

企业网站开发真实商业案例



HTML5 网页前端设计实战

◎ 周文洁 编著

知识全面 | 循序渐进 |

项目驱动 | 实用性强 |

步骤详细 | 易于理解 |

清华大学出版社





HTML5 网页前端设计实战

◎ 周文洁 编著



清华大学出版社
北京

内 容 简 介

本书是《HTML5 网页前端设计》一书的配套实战项目教程,也可单独为具有一定 Web 前端基础的读者使用。本书的电子资源包括全套例题源代码可供读者下载。

全书共包含 10 章,每章均配套两个实例项目。全部内容可分为以下 4 个部分:

第一部分是基础知识篇,包括第 1、2 章的内容。其中,第 1 章是 HTML+CSS 基础项目,介绍导航菜单与商务风格表格的设计与实现;第 2 章是 JavaScript 基础项目,介绍电子时钟与电子日历的设计与实现。

第二部分是重点篇,包括第 3~8 章的内容。这 6 章分别介绍基于 HTML5 拖放 API、表单 API、画布 API、媒体 API、地理定位 API 以及 Web 存储 API 的项目案例。其中较有特色的综合项目有手绘时钟、拼图游戏、网页日志本、音乐播放器、在线教学视频等。

第三部分是提高篇,包括第 9 章的内容。第 9 章是 CSS3 基础项目,主要讲解使用 CSS3 制作火焰和霓虹文字特效以及使用 CSS3 动画制作响应式放大菜单。

第四部分是综合篇,包括第 10 章的内容。第 10 章提供了两个完整的项目实例,包括贪吃蛇游戏的开发和企业文化用品展示网站的设计与实现。这两个项目实例综合应用了全书所学的知识,让读者所学即所用。

本书可作为高校计算机相关专业 HTML5 课程的实践教材,也可作为学习 HTML5 开发的自学教材或培训教材。

本书封面贴有清华大学出版社防伪标签,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

HTML5 网页前端设计实战/周文洁编著.—北京:清华大学出版社,2017

(Web 前端开发技术丛书)

ISBN 978-7-302-46468-6

I. ①H… II. ①周… III. ①超文本标记语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 024639 号

责任编辑:魏江江 王冰飞

封面设计:刘 键

责任校对:时翠兰

责任印制:沈 露

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

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:15

字 数:372 千字

版 次:2017 年 7 月第 1 版

印 次:2017 年 7 月第 1 次印刷

印 数:1~2000

定 价:35.00 元

产品编号:073581-01

前 言

本书是《HTML5 网页前端设计》一书的配套实战项目教程，也可单独为具有一定 Web 前端基础的读者使用。本书的电子资源包括全套例题的源代码可供读者下载。

全书共包含 10 章，每章均配套两个实例项目。全书内容可分为以下 4 个部分：

第一部分是基础知识篇，包括第 1、2 章的内容。其中，第 1 章是 HTML+CSS 基础项目，介绍导航菜单与商务风格表格的设计与实现；第 2 章是 JavaScript 基础项目，介绍电子时钟与电子日历的设计与实现。

第二部分是重点篇，包括第 3~8 章的内容。这 6 章分别介绍基于 HTML5 拖放 API、表单 API、画布 API、媒体 API、地理定位 API 以及 Web 存储 API 的项目案例。其中较有特色的综合项目有手绘时钟、拼图游戏、网页日志本、音乐播放器、在线教学视频等。

第三部分是提高篇，包括第 9 章的内容。第 9 章是 CSS3 基础项目，主要讲解使用 CSS3 制作火焰和霓虹文字特效以及使用 CSS3 动画制作响应式放大菜单。

第四部分是综合篇，包括第 10 章的内容。第 10 章提供了两个完整的项目实例，包括贪吃蛇游戏的开发和企业文化用品展示网页的设计与实现。这两个项目实例综合应用了全书所学的知识，让读者所学即所用。

本书有以下几个特点。

1. 知识全面，循序渐进

本书首先循序渐进地介绍了一些基于 HTML、CSS 和 JavaScript 的基础项目，帮助读者打好基本功；然后正式进入 HTML5 新增 API 的相关项目介绍，让读者有针对性地逐步巩固常用 HTML5 新增 API 的用法；接着在提高篇补充了 CSS3 相关技术为页面增加特效；最后提供了两个完整项目实例，让读者进一步提高自己对于知识的综合应用能力。

