



WWW.IT315.ORG

IT 资讯交流网

在线答疑 视频光盘



JavaScript 网页开发

——体验式学习教程

张孝祥 张红梅 编著

```
用户名: <input type="text" name="username">
密码: <input type="password" name="password">
<input type="submit" name="submit" value="提交">
</form>

<script language="javascript">
    function checkName(frm)
    {
        if(frm.user.value == "")
        {
            frm.user.focus();
        }
    }
</script>
```



清华大学出版社

www.it315.org 提供在线答疑和教学视频

JavaScript 网页开发

——体验式学习教程

张孝祥 张红梅 编著

清华大学出版社

北 京

内 容 简 介

本书结合大量应用实例,详细地讲解了 HTML 语言、CSS、JavaScript 语法、DOM 对象模型编程、正则表达式,并介绍了网页脚本编程的其他相关技术和知识,例如,VBScript、NetScape 控件、ActiveX 控件、Java Applet 小程序等。本书力求在不减少知识信息量的情况下,能够把书写薄,同时又能把问题说透,让读者能够迅速上手,并尽最大可能地扩展读者的知识面,启发读者自我思考和学习的能力,让读者感受到技术学习所带来的快乐。

本书主要面向网站开发人员,也适于普通前端网页设计人员阅读。

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

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。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

JavaScript 网页开发——体验式学习教程/张孝祥,张红梅编著. —北京:清华大学出版社,2004.11
ISBN 7-302-09846-8

I. J… II. ① 张… ② 张… III. Java 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 112071 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客户服务:010-62776969

组稿编辑:刘利民

文稿编辑:刘 丽

封面设计:姜凌娜

版式设计:张红英

印 刷 者:清华大学印刷厂

装 订 者:三河市兴旺装订有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:17.75 字数:388 千字

版 次:2004 年 11 月第 1 版 2004 年 11 月第 1 次印刷

书 号:ISBN 7-302-09846-8/TP·6788

印 数:1~5000

定 价:28.00 元(附教学光盘)

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

序

本书作者自去年在清华大学出版社出版《Java 就业培训教程》以来，收到了来自全国各地的读者来信，广大读者对该书给予了很高的评价，认为该书通俗易懂、实践性强，切切实实让读者学到了东西。该书在一年内实现了 4 次印刷，在同类图书市场中位居前列。另外，作者还专门在 www.it315.org 上提供了该书的在线答疑，读者朋友在这里进行了热烈的讨论。这些都让作者倍感振奋，于是，应广大读者的要求，作者历时一年，字斟句酌，几易其稿，又推出新作《JavaScript 网页开发——体验式学习教程》。

《JavaScript 网页开发——体验式学习教程》除了继承了《Java 就业培训教程》的风格以外，还有以下特点：

1. 强调用最短的时间和最浅显易懂的例子说明问题。

作者致力于让一个从零起步的读者能够在最短的时间内，理解编程的基本思路，所以本书语言非常通俗，而且运用了比喻、对比等多种手法，本书的例子也都非常浅显、典型、简洁，目的就是让读者“一看就懂，一学就会”。

2. 启发和培养读者思考问题和自我解决实际问题的能力。

对于编程，学生普遍缺乏的是思路、解决问题的能力以及遇到问题不知道如何下手，而不是缺乏对语法的掌握。但常见的编程类的教材重点放在语法上，学生花大量时间去死记硬背这些繁琐的规则、细节，而对基本的编程思想没有理解掌握，所以很多学生毕业以后，除了记住个别语法以外，对应用编程一无所知。本书力图改变这种状况，着重阐明解决问题的思路，给出解决办法，然后再用语言去实现它，从而真正培养学生用语言工具解决实际问题的能力。

3. 解决初学者学习中可能碰到的大多数问题，让读者感受到技术学习所带来的快乐。

以往教材往往从知识体系结构出发，按部就班，忽视了读者的接受能力，以至于经常出现读不下去、半途而废的情况。本书作者有过多年的培训经验，记录了不少读者经常遇到的问题，这些问题都在本书中有所体现，目的是不留给读者任何学习障碍，让他们学习起来轻松愉快，在学习中感受快乐。

