



软件系统开发指导教程系列丛书

计算机编程导论

主编 朱志良 李丹程
编者 毛克明 王 爽 王蓓蕾

TEXTBOOK FOR HIGHER EDUCATION



西北工业大学出版社

NORTHWESTERN POLYTECHNICAL UNIVERSITY PRESS

软件系统开发指导教程系列丛书

计算机编程导论

主编 朱志良 李丹程

编者 毛克明 王 爽 任蓓蕾

西北工业大学出版社

【内容简介】 本书为软件工程专业软件系统开发指导教程系列丛书之一。全书共分4章。其中,第一章介绍了HTML语言的基本原理以及如何用HTML语言编写网页的方法;第二章介绍了面向对象的基本概念和原理,以及Java编程语言的基本用法;第三章介绍了面向对象的各種特点及其应用的环境;第四章简要介绍了Servlet的基础知识,包括Servlet的基本原理、Servlet结构及Servlet的基本用法。

本书可作为高等院校计算机系统导论课程教材,也可作为计算机相关专业以及有志了解计算机知识的各类人员的参考书。

图书在版编目(CIP)数据

计算机编程导论/朱志良,李丹程主编. —西安:西北工业大学出版社,2010.12
(软件系统开发指导教程系列丛书)
ISBN 978-7-5612-2962-0

I. ①计… II. ①朱… ②李… III. ①程序设计—教材 IV. ①TP311

中国版本图书馆CIP数据核字(2010)第232787号

出版发行:西北工业大学出版社

通信地址:西安市友谊西路127号 邮编:710072

电 话:(029)88493844 88491757

网 址:www.nwpu.com

印 刷 者:陕西丰源印务有限公司印刷

开 本:787 mm×960 mm 1/16

印 张:12.5

字 数:275千字

版 次:2010年12月第1版 2010年12月第1次印刷

定 价:23.00元

《软件系统开发指导教程系列丛书》编委会

- 主任：沈绪榜 中国科学院院士
计算机专家
- 委员：朱怡安 教授、博导
西北工业大学软件学院 常务副院长
- 朱志良 教授、博导
东北大学软件学院 院长
- 陈珉 教授、博导
武汉大学国际软件学院 常务副院长
- 李耀国 教授、博导
南开大学软件学院软件工程硕士中心 主任
- 林亚平 教授、博导
湖南大学软件学院 院长
- 张红延 北京交通大学软件学院 院长助理
中国软件行业协会系统与过程领域专家委员
- 洪玫 教授
四川大学软件学院 副院长
- 黄虎杰 教授
哈尔滨工业大学软件学院 常务副院长
- 朝红阳 教授、博导
中山大学软件学院 常务副院长

出版说明

2001年12月，教育部批准成立了35所国家示范性软件学院，旨在为国家软件产业的发展培养多层次、实用型、高水平、具有国际竞争力的专业人才，以适应社会对软件高端人才的需要。各软件学院在这样的大环境下，纷纷挖掘自身的优势，采用各种先进的教学模式，注重教育与教学改革，已形成了各具特色的软件工程教育体系。广大教师也在这样的教育教学中不断对传统软件工程教学进行总结，并汲取国外先进教学中的精髓，取长补短，积累了丰富的教学实践经验。

有鉴于此，我们组织策划了《软件系统开发指导教程系列丛书》。本系列丛书共10册，包括《计算机编程导论》《计算机系统导论》《面向对象编程基础——Java语言描述》《交互式用户界面设计与测试》《数据库系统——设计与应用》《数据结构与算法》《系统级编程》《网络与分布式计算》《软件规范测试与维护》《软件项目组织与管理》。本系列丛书的出版意欲将广大教师在培养国际化、应用型软件工程化人才的教育教学中积累的经验进行推广与传播。特别是将这种教学理念在一些外语基础薄弱，还不能适应双语教学的学生中推广。

