



图灵程序设计丛书

Apress®

The Essential Guide to HTML5

Using Games to Learn HTML5 and JavaScript

# HTML5游戏开发



[美] Jeanine Meyer 著  
徐阳 荆涛 等译



人民邮电出版社  
POSTS & TELECOM PRESS



### **Jeanine Meyer**

纽约州立大学帕切斯学院教授，主要讲授数学/计算机科学和新兴媒体专业的课程，另外还为主修人文的学生主讲数学课程。要了解她参与的学术活动，可以访问网站<http://faculty.purchase.edu/jeanine.meyer>。进入大学任教之前，她曾任IBM研究中心经理，主要从事机器人和机械制造方面的研究，后来还担任了IBM教育授权计划的顾问。

TURING

图灵程序设计丛书

The Essential Guide to HTML5

Using Games to Learn HTML5 and JavaScript

# HTML5游戏开发

[美] Jeanine Meyer 著  
徐阳 荆涛 等译

人民邮电出版社  
北京

## 图书在版编目 (C I P) 数据

HTML5游戏开发 / (美) 迈耶 (Meyer, J.) 著 ; 徐阳等译. -- 北京 : 人民邮电出版社, 2011. 10

(图灵程序设计丛书)

书名原文: The Essential Guide to HTML5: Using Games to Learn HTML5 and JavaScript

ISBN 978-7-115-26363-6

I. ①H… II. ①迈… ②徐… III. ①超文本标记语言, HTML—游戏—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2011)第188112号

## 内 容 提 要

本书共 10 章, 通过 10 个具体的游戏示例详细介绍 HTML5 的用法。每章都先列出相关的技术特性并给出了应用的描述, 然后讨论了实现这个应用的关键需求, 接着强调了满足这些需求的 HTML5、CSS 和 JavaScript 特性或者通用编程方法, 最后详细介绍了代码的实现。另外, 每一章都对如何把这些游戏变成你自己的应用给出了建议, 并指出如何测试应用并将其上传到网站上。

本书适合所有希望了解如何利用 HTML5 构建令人兴奋的动态网站的人阅读。

图灵程序设计丛书

## HTML5游戏开发

- 
- ◆ 著 [美] Jeanine Meyer  
译 徐 阳 荆 涛 等  
责任编辑 王军花
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京 印刷
- ◆ 开本: 800×1000 1/16  
印张: 16.75  
字数: 396千字 2011年10月第1版  
印数: 1-3 000册 2011年10月北京第1次印刷  
著作权合同登记号 图字: 01-2011-5244 号  
ISBN 978-7-115-26363-6
- 

定价: 49.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

# 前言

人们对 HTML5 的新功能热情高涨，甚至有人建议就用 HTML5 构建吸引人的动态、交互式网站，而不再需要其他的技术或产品。这可能有些夸张，不过这些新特性让人兴奋不已确实是不争的事实。如今，只使用 HTML5、CSS（Cascading Style Sheet，层叠样式表）和 JavaScript 就完全可以在屏幕上画出直线、弧线、圆和椭圆，还可以指定事件和事件处理来生成动画，并对用户的动作作出响应。可以使用标准控件在网站上加入视频和音频，或者也可以根据需要在应用中放入视频或音频。可以创建提供输入验证的表单，并立即向用户提供反馈。还可以使用一个类似于 cookie 的工具在客户计算机上存储信息。另外，可以使用一些新元素（如 header 和 footer）帮助建立文档的结构。

本书是在我的教学实践和以前写的一些文章的基础上完成的。要深入了解一项技术的特性或者通用的编程概念，最好首先有具体的需求。游戏（特别是我们熟悉的一些简单游戏）就能提供这种需求，它可以很好地解释为什么要学习有关的技术，也让我们有了动力去钻研这些技术的特性。学习一种新的编程语言时，第一步我会编写一个骰子游戏。如果能构建一个带动画的弹道仿真应用，如弹弓游戏，并在出现某个特定条件时播放一个视频或音频片段，这会让我很高兴。如果可以构建我自己的迷宫，绘制上吊小人简笔画，并在玩家的计算机上存储信息，这更会让我欣喜若狂。这正是我们将在本书中做的。在了解如何构建这些简单游戏的同时，你也将逐步积累自己的经验。

在 friends of ED 全体人员和技术审校人员的大力帮助之下，本书得以完成。而撰写本书的目的就是让你做好准备，可以着手建立自己的网站，这包括游戏以及其他动态应用，另外还会简要介绍 HTML5 和编程的精髓。

写这本书时，并不是所有浏览器都支持 HTML5 的全部特性。我们已经使用 Chrome、Firefox 和 Safari 对书中开发的这些应用做了测试。

## 本书的读者对象

本书适合所有希望了解如何利用 HTML5 构建令人兴奋的动态网站的人阅读。如果你对编程有所了解，想知道 HTML5 能带来多大帮助，那么本书非常适合你。如果你没有任何编程经验，本书也同样适合你。也许你是一名 Web 设计人员或者网站所有者，只是想知道如何在后台做一些工作。通过本书，我们希望突出 HTML5 的新特性，揭开编程艺术的神秘面纱。编程是一种艺

术，要想创建吸引人的游戏以及其他应用，这确实需要天赋。不过，只要你能把单词汇集在一起构成句子，能够把句子组织成段落，而且有一定的逻辑性，你就能编程。

## 本书的组织结构

本书共有 10 章，分别围绕我们熟悉的一个游戏或者类似的应用展开介绍。各章的内容有很多冗余，所以如果你愿意，完全可以跳过冗余的部分，不过这些游戏确实越来越复杂。每一章最前面会列出这一章将要介绍的技术特性，并给出应用的描述。首先我们会考虑一般意义上（不依赖任何特定的技术）的关键需求：实现这个应用需要些什么。然后强调满足这些需求的 HTML5、CSS 和 JavaScript 特性或通用的编程方法。最后，我们会详细分析应用的实现。我把代码逐行列在一个表格中，并在旁边分别给出各行的注释。如果要介绍一个游戏的多个版本，将只对新的代码行给出注释。这样做并不是因为我不愿意为你提供有关信息，而是希望你能从中看出哪些代码是类似的，而哪些不同，另外还可以了解到如何分阶段地构建应用。每一章都对如何把这些游戏变成你自己的应用给出了建议，并指出如何测试应用并将其上传到网站上。各章最后的小结会强调在这一章中你学到了什么以及接下来还会看到哪些内容。本书的源代码可在图灵社区 ([ituring.com.cn](http://ituring.com.cn)) 的本书页面下载。

## 本书约定

本书中的应用都是 HTML 文档。JavaScript 放在 `head` 元素的 `script` 元素中，CSS 放在 `head` 元素的 `style` 元素中。`body` 元素包含静态 html，其中还包括 `canvas` 元素。一些示例要依赖外部图像文件，还有一个例子需要用到外部视频文件，另外有一个例子需要一些外部音频文件。

## 排版约定

为了保证这本书尽可能简洁易懂，本书将使用以下排版约定。

- 重要的单词或概念第一次出现时通常会用楷体来强调。
- 代码用 Courier 列出。
- 利用表格给出每个应用的完整代码，左列列出每一条语句，右列给出相应的注释。
- 有时代码很长，在书中无法用一行显示，此时我会使用这样一个箭头：➡