为了帮助读者尽快掌握本书内容，作者专门录制了本书的教学**视频录像**，由于篇幅所限，本书光盘只能载一部分，读者可以登录 www.it315.org 咨询光盘的有关事宜。该网站还提供了本书的**在线答疑**，作者力求在第一时间解答读者的疑问。最后，欢迎读者朋友经常到 www.it315.org 来看看。

作者
2004.11

目 录

第 1 章 HTML 语言基础	1
1.1 HTML 的背景知识	1
1.1.1 什么是 HTML	1
1.1.2 HTML 规范与版本	2
1.1.3 IETF 与 W3C 组织	3
1.2 HTML 的全局架构标签	5
1.2.1 <html></html>	5
1.2.2 <head></head>	6
1.2.3 <title></title>	6
1.2.4 <body></body>	6
1.2.5 使用 Visual Studio .NET 产生网页文档	6
1.2.6 文档类型定义 (DTD)	9
1.3 注释与特殊字符	10
1.3.1 HTML 文档中的注释	10
1.3.2 HTML 文档中的符号和特殊字符	11
1.4 格式标签	13
1.4.1 	13
1.4.2 	13
1.4.3 <nobr></nobr>	13
1.4.4 <blockquote></blockquote>	13
1.4.5 <center></center>	13
1.4.6 <marquee></marquee>	13
1.4.7 <dl></dl><dt></dt><dd></dd>	14
1.4.8 	15
1.4.9 <pre></pre>	16
1.5 文本标签	19
1.5.1 	20
1.5.2 <i></i><u></u>	20
1.5.3 <tt></tt><cite></cite>	21
1.5.4 	21
1.6 超链接标签	22
1.6.1 统一资源定位符 (URL)	23

1.6.2	a href="">.....	27
1.6.3	a name="">.....	28
1.7	图像标签.....	29
1.7.1		29
1.7.2	<hr>.....	30
1.8	客户端图像地图.....	30
1.9	思考与实践.....	32
第 2 章	HTML 高级部分.....	33
2.1	表格标签.....	33
2.1.1	<table></table>.....	34
2.1.2	<tr></tr><td></td>.....	34
2.1.3	<th></th><caption></caption>.....	37
2.2	帧标签.....	38
2.2.1	<frameset></frameset>.....	39
2.2.2	<frame>.....	40
2.2.3	<noframes></noframes>.....	41
2.2.4	<iframe></iframe>.....	43
2.3	表单标签.....	45
2.3.1	<form></form>.....	45
2.3.2	<input type="submit">.....	47
2.3.3	<input type="reset">.....	48
2.3.4	<input type="text">.....	48
2.3.5	<input type="checkbox">.....	48
2.3.6	<input type="radio">.....	49
2.3.7	<input type="hidden">.....	49
2.3.8	<input type="password">.....	49
2.3.9	<input type="button">.....	49
2.3.10	<input type="file">.....	49
2.3.11	<input type="image">.....	50
2.3.12	<select></select><option></option>.....	50
2.3.13	<textarea></textarea>.....	51
2.3.14	<label>标签.....	53
2.4	头元素.....	56
2.4.1	<base>标签.....	56
2.4.2	<link>标签.....	58
2.4.3	<meta>标签.....	60
2.5	分区标签.....	63

2.5.1		63
2.5.2		64
2.6	思考与实践.....	65
第3章	CSS.....	66
3.1	什么是CSS（层叠样式表）	66
3.2	CSS的几种设置方式.....	68
3.2.1	内联样式表.....	68
3.2.2	嵌入样式表.....	68
3.2.3	外部样式表.....	70
3.2.4	输入样式表.....	70
3.3	样式规则的选择器.....	71
3.3.1	HTML selector	71
3.3.2	class selector.....	71
3.3.3	ID selector	73
3.3.4	关联选择器.....	74
3.3.5	组合选择器.....	74
3.3.6	伪元素选择器.....	74
3.4	样式规则的注释与有效范围.....	75
3.4.1	样式表中的注释.....	75
3.4.2	样式规则的继承.....	75
3.4.3	样式规则的优先级.....	75
3.5	用Microsoft Visual Studio .NET产生CSS.....	76
3.5.1	产生内联样式表.....	76
3.5.2	产生嵌入样式表.....	77
3.5.3	产生外部样式表.....	80
3.5.4	连接外部样式表.....	81
3.6	样式属性详解.....	82
3.6.1	字体.....	82
3.6.2	背景.....	84
3.6.3	文本.....	84
3.6.4	位置.....	85
3.6.5	布局.....	86
3.6.6	边缘.....	87
3.6.7	列表.....	88
3.6.8	其他.....	89
3.6.9	CSS滤镜.....	90
3.7	一些建议的学习方法.....	94