本系列丛书的分册主编均是各软件学院活跃在教学第一线的教师。他们都具有多年的教学经验、深厚的专业功底和丰富的软件开发实践经验，因而保证了这套丛书理论与实践兼备，教与学互动，特色鲜明。

为确保本系列丛书的质量，我们邀请了软件学院的一些专家、教授，成立了《软件系统开发指导教程系列丛书》的编委会。他们当中有全国知名的计算机专家、科学院院士，也有在软件工程教育中有着丰富工作经验的教授、博导，也不乏具有出国留学经历的教师，他们对国际与国内该领域的技术状况、应用环境都十分了解。这些均有利于从总体上把握本系列丛书内容向着适应国内需要并与国际接轨的方向发展。

我们将以高度的社会责任感，投入满腔的工作热忱，精益求精，为广大读者提供高质量的精品图书。

西北工业大学出版社

2009年3月

前 言

目前,许多高校的计算机专业逐步开始使用国外原版教材进行双语教学,甚至引进了国外软件系统开发系列课程。我们所在高校自 2004 年也开展了该项工作。几年来,我们一直负责计算机系统导论课程的教学,该课程全面介绍计算机相关知识,为后续计算机专业基础课和专业课做引导。但在实际教学过程中,教师和学生普遍感到中国的教育模式和国外的教材不能很好地结合,同时,由于原版教材是全英文,容易造成课程教学质量两极分化现象。为了改变上述不足,我们编写了本书,目的是适应国内实际情况,吸收国外教材内容的精华,为广大读者更好地学习、掌握。

本书是我们在多年从事计算机教学实践,并参考国内外多种教材的基础上编写而成的。其中,去掉了一些不符合国内实际情况的信息,增加了一些适合计算机基础教学的内容。在内容编排上力求与目前计算机科学和技术的发展相结合;对书中的章节、实例、图表都通过精选,按照步步启发的模式设计安排。

本书涵盖了计算机系统导论课程的所有知识结构,其内容新颖、概念清晰、通俗易懂,既有利于教师在教学过程中发挥,也有利于学生在学习中拓展。

为了便于读者使用,配合本书制作有电子课件,需要的读者可登录西北工业大学出版社网站下载。

本书由朱志良、李丹程任主编。全书共分 4 章,第一、四章由王爽编写,第二、三章由毛克明、王蓓蕾编写。

由于水平有限,书中难免存在不当和疏漏之处,恳请读者批评指正。

编 者
2010 年 6 月

目 录

第一章 HTML 语言	1
1.1 绪论	1
1.2 HTML 的基本概念	5
1.3 文本处理	13
1.4 超级链接	27
1.5 插入图片	33
1.6 列表	36
1.7 表格	42
1.8 表单	46
1.9 多框架文档	53
第二章 面向对象编程基础	59
2.1 Java 编程及环境配置	59
2.2 面向对象的基本概念	65
2.3 Java 编程基础	91
第三章 面向对象特点	124
3.1 类的继承	124
3.2 类的封装	142
3.3 对象的多态性	148
3.4 包	157
3.5 类设计中需要注意的问题	163
3.6 Java 开发中常见的异常问题	164
第四章 初识 Servlet	166
4.1 Servlet 概述	166
4.2 Servlet 基础知识	169
4.3 Servlet 程序编写基础	172
4.4 客户请求的处理	178
参考文献	192

第一章 HTML 语言

要让设计者在网络上发布的网页能够被世界各地的浏览者所阅读,需要一种规范化的语言。在万维网(WWW)上,文档的发布语言是 HTML。HTML 语言是 HyperText Markup Language 的缩写,中文名为超文本标记语言。正是 HTML 这种规范化的发布语言使得网络上的网页能够被世界各地的浏览者所阅读。

