

JavaScript & jQuery

交互式 Web 前端开发

[美] Jon Duckett

著

杜 伟 柴晓伟 涂曙光

译



BOO清华大学出版社COM

北 京

Jon Duckett

JavaScript and jQuery: Interactive Front-End Web Development

EISBN: 978-1-118-53164-8

Copyright © 2014 by John Wiley & Sons, Inc., Indianapolis, Indiana

All Rights Reserved. This translation published under license.

Trademarks: Wiley, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Access is a registered trademark of Microsoft Corporation. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc., is not associated with any product or vendor mentioned in this book.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2014-6430

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

JavaScript & jQuery 交互式 Web 前端开发/(美) 达克特(Duckett,J.) 著; 杜伟, 柴晓伟, 涂曙光 译. —北京: 清华大学出版社, 2015

书名原文: JavaScript and jQuery: Interactive Front-End Web Development

ISBN 978-7-302-39763-2

I. ①J… II. ①达… ②杜… ③柴… ④涂… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 077130 号

责任编辑: 王 军 李维杰

封面设计: 牛艳敏

版式设计: 思创景点

责任校对: 曹 阳

责任印制:

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

装 订 者: 河北三河市新茂装订厂

经 销: 全国新华书店

开 本: 148mm×210mm

印 张: 19.625

字 数: 924 千字

版 次: 2015 年 6 月第 1 版

印 次: 2015 年 6 月第 1 次印刷

印 数: 1~4000

定 价: 79.80 元

产品编号:

译者序

今天的Web网站，已经和10年前(甚至5年前)的Web网站有了很大的区别。无论是网站具有的交互性、面对用户的友好性，还是网页上所展现内容的丰富性，相比以前都有了巨大飞跃。这一切都拜JavaScript这门强大且灵活的编程语言所赐。JavaScript已经从以前的一门仅提供某些有趣特效的玩具脚本语言，转变为在Web网站开发中承担着核心角色的重要工具。无论使用何种服务器技术来创建Web网站，JavaScript都是Web工程师必须使用的前端语言。

但是，由于JavaScript语言本身特有的灵活性、动态性等特点，学习JavaScript语言无论是对一名在其他编程语言领域具有丰富经验的软件工程师，还是一名刚刚踏入编程世界的新手，都是一个不小的挑战。本书以通俗易懂、由浅入深的方式，向读者一步一步介绍了JavaScript，是一本非常好的JavaScript前端开发入门书籍。

使用大量的图示是本书的一大特点。常言道，“一图胜万言”，通过使用这些图示，能够让读者更加清晰明了地理解本书所要阐述的概念和知识。丰富的代码示例和代码流程图，则是本书的另一大特点。代码示例可以让读者更多地了解Web网页上那些特效和功能是如何通过代码实现的，与代码示例配套的代码流程图则降低了阅读代码的难度，也让读者对代码的执行过程和逻辑有了更直观的理解。

由于jQuery几乎已经成为Web前端代码的“标配”，因此本书将jQuery库也纳入进来。jQuery可以让Web前端工程师事半功倍，更轻松地达成自己的目标。

本书由杜伟、柴晓伟、涂曙光翻译，参与本次翻译活动的还有梁祝权、钟凤华、毛士之、张杉杉、张文旭、彭康、尼春雨、李明、陈龙、董欢。

总而言之，如果想要找到一本几乎“零难度”的JavaScript前端开发入门书籍，通过阅读而快速成为一名Web前端工程师，那么此书将是不二之选。

祝阅读快乐，学习快乐！

译者(杜伟/柴晓伟/涂曙光)

BOOKASK.COM

源代码下载

可访问网址<http://javascriptbook.com>来下载本书所有代码，也可以登录<http://www.tupwk.com.cn/downpage/>页面，输入本书的书名或ISBN来下载。



BOOKASK.COM

前言

书问

BOOKASK.COM

本书讲解如何将JavaScript应用于浏览器中，以创建出更富交互性、更有趣、对用户更友好的Web网站。你还将从本书中学习jQuery，因为它使得JavaScript代码的编写容易了许多。

要充分吸收本书中的内容，在阅读本书之前，你需要了解如何使用HTML和CSS来构建Web页面。除了这个要求以外，并不需要你有任何编程方面的经验。学习使用JavaScript编程包含：

1

理解一些基本的编程概念，以及JavaScript程序员用来描述这些概念的术语。

2

学习JavaScript这门语言本身，这就像学习其他任何语言一样，你需要知道这门语言的词汇，以及使用词汇来组织语句。

3

通过学习JavaScript如何被用在现今的Web网站中的例子，来熟悉如何使用它。

在阅读本书过程中，你唯一需要的设备就是一台安装有现代Web浏览器的计算机，以及你最喜欢的代码编辑器(例如记事本、TextEdit、Sublime Text、Coda等)。

