

Web 编程丛书

HTML 编程指南

武 焰 等编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细介绍了 HTML 4.0 规范中的每一个元素及其属性,并给出了编程实例。

本书内容由浅入深,由易而难,很适合于那些不了解 HTML 或经验很少的用户学习。而对于已具有相当的 HTML 编程经验的读者,则完全可以从感兴趣的内容中获得新的启示。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,翻版必究。

图书在版编目(CIP)数据

HTML 4.0 编程指南/武焰编著. - 北京:电子工业出版社,1999.6 (Web 编程丛书)

ISBN 7-5053-5186-9

I . H… II . 武… III . 超文本标记语言,HTML 4.0-程序设计 IV . TP312

中国版本图书馆 CIP 数据核字(1998)第 32023 号

丛 书 名: Web 编程丛书

书 名: HTML 编程指南

编 著 者: 武 焰 等

责任编辑: 秦 梅

特约编辑: 许波建

排版制作: 电子工业出版社计算机排版室

印 刷 者: 北京京安达明印刷厂

出版发行: 电子工业出版社 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787 × 1092 1/16 印张: 18.5 字数: 492.8 千字

版 次: 1999 年 6 月第 1 版 1999 年 6 月第 1 次印刷

书 号: ISBN 7-5053-5186-9
TP·2574

定 价: 28.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换。

若书店售缺,请与本社发行部联系调换。电话 68279077

《Web 编程丛书》序

目前,许多公司和机构都在建立可供对外访问的 Internet 站点以及仅供公司内部使用的 Intranet。而且,这种势头正在迅猛增长。这样,对于建立 Web 站点所需要的各种软件技术,包括程序设计语言、开发和设计工具,则是各类 Web 站点开发和设计人员急切了解的热门主题。本套丛书正是基于这种市场需求,组织有关专业技术人员编写而成。

本套丛书大致分为两类:程序设计语言和开发工具。程序设计语言包括了 HTML 4.0、JavaScript 和 CGI; Web 站点开发和设计工具包括 FrontPage 98、Visual InterDev 1.0、IntraBuilder 1.51和简易 Intranet。下面分别介绍各个主题,读者可从中把握各种技术、各书的主要内容以及各书之间的相互关系,从而决定自己的选择。

《HTML 编程指南》

HTML(超文本标志语言)是 Web 页面描述语言,不管是在 Web 上发布信息,或者编写可供用户交互的程序,都需通过 HTML 语言来实现。

本书详细介绍了 HTML 4.0 规范中的每一个元素及其属性,并给出了相应的编程实例。

由于 HTML 是 Intranet 的基础,本套丛书中的其他主题不可避免要涉及到 HTML 的运用。我们在策划本套丛书时,为了保持丛书的一致性,各主题没有或仅用了非常有限的篇幅介绍 HTML。但这并不是说,读者不需要具备 HTML 的知识。所以,读者在阅读其他主题的过程中遇到有关 HTML 的内容时,请参考本书。

《JavaScript 编程指南》

JavaScript 是 Netscape 在 Java 的基础上开发出的一种脚本编程语言,这种语言功能比 HTML 强大,但比 HTML 复杂,由 JavaScript 编写的代码可直接嵌入到 HTML 文档中,由浏览器解释执行。

本书主要介绍了 JavaScript 的语法规则、对象应用和服务端编程等内容,并在讲解基本概念和原理时,辅以代码实例以及实用程序编程。

在阅读本书之前,读者应具备相当的 HTML 知识;否则请参考《HTML 编程指南》;同时,本书是《IntraBuilder 编程指南》的基础,因为 IntraBuilder 把 JavaScript 当做开发语言。

《CGI 编程指南》

CGI 是 Web 服务器提供交互式信息服务的标准接口。通过 CGI,Web 服务器就能够执行应用程序,并将它们的输出(如文字、图形、声音等)传送给一个 Web 浏览器。

本书首先介绍 CGI 基本知识,包括 CGI 简介、编程初步、运行准备,然后通过两个实用的例子“用户意见簿”和“网络考试”来讲解 CGI 编程应用,最后讲解了 CGI 编程的高级话题:数据库访问、搜索引擎、SSI、调试 CGI 程序和提高 CGI 程序的安全性。

由于 CGI、Visual InterDev 和 IntraBuilder 都是 Web 服务器编程的方式,请读者在阅读这三本书时注意相互参考,选择一种最适合当前项目的开发形式。

《FrontPage 98 编程指南》

FrontPage 98 是 Microsoft 公司的 Web 站点创建和管理的第三个版本,也是目前主页制作市场上占有率最高的产品。Microsoft FrontPage 98 包括 FrontPage 管理器和 FrontPage 编辑器。使用 FrontPage 管理器可用来创建、设计、查看和管理整个 Web 站点;使用 FrontPage 编辑器可创建和编辑任意复杂的 Web 页面,甚至可以不需要知道 HTML 的知识。本书以创建和管理 Web 页面的各种应用为线索,全面展示了 FrontPage 管理器和编辑器的各项功能和操作方法。