HTML 是搭建网页的基础语言,学习 HTML 不见得是为了要从头到尾编写网页,主要是为了了解“如何做”和“怎样做好”。如果不了解 HTML,在制作网页之前,就不知道网页的代码是如何工作的;在制作过程中,就不能灵活地实现想要的网页效果。

1.1 绪 论

所谓超文本文件,就是 HTML 文档有别于纯文本的单个文件的浏览形式,超文本文档中提供的超级链接能够让浏览者在不同的页面之间跳转。

标记语言是一种基于源代码解释的访问方式,它的源文件由一个纯文本文件组成,代码由许多元素组成,而前台浏览器通过解释这些元素显示各种样式的文档。也就是说,浏览器就是把纯文本文件的后台源文件以赋有样式定义的超文本文件方式显示出来。

1.1.1 万维网

Web 技术是随着 Internet 使用的普及和发展而兴起的一门新兴的技术,它的发展速度之快和应用范围之广超出了人们的预料。而 Web 发明的同时又进一步促进了 Internet 使用的普及和发展。因此,了解和掌握 Web 技术已成为网络应用程序开发人员的当务之急。

1. Internet 和 Web

简单地说,Internet(因特网,也称国际互联网)就是由 ARPANET 演变而成的、采用 TCP/IP 协议的、由网络互连而成的全球范围的网络。在 WWW(World Wide Web 的简称)发明之前,Internet 提供的服务主要为电子邮件 E-mail、文件传输 FTP(File Transfer Protocol)、电子公告 BBS(Bulletin Board System)以及远程登录 Telnet(Telecommunication Network)等。借助于这些服务,人们可以在 Internet 上相互通信、传送文件、发布信息和查找资料。但由于使用的手段过于专业,因此,Internet 的普及使用受到了极大的限制。

Web 是在 Internet 上的一种新的资源组织形式,也是一种新的信息服务方式。Web 服务

系统通过超文本链接功能和 HTTP 协议,将文本、图像、声音和其他 Internet 上的资源紧密地结合起来,人们只要在 Web 浏览器的地址栏中输入 Web 资源的 URL 地址,即可以 Web 页面的形式显示在 Web 浏览器上。组成 Web 的超文本文件通过超链接相互连接,只要在链接点上单击一下,就可从当前页面跳转到某个其他相关页面。Web 资源的这种方便的访问方式大大加速了 Internet 的普及使用。

2. Web 的架构

Web 既是一种资源的组织形式,也是一种应用开发的平台或架构。它由 Web 服务器和 Web 浏览器两部分组成。

(1)Web 服务器。Web 服务器是构成 Web 的必不可少的服务器端软件。Web 服务器接收 Web 浏览器传送来的 Web 资源的 URL 地址,并将相应的文本或信息提供给客户端的 Web 浏览器。常用的 Web 服务器有: Internet Information Server, Website 和 Tomcat 等。

(2)Web 浏览器。Web 浏览器是构成 Web 的客户机端软件,主要用于在用户机器上处理和显示超文本文件(即 Web 页面)。最常用的两种 Web 浏览器软件是微软公司的 Internet Explorer 和网景公司的 Netscape Navigator。

3. Web 的技术基础

HTML 语言和 HTTP 协议是构成 Web 的主要技术基础。

(1)HTML。超文本标记语言 HTML 是一种用于编制超文本文件的文档格式化语言。它对文件显示的具体格式进行了详细的规定和描述,当浏览器读取某 HTML 文件时,就会显示出相应的页面。HTML 语言具有平台无关性,无论用户使用何种操作系统,只要拥有相应的 Web 浏览器,就可以解释运行 HTML 文件。

(2)HTTP。超文本传输协议 HTTP(Hypertext Transport Protocol)是 Web 服务器与 Web 浏览器之间进行超文本传输的应用层通信协议。它建立在 TCP/IP 协议的基础上,规定了 Web 浏览器在运行 HTML 文件时所遵循的规则和要进行的操作。