BOOKASK.COM



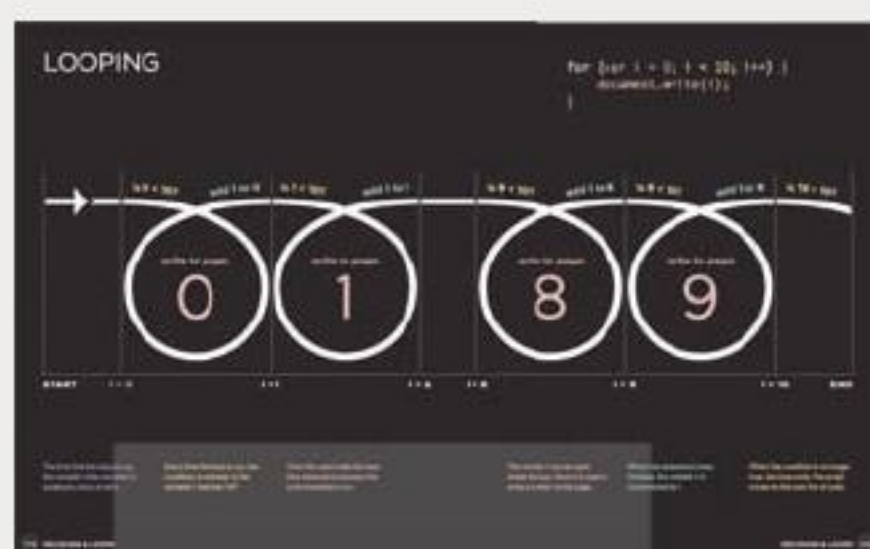
每一章的开头是介绍部分。它们将介绍每一章你要学习的关键概念。



参考部分介绍关键的JavaScript代码段。HTML代码使用蓝色字体显示，CSS代码使用粉红色字体显示，JavaScript则使用绿色字体显示。



背景信息部分显示成白色，介绍在每章将要讨论的主题的一些上下文信息。



图表和信息图部分使用黑色背景，对要讨论的主题提供简单、可视化的参考。



示例部分将你要从每一章中学到的主题归纳在一起，并且展示如何应用它们。



总结部分位于每一章的结尾，它们提醒你每一章中涵盖的关键主题。

JavaScript如何使Web浏览器更富有交互性

1

访问内容

可以使用JavaScript选择HTML页面上的任意元素、属性或文本。例如：

- 选择页面上所有<h1>元素中的文本
- 选择拥有class属性且该属性包含某个值的所有元素
- 查看id属性值等于email的文本输入框里面输入了什么样的内容

当一个Web页面被展现在浏览器中时, JavaScript能让你通过访问和修改页面上的内容和标记, 使得Web页面更富有交互性。

2

修改内容

可以使用JavaScript将元素、属性和文本添加到页面上, 或者从页面上删除这些内容。例如：

- 在第一个<h1>元素的后面添加一段文本
- 更改class属性的值, 将新的CSS规则应用到那些元素上
- 更改元素的大小和位置

3

编制规则

可以为浏览器指定一组要执行的步骤(就像食谱上的烹饪步骤一样), 这些步骤使浏览器能够访问或修改页面上的内容。例如:

- 图库脚本可以检测用户点击了哪张图片, 然后显示被点击图片的放大版。
- 房贷计算器可以从表单中收集数据, 执行运算, 然后显示每月的还款额。
- 动画可以检查浏览器窗口的大小, 然后将一张图片移到可视区域(也被称为 viewport)的底部。

JavaScript涵盖编程中许多传统的规则。通过响应用户的操作, 使得Web页面更具交互性。

4

响应事件

可以指定一段脚本, 让它在某个特定的事件发生时运行。例如, 脚本可以在如下事件发生时运行:

- 按下按钮
- 点击(或按下)链接
- 光标移动到元素上
- 向表单中添加信息
- 一定时间之后
- Web页面已完成加载

浏览器中的JavaScript示例

能在一个HTML页面被载入浏览器时修改它的内容，是一项非常强大的功能。下面这些示例就基于JavaScript的如下功能：

- 访问页面的内容
- 修改页面的内容
- 编制浏览器要遵循的规则或步骤
- 响应由用户或浏览器引发的事件



幻灯片特效

在第11章中展现

幻灯片特效可以在给定页面的同一块区域内显示一系列不同的图片(或其他HTML内容)。这些图片可以按照顺序自动播放，或在用户人工点击后播放。这种特效使得在有限区域内可以显示更多的内容。

响应：脚本在页面载入后执行

访问：获取要展现的每张幻灯片

修改：只显示第一张幻灯片(隐藏其他的幻灯片)

编制：设置计时器，决定何时显示下一张幻灯片

修改：更改要显示的幻灯片

响应：用户点击不同幻灯片的按钮

编制：检测要显示哪张幻灯片

修改：显示用户要查看的幻灯片



表单

在第13章中展现

当用户在页面上输入表单中的信息时，对表单进行验证(检查表单的填写是否正确)是很重要的。JavaScript可以在发生错误时警告用户。可以基于输入的任何数据进行精巧的运算，并将结果展现给用户。