2. 项目驱动，实用性强

全书将主教材各章节的知识点融入综合项目案例中，帮助读者更好地理解所学知识。在本书的最后一章还包含了游戏开发以及企业网站开发的真实商业案例，具有较高的实用价值，也适合培养读者的动手能力。

3. 步骤详细，易于理解

本书思路清晰、知识点循序渐进展开，每章的项目案例均分步骤讲解，读者可以看到从界面设计开始到样式美化以及特效完成的整个变化过程。读者跟着每章的综合案例独立完成开发过程即可基本达到前端开发的要求。

本书全部例题均在浏览器中调试通过，由于很多 HTML5 和 CSS3 的代码需要较高版本的浏览器才能提供更好的体验效果，建议读者使用但不限于 Internet Explorer 10.0、

Chrome 17.0、Firefox 10.0、Safari 5.0 或 Opera 11.1 以上版本的浏览器。

最后感谢家人和朋友给予的关心和大力支持，本书能够完成与你们的鼓励是分不开的。

愿本书能够对读者学习 Web 前端新技术有所帮助，并真诚地欢迎读者批评指正，希望能与读者朋友们共同学习成长，在浩瀚的技术之海不断前行。

作者

2017年3月

目 录

第 1 章 HTML+CSS 基础项目	1
1.1 导航栏菜单的设计与实现	1
1.1.1 界面设计	1
1.1.2 鼠标事件	4
1.1.3 完整代码展示	5
1.2 商务风格表格的设计与实现	6
1.2.1 创建表格	6
1.2.2 样式设计	7
1.2.3 完整代码展示	11
第 2 章 JavaScript 基础项目	13
2.1 电子时钟的设计与实现	13
2.1.1 界面设计	13
2.1.2 时钟动态效果的实现	16
2.1.3 完整代码展示	18
2.2 电子日历的设计与实现	19
2.2.1 界面设计	19
2.2.2 显示状态栏中的年份和月份	22
2.2.3 显示当前月份的所有日期	23
2.2.4 按钮控件功能的实现	26
2.2.5 完整代码展示	27
第 3 章 HTML5 拖放 API 项目	31
3.1 仿回收站效果的设计与实现	31
3.1.1 界面设计	31
3.1.2 文件拖曳与回收功能的实现	34
3.1.3 完整代码展示	37
3.2 图片相框展示的设计与实现	39
3.2.1 界面设计	39
3.2.2 相框自动生成功能的实现	41
3.2.3 完整代码展示	46

第 4 章 HTML5 表单 API 项目	49
4.1 用户注册页面的设计与实现.....	49
4.1.1 界面设计.....	50
4.1.2 表单设计.....	51
4.1.3 提示与验证功能的实现.....	54
4.1.4 完整代码展示.....	58
4.2 问卷调查页面的设计与实现.....	60
4.2.1 界面设计.....	61
4.2.2 表单设计.....	62
4.2.3 验证功能的实现.....	69
4.2.4 完整代码展示.....	73
第 5 章 HTML5 画布 API 项目	79
5.1 手绘时钟的设计与实现.....	79
5.1.1 界面设计.....	79
5.1.2 实时更新效果.....	87
5.1.3 完整代码展示.....	88
5.2 拼图游戏的设计与实现.....	91
5.2.1 界面设计.....	91
5.2.2 实现游戏逻辑.....	95
5.2.3 游戏成功与重新开始.....	102
5.2.4 完整代码展示.....	105
第 6 章 HTML5 媒体 API 项目	111
6.1 音乐播放器的设计与实现.....	111
6.1.1 界面设计.....	111
6.1.2 媒体文件的载入.....	115
6.1.3 控件功能的实现.....	115
6.1.4 完整代码展示.....	119
6.2 在线教学视频的设计与实现.....	122
6.2.1 界面设计.....	122
6.2.2 视频文件的载入与播放.....	127
6.2.3 视频时间跳转的实现.....	128
6.2.4 完整代码展示.....	130
第 7 章 HTML5 地理定位 API 项目	133
7.1 运动数据记录页面的设计与实现.....	133
7.1.1 界面设计.....	134