了解了以上格式规范后，下面进入正题。

# 目 录

第 1 章 基础知识.....1	3.5 测试和上传.....74
1.1 引言.....1	3.6 小结.....74
1.2 关键需求.....3	第 4 章 炮弹和弹弓.....75
1.3 HTML5、CSS 和 JavaScript 特性.....3	4.1 引言.....75
1.3.1 基本 HTML 结构和标记.....4	4.2 关键需求.....78
1.3.2 JavaScript 编程.....9	4.3 HTML5、CSS 和 JavaScript 特性.....79
1.4 构建自己的应用.....10	4.3.1 数组和程序员自定义对象.....79
1.5 测试和上传应用.....16	4.3.2 绘图旋转和平移.....80
1.6 小结.....16	4.3.3 绘制线段.....84
第 2 章 骰子游戏.....17	4.3.4 拉弹弓的鼠标事件.....85
2.1 引言.....17	4.3.5 使用数组接合改变显示元素 列表.....87
2.2 关键需求.....20	4.3.6 点之间的距离.....87
2.3 HTML5、CSS 和 JavaScript 特性.....20	4.4 构建自己的应用.....88
2.3.1 伪随机处理和数学表达式.....20	4.4.1 有大炮、角度和速度的炮弹 应用.....92
2.3.2 变量和赋值语句.....21	4.4.2 弹弓：使用鼠标设置飞行参数.....98
2.3.3 程序员自定义函数.....22	4.5 测试和上传应用.....106
2.3.4 条件语句：if 和 switch.....23	4.6 小结.....106
2.3.5 在画布上绘图.....25	第 5 章 记忆力（注意力）游戏.....107
2.4 构建自己的应用.....34	5.1 引言.....107
2.4.1 掷一个骰子.....35	5.2 关键需求.....111
2.4.2 掷两个骰子.....40	5.3 HTML5、CSS 和 JavaScript 特性.....112
2.4.3 完整的 craps 游戏.....44	5.3.1 表示扑克牌.....112
2.5 测试和上传应用.....50	5.3.2 使用 Date 确定时间.....113
2.6 小结.....51	5.3.3 提供暂停.....114
第 3 章 弹跳球.....52	5.3.4 绘制文本.....114
3.1 引言.....52	5.3.5 绘制多边形.....116
3.2 关键需求.....54	5.3.6 洗牌.....117
3.3 HTML5、CSS 和 JavaScript 特性.....55	
3.4 构建自己的应用.....64	

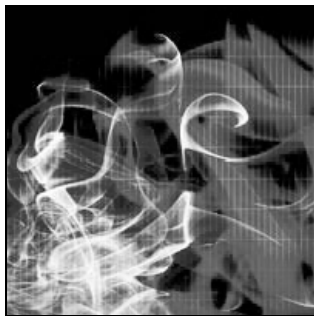


5.3.7 单击牌的实现 .....	118	第 8 章 石头剪刀布 .....	191
5.3.8 防止某些作弊行为 .....	119	8.1 引言 .....	191
5.4 构建自己的应用 .....	119	8.2 关键需求 .....	194
5.5 测试和上传应用 .....	132	8.3 HTML5、CSS 和 JavaScript 特性 .....	195
5.6 小结 .....	132	8.3.1 为玩家提供图片按钮 .....	195
第 6 章 猜谜游戏 .....	133	8.3.2 生成计算机动作 .....	199
6.1 引言 .....	133	8.3.3 启动 .....	206
6.2 关键需求 .....	137	8.4 构建自己的应用 .....	206
6.3 HTML5、CSS 和 JavaScript 特性 .....	138	8.5 测试和上传应用 .....	212
6.3.1 存储和获取数组信息 .....	138	8.6 小结 .....	213
6.3.2 程序执行时创建 HTML .....	140	第 9 章 上吊小人游戏 .....	214
6.3.3 使用 JavaScript 代码修改 CSS 来改变元素 .....	142	9.1 引言 .....	214
6.3.4 使用 form 和 input 元素的 文本反馈 .....	144	9.2 关键需求 .....	221
6.3.5 表现视频 .....	144	9.3 HTML5、CSS 和 JavaScript 特性 .....	222
6.4 构建自己的应用 .....	146	9.3.1 将单词表存储为一个在外部 脚本文件中定义的数组 .....	222
6.5 测试和上传应用 .....	157	9.3.2 生成和定位 HTML 标记, 使 标记作为按钮并禁用这些按 钮 .....	223
6.6 小结 .....	157	9.3.3 在画布上逐步绘制 .....	225
第 7 章 迷宫 .....	158	9.3.4 维护游戏状态并确定输赢 .....	226
7.1 引言 .....	158	9.3.5 检查猜测, 设置 textContent 显示秘密词中的字母 .....	227
7.2 关键需求 .....	162	9.4 构建自己的应用 .....	228
7.3 HTML5、CSS 和 JavaScript 特性 .....	162	9.5 测试和上传应用 .....	236
7.3.1 墙和 token 的表示 .....	163	9.6 小结 .....	236
7.3.2 建立和定位墙的鼠标事件 .....	163	第 10 章 黑桃 J .....	237
7.3.3 检测箭头按键 .....	164	10.1 引言 .....	237
7.3.4 token 与墙的碰撞检测 .....	165	10.2 关键需求 .....	242
7.3.5 使用本地存储 .....	167	10.3 HTML5、CSS 和 JavaScript 特性 .....	242
7.3.6 为本地存储编码数据 .....	172	10.4 构建自己的应用 .....	249
7.3.7 单选按钮 .....	174	10.5 测试和上传应用 .....	258
7.4 构建自己的应用 .....	174	10.6 小结 .....	258
7.5 测试和上传应用 .....	189		
7.6 小结 .....	189		



## 第 1 章

# 基础知识



### 本章内容

- HTML文档的基本结构
- html、head、title、script、style、body、img和a元素
- 一个CSS示例
- 一个使用了Date和document.write的JavaScript代码示例

## 1.1 引言

HTML (Hypertext Markup Language, 超文本标记语言) 是Web上用来传送内容的语言。HTML并非出自某一个人之手, 而是很多国家、很多机构的很多人共同的成果, 他们对定义这个语言的特性都作出了贡献。HTML文档是一个文本文档, 可以使用任何文本编辑器生成。HTML文档包含有元素, 元素的前后“包围”着标记, 标记就是以<符号开始并以>符号结束的文本。比如说, ``就是标记的一个例子。这个标记会显示文件home.gif中包含的图像。这些标记正是HTML中“M”所代表的标记 (markup)。通过使用标记, 才得以在网页中包含超链接、图像和其他媒体。

基本HTML可以采用CSS语言包含指定格式的指令, 另外还可以采用JavaScript语言包含程序来完成交互。浏览器 (如Firefox和Chrome) 会解释HTML以及其中包含的CSS和JavaScript, 生成我们访问网站时所得到的体验。HTML包含网站的内容, 标记提供了有关内容特性和结构的信息, 另外还会提供图像和其他媒体的引用。CSS用来指定格式。同样的内容可以有不同的格式。JavaScript是一种编程语言, 用来建立动态、交互式的网站。即使是最小规模的工作组, 也会由不同的人分别负责HTML、CSS和JavaScript方面的工作, 但最好能对这些工具如何合作有一个基本的认识。如果你已经很熟悉HTML的基础知识, 也知道如何加入CSS和JavaScript, 可能希望跳过这一章, 直接读下一章的内容。不过还是很有必要对这一章的内容大致浏览一遍, 保证确实已经

