

第1章

JavaScript 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。

JavaScript 语言是按照 ECMAScript 标准去设计实现的，ECMAScript 6.0（简称 ES6）已经在 2015 年 6 月正式发布了。它的目标是使 JavaScript 语言可以用来编写复杂的大型应用程序，成为企业级开发语言。ES 标准不断更新，2018 年 6 月发布的新版本是 ECMAScript 2018（简称 ES9）。ES6 以上标准所涉及的新功能已被目前绝大多数的主流浏览器实现。

课堂学习目标：

- ☐ 了解 JavaScript 脚本语言的特色；
- ☐ 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 是目前互联网比较流行的脚本语言，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 也在日益普及，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（如 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、容易上手等优势而大行其道。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，及其事件驱动和异步 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

本书从第 2 章开始将详细介绍 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。

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，希望读者在学习过程中能准确地抓住重点，少走弯路，愉快学习。

- ❑ 不要放过任何一个看上去很简单的小问题，它们往往可以引申出很多知识点；
- ❑ 学习 JavaScript 并不难，难的是长期坚持实践和不遗余力地博览群书；
- ❑ 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；
- ❑ 看得懂的问题要仔细看，看不懂的问题要硬着头皮看；
- ❑ 不要停留在集成开发环境（IDE）的摇篮上；
- ❑ 请亲自把本书的程序例子输入计算机进行实践，即使配套资源中有源代码；
- ❑ 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，而不是一次性记住的；
- ❑ 记录下在和别人交流时发现而自己忽视或不理解的知识点；
- ❑ 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 常被用来完成以下任务：

- ❑ 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，例如：

- ❑ 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，运行效果如图 1-1 所示。



图 1-1

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，运行效果如图 1-2 所示。



图 1-2

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，运行效果如图 1-3 所示。



图 1-3

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

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

【演示 5】网页游戏

本游戏名为“守护公主”，由一家法国公司使用 JavaScript 开发。如果公主从建筑物上落下，则游戏失败。本书并不鼓励读者沉迷于游戏，所以并未提供源代码。程序演示网址：<http://lorancou.free.fr/bulk/princesskeeper>。运行效果如图 1-5 所示。

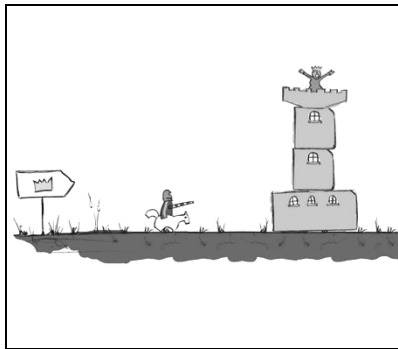


图 1-5

【演示 6】布料质感模拟

本例是由 Kaimibu 公司提供的基于 WebGL、GPU 加速的布料模拟程序。WebGL 是一种 3D 绘图技术标准，这种绘图技术标准允许把 JavaScript 和 OpenGL ES2.0 结合在一起，为 HTML5 Canvas 提供硬件 3D 加速渲染。WebGL 技术标准免去了开发网页专用渲染插件的麻烦，可被用于创建具有复杂 3D 结构的网页页面，甚至可以用来设计 3D 网页游戏等。WebGL 的发展必将给网页技术带来新的冲击。运行效果如图 1-6 所示。

提示：请使用最新版本的 Chrome 浏览器或 Firefox 浏览器运行本演示程序。

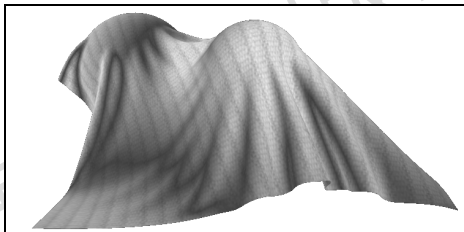


图 1-6

1.3 开发 JavaScript 应用程序

了解 JavaScript 语言后，本章将继续介绍如何使用 JavaScript 编写程序。将 JavaScript 嵌入到 HTML 代码中，是进行 JavaScript 开发的关键步骤。另外，选择适合的开发工具及浏览器也非常重要，每种浏览器对于 JavaScript 的支持情况不尽相同。

1. 选择开发工具

JavaScript 是一种脚本语言，可以使用任何文本编辑器编写代码，也可以使用一些专用的 JavaScript 集成开发环境（Integrated Development Environment, IDE）来辅助开发 Web 应用程序。“工欲善其事，必先利其器”。在熟悉 JavaScript 语言后，选择一种适合自己的开发工具，可以减少代码的输入量，增加代码输入的正确率，提高开发效率。常见的开发工具如下。