4. 网络计算模式的演变和 B/S 模式

网络环境下的应用程序的结构模式大致可分为:服务器中心模式、客户机/服务器模式和浏览器/服务器模式这三种类型。

(1)服务器中心(Server-Centric)模式。服务器中心模式是早期网络环境下应用程序的主要模式。这时的服务器就好比一个集中管理的仓库,共享的程序和数据均放在服务器上,使用时须通过网络将程序和数据传送至用户的工作站上,程序的运行和数据的处理都在工作站上进行。

这种模式的两个显而易见的缺点是:网络上传输的是整个应用程序和数据文件本身,通信量较大;整个应用程序和数据文件都在作为工作站的客户机上运行和处理,对工作站的配置要求较高。这在应用程序和数据文件都不太大的早期是可行的,但随着计算机应用的发展,这种模式显然是不适宜的。

(2)客户机/服务器(Client/Server)模式。客户机/服务器(C/S)模式把整个应用程序分为

两部分:客户机端的部分和服务端的部分。客户机端软件安装在用户端的客户机上,而服务器端软件则安装在服务器上。两部分软件协同工作:客户机端软件主要是用户界面程序,一方面用于收集用户的请求,并把这种请求传送给服务器端软件处理;另一方面则用于将服务器端送回的结果显示给客户。服务器端软件主要在服务器上完成来自客户端的请求,并把结果送回至客户端。

这种模式在网络上传输的只是请求(Requests)和结果(Results),大大减轻了网络的通信负担;另外,由于在客户机上运行的只是较简单的界面程序,也降低了对客户机的配置要求。这种客户机/服务器模式已成为网络环境下应用程序的主流模式。

(3)浏览器/服务器(Browser/Server)模式。Web 应用程序采用的是浏览器/服务器(B/S)模式,这是一种特殊的 C/S 模式。其客户机部分采用的是 Web 浏览器,克服了 C/S 模式的客户机端界面程序不一致的缺点,使得用户所在的客户机只要拥有 Web 浏览器即可方便地访问任何 Web 资源。另外,Web 应用程序更新时只须更新服务器端的资源,免去了 C/S 模式应用程序往往须同时更新客户机端界面程序的麻烦,从而大大减轻了应用程序更新的负担。

5. HTML 相关术语

(1)统一资源标识符(URI)。访问和操纵通过 Web 发布的资源时,需要一种方法来标识它们,Web 资源是由 URI 标识的。URI 不是代表什么物理的东西,它可以看做是前往一个黑盒的指针。

最流行的一种 URI 形式是统一资源定位地址(URL)。在本书中,我们将使用流行的术语 URL,而不是更一般的 URI。

URL 的一般格式如下:

协议名://主机名[:端口号]/路径名/文件名

上述 URL 各组成部分中,协议名常使用 http;主机名可以是 Web 服务器的域名或 IP 地址;端口号指的是 Web 服务器运行的端口号,为 80 时可以省略。

(2)超文本。超文本(Hypertext)是指具有超链接功能的文本。与传统线性浏览的文本不同,超文本中可包含有链接到其他文件的文字或图像,通过点击这些文字或图像,可以从一个浏览空间跳转至另一个浏览空间。

(3)超文本链接。超文本链接(Hypertext Link)是指超文本中用以链接到其他文件的文字或图像,简称超链接。超文本中的任何文字或图像都可以被指定为超链接。

1.1.2 HTML 语言的发展

HTML 语言最初是由 Tim Berners-Lee 在 CERN(欧洲粒子物理实验室,European Laboratory for Particle Physics)期间开发,并且得益于 NCSA(国家超级计算应用中心,National Center for Supercomputing Applications)的 Mosaic 浏览器的流行和推广。20 世纪 90 年代所谓“网络爆炸”年代令 HTML 语言一跃成为制作网页的标准语言。