响应：当用户输入他们的名字，点击提交按钮时

访问：从表单字段中获取用户输入的值

编制：检查名字是否足够长

修改：如果名字的长度不够，显示一条警告消息

这两页的示例能让你领略在一个Web页面上JavaScript可以完成的事情，以及通过本书将要学到的技巧。

在接下来的章节中，你将学习如何以及何时访问或修改内容、添加编程规则以及响应事件。



重新载入页面的一部分

在第8章中展现

你可能不想强制访问者重载整个Web页面的内容，特别是在只有页面的一小部分需要刷新的情况下。只重载页面的一部分可以让网站看起来载入速度更快，并更像应用程序。

响应：脚本在用户点击链接时执行

访问：用户点击的链接

编制：载入链接指向的新的页面内容

访问：寻找页面上要被替代的元素

修改：使用新内容替代页面上的旧内容

过滤数据

在第12章中展现

如果有一大堆数据要显示在一个页面上，可以通过过滤来帮助用户找到他们需要的信息。按钮基于HTML元素的属性中的数据生成。当用户点击某个按钮时，只有带有那个关键字的图片才被显示出来。

响应：脚本在页面载入时执行

编制：从图片中收集关键字

编制：将关键字转换为用户可以点击的按钮

响应：用户点击某个按钮

编制：找到应当显示的图片组

修改：显示使用相应标签的图片组

本书结构

为了让你掌握JavaScript，本书分成以下两部分：

核心概念

前面的9章将介绍编程和JavaScript语言的基础知识。与此同时，你将学习如何使用JavaScript创建更有趣、更富交互性、更有用的网站。

第1章将介绍计算机编程中的一些关键概念，向你展现计算机如何使用数据来创建世界万物的模型，以及如何使用JavaScript来修改HTML页面的内容。

第2至第4章将涵盖JavaScript语言的基础知识。

第5章解释文档对象模型(DOM)如何在将文档载入浏览器时，使你可以访问和修改文档的内容。

第6章讨论如何使用事件来激活代码的运行。

第7章将展示jQuery如何使得编写脚本的过程更快、更容易。

第8章将介绍Ajax，它是一组使你可以不重新载入整个页面的情况下，修改Web页面部分内容的技术。

第9章涵盖应用编程接口(Application Programming Interfaces, API)，包括HTML中的一些新API，以及类似Google Maps之类的网站所提供的API。

实用应用程序

学完前面的部分，你应该已经看到了许多JavaScript是如何用于各种网站上的示例。该部分会把你已经掌握的各种技术综合起来，给你一些实用的示范来展现专业程序员是如何使用JavaScript的。你不但可以看到一系列深入的示例，还可以学习更多如何从头设计和编写脚本的过程。

第10章讲述错误处理和调试，并解释有关JavaScript是如何被执行的更多信息。

第11章将教你如何创建诸如滑块、模式窗口、选项卡面板和折叠式面板之类的内容面板。

第12章演示多个过滤和排序数据的技巧，包括过滤图片库，以及通过点击列的抬头来表格中的数据行进行重新排序。

第13章讲述表单增强技术以及如何验证表单数据。

除非已经是十分自信的程序员，否则可能会发现在第一次阅读本书时，从头到尾通读一遍将是非常有帮助的。然而，即使已经了解编程的基础知识，我们也希望能在你创建自己的脚本时，提供有帮助价值的参考。

HTML和CSS: 快速参考

在开始学习JavaScript之前, 先介绍一些HTML和CSS术语, 了解HTML属性和CSS属性是如何使用键/值对的。

HTML元素

HTML元素被添加到一个页面上以描述页面的结构。一个元素包含起始和结束标签, 以及标签内的内容。

标签通常成对出现, 一个是起始标签, 一个是结束标签。有少数不包含任何内容的元素(例如, ``), 它们有自结束的标签。

起始标签可以包含属性, 这些属性通常描述有关这个元素的更多信息。属性有名称和值。值通常包含在一对双引号中。



CSS规则

CSS使用规则来指定一个或多个元素的内容在浏览器中的显示方式。每个规则都有选择器和声明块。

CSS选择器指定哪些元素需要应用这个规则。声明块包含规则, 这些规则指定如何显示那些元素。

声明块中的每一个声明都有属性(想要控制什么东西)和值(属性的设置值)。



浏览器支持

本书前面部分的一些示例不能在Internet Explorer 8或更早版本的IE浏览器中工作(但这些示例提供了能工作于IE8中的版本, 这些兼容版本可以通过<http://javascriptbook.com>下载)。在本书后面部分的章节中, 我们介绍了对旧版浏览器进行兼容的技术。