完全掌握了所有内容，然后再开始学习第一组核心示例。

HTML（及相关的CSS和JavaScript）的最新版本是HTML5。由于它提供了大量新特性，如显示图片和动画的画布，对视频和音频的支持，以及定义一些常用文档元素的新标记（如header、section和footer），这让人们很受鼓舞。利用新的HTML5完全可以创建复杂的、高度交互的网站。写这本书时，并不是所有浏览器都能够接受HTML5的全部特性，不过你可以现在开始学习HTML5、CSS和JavaScript。通过学习JavaScript，你会了解通用的编程概念。如果你想学习其他编程语言，或者想要作为团队一员与其他程序员合作，这会很有好处。

本书中我采用的方法是以特定的示例为背景来解释HTML5、CSS和JavaScript的概念，其中大多数示例都是我们耳熟能详的游戏。在分析这些示例的过程中，我会利用一些小例子介绍具体特性。希望这不仅能帮助你了解你想做什么，还能清楚地知道该如何去做。这样一来，在我解释这些概念和细节时，你就能把握住我们前进的方向。

本章的任务是构建一个网页，其中包含其他网站的链接。在这个过程中，你会对HTML文档（包含少量CSS代码和JavaScript代码）的结构有一个基本的认识。对于这样一些例子，请考虑如何让它成为对你有意义的项目。这个页面可以是你自己的项目、你喜欢的网站或者关于某个特定主题的网站的清单。每个网站中你都会看到文本和一个超链接。第二个例子包括一些额外的格式化设置，文本周围加有方框，还显示了图片和当天的日期与时间。图1-1和图1-2显示了我创建的这两个不同示例。

重新加载Favorite Sites（最喜爱网站）页面时，日期和时间会变为计算机的当前日期和时间。

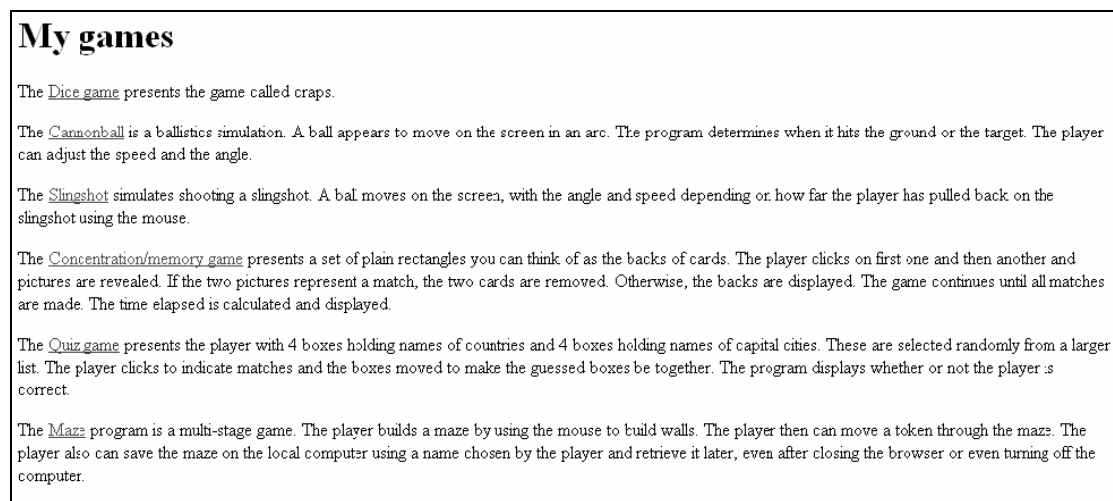


图1-1 给出了注解的游戏清单

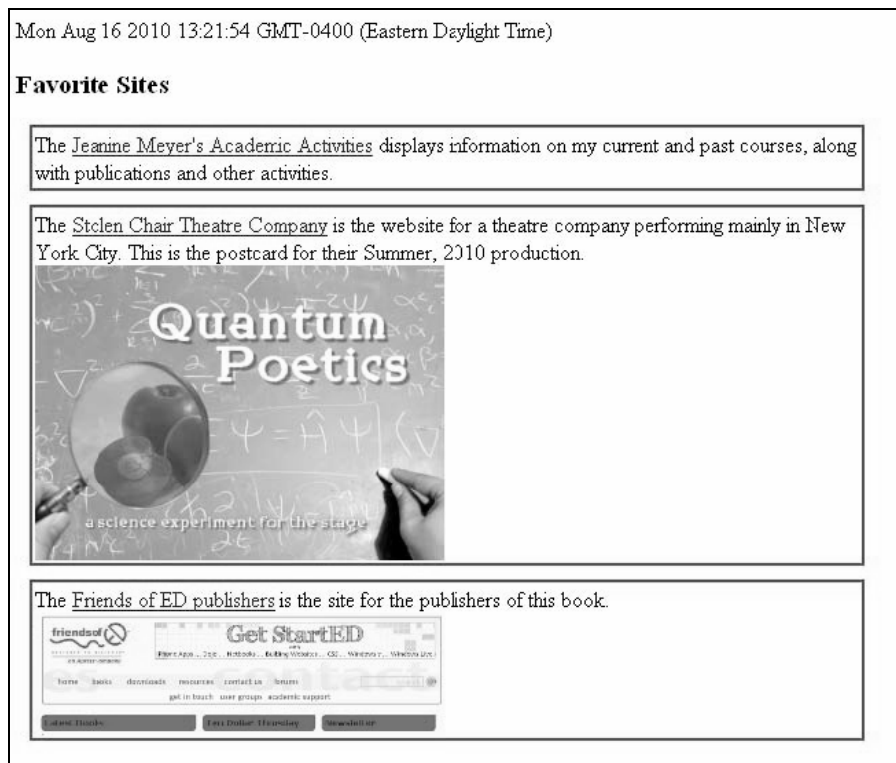


图1-2 最喜爱网站的清单，其中包括额外的格式化设置

## 1.2 关键需求

这个链接清单应用的需求都是构建一个包含文本、链接和图像的Web页面的非常基本的需求。在图1-1所示的例子中，每一项都作为一个段落出现。图1-2所示的例子有所不同，每一项周围都有一个方框。第二个例子还包含一些图像，另外提供了一种方法来得到当天的日期和时间。后面的应用会更为复杂，需要对关键需求做更多讨论，不过这个例子很简单，所以下面直接介绍如何使用HTML、CSS和JavaScript来实现它。

## 1.3 HTML5、CSS 和 JavaScript 特性

前面已经提到，HTML文档是文本，那么如何指定链接、图片、格式和编码呢？答案就是利用标记。除了定义内容的HTML外，通常你还会看到CSS样式，这可以在HTML文档中指定，也可以在一个外部文档中指定。可能还会包括JavaScript来提供交互性，同样可以在HTML文档或一个外部文档中指定。首先来看如何建立简单的HTML标记，以及如何在同一个文档中增加内联的CSS和JavaScript代码。

### 1.3.1 基本 HTML 结构和标记

HTML元素最前面有一个开始标记，后面是元素内容和一个结束标记。结束标记包括一个/符号，后面是元素类型，例如/head。元素间还可以嵌套。标准的HTML文档如下所示：

```
<html>
  <head>
    <title>Very simple example
  </title>
  </head>
  <body>
    This will appear as is.
  </body>
</html>
```

注意这里我把嵌套的标记缩进显示，从而可以更明显地看出这种嵌套，不过HTML本身会忽略这些缩进（或所谓的空白符），你不需要在你自己的文档中增加这些缩进。实际上，在本书的大多数例子中，我都没有对代码做这种缩进。