HTML 版本经历了 2.0 和 3.0 之后,1997 年推出的 HTML 3.2 已经极大丰富了 HTML

的功能。1997 年 12 月的 HTML 4.0 将 HTML 语言推向一个新高度。由于同期 CSS 层叠样式的配套推出,更使得 HTML 和 CSS 对于网页制作的能力达到前所未有的高度。

HTML 4.0 发布后,W3C 网络标准化组织于 1999 年 12 月推出改进版 HTML 4.01,对 HTML 4.0 的一些功能作了进一步完善,之后变化不大,可见该语言非常成熟可靠。

1.1.3 编辑、保存和查看 HTML 文档

编辑 HTML 文档最简单最常用的方法就是采用 Windows 系统自带的记事本。只要在 Windows 的“开始”菜单中,选择“程序”,选择“附件”,然后选择“记事本”命令,即可编辑 HTML 文档。

保存 HTML 文档非常简单,编辑完成后,只要在记事本菜单中选择“文件”,然后选择“保存”或者“另存为”即可。但是要保存成“.html”或“.htm”格式。

查看 HTML 文档也非常简单,在 Windows 中打开 Internet Explorer 浏览器,然后用浏览器打开即可。简单示例如下。

如图 1.1 所示,首先在记事本中输入代码,然后存为“html 示例 1.html”文件。值得注意的是,在记事本中输入代码的时候,所有<>符号和<>符号内的字母都必须使用英文半角字符,其他区域内的文本没有此限制。

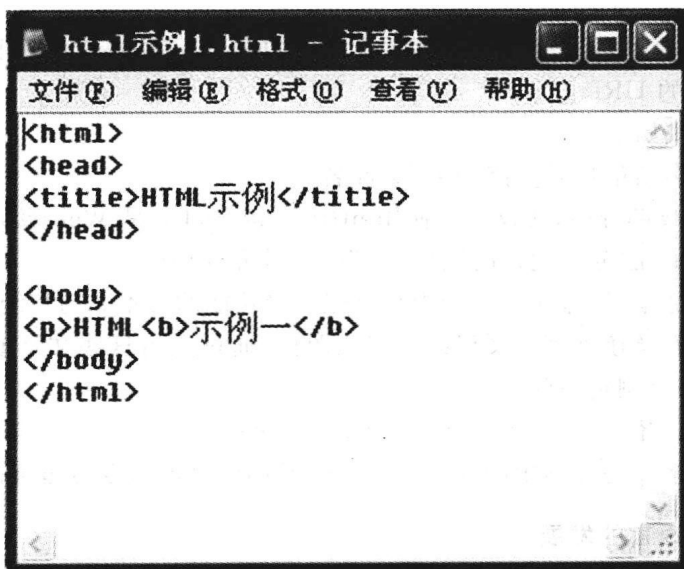


图 1.1 在记事本中输入代码

用浏览器打开“html 示例 1.html”文件,其效果如图 1.2 所示。

用记事本作为 HTML 文档的编辑器,最大的优点就是无须安装其他应用程序,每个版本的 Windows 都带有记事本。除了记事本还有很多的 HTML 编辑工具,其中常见的有 Dream-

weaver, FrontPage, Edit Plus 和 Ultra Edit, 等等。它们各有长处, 前两种称为“所见即所得”工具, 简单易学, 但削弱了设计者对源代码的控制和学习能力。对于学习者而言, 最好不使用 Dreamweaver, FrontPage 等“所见即所得”工具。



图 1.2 用浏览器打开 HTML 文档

目前国内浏览器基本被微软公司的 IE 所垄断, 但是在全球范围内, Netscape, Opera 等浏览器仍然有很大市场。因此, 设计代码时也要考虑浏览器的兼容问题。本书默认的查看浏览器为 IE。

