要求较高、对客户端的要求较低。
- 客户端脚本语言：可以直接在客户端上运行，减少了服务器的负担。这类语言有 JavaScript、VBScript、JScript 等。

JavaScript 是一种脚本语言，它与其他编程语言 (C、C++、Java、C#等) 相比存在以下特点：

- JavaScript 由浏览器负责解释运行，源代码不需要编译；
- JavaScript 以文本文件形式存在，不能生成可执行文件；
- JavaScript 的语言结构比较松散、简单；

□ JavaScript 是一种基于对象的语言。

提示：虽然随着服务器的日益强壮，现在的一些程序员更喜欢运行于服务器端的脚本以保证程序安全，但 JavaScript 仍然以其跨平台、容易上手等优势大行其道。同时，有些特殊功能（如 AJAX）必须依赖 JavaScript 在客户端进行支持。随着 V8 引擎和框架 Node.js 的发展，及其事件驱动及异步 IO 等特性，JavaScript 也可以被用来编写服务器端程序。

2. 学习 JavaScript 前应具备的知识

本书第 2 章将开始详细介绍 JavaScript 的语法、对象、框架等知识，在学习这些内容之前读者应掌握以下知识：

- 具备基本的计算机操作技能；
- 对因特网（Internet）和 WWW（World Wide Web）有一定的了解；
- 会使用浏览器浏览网页；
- 掌握 HTML/XHTML 超文本标记语言；
- 掌握 CSS 层级样式表。

另外，如果读者具有其他编程语言（如 C、Java、PHP 等）的基础，将会对学习 JavaScript 产生莫大的帮助，但这不是必需的。

2

3. 学习 JavaScript 的建议

每种技术都有不同的特点，学习方法也不尽相同。本书为编程初学者提供一些学习 JavaScript 的建议，希望在读者学习过程中准确地抓住重点，少走弯路，学习愉快。

- 不要放过任何一个看上去很简单的小问题。它们往往可以引伸出很多知识点；
- 学习 JavaScript 并不难，难的是长期坚持实践和不遗余力的博览群书；
- 不要抵触英文学习资料，它们有时能够做到比中文更直观的表述；
- 学习编程最好的方法之一是阅读源代码；
- 看得懂的问题，请仔细看；看不懂的问题，硬着头皮看；
- 不要停留在集成开发环境（IDE）的摇篮上；
- 请把本书的程序例子亲自输入实践，即使配套光盘中有源代码；
- 如果学习者对书中的例子有新的想法，请及时将它扩充；
- 经常回顾自己以前写过的程序，并尝试重写，把自己学到的新知识运用进去；
- 不要漏掉书中任何一个练习题，请全部做完并记录下解题思路和困难；
- 不因为程序“很小”就不遵循某些规则，好习惯是培养出来的，而不是一次记住的；
- 记录下在和别人交流时发现的自己忽视的或不理解的知识；
- 请学习者不断地对自己写的程序提出更高的要求；
- 保存好编写过的所有程序，那是最好的积累和财富；
- 如果是浮躁的人，请不要学习 JavaScript；
- 如果学习 JavaScript，请热爱它。

1.2 JavaScript 简介

在本节中读者可以了解到 JavaScript 的发展过程和功能特点，以及一些具有 JavaScript

语言特色的演示程序。

► 1. JavaScript 的作用

虽然 JavaScript 是客户端脚本语言，而且结构松散、语法简单，但它能实现的功能却并不简单。JavaScript 常用来完成以下任务：

- 动态网页效果；
- 对浏览器事件做出响应；
- 读写 HTML 元素；
- 记录访问者状态；
- 网页游戏；
- 处理图形（WebGL）。

► 2. JavaScript 的局限性

JavaScript 并非无所不能，它的功能也存在一定的局限性：

- 大部分应用无法脱离浏览器；
- 不能访问网络数据库；
- 不包含联网技术；
- 虽然可以通过 WebGL 处理图形，但与本地应用程序相比还有一定的距离。

► 3. JavaScript 演示

以下示例展现了一些 JavaScript 语言的功能特点。

【演示 1】：Lomo 风格的电子相册

本例展示了一个 Lomo 风格的电子相册，由 HTML+JavaScript+CSS+jQuery 技术混合开发，兼容当前主流浏览器，能够响应网页事件，支持左右翻页、缩略图等功能，如图 1-1 所示。



图 1-1 演示 1 运行效果截图

演示网页所在位置：配套资源/source/chapter1/Slider/index.html。

【演示 2】：放大镜特效

本例演示了响应用户事件，读写 HTML 元素，使用 JavaScript 获取鼠标在左侧小图中的位置，并在右侧显示相应的大图，来模仿购物网站的放大镜效果，如图 1-2 所示。

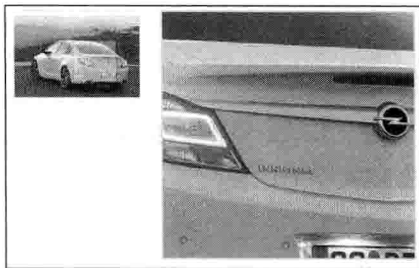


图 1-2 演示 2 运行效果截图

演示网页所在位置：[配套资源/source/chapter1/insignia/index.html](#)。

【演示 3】：全角度展示

本例演示了鼠标拖动图片进行旋转，360° 展示物体，用若干角度不同的图片切换来模仿 3D 效果，如图 1-3 所示。

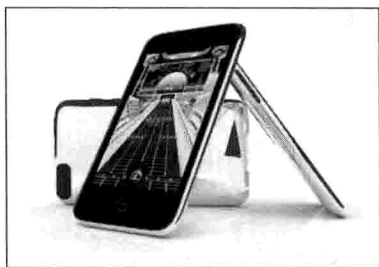


图 1-3 演示 3 运行效果截图

演示网页所在位置：[配套资源/source/chapter1/360Show/index.html](#)。

【演示 4】：表单数据验证

表单验证是 JavaScript 在网页中的重要应用之一。本例演示了对几种常见输入数据的验证，默认值的判断，以及回调函数的使用。在数据发送到服务器前进行验证过滤，可以有效减轻服务器端的压力，运行效果如图 1-4 所示。

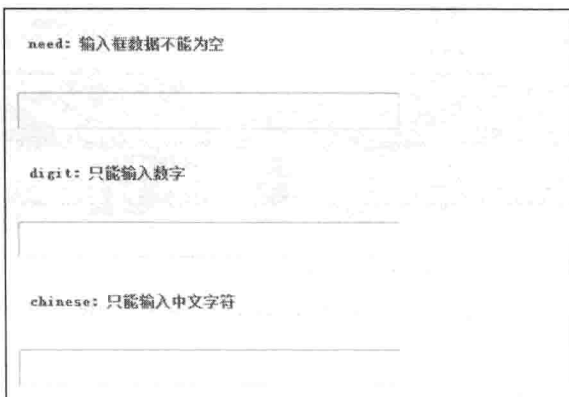


图 1-4 演示 4 运行效果截图

演示网页所在位置：[配套资源/source/chapter1/formValidate/index.html](#)。