这个文档包含一个html元素，由开始标记<html>指示，到结束标记(</html>)结束。

就像这个文档一样，HTML文档通常有一个head元素和一个body元素。这个head元素包含一个元素title。在不同的浏览器中，HTML标题会出现在不同的位置。图1-3显示了在Firefox浏览器中，标题“Very simple example”不仅会出现在屏幕的左上方，还会出现在一个标签页中。



图1-3 Firefox中HTML标题出现在两个位置

大多数情况下，你可能会在网页中创建一个你所认为的“标题”，但它并不是HTML标题！图1-3还显示了网页的内容：一小段文本。注意html、head、title和body等单词都没有出现。这些标记只是“告诉”浏览器如何显示HTML文档。

对文本还可以做更多工作，不过下面先来看如何显示图像。这需要一个img元素。与html、head和body元素（它们都要使用开始标记和结束标记）不同，img元素只使用一个标记。这称为一个单例标记。其元素类型是img（而不是image），要使用属性（attribute）将所有信息放入这个标记本身。什么信息呢？最重要的就是包含图像的文件的名称。标记

```

```

告诉浏览器要查找一个名为frog而且文件类型为jpg的文件。在这里，浏览器会在HTML文件所在

的同一目录或文件夹下查找这个文件。也可以指示其他位置的图像文件，稍后就会介绍。src代表源（source）。这称为元素的一个属性。>前面的斜线指示这是一个单例标记。不同的元素类型有一些共同的属性，不过大多数元素类型还有一些额外的属性。img元素的另一个属性是width属性。

```

```

这个属性指定了这个图像应当以200像素的宽度显示，高度要保证图像原有的宽高比。如果希望指定特定的宽度和高度，可以同时指定width和height属性（不过这有可能会让图像变形）。

**提示** 你会看到有些例子（甚至包括我给出的一些例子）会省略斜线，而且也能正常工作，不过一般认为最好包含斜线。类似地，你会看到有些例子中文件名两边没有引号。相对于大多数其他编程系统来说，HTML对于语法（拼法）要更为宽松。最后一点，你会看到有些HTML文档会从一个类型为!DOCTYPE的奇怪标记开始，而且允许HTML标记包含其他信息。目前我们还不需要这样做，所以我的原则是尽可能简单（而不是过于简单，这里援引了爱因斯坦的名言<sup>①</sup>）。

生成超链接与生成图像类似。超链接的元素类型为a，它的重要属性是href。

```
<a href="http://faculty.purchase.edu/jeanine.meyer">Jeanine Meyer's Academic
Activities </a>
```

可以看到，这个元素有一个开始标记和一个结束标记。元素的内容，也就是这两个标记之间的所有文本（这里就是Jeanine Meyer's Academic Activities）会用蓝色显示，而且有下划线。开始标记以a开头。要想记住这一点，一种办法是可以认为它是HTML中最重要的元素，所以使用了字母表中的第一个字母。也可以把它想成是一个锚（anchor），这才是a真正代表的含义，不过对我来说这种解释没有太大意义。href属性（可以认为是超文本引用）指定了单击这个超链接时浏览器要导航到哪个网站。注意这是一个完全Web地址[称为统一资源定位符（Universal Resource Locator），或简称为URL]。

可以把hyperlink元素与img元素结合起来，在屏幕上生成一个允许用户单击的图片。要记住，元素可以嵌套在其他元素中。在开始标记<a>后可以不放文本，而是放置一个<img>标记：

```
<a href="http://faculty.purchase.edu/jeanine">

</a>
```

下面把这些例子集成在一起：

```
<html>
<head>
<title>Second example </title>
</head>
```

① 爱因斯坦名言的原文为“Everything should be made as simple as possible, but not simpler”，即一切事物应该变得尽可能简单，而不是过于简单。——译者注

```

<body>
This will appear as is.


<a href=http://faculty.purchase.edu/jeanine.meyer>Jeanine Meyer's Academic
Activities </a>
<a href=http://faculty.purchase.edu/jeanine.meyer></a>
</body>
</html>

```

我创建了这个HTML文件，保存为second.html，然后在Chrome浏览器中打开这个文档。图1-4给出了显示结果。

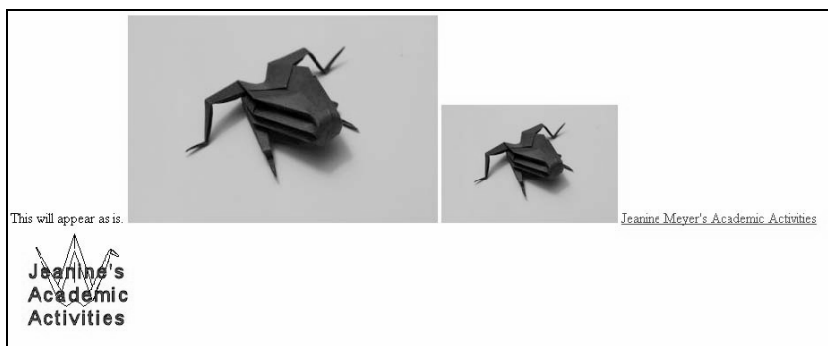


图1-4 包含图像和超链接的示例

这个文档会生成以下内容：一个文本，一个按原始宽度和高度显示的图像，一个宽度固定为200像素、高度按比例调整的图像，一个超链接[它会带你进入我的网页（我可以保证这一点）]，以及一个图像链接，它也可以把你带到我的网页。不过，这还不是我期望的效果，我希望这些元素在页面上自上向下排列。

由此说明HTML会忽略换行符和其他空白符，你要记住这一点。如果想增加一个换行，必须特别指定它。一种办法是使用br单例标记。后面我还会介绍其他方法。下面来看看修改过的代码。注意并不要求<br/>标记单独成一行。

```

<head>
<title>Second example </title>
<body>
This will appear as is. <br/>

<br/>

<br/>
<a href=http://faculty.purchase.edu/jeanine.meyer>Jeanine Meyer's Academic
Activities </a>
<br/>
<a href=http://faculty.purchase.edu/jeanine.meyer></a>
</body>
</html>

```



图1-5显示了这个代码生成的结果。

有很多种HTML元素类型：h1到h6标题元素可以生成不同大小的文本；另外还有很多表示列表和表格的元素，以及一些表示表单的元素。稍后会看到，CSS也用于指定格式。可以选择不同的字体、背景色、文本颜色，还可以控制文档的布局。由CSS完成格式化，由JavaScript提供交互性，而让HTML负责内容，这是一种很好的实践做法。HTML5提供了一些新的结构元素，如article、section、footer和header，这使得由CSS完成格式化更为容易。这样一来，你可以很轻松地改变格式和调整交互。格式化（包括文档布局）是一个范围很广的主题。本书中我只会介绍有关的基础知识。

### 使用CSS

CSS是一种专门用于格式化的特殊语言。样式实际上就是一个规则，指定了如何格式化一个特定元素。这说明，样式信息可以放在很多不同位置：可以放在

一个单独的文件中，可以放在head元素中的一个style元素里，或者放在HTML文档中的一个样式里（例如，假设你希望以一种特定的方式格式化一个元素，可能样式信息就放在这个元素中）。样式信息可以向下层叠，除非下一级指定了一个不同的样式。换句话说，与元素离得最近的样式就是这个元素要使用的样式。例如，大多数文本都使用head元素style节中指定的统一字体，不过另外会在local元素中包含规范，为某段特定文本指定样式。由于这个样式与元素最接近，所以这就是这个元素要使用的样式。