7.1.2	实时监控地理定位	135
7.1.3	开始与结束按钮的切换	136
7.1.4	计算与显示距离	138
7.1.5	完整代码展示	140
7.2	运动轨迹绘制页面的设计与实现	144
7.2.1	界面设计	144
7.2.2	开始与结束按钮的切换	146
7.2.3	绘制地图与运动轨迹	146
7.2.4	计时功能的实现	151
7.2.5	完整代码展示	153
第 8 章	HTML5 Web 存储 API 项目	157
8.1	基于 Web 存储技术的网页主题设置	157
8.1.1	界面设计	157
8.1.2	重置网页主题颜色的实现	160
8.1.3	加载网页主题颜色的实现	162
8.1.4	完整代码展示	163
8.2	基于 Web 存储技术的网页日志本	165
8.2.1	界面设计	165
8.2.2	读取日志功能的实现	168
8.2.3	保存日志功能的实现	169
8.2.4	删除日志功能的实现	170
8.2.5	完整代码展示	171
第 9 章	CSS3 基础项目	175
9.1	使用 CSS3 文本阴影制作特殊字体效果	175
9.1.1	整体设计	175
9.1.2	火焰文字效果的实现	177
9.1.3	霓虹文字效果的实现	178
9.1.4	完整代码展示	179
9.2	使用 CSS3 动画制作响应式放大菜单	180
9.2.1	整体设计	180
9.2.2	动画效果的实现	182
9.2.3	完整代码展示	183
第 10 章	综合应用设计实例	185
10.1	基于 HTML5 的贪吃蛇游戏的设计与实现	185
10.1.1	贪吃蛇游戏简介	185
10.1.2	界面布局设计	185

10.1.3	数据模型设计	190
10.1.4	游戏的逻辑实现	192
10.1.5	完整代码展示	203
10.2	实战项目——企业文化用品展示网页的开发	207
10.2.1	项目简介	207
10.2.2	整体布局设计	208
10.2.3	页眉和页脚的实现	210
10.2.4	主体内容的实现	212
10.2.5	完整代码展示	225

本章主要包含两个基于 HTML+CSS 的应用设计实例，一是导航栏菜单的设计与实现，二是商务风格表格的设计与实现。在导航栏菜单项目中，主要难点为超链接样式的设计以及鼠标悬浮与点击事件的处理；在商务风格表格项目中，主要内容包括表格总体样式、单元格样式以及单元行样式的设计实现。

本章学习目标：

- 学习如何综合应用 HTML 与 CSS 开发导航栏菜单样式；
- 学习如何综合应用 HTML 与 CSS 开发商务风格表格样式。

1.1 导航栏菜单的设计与实现

【例 1-1】 简单菜单导航栏的设计与实现

功能要求：使用 CSS 样式可以制作较为美观的导航栏效果，试设计一款横向的菜单导航栏用于页面，设计图如图 1-1 所示。

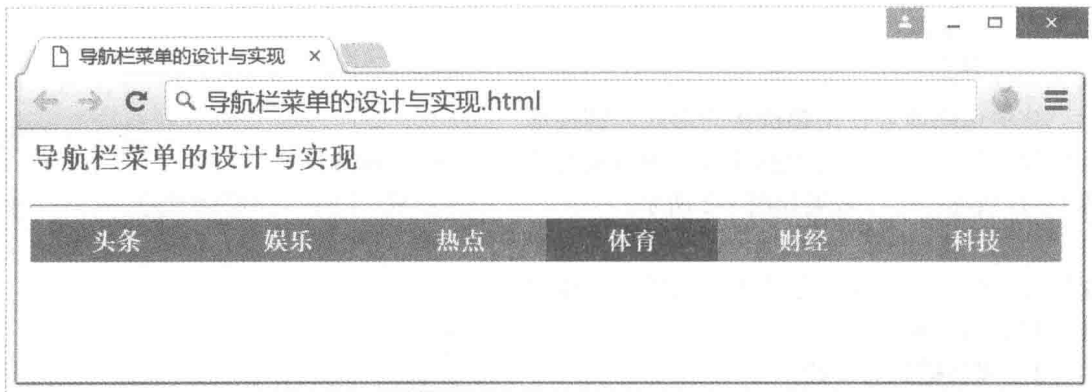


图 1-1 导航栏菜单的显示效果

1.1.1 界面设计

本节主要介绍导航栏菜单的界面设计，包括以下内容：

- 导航栏的创建；
- 列表样式的设计；
- 超链接样式的设计。

1. 导航栏的创建

使用<nav>标签在页面上创建导航栏菜单区域。相关 HTML5 代码片段如下：

```
<body>
  <h3>导航栏菜单的设计与实现</h3>
  <hr/>
  <!--导航栏-->
  <nav>
  </nav>
</body>
```