每个版本的Web浏览器都会增加新的功能。这些新功能通常会让编程更简单, 或相比起使用旧技术看起来更好。

但是, Web网站的访问者并不总是使用最新版本的浏览器, 所以Web网站的开发人员不能依赖于最新的技术。

正如你将要看到的, 不同的浏览器之间存在许多不一致性, 这些不一致性会影响JavaScript开发人员。jQuery可以帮助你处理跨浏览器不一致性的问题(这也是jQuery被如此多的Web开发人员如此广泛使用的最主要原因之一)。但是, 在学习jQuery之前, 先了解这些浏览器兼容性问题十分有用。

为了让学习JavaScript更容易, 前面的几章使用了一些在IE8中不受支持的JavaScript特性。但是:

- 在后面的章节中你将学习如何处理IE8和旧版浏览器的兼容性问题(因为我们知道仍有许多客户期待网站能工作于IE8中)。处理这些问题只需要你了解一些额外的代码, 或是需要你知道一些额外的问题。
- 在网上, 可以找到每章中那些不工作于IE8中的示例的兼容性版本。但是请查看那些代码示例中的注释, 以确保了解使用它们时所引发的问题。

BOOKASK.COM

目录

第1章 编程基础知识	1
第2章 JavaScript基础指令	43
第3章 函数、方法与对象	75
第4章 判断和循环	135
第5章 文档对象模型	173
第6章 事件	233
第7章 jQuery	283
第8章 Ajax与JSON	357
第9章 API	399
第10章 错误处理与调试	439
第11章 内容面板	477
第12章 筛选、搜索与排序	517
第13章 表单增强与验证	557



BOOKASK.COM

第1章

编程基础知识

书问

BOOKASK.COM

在学习如何阅读和编写JavaScript语言之前，需要先熟悉计算机编程中的一些关键概念。它们将包含以下三个部分：

A

什么是脚本？如何创建一段脚本？

B

计算机如何融入它们周围的世界之中？

C

如何为Web页面编写一段脚本？

一旦学完这些基础知识，后面的章节就将向你展示如何使用JavaScript语言来告诉浏览器，让它做你想要完成的事情。

BOOKASK.COM

1/a

什么是脚本？
如何创建一段
脚本？

BOOKASK.COM

一段脚本是一系列的指令

一段脚本是一系列的指令，一台计算机可以执行这些指令来达成目标。可以将脚本比作：

食谱

按照一份食谱中的步骤，一步一步来做，人们可以做出一道之前从来没有做过的菜肴。

有一些脚本很简单，它们仅仅用来处理某个特定场景，就像一道简单菜肴的简单食谱一样。有些脚本则可以执行许多任务，就像一份包含三道菜肴的复杂食谱。

脚本和食谱的另一个相似之处是，如果你从未做过菜或从未写过程序，将会有许多新名词需要学习。

手册

大型公司通常会为新员工提供手册，这些手册里包含了在某些情况下要遵循的流程。

例如，酒店的手册会包含诸如如何办理顾客的入住手续、如何清洁房间、如何在有火警发生时离开酒店之类的步骤。

在任何此类场景中，员工只需要根据事件的类型去遵循特定的步骤(你肯定不想接待你的酒店员工在你等着办理入住时，还要去从头查阅整本手册)。同样，在一段复杂的脚本中，浏览器也可能只会在特定时间执行某部分的代码。

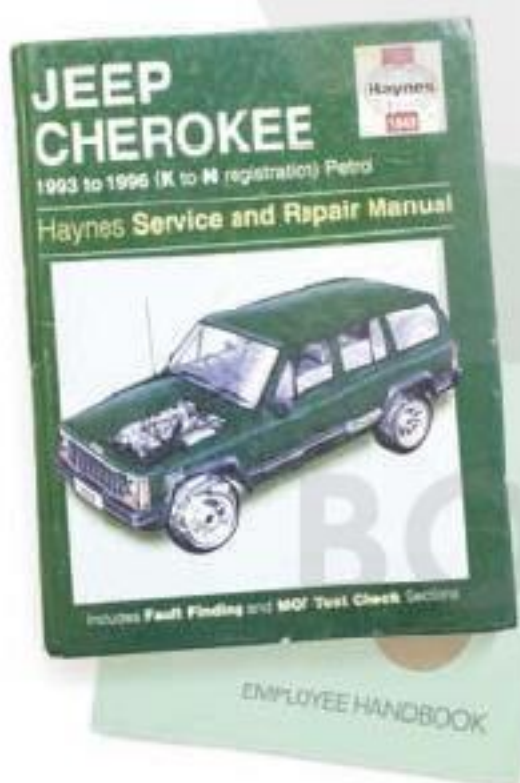
说明书

机械师经常需要在修理一辆自己不太熟悉的汽车型号时，查询汽车维修说明书。这些说明书包含一系列的测试，用来检查汽车的关键功能是否工作正常，以及如果在发生任何问题时，如何进行修理。

例如，说明书里会有如何测试刹车的详细信息。如果汽车通过这个测试，机械师就无须担心刹车有问题，从而继续进行下一项测试。但是，如果刹车测试失败，机械师就需要按照说明书上的指导来修理刹车。