基本格式包括一个指示符，指出要对什么元素格式化，后面是一个或多个指令。在本章的应用中（可以从[www.friendsofed.com/downloads.html](http://www.friendsofed.com/downloads.html)得到），我为section类型的元素指定了格式，具体来说就是在每一项周围加一个边框（或方框），并指定了外边距、内边距和对齐方式，还指定背景色为白色。完整的HTML文档（见代码清单 1-1）混合了多项特性（可能有人会说这太乱了）。元素body和p（段落）是HTML原先版本就有的元素。section元素是HTML5新增的元素类型之一。section元素确实需要格式化，这与body和p元素不同（它们会采用默认格式，即body和各个p元素都会另起一行）。CSS可以修改原有元素类型以及新元素类型的格式。注意节(section)中文本的背景色与节之外文本的背景色不同。

在代码清单1-1中，我指定了body元素（只有一个）以及section元素的样式。如果有多个section元素，这里的样式会分别应用到每一个元素。body元素的样式指定了背景色和文本颜色。CSS接受一组16个命名颜色，包括黑(black)、白(white)、红(red)、蓝(blue)、绿(green)、青(cyan)和粉红(pink)。也可以使用RGB(Red Green Blue)十六进制码指定颜色，不过为此需要使用一个图像程序，如Adobe Photoshop、Corel Paint Shop Pro或Adobe Flash Professional来得

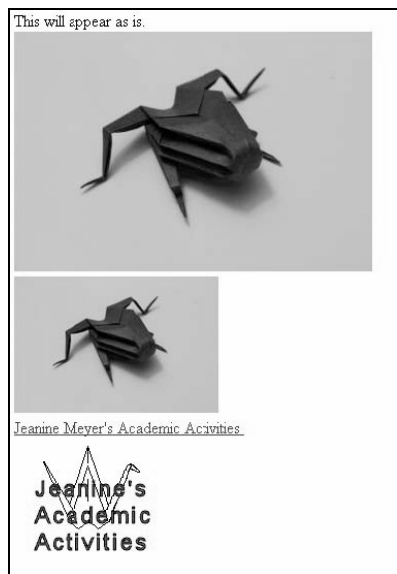


图1-5 文本、图像和链接（加入换行）



出RGB值，或者你也可以自己试验得出。我自己使用了Paint Shop Pro来确定“青蛙头”图片中绿色的RGB值，另外边框也使用这个颜色。

text-align指令顾名思义：指示将内容居中还是左对齐。font-size按像素设置文本的大小。边框要麻烦一些，在不同浏览器中的显示可能不一致。这里我指定了一个绿色的4像素实线边框。section的width规范指示浏览器应当使用窗口的85%（不论是什么浏览器）。p的规范设置段落宽度为250像素。内边距（padding）是指文本和节边框之间的间距。外边距（margin）是节与周围其他节之间的间距。

#### 代码清单1-1 一个包含样式的完整HTML文档

```
<html>
<head>
<title>CSS example </title>
<style>
body {
    background-color:tan;
    color: #EE015;
    text-align:center;
    font-size:22px;
}
section {
    width:85%;
    border:4px #00FF63 solid;
    text-align:left;
    padding:5px;
    margin:10px;
    background-color: white;
}

p {
    width: 250px;
}
</style>
</head>
<body>
The background here is tan and the text is the totally arbitrary RED GREEN BLUE
value #EE015. <br/>
<section>Within the section, the background color is white. There is text with
additional HTML markup, followed by a paragraph with text. Then, outside the
section there will be text, followed by an image, more text and then a
hyperlink. <p>The border color of the section matches the color of the
frog image. </p></section>
<br/>
As you may have noticed, I like origami. The next image represents a frog head.<br/>
 <br/>If you want to learn how to fold it, go to

<a href=http://faculty.purchase.edu/jeanine.meyer/origami>the Meyer Family
Origami Page </a>

</body>
</html>
```

这会生成如图1-6所示的屏幕。

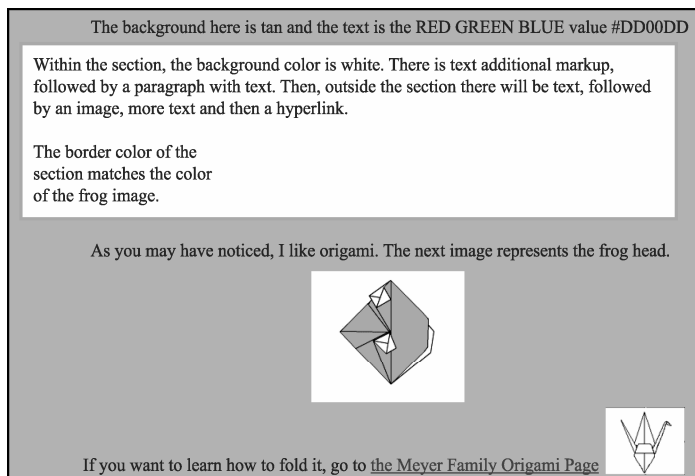


图1-6 示例CSS样式

**提示** 如果你不能马上完全理解，也不用担心——可以在网上得到很多帮助。尤其可以参考HTML5的官方资源：<http://dev.w3.org/html5/spec/Overview.html>。

利用CSS可以做很多事情。可以用它指定不同类型元素的格式（如上面所示），可以指定元素是某个类的一部分，可以使用id属性标识单个元素。第6章中我们创建了一个猜谜游戏，其中就使用CSS指定特定元素在窗口中的位置，然后使用JavaScript移动这些元素。

### 1.3.2 JavaScript 编程

JavaScript是一种编程语言，提供了很多内置特性来访问HTML文档的各个部分，包括CSS元素中的样式。这种语言被称为一种脚本语言，以区别编译语言（如C++）。编译语言会在使用前一次性完全得到“翻译”，而脚本语言会由浏览器逐行解释。下面的介绍假设你之前没有任何编程经验，或者完全不了解JavaScript，不过可以参考一些其他图书，如Terry McNavage著的*Getting Started with JavaScript*（friends of ED，2010），或者查看一些在线资源，如<http://en.wikipedia.org/wiki/JavaScript>，这会很有帮助。每个浏览器都有自己的JavaScript版本。

HTML文档将JavaScript放在一个script元素里，script元素则位于head元素中。要显示如图1-2所示的时间和日期信息，我在HTML文档的head元素中加入以下代码：

```
<script>
document.write(Date());
</script>
```

类似于其他编程语言，JavaScript也是由不同类型的语句构成的。在后面几章中，我会介绍赋值语句、复合语句（如if、switch和for语句），以及创建所谓“程序员自定义函数”的语句。函数就是可以一起构成块的一个或多个语句，只要需要相应功能就可以调用这个函数。利用函数

可以避免反复写相同的代码。JavaScript提供了很多内置函数。有些函数与对象关联（稍后会介绍更多有关内容），这些函数称为方法。以下代码

```
document.write("hello");
```