<nav>标签是 HTML5 新增的文档结构标签，用于标记导航栏，以便后续与网站的其他内容整合。

在<nav>的首尾标签之间使用无序列表标签配合列表选项创建菜单选项。相关 HTML5 代码片段如下：

```
<body>
  <h3>导航栏菜单的设计与实现</h3>
  <hr/>
  <!--导航栏-->
  <nav>
    <ul>
      <li><a href="#">头条</a></li>
      <li><a href="#">娱乐</a></li>
      <li><a href="#">热点</a></li>
      <li><a href="#">体育</a></li>
      <li><a href="#">财经</a></li>
      <li><a href="#">科技</a></li>
    </ul>
  </nav>
</body>
```

其中选项内容使用超链接的形式，链接地址当前设置为#，用户可根据实际情况改成具体的 URL 地址。运行效果如图 1-2 所示。

由图 1-2 可见，导航菜单中的文字内容和超链接已经完成，用户单击不同的选项可以跳转至指定的页面。

2. 列表样式的设计

此时会带有元素的默认样式，即每个列表选项前面有实心点标记，可以使用 list-style 属性将其去掉，采用 CSS 内部样式表的形式在<head>标签中进行声明，相关代码如下：

```
<head>
  <meta charset="utf-8">
  <title>导航栏菜单的设计与实现</title>
  <style>
    ul {
      list-style: none; /*用于去掉列表的实心点标记*/
    }
  </style>
</head>
```

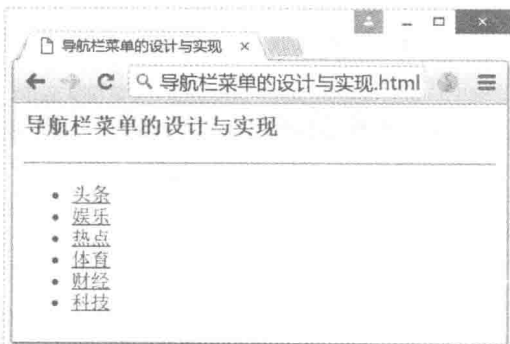


图 1-2 导航栏菜单的初步显示效果

```
margin: 0;
padding: 0;
}
</style>
</head>
```

其中, `margin` 和 `padding` 属性设置为 0 是为了预防不同的浏览器会出现预设值, 以避免最终布局效果可能产生的误差。

为列表元素 `` 定义浮动效果, 使其能够在同一行显示。相关代码如下:

```
<head>
  <meta charset="utf-8">
  <title>导航栏菜单的设计与实现</title>
  <style>
    ... (ul样式代码略)
    li {
      float: left;
    }
  </style>
</head>
```

运行效果如图 1-3 所示。

由图 1-3 可见, 列表已经去掉了实心点标记, 并且每个列表选项均在同一行显示。

3. 超链接样式的设计

首先在 CSS 内部样式表中为超链接重新设置样式, 具体要求如下。

- 颜色: 背景颜色为蓝色, 字体颜色为白色;
- 尺寸: 宽度为 100 像素;
- 边距: 各边的内边距为 5 像素;
- 文本: 居中显示, 并且去掉了文本的下划线样式。

相关代码片段如下:

```
<head>
  <meta charset="utf-8">
  <title>导航栏菜单的设计与实现</title>
  <style>
    ... (ul和li样式代码略)
    a:link, a:visited {
      display: block;           /*设置为块级元素*/
      width: 100px;           /*设置宽度为100像素*/
      font-weight: bold;      /*设置字体为加粗*/
      color: white;           /*设置字体为白色*/
      background-color: #5AF; /*设置背景颜色为蓝色*/
      text-align: center;     /*设置文本居中显示*/
      padding: 5px;          /*设置各边的内边距为5像素*/
      text-decoration: none;  /*去掉文本的下划线*/
    }
  </style>
</head>
```