修理完成后，机械师可以重新对刹车进行测试，以确认是否修复了问题。如果刹车通过测试，机械师就知道刹车修好了，现在可以进行下一项测试了。

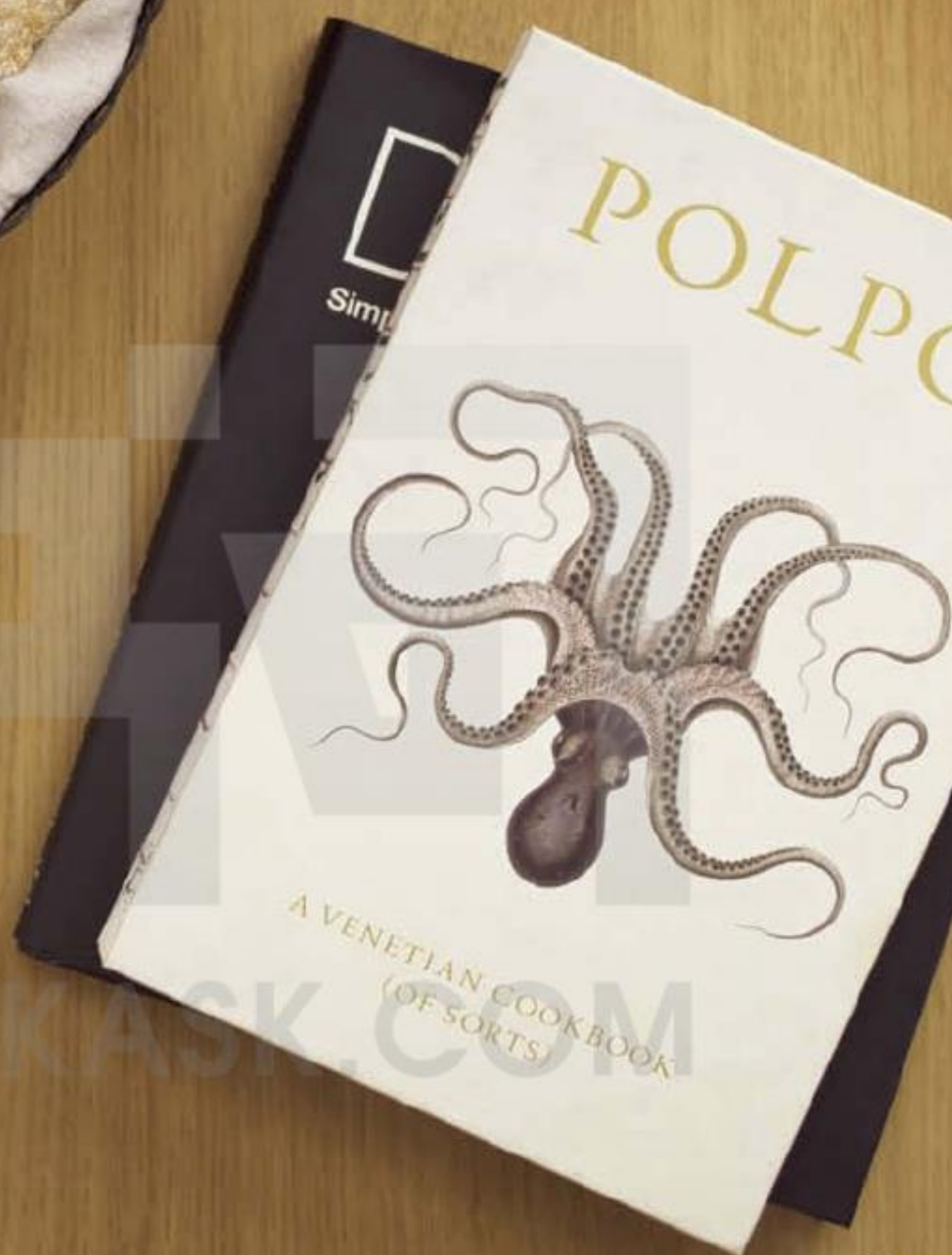
类似的，脚本可以让浏览器检查当前的状况，并且只执行相应的步骤。



脚本由一台计算机可以一步一步执行的指令组成。

根据用户如何与Web页面进行交互，浏览器可以只执行脚本的某个部分。

脚本可以运行代码中的不同部分来响应它周围的情景。



编写一段脚本

要编写一段脚本，你需要首先确定自己的目标，然后列出为了达成目标而需要完成的任务。

人们在完成某个复杂的目标时不用总是想着它。例如，你可以开一辆车、做一顿早餐或发送一封电子邮件，在干这些事情时，你不用一板一眼地按照一份详细的说明去做。但是在你第一次做这些事情时，它们很可能看起来复杂得让人气馁。因此，当学习一项新技能时，我们通常将它分解成更小的任务，然后每次学习其中的一个任务。多做几次之后，这些任务会越来越熟悉，看起来也会简单很多。