在使用《HTML 编程指南》时可以参考本书,因为 FrontPage 98 提供了一个 HTML 编辑和预览非常简便的环境;在使用《Visual InterDev 编程指南》时也可参考本书,因为 Visual InterDev 自带一个 FrontPage Editor for InterDev 版本,在《Visual InterDev 编程指南》中只用了非常有限的篇幅介绍 FrontPage,要深究 FrontPage 的应用,需参考本书。

《Visual InterDev 编程指南》

与 FrontPage 98 不同,Visual InterDev 是 Microsoft 公司的站点开发工具,主要用于 Web 客户端和服务端各种交互性功能的实现。

本书主要介绍了 Visual InterDev 的基本操作、使用 FrontPage Editor 来开发页面、用 VBScript 开发客户端和服务端脚本,以及如何进行 Web 应用程序与数据库集成等内容。

本书以 HTML 和主页制作为基础,这样,如果没有这些方面基础的读者,请注意参考《HTML 编程指南》和《FrontPage 98 编程指南》。

《IntraBuilder 编程指南》

IntraBuilder 是 Borland 公司推出的站点建立、Web 与数据库集成的解决方案,其采用 JavaScript 为开发语言,可支持多种数据库,包括 Paradox 及各类支持 ODBC 的数据库系统。

本书紧密结合数据库的开发和应用,详细介绍 IntraBuilder 的使用方法,主要内容有生成数据库表格、构造窗体、设计报表、数据库查询、安全管理、JavaScript 面向对象编程等。另外,在本书的最后,还给出了综合性的应用示例。

IntraBuilder 以 HTML 和 JavaScript 为基础,如果没有这些方面基础的读者,请注意参考《HTML 编程指南》和《JavaScript 编程指南》。

《简易 Intranet 开发指南》

采用 Office 97 和 FrontPage 98 可以创建一个简单实用的 Intranet,这种方法特别适合于那些包含几台到几十台电脑、由非计算机专业人员组成的办公室进行企业网组建。

本书专为那些想快速建立单位或工作组内部网络的用户编写。全书以通俗的语言、丰富的内容和耐心细致的操作指导,着重介绍了如何使用 Office 97 和 FrontPage 98 来快速构建 Web 服务器和 Intranet。

电子工业出版社

1999 年 1 月

前 言

在 World Wide Web 掀起全球性浪潮的年代,谁都想成为在 World Wide Web 上出版的专家,可是如何才能跟上不断涌现的新技术,成为浪尖上的弄潮儿呢?本书或许能够在这个方面给予你一些帮助。

本书详尽地介绍了 HTML 4.0 语言规范。众所周知,HTML 是专用于 World Wide Web 的出版语言,到目前为止,环球网联盟共发布了四个 HTML 规范:HTML 1.0、HTML 2.0、HTML 3.2、HTML 4.0,而 HTML 4.0 是其最新版本。

在 HTML 4.0 中增加了 HTML 文档在风格、国际化及嵌入对象等方面许多新的特性。除了包容先前版本的 HTML 规范在文本、多媒体和超链接等方面的特征,HTML 4.0 还具有以下的新特征:可以在文档中使用更加丰富多彩的多媒体内容;支持脚本语言;支持利用风格页来设置文档的风格;使 HTML 文档更便于打印;对于有缺陷(残疾)的用户,文档具有更好的可操作性。HTML 4.0 在文档的国际化方面也作出了很大的努力,使得 World Wide Web 真正名副其实。

本书详细介绍了 HTML 4.0 的特点和编程方法,内容的安排大致是:首先介绍了 HTML 4.0 文档的基本结构和代码编写思想,接着详细介绍了 HTML 4.0 规范中的每一个元素及其属性,并给出了编程实例。在附录中,给出了元素和属性的索引,方便读者检索,并对常用的 HTML 编辑工具做了简单介绍。根据附录中所提供的与 HTML 相关的 Web 站点,读者还可以获得有关 HTML 的更多信息及其最新发展动向。

读者既可以从头至尾顺序学习本书,也可以挑选感兴趣的章节来阅读。本书的内容由浅入深,由易而难,很适于那些不了解 HTML 或经验很少的用户学习。读者若已具有相当的 HTML 编程经验,则完全可以仅学习感兴趣的内容,可从目录中按功能顺序进入相应章节。

本书是集体创作的结晶,除了封面上署名的主要编著者外,王乃卫、杜青云、万俊伟、史湘宁、王炜、任伟、洪波等都参加了本书的编著工作。另外,王琨、何芳参加了本书的录入工作,在此表示衷心的感谢。