3.8 思考与实践.....	97
第4章 JavaScript 语法	99
4.1 JavaScript 是什么	99
4.1.1 什么是脚本程序	99
4.1.2 JavaScript 简介	101
4.1.3 JavaScript 与 Java	101
4.1.4 JavaScript、JScript 与 ECMAScript	101
4.2 JavaScript 的基本语法	102
4.2.1 脚本代码的位置	102
4.2.2 JavaScript 中的标识符	104
4.2.3 JavaScript 中的保留关键字	104
4.2.4 JavaScript 严格区分大小写	105
4.2.5 JavaScript 程序代码的格式	105
4.2.6 JavaScript 程序的注释	106
4.2.7 基本数据类型与常量	106
4.2.8 变量	107
4.3 运算符	108
4.3.1 算术运算符	108
4.3.2 赋值运算符	108
4.3.3 比较运算符	109
4.3.4 逻辑运算符	109
4.3.5 位运算符	109
4.4 程序的流程控制	110
4.4.1 顺序结构	110
4.4.2 if 条件选择语句	111
4.4.3 switch 选择语句	115
4.4.4 while 循环语句	116
4.4.5 do while 语句	117
4.4.6 for 循环语句	118
4.4.7 break 与 continue 语句	119
4.5 函数	120
4.5.1 函数的定义与调用	121
4.5.2 全局变量与局部变量	123
4.5.3 参数个数可变的函数	124
4.5.4 创建动态函数	125
4.5.5 JavaScript 中的系统函数	125
4.6 对象	130

4.6.1 对象与对象实例	130
4.6.2 构造方法与 this 关键字	132
4.6.3 在函数中修改参数值的问题	133
4.7 JavaScript 的内部对象	134
4.7.1 Object 对象	135
4.7.2 String 对象	135
4.7.3 Math 对象	137
4.7.4 Date 对象	138
4.7.5 toString 方法	139
4.8 专门用于对象的语句	140
4.8.1 with 语句	140
4.8.2 for...in 语句	141
4.9 数组	142
4.9.1 数组列表	142
4.9.2 用对象的方式实现数组	144
4.9.3 Array 对象	145
4.10 思考与实践	146
第 5 章 DOM 编程	148
5.1 DHTML	148
5.1.1 DOM 与 DHTML 介绍	148
5.1.2 如何编写事件处理程序	149
5.2 window 对象	152
5.2.1 window 对象的方法	153
5.2.2 window 对象的属性	154
5.2.3 window 对象的事件	157
5.2.4 window 对象的对象属性	158
5.3 document 对象	165
5.3.1 document 对象的方法	166
5.3.2 document 对象的属性	169
5.3.3 document 对象的事件	170
5.3.4 document 对象的对象属性	170
5.3.5 <script>标签的属性	176
5.3.6 Cookie 属性	178
5.4 body 对象	183
5.4.1 body 对象的方法	183
5.4.2 body 对象的属性	184
5.4.3 body 对象的事件	187