1.2 HTML 的基本概念

HTML 语言不同于 C 语言、Java 语言等编程语言, 而是一种标记语言, 其作用是规定网页以何种形式展现给使用者。它也不同于一般的文本, 一个 HTML 文件不仅包含文本内容, 还包含一些 Tag, 中文称“标记”, 用于控制文件的显示。一个 HTML 文件的后缀名是 .htm 或者是 .html。

1.2.1 HTML 文档的基本结构

一个完整的 HTML 文档必须包括 3 个部分: 一个 `<html>` 元素、一个 `<head>` 元素定义各项声明的文档头部和一个 `<body>` 元素定义文档的主体部分。其中, 后两个部分应该放在 `<html>` 元素中。

HTML 文档的基本结构如下:

`<html>` 为顶级元素, 所有 HTML 页面的内容都包含在这个元素中, `<head>` 作为各种声明信息的包含元素, 出现在文档的顶端, 要先于 `<body>` 出现。而 `<body>` 分别用来显示文档主体内容。

```

<html>..... HTML 文档标记
<head>.....文档首部标记
    <title>主页标题</title>.....标题标记
</head>
<body>.....文档内容标记
    主页文档主体部分
</body>
</html>
    
```

1. 根元素<html>

由 HTML 文档的基本结构可以看出,<html>标记用于 HTML 文档的最前边,用来标识 HTML 文档的开始。而</html>标记恰恰相反,它放在 HTML 文档的最后边,用来标识 HTML 文档的结束,两个标记必须成对使用。

2. 文档头部信息<head>

<head>和</head>构成 HTML 文档的开头部分。在此标记对之间可以使用<title></title>,<meta></meta>等标记对,这些标记对都是描述 HTML 文档相关信息的标记对。<head></head>标记对之间的内容是不会在浏览器的框内显示出来的,但是这并不表示这些信息没有用处,比如你可以在 head 信息里加上一些关键词,有助于搜索引擎搜索到你的网页。两个标记必须成对使用。

3. 文档主体<body>

<body></body>标记对之间是 HTML 文档的主体部分。在此标记对之间可包含<p>,
,<hr>等众多的标记,它们所定义的文本、图像等将会在浏览器的框内显示出来。两个标记必须成对使用。<body>标记中还可以有众多属性,如:

<body bgcolor="#rrggbb">:设置背景颜色。如:<body bgcolor="red">:红色背景。

<body text="#rrggbb">:设置文本颜色。如:<body text="#0000ff">:蓝色文本。

<body link="#rrggbb">:设置链接颜色。如:<body link="blue">:链接为蓝色。

说明:以上各个属性可以结合使用,如<body bgcolor="red" text="#0000ff">。引号内的 rrggbb 是用 6 个十六进制数表示的 RGB(即红、绿、蓝三色的组合)颜色,如 #ff0000 对应的是红色。此外,还可以使用 HTML 语言所给定的常量名来表示颜色:Black,White,Green,Maroon,Olive,Navy,Purple,Gray,Yellow,Lime,Agua,Fuchsia,Silver,Red,Blue 和 Teal,如 <body text="Blue">表示<body></body>标记对中的文本使用蓝色显示在浏览器的框内。

4. 标题<title>

使用过浏览器的人可能都会注意到浏览器窗口最上边蓝色部分显示的文本信息,那些信息一般是网页的“主题”。要将您的网页的主题显示到浏览器的顶部其实很简单,只要在

`<title></title>` 标记对之间加入您要显示的文本即可。注意：`<title></title>` 标记必须成对出现，并且只能放在 `<head></head>` 标记对之间。

5. 元信息定义 `<meta>`

元信息用来定义文档本身的信息而非文档内容信息，它可以定义很多信息，如作者信息、关键字和字符解码等。

一个元信息的例子：

```
<meta name="author" content="Tom">
```

在这段声明中，`name` 属性说明这个元信息用来定义作者名字，而 `content` 属性给出具体作者的名字。`name/content` 总是成对出现，具体属性值并没有限制。