□ 记事本

在 Windows 系统中，记事本是一个基本的文本编辑程序，最常用于查看或编辑文本文件，也是最简单、最方便的编辑器。有些开发人员在修改代码片段时经常会用到它。

□ Vi 文本编辑器

Vi 是 UNIX、Linux 系统中非常普遍的文本编辑器。Vi 命令集众多，有些人不习惯使用它，但只需要掌握基本命令并加以灵活运用，Vi 就会显现出它的优势。

□ UltraEdit

UltraEdit 是用于 Windows 系统上的一套商业性文本编辑器，由 IDM Computer Solutions 于在 1994 年创造。UltraEdit 有很强大的编程功能，支持宏、语法高亮度显示和正则表达式等功能。UltraEdit 也支持 Unicode 和 Hex 编辑的模式。

□ EditPlus

EditPlus 是一款小巧但功能强大的编辑器。EditPlus 具有无限制的撤销与重做、英文拼字检查、自动换行、列数标记、搜寻取代等功能，同时支持编辑多文件、全屏幕浏览，甚至监视系统剪贴板等功能。EditPlus 支持语法高亮显示，通过定制语法文件，可以扩展到其他程序语言。还可以在 EditPlus 的工作区域中打开浏览器窗口。

□ JetBrains WebStorm

WebStorm 是 JetBrains 公司旗下一款 JavaScript 开发工具，被广大中国开发者誉为“Web 前端开发神器”“最强大的 HTML5 编辑器”“最智能的 JavaScript IDE”等。WebStorm 与 IntelliJ IDEA 同源，继承了 IntelliJ IDEA 强大的 JavaScript 部分的功能，支持编码导航和用法查询、JavaScript 单元测试、代码检测和快速修复、FTP 远程文件同步以及版本的控制系统等。

□ Aptana Studio

Aptana 是一个基于 Eclipse 的集成开发环境，其最广为人知的是它非常强悍的 JavaScript 编辑器和调试器。Aptana Studio 具有如下特点：代码语法错误提示，支持 Aptana UI 自定义和扩展，支持跨平台，支持流行 AJAX 框架的 Code Assist 功能（AFLAX、Dojo、jQuery、MochiKit、Prototype、Rico、script.aculo.us、Yahoo UI）。

□ 1st JavaScript Editor

1st JavaScript Editor 是一款 JavaScript 开发、校验和调试工具，简单易用，同时它又是

AJAX、CSS、HTML、DOM 开发工具，支持内置预览功能，提供了 HTML 标记、HTML 属性、HTML 事件、JavaScript 事件和 JavaScript 函数等完整的代码库。

❑ NetBeans

NetBeans IDE 是一个屡获殊荣的集成开发环境，可以方便地在 Windows、Mac、Linux 和 Solaris 中运行。NetBeans 包括开源的开发环境和应用平台，目前支持 PHP、Ruby、JavaScript、AJAX、Groovy、Grails 和 C/C++ 等开发语言。

❑ Dreamweaver

Dreamweaver 是 Adobe 公司著名的网站开发工具。它使用所见即所得的接口，亦有 HTML 编辑的功能。Dreamweaver 可以用最快速的方式将 Fireworks、FreeHand 或 Photoshop 等文档移至网页上，可以简单迅速地更新网站，并拥有较强的网站控制能力。

支持 JavaScript 的 IDE 数不胜数，受限篇幅，此处不一一列举。读者可以根据自己的喜好或开发要求来选择对应的 IDE，本书并不指定或推荐某一款 IDE 产品。使用 IDE 并不是技术难点，当读者熟练掌握了 JavaScript 开发技术后，任何一款 IDE 都可以驾轻就熟。如果读者是零起点学习 JavaScript 的，建议使用最简单的文本编辑器（如记事本），它们并不提供任何输入帮助，有助于初学者学习并掌握 JavaScript 语法，养成良好的代码书写习惯。

2. 在 HTML 文档中嵌入 JavaScript 程序

JavaScript 可以直接嵌入 HTML 网页中，由浏览器解释并执行。网页中的所有 JavaScript 代码必须放在 `<script></script>` 标签之间。

【课堂案例 1-1】第 1 个 JavaScript 程序

本案例在 HTML 代码中嵌入 JavaScript 程序，并使用 JavaScript 程序在网页上输出“Hello JavaScript.” 字符串。程序运行结果如图 1-7 所示。



图 1-7

案例学习目标：

- ❑ 能够在网页中嵌入 JavaScript 程序；
- ❑ 能分辨 HTML 标签与 JavaScript 程序代码。

程序代码（1-1.html）：

```
01 <html>
02 <head><title>第 1 个 JavaScript 程序</title></head>
03 <body>
04 <script>
05     document.write("Hello JavaScript. <br />");
06 </script>
07 </body>
08 </html>
```

案例分析：<script></script>之间是 JavaScript 程序。第 05 行代码“document.write(...)”