5.4.4	body 对象的对象属性	188
5.4.5	在网页上实现图标的漂浮移动效果	190
5.5	form 对象	192
5.5.1	form 对象的方法	192
5.5.2	form 对象的属性	193
5.5.3	form 对象的事件	197
5.5.4	form 对象的对象属性	200
5.6	form 表单字段元素对象	200
5.6.1	form 表单字段元素对象的方法	200
5.6.2	form 表单字段元素对象的属性	201
5.6.3	form 表单字段元素对象的事件	202
5.6.4	表单字段元素对象的几个应用例子	202
5.7	思考与实践	208
第 6 章	脚本编程的相关技术	209
6.1	用 Microsoft Visual Studio .NET 编写 JavaScript 程序	209
6.1.1	查看 Microsoft Visual Studio .NET 文档	209
6.1.2	使用 Microsoft Visual Studio .NET 编写脚本程序	209
6.2	编写兼容 IE 和 NS 的 JavaScript 代码	214
6.2.1	JavaScript 实现 marquee 功能	214
6.2.2	检查浏览器的类别	217
6.2.3	层对象的引用	218
6.2.4	CSS-P 属性的引用	219
6.2.5	层的显示和隐藏	219
6.2.6	层的定位	220
6.2.7	修改和引用层中的内容	221
6.2.8	层的嵌套	221
6.2.9	响应鼠标及键盘事件	222
6.2.10	综合示例	223
6.3	VBScript	225
6.3.1	变量的定义	225
6.3.2	过程和函数	227
6.3.3	事件处理	229
6.3.4	VBScript 应用举例	231
6.3.5	学习和运用 VBScript 编程的方法	234
6.4	插件技术	236
6.4.1	什么是插件	236
6.4.2	Netscape 插件	236

6.4.3	Java Applet.....	237
6.4.4	ActiveX 控件	240
6.4.5	用工具软件添加插件.....	244
6.4.6	Windows 窗体控件.....	246
6.5	建议的学习和应用方法.....	247
6.6	思考与实践.....	247
第 7 章	正则表达式	248
7.1	正则表达式及其作用	248
7.2	RegExp 对象	249
7.2.1	创建 RegExp 对象实例的两种方式	249
7.2.2	RegExp 对象的属性	250
7.2.3	RegExp 对象的方法	252
7.3	String 对象中与正则表达式有关的方法	254
7.4	正则表达式的语法参考	256
7.4.1	限定符	256
7.4.2	选择匹配符	256
7.4.3	分组组合和反向引用符	257
7.4.4	特殊字符	258
7.4.5	字符匹配符	259
7.4.6	定位符	260
7.4.7	原义字符	261
7.4.8	正则表达式语法综合举例	262
7.5	实用程序举例	263
7.5.1	校验 IP 地址是否合法	263
7.5.2	从统一资源定位符 (URL) 中提取各组成元素信息	266
7.5.3	创建能对多种常用格式进行验证的函数	268
7.6	思考与实践	269

第 1 章 HTML 语言基础

1.1 HTML 的背景知识

1.1.1 什么是 HTML

网站的开发离不开网页，而网页的核心技术就是 HTML 语言。尽管在网站开发老手们看来，HTML 语言恐怕是众多 Web 技术中最显得微不足道的部分，但是，HTML 的确是精彩的 Web 世界里必不可少的基石。下面先用一个实验来快速了解什么是网页和 HTML 语言，用记事本程序创建一个 test.txt 文本文件，文件内容如下：

```
<marquee behavior="alternate"><font size=30 color='red'>www.it315.org  
</font></marquee>
```

编写完上述代码后，将它存盘并将文件名改为 test.htm。然后用 IE 浏览器打开这个文件，可以看到在浏览器中显示的效果是：字体大小为 30 个像素，颜色为红色，内容为“www.it315.org”的文本串在不停地水平移动。但放在尖括号对（<>）中的字符序列，如 <marquee>、<font...> 并没有在浏览器中显示出来，它们指定了“www.it315.org”文本串的显示效果。这些尖括号对（<>）与其中的字符序列就是 HTML 标签，一个 HTML 标签必须是由“<”开头，由“>”结尾。HTML 是英文 HyperText Markup Language 的缩写，中文意思是“超文本标签语言”，使用 HTML 语言编写的文件的扩展名是 .html 或 .htm，这就是网页文件。可以使用记事本程序来编写网页文件，也可以使用 FrontPage Editor 等专门的工具软件来编写 HTML 文件。HTML 语言中的标签通常是成对使用的，它使用一个开始标签和一个结束标签来标识文本，结束标签是在标签名称前加一个“/”，也就是以“<标签名>”表示标签的开始，以“</标签名>”表示标签的结束。一对标签中还可以嵌套其他的标签，所以，成对标签又称之为容器。HTML 中也有单独标签，单独标签不需要与之配对的结束标签，它们可以单独使用，又称之为空标签，空标签只能单独使用，不能用于格式化文本（例如，
）。一个 HTML 标签及标签中嵌套的内容形成了网页中的一个元素，很多人喜欢直接用“HTML 元素”来等同“HTML 标签”，这本身并不是一个非常严谨的问题，读者也不必严格区分两者的概念。