当阅读完本书时，你将会阅读到的和编写的一些脚本会相当复杂，它们初看起来会相当“恐怖”。然而，一段脚本只不过是一系列的指令，每一个指令都会按序执行，以解决手头的某个问题。这就是为什么创建一段脚本就像在编写食谱或说明书，这段脚本让一台计算机可以一步一步地解决一个难题。

值得注意的是，一台计算机不能像你我那样去学习如何执行任务，它每次执行任务时都需要遵照指令而行。所以一个程序必须给计算机足够的细节让它可以执行任务，对于计算机而言，每一次都像第一次一般。



move(index)
image
FILES
Left
ING
IMAGE
Y
show active
NEW IMAGE
IMAGE
Y
position
new slide
to right
animate
current slide
to left
or right
slide
de)
moved
(left:0)
(left:0)
index to
of new slide

从你要达成的目标的大局出发，然后将它分解成更小的步骤。

1: 定义目标

首先，你需要定义你要达成的任务。可以将这想象成一个需要计算机解决的拼图游戏。

2: 设计脚本

要设计一段脚本，你需要将目标分解成一系列解决拼图游戏所需的任务。这可以通过一个流程图来表现。

然后编写计算机完成每个任务所需执行的各个步骤(以及执行任务时所需的其他任何信息)，就像编写一个可以一步一步照着做的菜谱一样。

3: 写出每个步骤的代码

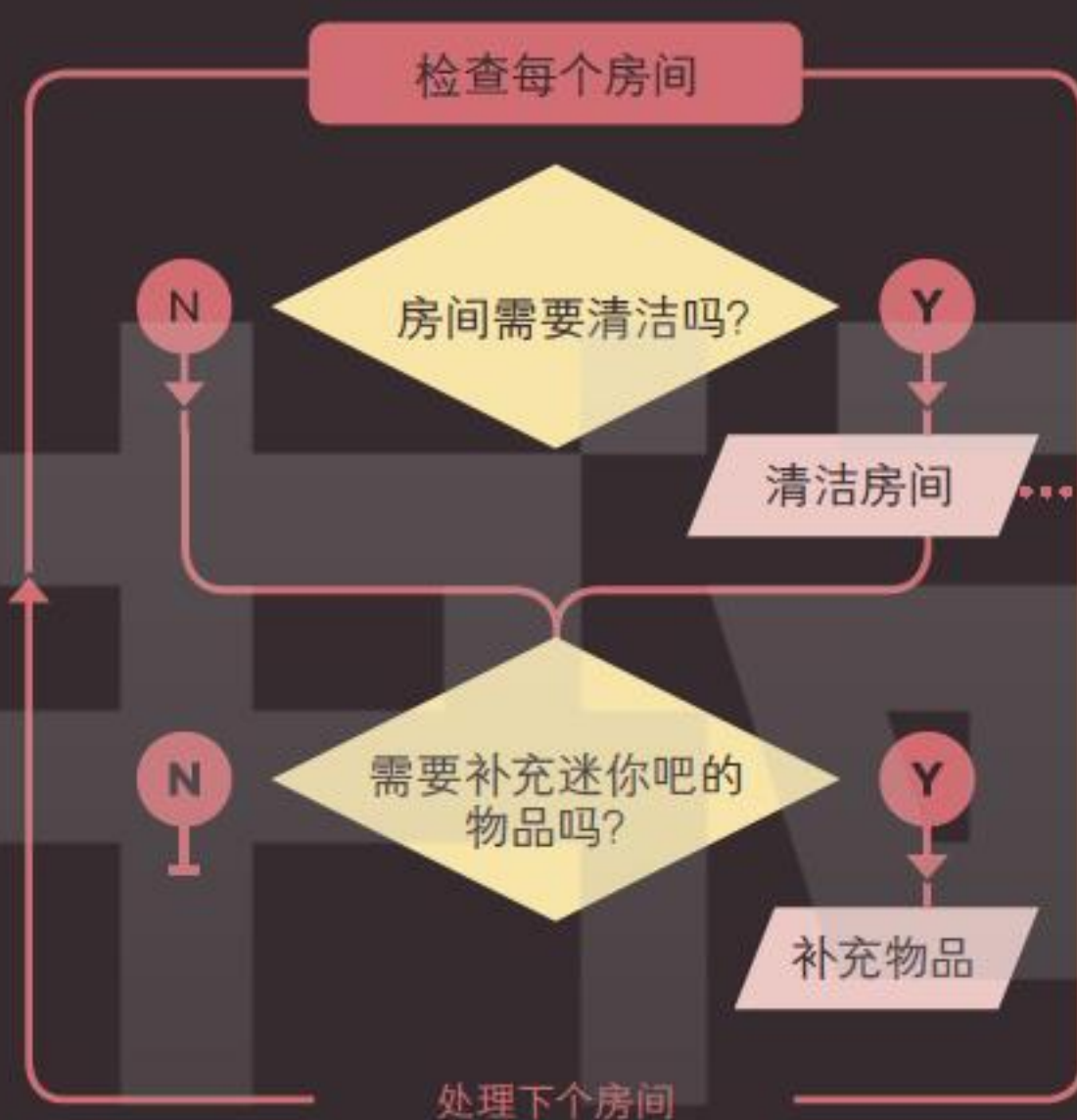
每个步骤都需要使用计算机能够理解的编程语言编写出来。在我们的场景中，JavaScript就是我们要使用的编程语言。