的作用是在网页上输出“Hello JavaScript.”字符串。JavaScript 代码可以放在 HTML 文档的任何位置，例如，很多开发者习惯将 JavaScript 代码放在<head></head>标签之间。也可以将多段 JavaScript 代码嵌入 HTML 文档中。

另外，还可以使用<script>标签的 src 属性，在 HTML 文档中引用外部 JavaScript 文件中的程序，使 HTML 标签与 JavaScript 程序分离。用法如下：

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

JavaScript 文件的扩展名是 js。例如，引用 example.js 的 JavaScript 标签的代码如下：

```
<script type="text/javascript" src="example.js"></script>
```

3. 运行 JavaScript 程序

使用任何浏览器打开网页，即可运行网页中的 JavaScript 程序。通过浏览器的设置，也可以启用或禁用 JavaScript。目前主流的浏览器有 IE、Chrome、Opera、Firefox、Safari、Edge 等，这些浏览器都支持对 JavaScript 的解释和运行。2019 年 10 月 NetMarketShare 统计的各浏览器在桌面市场上的占有率如图 1-8 所示。

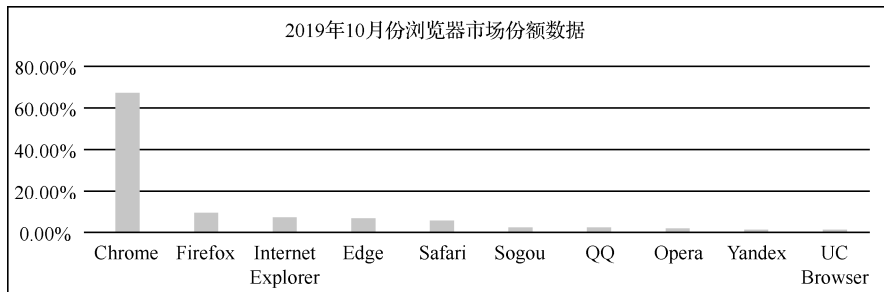


图 1-8

提示：本书中所有示例并不限浏览器，均可运行。

4. 调试 JavaScript 程序

目前主流的浏览器都可以调试 JavaScript 程序，如 IE 浏览器的“开发人员工具”、Firefox 浏览器的 FireBug、Opera 浏览器的 Red Dragonfly 等。单击 IE 浏览器“工具”菜单中的“开发人员工具”命令，可以打开 IE 浏览器的 JavaScript 调试窗口，如图 1-9 所示。

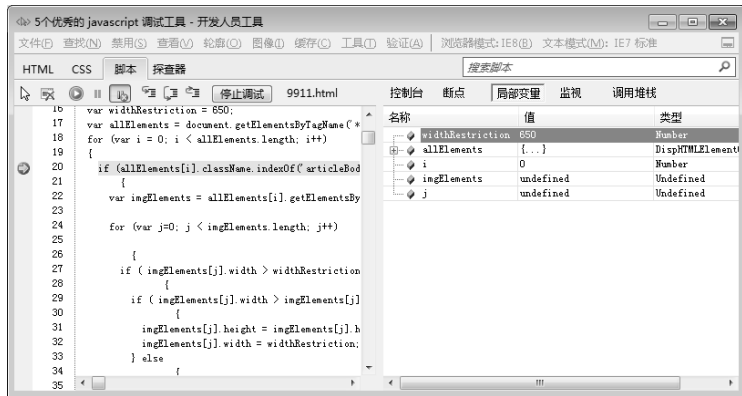


图 1-9

5. 良好的代码书写习惯

为了使程序代码美观大方、方便阅读，需要开发人员养成良好的代码书写习惯。书写习惯不好的代码层次不清、难以阅读。请读者比较如图 1-10 所示两段代码的写法。

<pre><script type="text/javascript">function javascript_ study(){//////////进行 javascript 学习 ////////// return "会使用 javascript ";} js = javascript_study(); document.write("掌握: ");document.write(js);</script></pre>	<pre><script type="text/javascript"> function javascript_study() { ////////// //进行 javascript 学习 ////////// return "会使用 javascript "; } js = javascript_study(); document.write("掌握: "); document.write(js); </script></pre>
--	--

图 1-10

通常开发人员在遇到“{}”时会键入一个 Tab 或若干空格，使代码缩进，并为程序代码加入适当的注释。这样做并不会对程序运行结果产生任何影响，但会使程序代码格式美观、有层次感，方便自己与他人阅读程序代码的含义。正确的代码缩进与适当的注释是书写程序代码的良好习惯。

另外，由于篇幅所限，本书中有些代码片段节省了一些缩进及换行，请读者谅解。

电子工业出版社有限公司
版权所有
盗版必究