许多 HTML 标签都可以设置一个或多个属性来控制标签的显示效果，例如，<marquee> 标签中的 behavior、 标签中的 size 和 color 就是 HTML 标签属性。属性设置的一般格式为：属性名=属性值，属性值部分可以用英文的双引号（" "）或单引号（' '）引起来，

也可以不使用任何引号。对于有些只有两种状态（有或无）的属性不用设置属性值，写上该属性名表示启用该属性；反之，不写则表示不使用该属性。每个属性的位置必须跟在起始标签名的后面，且位于尖括号之内。标签名与属性之间，属性与属性之间必须用空格分隔。有些属性是公共的，这些属性的名称和作用在每个 HTML 标签中都完全相同，有些属性是某个 HTML 标签专用的。HTML 标签、属性名与属性值都是大小写不敏感的，即、、以及的效果是一样的，但网页文档编写者应该养成大小写统一的习惯，不要随心所欲、忽大忽小。

浏览器打开网页文件的过程与用记事本程序打开文本文件的过程是一样的，只是浏览器会对网页文件中的内容用特殊的方式显示。浏览器除了从本地硬盘上打开网页文件外，还可以使用HTTP网络协议从网络上的WWW服务器（也叫Web服务器）上获取网页文件的内容，如图1.1所示。

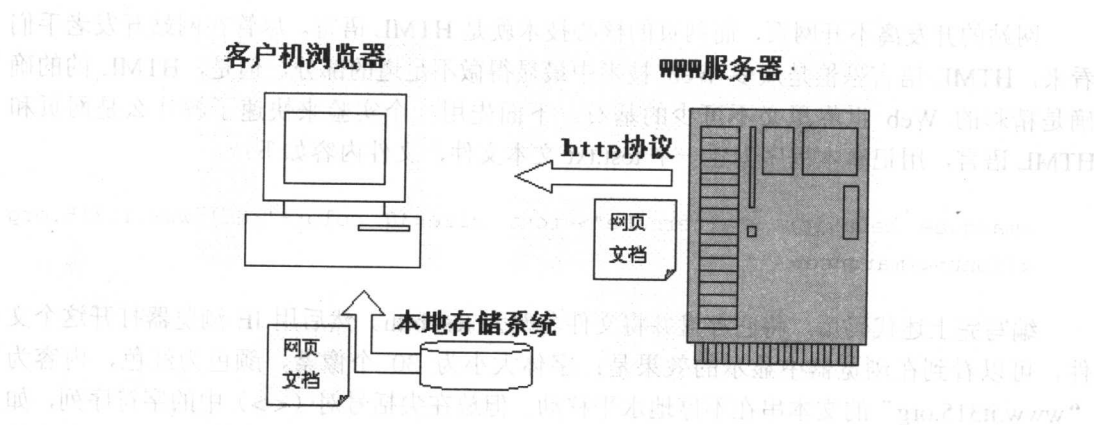


图 1.1

网页文件就是一个普通的文本文件，这个文本文件里的一些特殊字符序列被当作一种 HTML 标记，当浏览器打开网页文件时，不是像记事本程序那样简单地显示文本文件里的内容，而是根据其中的标记来控制文件内容的显示效果和执行某种功能。单击 IE 浏览器上的“查看”→“源文件”菜单，可以看到 test.htm 中的原始文本内容，可见，浏览器的基本功能就是根据 HTML 标签的含义，用特殊的效果去显示和控制一对 HTML 标签之间所引用的文本内容，HTML 标签的作用就是告诉浏览器应该如何显示有关的文本。有各种各样的 HTML 标签，它们定义了网页中文字的大小、颜色、效果，段落的排版方式，以及用户如何通过一个网页导航到另外的网页等各方面的内容，这些 HTML 标签的组合就是 HTML 语言。HTML 不是程序设计语言，而是一种标记语言，也就是用一些标记来说明文本的显示效果。要建立网站和制作网页，就必须对 HTML 语言有所了解。