图 1-3 导航栏菜单的初步显示效果

```
</style>
</head>
```

首先对超链接的 `a:link` 和 `a:visited` 进行设置，表示超链接未访问和已访问状态。然后将其 `display` 属性设置为 `block`，使得超链接成为块级元素，这样才可以为其设置宽度 100 像素。

此时页面效果如图 1-4 所示。

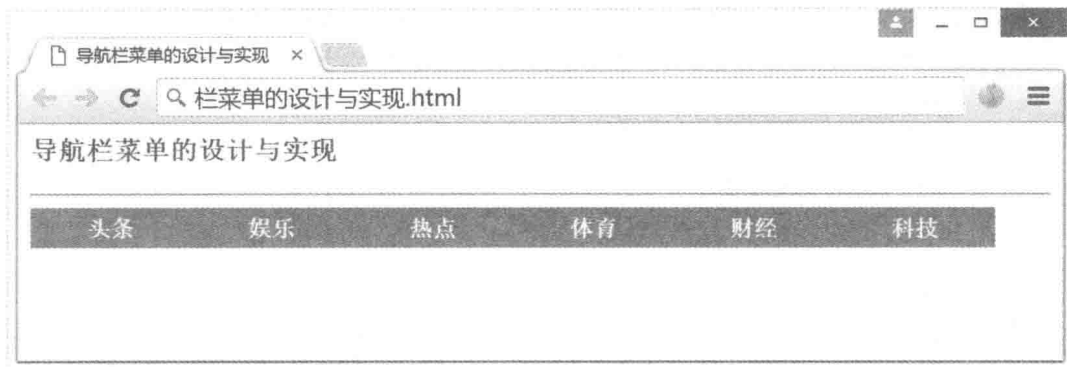


图 1-4 设置超链接样式后的效果图

由图 1-4 可见，关于导航菜单的样式要求已初步实现。下一节将介绍如何为菜单选项设计鼠标悬浮事件，当鼠标悬浮在某选项上时该选项的背景颜色发生改变。

1.1.2 鼠标事件

在 CSS 内部样式表中为 `a:hover` 和 `a:active` 进行样式设置，表示鼠标悬浮和单击未释放状态时的样式变化。相关代码片段如下：

```
<head>
  <meta charset="utf-8">
  <title>导航栏菜单的设计与实现</title>
  <style>
    ... (ul和li样式代码略)
    ... (a:link和a:visited代码略)
    a:hover, a:active{
      background-color: #006FDD;
    }
  </style>
</head>
```

上述代码表示将鼠标在超链接上悬浮和单击未释放的样式更新成背景颜色为深蓝色（十六进制码为#006FDD）。

此时页面效果如图 1-5 所示。

由图 1-5 可见，鼠标悬浮的效果已实现。如果需要竖向排列的菜单栏，去掉 `li` 元素的浮动样式即可。至此导航栏菜单的设计与实现已全部完成，用户可根据实际需要将其应用到网站开发中。