虽然尽快地开始写代码很吸引人，但在开始写代码之前，还是要花时间设计一下你的代码。

设计一段脚本：任务

一旦了解了脚本要完成的目标，就可以开始拟定一下需要完成的单个任务了。可以通过使用一个流程图来“鸟瞰”所有的任务。

流程图：一名酒店清洁工的任务列表



BOOKASK.COM

设计一段脚本: 步骤

每一个任务都可以被分解成一系列步骤。当你准备开始编写脚本时，这些步骤可以被“翻译”为一行一行的代码。

列表: 清洁一个房间所需的步骤

- 步骤 1: 清除用过的床上用品
- 步骤 2: 清扫所有家具的表面
- 步骤 3: 给地板吸尘
- 步骤 4: 铺上新的床上用品
- 步骤 5: 清除用过的毛巾和香皂
- 步骤 6: 清扫卫生间、浴池、水槽和台面
- 步骤 7: 补充新毛巾和香皂
- 步骤 8: 清扫浴室地板

BOOKASK.COM

如你将在下一页所见，计算机为了执行一个任务所需按序执行的步骤，和你我平时做事时要做的步骤相比会有很大的区别。

从步骤到代码

流程图中每个任务的每个步骤，都需要使用一门计算机能够理解和执行的语言来编写。

在本书中，我们将关注JavaScript语言以及如何在Web浏览器中使用JavaScript语言。

就像学习任何一门新的(人类)语言一样，你需要努力地学习：

- 词汇：计算机能够理解的单词。
- 语法：将单词组合起来，创建计算机能够执行的指令。

除了学习语言本身之外，如果你是编程新手，那你同时还需要学习计算机如何使用一种程序化的方式来解决各种问题，以达成各种不同的目标。

计算机是非常逻辑化且非常听话的。它们需要被告知需要它们做的事情的每一个细节，然后它们就会严格遵照执行。由于计算机需要的指令和你我所能理解的不同，每一个刚开始学习编程的人在一开始都会犯很多错误。别由此灰心丧气，在第10章你会看到查找错误的多种方法，程序员管这种事儿叫作调试。





你需要学习像计算机那样去“思考”，因为它们使用与你我不同的方式来解决问題。

计算机用程序化的方式来解决问題；它们一个接一个地遵照执行一系列的指令。它们需要的指令的类型，通常和你交给另外一个人的指令有所不同。因此，在本书中你不仅将学到JavaScript所使用的词汇和语法，还将学习如何编写计算机能够执行的指令。

例如，当你看着左侧的图片时，你要如何找到个头最高的那个人呢？计算机需要显式的、一步一步的指令，如下所示：

1. 找到第一个人的身高。
2. 假设他(或她)是“最高者”。
3. 逐个查看其他人的身高，将他们的身高和你找到的“最高者”进行比较。
4. 在每次和一个人进行比较时，如果你发现某人的身高比当前“最高者”的身高还要高，他(或她)就成为新的“最高者”。
5. 在检查完所有人的身高之后，告诉我何人是“最高者”。

所以计算机需要依序查看每个人的身高，每一次它都执行一个测试(“他比当前的“最高者”还要高吗?”)。完成对所有人的测试之后，计算机就可以给出答案了。

定义目标/设计脚本

现在看看如何创建一种不同类型的脚本。这个示例会计算铭牌(上面刻着某个人的名字)的价格。客户将被按照字母来收费。

你应当做的第一件事情就是详细地写出脚本要完成的目标(也就是你希望做到的事):

客户可以将一个名字放到铭牌上, 每个字母的价格是5美元。当用户输入名字时, 向他们显示这个名字要花多少钱。

接下来, 将目标分解成一系列可以执行并能最终达成目标的任务:

1. 当按钮被按下时, 脚本被触发执行。
2. 收集输入到表单字段里的名字。
3. 检查用户是否输入了内容。
4. 如果用户没有输入任何内容, 显示一条消息, 告诉用户输入一个名字。
5. 如果用户输入了一个名字, 将字母的数量乘上每个字母的价格, 算出整个名字的价格。
6. 将铭牌的价格显示给用户。

(上面这些步骤的数字编号, 对应下一页的流程图中的步骤。)



CUSTOM SIGNAGE

Enter name:

SHOW COST



CUSTOM SIGNAGE

Enter name: Please enter a name below...