1.1.2 HTML 规范与版本

目前使用得较多的浏览器软件是 Microsoft（微软公司）的 Internet Explorer 和 Netscape（网景公司）的 Navigator，读者也许听说过，这两种浏览器或者同一种浏览器的不同版本

之间存在着不兼容问题，这是怎么回事呢？我们需要从计算机软件开发和运行的原理上来解释和说明这个问题，浏览器是人们开发的应用软件，有多个公司都开发了这种软件，它根据网页文件中的 HTML 标签，来决定在它的窗口中绘制（对用户来说就是显示）什么样的信息，执行什么样的动作。有些浏览器软件的开发人员为了实现一些特殊的效果，让该浏览器可以接受网页中引入的一些特殊标签和标签属性，然后对这些特殊的标签和标签属性作一些特殊的处理，这些特殊的标签和标签属性就成了这个浏览器的“方言”。而另外的浏览器软件并不知道这些特殊的标签和标签属性是什么含义，也就是它并不知道有这些“方言”，所以它就不能对这些特殊的标签和标签属性进行处理，这就导致不同的浏览器打开一个使用了特殊标签和标签属性的网页文件时，会有不兼容的问题。

为了解决兼容性和互用性问题，需要一些组织和机构来制定 HTML 规范和标准。这些组织和机构根据当时的需求和应用情况，定义了许多 HTML 标签，这些标签就是某一版本的 HTML 规范。但随着情况的变化，时间的迁移，又出现了更多新的需求和应用，例如，我们以后可能要在网页中嵌入一段电影片断，而不再仅仅是一张图片，这就需要定义一个告诉浏览器播放电影片断的标签。标准化过程必须随着新技术的发展和应用而不断发展。因此，标准化组织必须不断地在以前的 HTML 规范的基础上，定义一些新的标签和新的内容，这就形成了新版本的 HTML 规范。

早期 HTML 是非常简单的，被称之为 HTML 1.0，后来由 IETF (Internet Engineering Task Force, Internet 工程任务组) 进一步扩展，并制定出对常用的 HTML 标签进行了详细说明的 HTML 规范，这就是 HTML 2.0。IETF 最终将负责 HTML 规范制定的权力移交给了一个比它后成立的、专门制定 Web 领域技术规范的组织，这就是 W3C (World Wide Web Consortium, 习惯称之为 WWW 联盟)，因为主要的厂商，如 Microsoft 和 Netscape 公司倾向于通过 W3C 工作。

本书编写时的最新 HTML 规范是 HTML 4.01，这也是 HTML 规范的最终版本，不可能再有更新的 HTML 规范了，HTML 将被 XHTML 所取代。虽然这么说，人们在相当长的一段时间内，还得使用 HTML，并且 HTML 是 XHTML 的基础，因此，学习 HTML 还是非常有必要的。

1.1.3 IETF 与 W3C 组织

Internet 的最大特点是管理上的开放性，它被每个用户所共同拥有，没有人和组织对 Internet 拥有实际的绝对控制权。Internet 没有集中的管理机构，但是为了促进 Internet 运行所需的标准兼容性，并确保 Internet 的持续发展，先后成立了一些机构和组织，它们自愿承担 Internet 的管理职责。

了解这些 Internet 组织及它们所制定的标准化文档，对于一个要掌握 Internet 网络应用细节、特别是要编写 Internet 网络应用程序的人来说，是很有必要的。目前主要由两个组织负责制定 Web 网站管理和开发相关的规范，这两个组织是 IETF 和 W3C，它们的主要职责是制定 Internet 网络连接和应用的协议标准，下面分别对这两个组织和它们所制定的相关文档进行介绍。