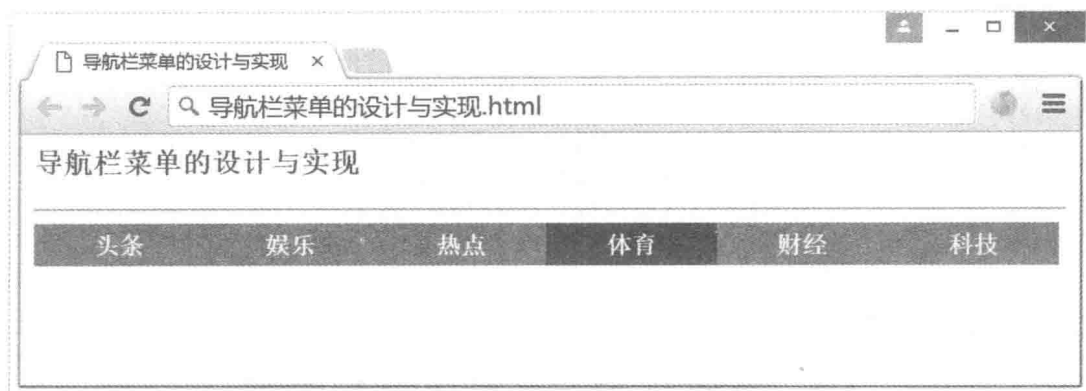


图 1-5 设置鼠标事件后的效果图

1.1.3 完整代码展示

完整的 HTML5 代码如下：

```
1.  <!DOCTYPE html>
2.  <html>
3.    <head>
4.      <meta charset="utf-8">
5.      <title>导航栏菜单的设计与实现</title>
6.      <style>
7.        ul {
8.          list-style: none; /*用于去掉列表的实心点标记*/
9.          margin: 0;
10.         padding: 0;
11.        }
12.        li {
13.          float: left;
14.        }
15.        a:link, a:visited {
16.          display: block; /*设置为块级元素*/
17.          width: 100px; /*设置宽度为100像素*/
18.          font-weight: bold; /*设置字体为加粗*/
19.          color: white; /*设置字体为白色*/
20.          background-color: #5AF; /*设置背景颜色为蓝色*/
21.          text-align: center; /*设置文本居中显示*/
22.          padding: 5px; /*设置各边的内边距为5像素*/
23.          text-decoration: none; /*去掉文本的下划线*/
24.        }
25.        a:hover, a:active {
26.          background-color: red;
27.        }
28.      </style>
29.    </head>
30.    <body>
31.      <h3>导航栏菜单的设计与实现</h3>
32.      <hr />
33.      <!--导航栏-->
34.      <nav>
```

```

35.         <ul>
36.             <li><a href="#">头条</a></li>
37.             <li><a href="#">娱乐</a></li>
38.             <li><a href="#">热点</a></li>
39.             <li><a href="#">体育</a></li>
40.             <li><a href="#">财经</a></li>
41.             <li><a href="#">科技</a></li>
42.         </ul>
43.     </nav>
44. </body>
45. </html>

```

1.2 商务风格表格的设计与实现

【例 1-2】 商务风格表格的设计与实现

功能要求：设计一款商务风格表格，运行效果如图 1-6 所示。

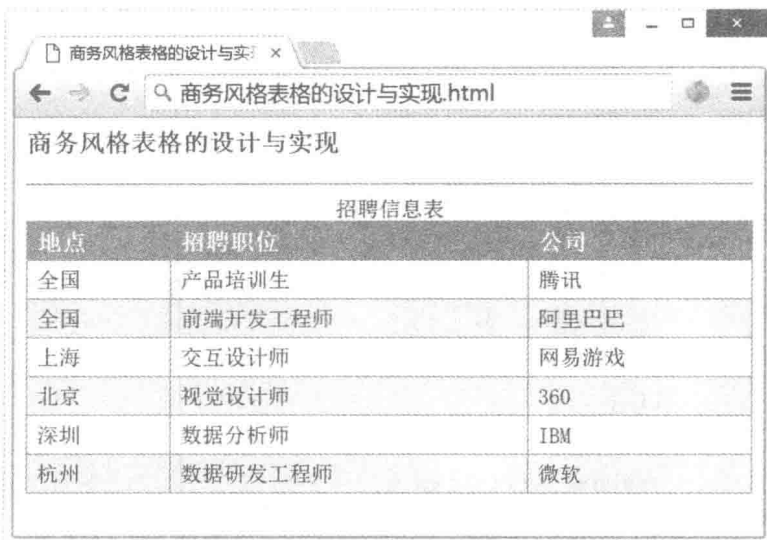


图 1-6 商务风格表格的显示效果

1.2.1 创建表格

使用<table>标签在页面上创建表格，并使用<tr>和<td>标签为其添加若干个单元行与单元格。相关 HTML5 代码片段如下：

```

<body>
  <h3>商务风格表格的设计与实现</h3>
  <hr />
  <table border="1">
    <caption>招聘信息表</caption>
    <tr><th>地点</th><th>招聘职位</th><th>公司</th></tr>
    <tr><td>全国</td><td>产品培训生</td><td>腾讯</td></tr>
    <tr><td>全国</td><td>前端开发工程师</td><td>阿里巴巴</td></tr>
    <tr><td>上海</td><td>交互设计师</td><td>网易游戏</td></tr>

```

```
<tr><td>北京</td><td>视觉设计师</td><td>360</td></tr>
<tr><td>深圳</td><td>数据分析师</td><td>IBM</td></tr>
<tr><td>杭州</td><td>数据研发工程师</td><td>微软</td></tr>
</table>
</body>
```