`<meta>` 元素的语法如下：

```
<meta 属性="属性值">
```

`<meta>` 元素只允许起始标记单个出现，它禁止结尾标记。

`<meta>` 元素的属性如表 1.1 所示。

表 1.1 `<meta>` 元素的属性定义

	属 性 名	说 明
元素固有属性	Name=名称	设置一个需要定义的名称，例如作者、关键字和描述等
	Content=字符串	为 <code>name</code> 定义的名称设定值，例如作者姓名、关键字列表和描述语句等
	Schema=字符串	命名一个方案，用来解释属性值
	http-equiv=名称	用来替代 <code>name</code> 属性， <code>http</code> 服务器用该属性为 <code>http</code> 相应信息收集信息

下面介绍一些关于 `<meta>` 元素属性的例子。

`http-equiv` 类似于 HTTP 的头部协议，它回应给浏览器一些有用的信息，以帮助正确和精确地显示网页内容。常用的 `http-equiv` 类型有：

(1) `expires`(期限)。

说明：可以用于设定网页的到期时间。一旦网页过期，必须到服务器上重新调阅。

用法：`<meta http-equiv="expires" content="Wed, 26 Feb 1997 08:21:57 GMT">`

注意：必须使用 GMT 的时间格式。

(2) `Pragma`(cach 模式)。

说明：禁止浏览器从本地机的缓存中调阅页面内容。

用法：`<meta http-equiv="Pragma" content="no-cache">`

注意:这样设定,访问者将无法脱机浏览。

(3)Refresh(刷新)。

说明:需要定时让网页自动链接到其他网页。

用法:<meta http-equiv="Refresh" content="5; URL=http://www. yahoo.com">

注意:其中的 5 是指停留 5 s 后自动刷新到 URL 网址。

(4)Content-Type(显示字符集的设定)。

说明:设定页面使用的字符集。

用法:< meta http - equiv = " Content - Type " content = " text/html; charset = gb2312">

其作用是指定了当前文档所使用的字符编码为 gb2312,也就是中文简体字符。根据这一行代码,浏览器就可以识别出这个网页应该用中文简体字符显示。类似地,如果将“gb2312”替换为“big5”,就是我们熟知的中文繁体字符了。

<meta>元素的 name 变量语法格式如下:

<meta name="xxx" content="xxxxxxxxxxxxxxxxxxxxx">

其中 xxx 主要有下面几种参数:

1)keywords(关键字)。

说明:keywords 用来告诉搜索引擎你的网页的关键字是什么。

举例:<meta name="keywords" content="life, universe, mankind, plants, relationships, the meaning of life, science">

2)description(简介)。

说明:description 用来告诉搜索引擎你的网站主要内容。

举例:<meta name="description" content="This page is about the meaning of life, the universe, mankind and plants. ">

3)robots(机器人向导)。

说明:robots 用来告诉搜索机器人哪些页面需要索引,哪些页面不需要索引。

content 的参数有 all, none, index ,noindex, follow, nofollow。默认是 all。

举例:<meta name="robots" content="none">

4)author(作者)。

说明:标注网页的作者

举例:<meta name="AUTHOR" content="ajie,ajie@netease.com">

以上是 meta 元素的一些基本用法,其中最重要的就是 keywords 和 description 的设定。因为这两个语句可以让搜索引擎能准确地发现你,吸引更多的人访问你的站点!