是一个JavaScript语句，它调用了document对象的write方法，并提供参数"hello"。参数是传入函数或方法的额外信息。语句用分号结束。这个代码会写出由字符h、e、l、l和o组成的字面量字符串作为HTML文档的一部分。

document.write方法会写出括号内的所有内容。由于我希望写出的信息随日期和时间改变，所以需要一种方法来访问当前日期和时间，因此我使用了JavaScript的内置函数Date。这个函数会利用日期和时间生成一个对象。后面你会了解到如何使用Date对象来计算玩家完成游戏所花费的时间。对现在来说，我只想显示当前的日期和时间信息，这正是代码

```
document.write(Date());
```

要做的工作。按正式的编程语言来讲：这个代码要调用document对象的write方法，这是一段内置代码。点号（.）指示所要调用的write是一个与HTML文件生成的文档相关的方法。所以，它会写出一些内容以作为HTML文档的一部分。那么会写些什么呢？就是开始括号和结束括号之间的所有内容。那具体是什么呢？这是调用内置函数Date的结果。Date函数会得到本地计算机维护的信息，并把它交给write方法。Date也需要使用括号，正是这个原因你才会看到这么多括号。write方法会显示日期和时间信息，作为HTML文档的一部分，如图1-2所示。以这种方式结合多种构造，这在编程语言中很典型。语句最后有一个分号。为什么不是点号呢？点号在JavaScript中还有其他用途，如指示方法，另外点号还要作为数字的小数点。

自然语言（比如英语）与编程语言有很多共同之处：有不同的语句类型，使用某些符号作为标点符号，另外有一种文法来规定放置元素的正确位置。在编程中，我们使用术语“记法”（notation）而不是“标点符号”，使用术语“语法”而不是“文法”。编程语言和自然语言都允许根据单独的部分建立相当复杂的语句。不过，这里有一个根本的区别：正如我对学生们所讲，课堂上我说的很多话在文法上可能并不正确，但是他们仍然能听懂我的意思。不过，通过一个编程语言与计算机“交谈”时，你的代码必须完全符合这种语言的语法规则，你才能达到目的。好消息是：与人类听众不同，计算机不会表现出不耐烦，也不会有人类的其他情绪，所以你可以根据你的需要花时间把语法改正确（哪怕时间很长）。坏消息是（可能需要花些时间来理解这一点）：如果你的HTML、CSS或JavaScript在文法上有错误（这称为语法错误），浏览器还是会试图显示一些结果。如果未能得到你想要的结果，要由你来找出到底出了什么问题，以及哪里出了问题。

## 1.4 构建自己的应用

使用文本编辑器建立一个HTML文档，并使用浏览器查看/测试/尝试执行这个文档。你当然可以使用任何文本编辑器程序来编写HTML，不过我建议在PC上使用TextPad，在Mac上使用TextWrangler。这些工具都是共享软件，因此相当廉价。不要使用字处理程序，这会插入一些非文本字符。也可以使用记事本，不过使用TextPad还有一些好处，如它会用不同颜色显示代码的不同部分（有关内容后面将会介绍）。要使用编辑器，需要打开编辑器，键入代码。图1-7显示了

TextPad的屏幕。

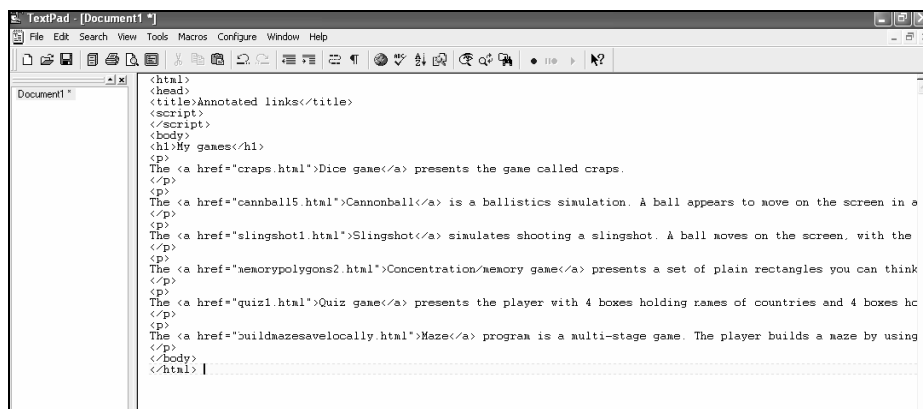


图1-7 从TextPad开始

要经常保存你的工作，最重要的是文件类型要保存为.html。在TextPad中，单击File → Save As，然后把Save as type（另存为类型）改为HTML，如图1-8所示。

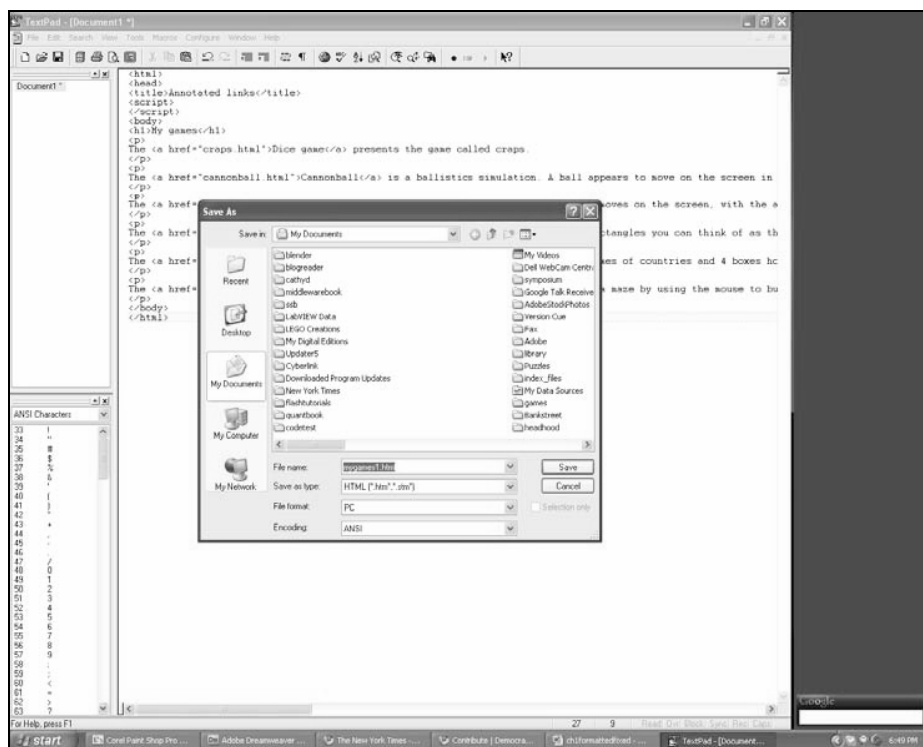


图1-8 文件保存为HTML类型

注意，我为这个文件指定了一个文件名，也可以把文件夹从My Documents改为我想要的任何其他文件夹。保存文件后，单击Configure → Word Wrap（使长代码行在屏幕上可见），窗口如图1-9所示。