IETF 是由网络设计人员、操作员、厂商、专家组成的民间组织，主要负责有关 Internet 的各种技术标准及接口规范的制定，其网址为<http://www.ietf.org>。参加 IETF 会议的人员都是个人代表，他们不代表任何组织、公司、学校、政府部门等。IETF 主要负责 8 个功能领域的规范和标准的制定，它们分别是：应用、Internet、网络管理、运行要求、路由、安全、传输与用户服务，每个领域都设有多个工作小组来开展相关工作。IETF 以 RFC (Requests for Comments, 请求注解文档) 定名所发布的各类标准与协议，RFC 实际上就是 Internet 有关服务的一些技术标准文档，是用于发布 Internet 标准和 Internet 其他正式出版物的一种网络文件或工作报告。RFC 的名字来源是历史原因造成的，现在看来，它的名字和实际上的内容并不一致。RFC 文档虽然是民间机构而不是官方制定的，但大多 RFC 都已成为业界的事实标准。

RFC 产生的过程是一种从下往上的过程，而不是从上往下的过程。它不是一个由主席，或者由工作组负责人下令做出来的，而是由下面的任何人自发地提出，然后在工作组里进行讨论，讨论了以后再交给有关组织进行审查通过后形成的。任何一个用户都可以对 Internet 某一领域的问题提出自己的解决方案或规范，作为 Internet 草案 (Internet Drafts, ID) 提交给 Internet 工程任务组 (IETF)，草案存放在美国、欧洲和亚太地区的工作文件站点上，供来自世界上多个国家的、自愿参加的 IETF 成员进行讨论、测试和审查。如果一个 Internet 草案被 IESG 确定为 Internet 的正式工作文件，则被提交给 Internet 体系架构委员会 (IAB)，并形成具有顺序编号的 RFC 文档，由 Internet 协会 (ISOC) 通过 Internet 向全世界颁布。TCP/IP 协议的一系列标准都是通过这种方式以 RFC 文档格式公布的。RFC 文档必须被分配 RFC 编号后才能在网络上发布，例如，RFC2616 是 HTTP/1.1 协议规范的文档，RFC1521 是 MIME 格式规范的文档。最初的 RFC 一直保留而从来不会被更新，如果修改了该文档，则该文档必须以一个新号码公布，用户可以通过遍布全世界的数个联机站点获得 RFC 文档。

W3C 于 1994 年成立，是与 Web 有关的企业机构成立的业界同盟，该组织是国际性的，在世界各地的许多研究机构中都设有办事处，其网址为<http://www.w3c.org>。W3C 目前的成员仅限于团体或组织，只要交纳一定的费用，并签署一份保证遵守规则的成员协议，任何公司均可加入。W3C 对 Web 的标准握有生杀大权，负责研究、审定、发布、管理有关 Web 的标准，例如，HTML、CSS 等。该组织致力于开发促进 Web 发展和确保其互操作性的基础性协议，引导进一步发掘 Web 的潜能，它还开发体现和推动标准的参考代码，以及各类展示新技术应用的源程序范例。W3C 不从事网络传输协议规范的制定，它将重点放在人们从 Web 上所看到的東西，例如，字体、图形和 3D 动画等。实际上，W3C 不具备强制执行能力，它的标准仅是建议，不具备任何法律效力，人们不必非得遵照执行。但是，如果电源插座厂家不按业界公认的标准来生产，其他电器产品就无法插接到这个厂家生产的插座上，这样的插座肯定卖不出去。显然，一个厂商不按公认的标准来制作相关产品，是没有出路的，所以，相关厂商都非常愿意与 W3C 合作，大多数加入 W3C 的成员都是为了在决定协议的未來内容时发表自己的意见，以便在标准制定过程中处于有利地位。IT 领域内的一些大公司，例如，Hewlett Packard、Netscape、Sun Microsystems、Microsoft 等都是 W3C 的成员。

当 W3C 工作组对即将准备制定的某个规范的初期成果感到相当满意时，他们就会在 W3C 的 Web 站点上以一份工作草案的形式发表供公众查阅。在对最初的反应进行评估之后，该工作组就将此草案作为所提出的建议发表在 Web 站点上，W3C 咨询委员会有一个月的时间投票决定它是否应成为一项实际建议。