根据现在流行搜索引擎(Lycos, AltaVista 等)的工作原理,搜索引擎首先派出机器人自动在 WWW 上搜索,当发现新的网站时,便检索页面中的 keywords 和 description,并将其加入到自己的数据库,然后再根据关键词的密度将网站排序。

也就是说：

(1)如果你的页面中根本没有 keywords 和 description 的 meta 标签,那么机器人是无法将你的站点加入数据库的,网友也就不可能搜索到你的站点。

(2)如果你的关键字选得不好,关键字的密度不高,被排列在几十甚至几百万个站点的后面,被点击的可能性也是非常小的。

所以,我们必须记住添加 keywords 和 description 的 meta 标签,并尽可能写好关键字和简介。

6. HTML 文档基本结构示例

下面是关于基本结构的例子,仔细阅读,你便可以了解以上各个标记对在一个 HTML 文档中的布局以及它们在浏览器中的对应位置。

代码如下：

```
<html>
<head>
<title>显示在浏览器最上边蓝色条中的文本</title>
</head>
<body bgcolor="black" text="white">
<p>黑色背景、白色文本</p>
</body>
</html>
```

代码运行结果如图 1.3 所示。



图 1.3 基本结构示例

1.2.2 HTML 元素、属性和数据类型

一个标准的 HTML 文件由 HTML 元素、元素属性和相关属性值 3 个基本部分组成。

1. HTML 元素

HTML 元素(HTML Element)用来标记文本,它通常由首尾两个标记(tag)组成。每个标记都由一组尖括号“<”和“>”包含。一组首尾标记组成一个元素。HTML 元素不区分大小写。

HTML 元素的出现形式有 3 种:

- (1)首尾标记必须成对出现的元素,如<h1>,<bottom>等;
- (2)起始标记必须出现,而结尾标记可选的元素,如<p>,<option>等;
- (3)只有起始标记,禁止使用结尾标记的元素,如<input>,等。

2. HTML 属性

每个元素都有一个或者几个属性定义该元素的显示样式、行为或者其他内容,同一个元素的各个属性之间没有顺序前后之分,属性的类型主要有等式型和布尔型两种。

属性全部都设置在 HTML 元素起始标记的尖括号内,一般格式为:

<起始标记 属性₁="属性值₁"属性₂="属性值₂"属性₃="属性值₃">

属性之间没有先后顺序,属性也没有大小写之分,每一对属性/属性值之间用空格分隔。布尔属性只有属性名称而没有属性值,指定了这个属性就代表它为“真(true)”,反之则代表“假(false)”,默认情况下,每一个布尔属性都是“假”,即不用指定,只有在需要时才添加该属性。按照一般规范而言,元素和属性都用小写字母显示。

按照 HTML 规范和标准,等号以后的属性值应该全部用英文半角双引号所包含。不过随着浏览器的不断发展和智能化,有些浏览器(IE)能够辨认单引号包含的属性值,甚至没有引号的属性值。考虑到网络标准化规范,属性值应该用英文半角双引号所包含。

3. HTML 数据类型

HTML 属性值就是各种数据类型的表现。HTML 常用的数据类型有长度、颜色、URL 链接地址和字符串等形式。下面简要介绍一些常用数据类型。更多的数据类型将结合具体元素和属性内容后续介绍。

(1)长度值。定义长度值的方法有多种,既有绝对数值定义也有相对数值定义。

1)像素值。像素是计算机所能显示的最小单位,不同计算机屏幕由于分辨率设置不同,所显示出像素的大小也不同。像素值定义长度值是一种绝对数值定义,像素值也是长度定义中使用的最为普遍的单位。像素值的英文名称为 pixel,“px”就成了单位名称,如“800px”就是 800 像素长度。像素值单位可以省略不写,如“800”就代表“800px”,即 800 像素长度。

2)相对长度值。百分比形式是最典型的相对长度值。如果对一张图片定义了百分比值,则它的参照对象是浏览器窗口宽度和高度,图片尺寸大小将会随着窗口大小的变化而变化。

3)比值长度定义。HTML 元素属性中还有一种长度定义是比值定义。它的格式为数字加上星号“*”,这些属性总是若干个一起出现,此时星号前面的数字代表这些数值之间的比值。如果星号前面的数字为 1,则 1 可以省略。

(2)颜色值。在 HTML 里,颜色有两种表示方式。一种是用颜色名称表示,比如 blue 表