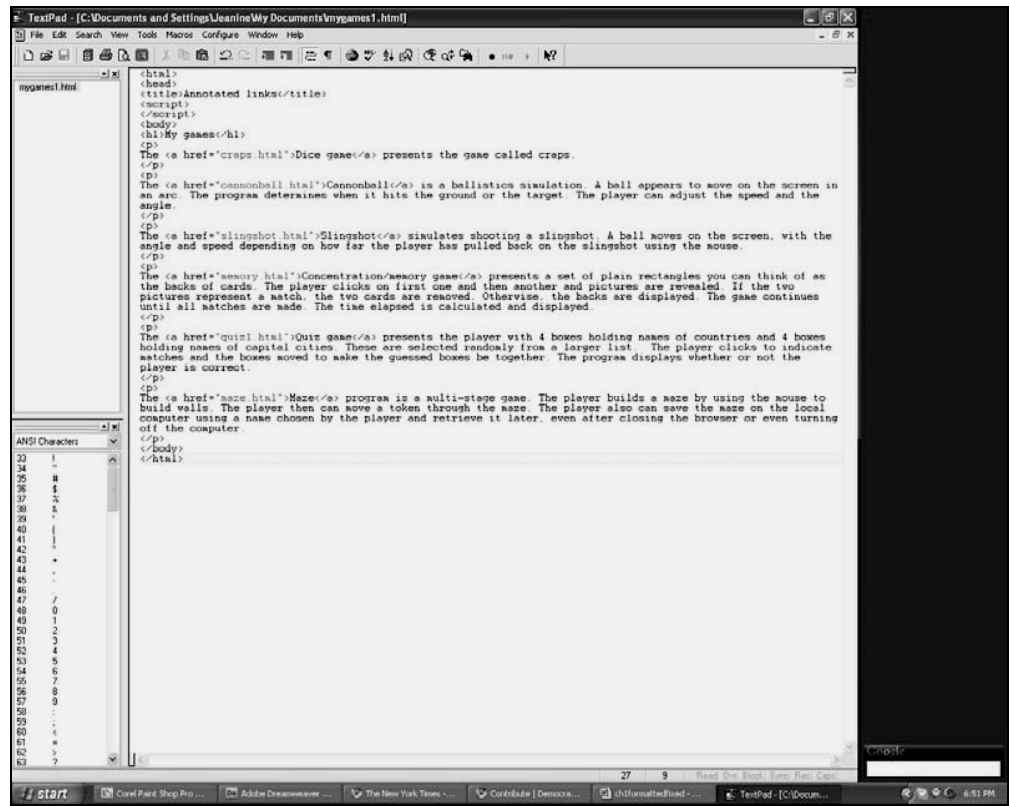


图1-9 文件保存为HTML并要求保证单词完整的条件下自动换行

代码的不同部分会用不同颜色显示（color coding），不过只有当文件保存为HTML之后才会看到这一点，这会指示出标记和加引号的字符串。这对于捕获错误很有意义。

现在来具体查看HTML代码，首先来看加注解的链接列表应用，然后考虑最喜爱网站列表应用。代码中使用了上一节介绍的特性。表1-1显示了这个应用的完整代码：文本段落包含指向不同文件的链接，这些文件都位于同一个文件夹下。

表1-1 加注解链接列表应用“ My games” 的代码

代 码	解 释
<html>	开始html标记
<head>	开始head标记
<title>Annotated links</title>	开始title标记、标题文本和结束title标记

(续)

代 码	解 释
<code>&lt;body&gt;</code>	开始body标记
<code>&lt;h1&gt;My games&lt;/h1&gt;</code>	开始h1标记、文本，然后是结束h1标记。这会 用一种大字体显示“My games”。具体字体取默 认设置
<code>&lt;p&gt;</code>	开始p标记，表示段落标记
The <code>&lt;a href="craps.html"&gt;Dice game&lt;/a&gt;</code> presents the game called craps.	包含一个a元素的文本。开始a标记的属性href 设置为值craps.html。假设这是与这个HTML文件 在同一个文件夹下的文件。a元素的内容（ <code>&lt;a&gt;</code> 和 <code>&lt;/a&gt;</code> 之间的部分）都会显示，开始时显示为蓝 色，一旦单击会显示为紫红色，而且有下划线
<code>&lt;/p&gt;</code>	结束p标记
<code>&lt;p&gt;</code>	开始p标记
The <code>&lt;a href="cannonball.html"&gt;Cannonball&lt;/a&gt;</code> is a ballistics simulation. A ball appears to move on the screen in an arc. The program determines when the ball hits the ground or the target. The player can adjust the speed and the angle.	见前面的说明。这里a元素指示cannonball.html 文件，显示的文本是Cannonball
<code>&lt;/p&gt;</code>	结束p标记
<code>&lt;p&gt;</code>	开始p标记
The <code>&lt;a href="slingshot.html"&gt;Slingshot&lt;/a&gt;</code> simulates shooting a slingshot. A ball moves on the screen, with the angle and speed depending on how far the player has pulled back on the slingshot using the mouse.	见前面的说明。这个段落包含指向slingshot.html 的超链接
<code>&lt;/p&gt;</code>	结束p标记
<code>&lt;p&gt;</code>	开始p标记
The <code>&lt;a href="memory.html"&gt;Concentration/memory game&lt;/a&gt;</code> presents a set of plain rectangles you can think of as the backs of cards. The player clicks on first one and then another and pictures are revealed. If the two pictures represent a match, the two cards are removed. Otherwise, the backs are displayed. The game continues until all matches are made. The time elapsed is calculated and displayed.	见前面的说明。这个段落包含指向memory.html 的超链接
<code>&lt;/p&gt;</code>	结束p标记
<code>&lt;p&gt;</code>	开始p标记
The <code>&lt;a href="quiz1.html"&gt;Quiz game&lt;/a&gt;</code> presents the player with 4 boxes holding names of countries and 4 boxes holding names of capital cities. These are selected randomly from a larger list. The player clicks to indicate matches and the boxes are moved to put the guessed boxes together. The program displays whether or not the player is correct.	见前面的说明。这个段落包含指向quiz1.html 的超链接
<code>&lt;/p&gt;</code>	结束p标记
<code>&lt;p&gt;</code>	开始p标记

(续)

代 码	解 释
The <a href="maze.html">Maze</a> program is a multi-stage game. The player builds a maze by using the mouse to build walls. The player then can move a token through the maze. The player can also save the maze on the local computer using a name chosen by the player and retrieve it later, even after closing the browser or turning off the computer.	见前面的说明。这个段落包含指向maze.html的超链接
</p>	结束p标记
</body>	结束body标记
</html>	结束html标记

最喜爱网站 (Favorite Site) 应用的代码同样提供了带注解的列表, 另外还增加了格式化设置: 每一项周围都有一个绿色的方框, 而且每一项中包含一个图片, 见表1-2。

表1-2 最喜爱网站应用代码

代 码	解 释
<html>	开始html标记
<head>	开始head标记
<title>Annotated links</title>	title元素: 开始标记和结束标记, 中间是Annotated links
<style>	开始style标记。这说明我们现在要使用CSS
Article {	样式开始。将对所有section元素应用这个样式。样式后有一个大括号{。开始和结束大括号之间是我们创建的样式规则, 类似于HTML中的开始和结束标记
width:60%;	width设置为包含元素 (包含当前元素的外层元素) 的60%。注意每个指令最后都有一个;
text-align:left;	文本左对齐
margin:10px;	外边距为10像素
border:2px green double;	边框是2像素宽的绿色双线
padding:2px;	文本与边框之间间隔为2像素
display:block;	article是一个块, 这说明前面和后面分别有换行
}	结束article样式
</style>	结束style标记
<script>	开始script标记。现在开始写JavaScript代码
document.write(Date());	一个代码语句: 写出Date()调用生成的结果
</script>	结束script标记
<body>	开始body标记
<h3>Favorite Sites</h3>	h3和/h3标记包围的文本。这会让文本比正常文本大一些
<article>	开始article标记



(续)