其中，<caption>标签用于显示表格的总标题，<th>标签用于显示表格的第 1 行标题。为方便显示运行效果，将<table>标签临时添加了行内样式 border="1"，表示表格具有 1 像素的边框线。此时运行效果如图 1-7 所示。

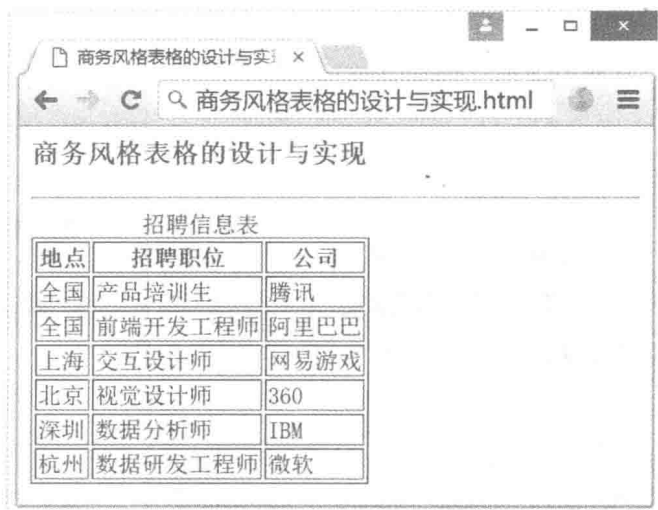


图 1-7 商务风格表格的显示效果

由图 1-7 可见，表格和相关数据已创建完成。下面介绍如何为表格设置样式。

1.2.2 样式设计

本节主要介绍表格的样式设计，包括以下内容：

- 表格的总体样式；
- 单元格样式；
- 单元行样式。

1. 表格的总体样式

为<table>标签增加自定义 id="recruit"，以便可以使用 CSS 的 ID 选择器为其规定样式。相关代码片段修改后如下：

```
<body>
  <h3>商务风格表格的设计与实现</h3>
  <hr />
  <table id="recruit" border="1">
    ... (内容略)
  </table>
</body>
```

采用 CSS 内部样式表的形式在<head>标签中为<table>标签设置样式：表格宽度为

100%，双线边框折叠为单线效果，文本对齐方式为左对齐。相关代码如下：

```
<head>
  <meta charset="utf-8">
  <title>商务风格表格的设计与实现</title>
  <style>
    /*设置表格的总体样式*/
    #recruit {
      width: 100%;
      border-collapse: collapse;
      text-align: left;
    }
  </style>
</head>
```

运行效果如图 1-8 所示。

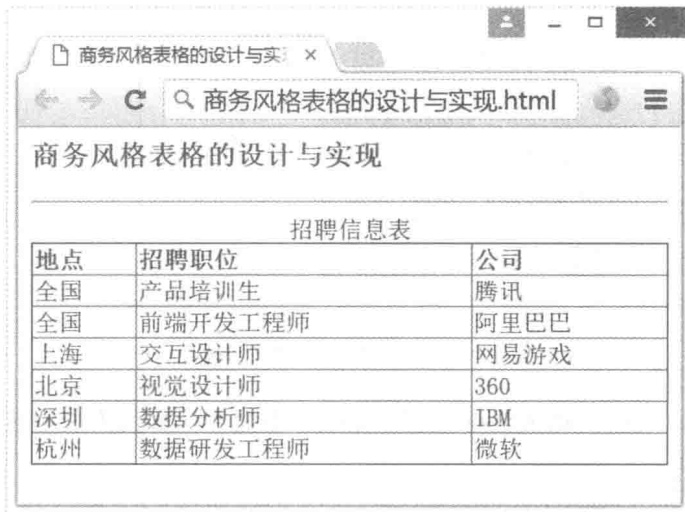


图 1-8 设置表格总体样式的显示效果

2. 单元格样式

为单元格标签<td>和标题单元格标签<th>重新设置样式：边框线条为 1 像素宽的橙色实线，并微调各边的内边距为 7 像素。相关代码如下：

```
<head>
  <meta charset="utf-8">
  <title>商务风格表格的设计与实现</title>
  <style>
    ... (表格的总体样式代码略)
    /*设置单元格样式*/
    #recruit td, #recruit th {
      border: 1px solid orange;
      padding: 7px;
    }
  </style>
</head>
```