SHOW COST



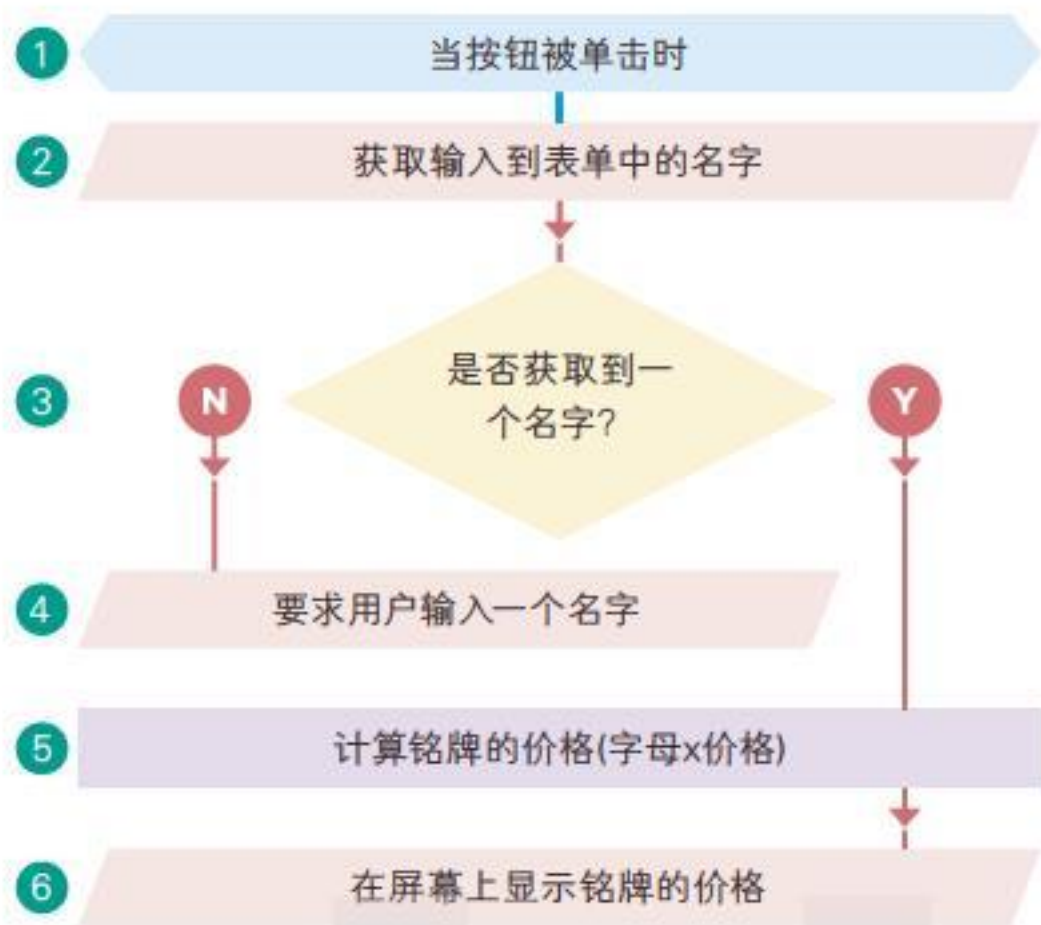
CUSTOM SIGNAGE

\$30

T H O M A S

在流程图中画出这些任务

脚本通常需要在不同的情况下执行不同的任务。可以使用流程图来表达任务是如何组合在一起的。流程图会显示出每个步骤之间的路径。



流程图上的箭头显示了脚本是如何从一个任务执行到下一个任务的。不同的形状代表不同类型的任务。在某些地方需要进行决策，这些决策会导致代码按照不同的路径来执行。

在第2章将介绍如何将这个示例转变为代码。你在本书中还将看到许多不同的流程图示例，以及能帮助你处理每种情况的代码。

一些有经验的程序员会使用更复杂的图形，特别在图形是用来表现代码时。然而，这些图形会有很陡的学习曲线。这里介绍的这些非正式的流程图会在你学习语言时，帮助你理解脚本是如何工作的。

流程图中的核心图形

总结

编程基础知识

问题：什么是脚本？如何创建一段脚本？

- ▶ 一段脚本是一系列计算机能够遵循的指令，这些指令都是为了达成某个目标。
- ▶ 脚本每次运行时，可能只会执行全部指令中的一部分。
- ▶ 计算机使用与人类不同的方式来完成任任务，所以你的指令必须让计算机能够程序化地解决掉任务。
- ▶ 要编写一段脚本，你需要将目标分解成一系列的任务，然后分解成一个一个的完成任务所需的步骤(流程图可以帮助我们完成这一点)。



本书试读到此结束啦！



[JavaScript & jQuery交互式Web前端开发](#)

作者：(美) 达克特 (Duckett, J.), 著

出版社：清华大学出版社

通过以下方式阅读更多 [Powered by 书问](#)



● 扫码分享到朋友圈可阅读更多



立即扫码



● 还不过瘾？购买书库畅读卡全本畅读此书！



[查看全部书库](#)



● 购买纸书也可畅读全本哦！

[广购书城](#)

[中国图书网](#)

[蔚蓝网](#)