代 码	解 释
The<a href="../../index.html">Jeanine Meyer's Academic Activities</a> displays information on my current and past courses, along with publications and other activities.	这个文本要应用指定的样式。它包括一个a元素。注意href属性的值是一个相对引用：这表示要前往当前文件夹的父文件夹，然后找到index.html文件。两个点号(..) 就是“退回上一层文件夹”的计算机说法，所以如果我们现在正位于tree/fruit/apple文件夹，那么../index.html会把我们带回到fruit文件夹来查找index文件， ../../index.html则会把我们带回到tree文件夹
</article>	结束article标记
<article>	开始article标记
The <a href="http://stolenchair.org">Stolen Chair Theatre Company</a> is the web site of a theatre company performing mainly in New York City. This is the postcard for their Summer, 2010 production. 	见前面的说明。注意href属性的值是一个完整的Web地址，而且HTML包含一个 标记。这会强制换行
	img标记。图像源为文件postcard.jpg。宽度设置为300像素
</article>	结束article标记
<article>	开始article标记
The <a href="http://friendsofed.com/">friends of ED publishers</a> is the site for the publishers of this book.  	见前面的说明。这也指示一个Web地址。 标记要求在图像前强制换行
	img元素。源为friendsofed.gif。宽度设置为300像素
</article>	结束article标记
</body>	结束body标记
</html>	结束html标记

把这个应用变成你自己的应用非常简单：只需使用你自己最喜欢的网站即可。在大多数浏览器中，如果你想使用一个网站logo作为超链接，可以下载并保存图像文件，也可以包含其他图片。在我看来，建立一个带注释的网站列表以及包含类似logo的图像应该算是“合法使用”，不过我不是个律师，法律问题不是我的本行。大多数情况下，大家都很乐意有人链接到他们的网站。这不存在法律问题，不过如果你不想把一个特定的图像文件下载到你的计算机，然后再将其上传到你的网站，也可以选择将img标记中的src设置为图像所在网站的Web地址。

Web地址可以是绝对地址，也可以是相对地址。绝对地址以http://开头。相对地址是相对于HTML文件的位置。在上面的例子中，postcard.jpg和friendsofed.gif都位于与HTML文件所在的同一个文件夹中，因为我特意把它们放在那里！对于大型项目，很多人会把所有图像都放在一个名为images的子文件夹下，并把地址写为“images/postcard.gif”。

还可以通过改变格式把这个应用变成你自己的应用。可以使用样式来指定字体，这包括特定字体、字体系列和大小。这允许你选择一种你喜欢的字体，也可以指定当用户计算机上没有首选

字体时将使用哪种替代字体。你还可以指定外边距和内边距，或者单独地调整上外边距（margin-top）、左外边距（margin-left）、上内边距（padding-top）等。

## 1.5 测试和上传应用

你要把所有文件（这里就是一个HTML文件以及所有图像文件）都放在同一个文件夹下，除非使用完全Web地址。要让链接正常工作，需要为所有href属性提供正确的地址。前面的例子展示了对相同文件夹中的HTML文件以及位于Web上其他位置的HTML文件如何指定地址。

即使应用还没有完全完成，也可以开始测试你的工作。例如，可以放入一个img元素或者一个a元素。打开一个浏览器，如Firefox、Chrome或Safari（我没有提到Internet Explorer，因为它还不支持我在其他教程中将要使用的一些HTML5特性，不过将来的Internet Explorer 9可能会提供支持）。在Firefox中，单击File，然后选择Open file，浏览到你的HTML文件。在Chrome中，如果是PC要按下Ctrl键（如果是Mac，则按下CMD键）和o，然后浏览到你的文件，并单击OK按钮打开这个文件。应该能看到与我的例子类似的结果。单击超链接访问其他网站。使用浏览器的重载图标来重新加载页面，可以观察到不同的时间。如果没有看到你期望的结果（类似我的例子），就需要检查你的代码。会有以下常见的错误。

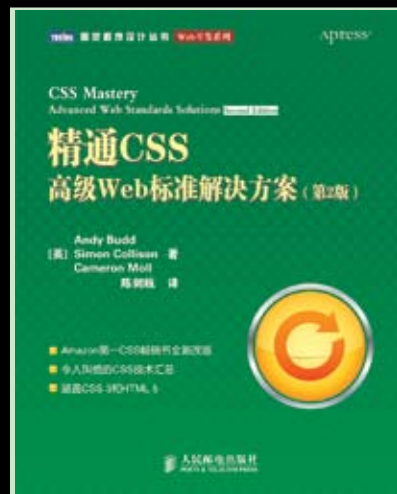
- ❑ 遗漏开始标记和结束标记，或者开始标记和结束标记不匹配。
- ❑ 图像文件或HTML文件的文件名不正确，或者图像文件的扩展名错误。可以使用JPG、GIF或PNG类型的图像文件，不过标记中指定的文件扩展名必须与图像的实际文件类型一致。
- ❑ 缺少引号。TextPad和其他一些编辑器提供的代码着色特性可以帮助你找出这种错误。

## 1.6 小结

本章中你已经了解了如何用文本、图像和超链接构成HTML文档。这包括：

- ❑ 基本标记，包括html、head、title、style、script、body；
- ❑ 显示图像的img元素；
- ❑ 提供超链接的a元素；
- ❑ 使用遵循CSS规则编写的style元素指定简单的格式；
- ❑ 用一行JavaScript代码提供日期和时间信息。

本章只是起点，不过不论是否使用CSS，都完全可以利用基本HTML生成富含信息的漂亮页面。下一章中，你会了解如何在应用中包含随机性和交互性，以及如何使用canvas元素，这是HTML5中一个非常重要的特性。



“本书内容全面，通过游戏示例介绍了HTML5的方方面面，涉及了碰撞检测、表单验证、生成和使用随机数、创建用户自定义的图片、鼠标事件、定时事件和使用localStorage在浏览器中存储/检索数据等，适合对HTML5、CSS和JavaScript感兴趣的人阅读。强烈推荐！”

——亚马逊读者评论

## The Essential Guide to HTML5

Using Games to Learn HTML5 and JavaScript

# HTML5游戏开发

今天，大多数现代浏览器都已经支持HTML5。学习体验HTML5带给Web开发的便捷、快速和强大功能，是每一位Web设计和开发人员的当务之急。

本书通过人们熟悉的一个个游戏让读者轻松掌握HTML5、CSS和JavaScript的新特性，并将这些特性灵活应用到Web开发中。书中主要介绍了使用canvas元素直接在屏幕上绘图、添加图像和编写文本，只使用HTML5、CSS和JavaScript在网站上加入视频和音频，使用事件处理响应鼠标事件和按键，使用定时事件生成动画，验证表单输入，在玩家的计算机上存储迷宫布局之类的信息并按需重新加载。跟随作者的指引，你也能开发出妙趣横生的小游戏。

- 引领下一代Web开发潮流
- 围绕10个游戏示例体验HTML5和JavaScript
- 寓教于乐，循序渐进，轻松上手

Apress®

图灵社区：[www.it-ebooks.com.cn](http://www.it-ebooks.com.cn)

反馈/投稿/推荐信箱：[contact@turingbook.com](mailto:contact@turingbook.com)

热线：(010)51095186转604

**分类建议** 计算机/Web开发

人民邮电出版社网址：[www.ptpress.com.cn](http://www.ptpress.com.cn)

1 978 978-7-115-26363-6



ISBN 978-7-115-26363-6

定价：49.00元