其实，读者从众多书籍上看到的各种 Internet 网络应用的规范，都是来源于 IETF 和 W3C 所发布的文档，本书中的很多内容也是如此。因为计算机技术的发展日新月异，人们完全可能在 6 个月的时间内以公布、开发、运用、批评以及放弃一种技术，对于一些具有时效性的标准规范，读者不能完全相信参考书，应该查看这两个组织所提供的最新相关文档。读者可以直接到 W3C 和 IETF 的网站上查找相关资料，但笔者最不爱记忆一些具体的事务，而喜欢记住一些解决问题的途径。例如，如果要查找 HTML 规范的最新资料，可去 www.google.com 网站上搜索“html w3c”关键字。同样，如果要了解 HTTP 协议的详细信息，可去 www.google.com 网站上去搜索“http rfc”关键字，很快就可以知道所要查看的文档资料在 Internet 上的位置，而且不用担心万一位置有变动的情况。

1.2 HTML 的全局架构标签

一个网页文件中的标签有一定的组成结构，不能随意颠倒和错乱这种关系，下面这段内容说明了一个最基本的网页文件的组成结构。

```
<html>

<head>
<title>显示在浏览器左上方的标题</title>
</head>

<body bgcolor="red" text="blue">
<p>红色背景、蓝色文本</p>
</body>

</html>
```

粗略阅读一下上面这段内容，将它们保存在一个.html 或.htm 文件中，然后用浏览器打开保存的文件并观察显示效果，结合下面的讲解，读者就很容易了解其中各个标签对在一个 HTML 文档组成结构中的位置及其自身的作用。

1.2.1 <html></html>

<html>标签用于HTML文档的最前边，用来标识HTML文档的开始。而</html>标签恰恰相反，它放在HTML文档的最后边，用来标识HTML文档的结束，两个标签必须成对使用，网页中所有其他的内容都要放在<html>和</html>之间。

1.2.2 <head></head>

一个网页文档从总体上可分为头和主体两部分。`<head>`和`</head>`定义了 HTML 文档的头部分，必须是结束标签与起始标签成对使用。在此标签对之间可以使用`<title></title>`、`<script></script>`等标签对，这些标签对都是描述 HTML 文档相关信息的标签对，`<head></head>`标签对之间的内容是不会在浏览器的文档窗口中显示出来的。

1.2.3 <title></title>

使用过浏览器的人可能都会注意到浏览器窗口的标题栏上显示的文本信息，那些信息一般是网页的“主题”，要将网页的主题显示到浏览器的顶部其实很简单，只要在`<title></title>`标签对之间加入主题文本即可。

注意：`<title></title>`标签对只能放在`<head></head>`标签对之间。

1.2.4 <body></body>

`<body></body>`定义了 HTML 文档的主体部分，必须是结束标签与起始标签成对使用。在`<body>`和`</body>`之间放置的是实际要显示的文本内容和其他用于控制文本显示方式的标签，如`<p>`、`</p>`、`<h1>`、`</h1>`、`<br>`、`<hr>`等，它们中间所定义的文本、图像等将会在浏览器的窗口内显示出来。对于`<body>`标签，有以下一些主要属性：

- ❖ `text` 用于设定整个网页中的文字颜色，关于颜色的取值，在稍后部分会有详细讲解。
- ❖ `link` 用于设定一般超链接文本的显示颜色。
- ❖ `alink` 用于设定鼠标移动到超链接上并按下鼠标时，超链接文本的显示颜色。
- ❖ `vlink` 用于设定访问过的超链接文本的显示颜色。
- ❖ `background` 用于设定背景墙纸所用的图像文件，可以是 GIF 或 JPEG 文件的绝对或相对路径。
- ❖ `bgcolor` 用于设定背景颜色，当已设定背景墙纸时，这个属性会失去作用，除非墙纸具有透明部分。
- ❖ `leftmargin` 设定网页显示画面与浏览器窗口左边沿的间隙，单位为像素。
- ❖ `topmargin` 设定网页显示画面与浏览器窗口上边沿的间隙，单位为像素。

`<body>`标签还有一些其他的公共属性，如 `class`、`name`、`id`、`style` 等，对这些属性的泛泛而谈必然过于空洞，读者也无法完全一下理解其确切的含义和作用。在后面章节的相关部分，结合一些具体的实际应用进行讲解，读者就会更容易理解。

1.2.5 使用 Visual Studio .NET 产生网页文档

如果使用专门的 HTML 编辑软件来编写 HTML 文件，上面这些基本的 HTML 标签都