目 录

第1章 HTML 4.0 简介	(1)
1.1 HTML 4.0 的新特点	(1)
1.2 HTML 4.0 基本数据类型	(3)
1.3 用 HTML 4.0 设计文档	(10)
第2章 文档结构	(12)
2.1 HTML 4.0 文档结构	(12)
2.2 HTML 版本信息	(12)
2.3 HTML 元素	(13)
2.4 HTML 文档的头部	(14)
2.5 HTML 文档正文	(19)
第3章 文本	(28)
3.1 空白文本	(28)
3.2 结构化文本	(29)
3.3 分段和断行	(33)
3.4 使用 INS 元素和 DEL 元素标记文档的改变	(37)
第4章 列表	(39)
4.1 有序列表	(39)
4.2 无序列表	(42)
4.3 定义列表	(45)
4.4 其他列表	(46)
4.5 列表的嵌套	(46)
第5章 表格	(49)
5.1 表格的结构	(49)
5.2 表格的行组合	(56)
5.3 表格的列组合	(58)
5.4 表格的格式化	(61)
5.5 表格的嵌套	(67)
5.6 用表格格式化页面	(69)
第6章 链接	(73)
6.1 链接和锚简介	(73)
6.2 A 元素	(76)
6.3 文档间的关系:LINK 元素	(80)
6.4 路径信息:BASE 元素	(82)
第7章 嵌入对象	(84)
7.1 对象、图像和小应用程序简介	(84)

7.2	加入一幅图像:IMG 元素	(85)
7.3	嵌入对象的一般解决方案:OBJECT 元素	(86)
7.4	嵌入小应用程序:APPLET 元素	(93)
7.5	嵌入文档	(95)
7.6	图像映射	(95)
7.7	图像、对象和小应用程序的外观	(101)
7.8	如何设置替换文本	(101)
第 8 章	图文框	(103)
8.1	图文框的结构定义	(103)
8.2	图文框属性定义	(107)
8.3	图文框的嵌套	(112)
8.4	在图文框中显示文档	(114)
8.5	不支持图文框的变通方法	(118)
8.6	行内图文框	(119)
第 9 章	表单	(122)
9.1	表单元素	(122)
9.2	表单控件元素	(124)
9.3	在表单中移动光标	(148)
9.4	表单控件的使能与只读	(151)
9.5	通用网关接口(CGI)	(153)
9.6	表单的布局技巧	(154)
第 10 章	脚本程序	(164)
10.1	脚本程序简介	(164)
10.2	脚本程序设计	(164)
10.3	为不支持脚本程序的浏览器设计文档	(170)
10.4	脚本程序实例	(172)
第 11 章	文档风格	(180)
11.1	概述	(180)
11.2	在文档中加入风格	(182)
11.3	外部风格文档	(186)
11.4	段落格式化	(189)
11.5	字体	(194)
11.6	水平线	(198)
第 12 章	语言信息与文本方向	(200)
12.1	定义语言信息	(200)
12.2	定义文本或表格的排版方向	(202)
第 13 章	DTD 文档	(207)
13.1	SGML 简介	(207)
13.2	HTML DTD 文档简介	(207)
13.3	DTD 文档示例	(211)

第 14 章 字符实体参考	(237)
14.1 字符参考	(237)
14.2 字符实体参考简介	(238)
14.3 ISO 8859 - 1 字符的字符实体参考	(238)
14.4 符号、数学符号和希腊字母的字符实体参考	(243)
14.5 有重要意义和与国际化有关的字符实体参考	(248)
第 15 章 HTML 文档的检测	(250)
15.1 使用浏览器检测 HTML 文档	(250)
15.2 使用语法分析程序检测 HTML 文档	(250)
15.3 在线检测	(252)
15.4 检测 HTML 文档中超链接的正确性	(252)
第 16 章 HTML4.0 与 HTML3.2 的差别	(254)
16.1 元素的差别	(254)
16.2 属性的差别	(255)
16.3 表格的改进	(255)
16.4 表单的改进	(255)
16.5 其他改进	(256)
附录 1 元素索引	(258)
附录 2 属性索引	(262)
附录 3 HTML 在线资源	(273)
附录 4 HTML 编辑工具简介	(276)
附录 5 词汇表	(279)

第 1 章 HTML 4.0 简介

要在全球范围内出版和发布信息,需要有一种能够被广泛理解的语言,即所有的计算机都能够理解的一种用于出版的“母语”。前面已经介绍过,World Wide Web 所使用的出版语言就是 HTML。

HTML 为出版者提供了以下的功能:

- 出版在线的文档,其中包含了标题、文本、表格、列表以及照片等内容。
- 通过超链接检索在线的信息。
- 为获取远程服务而设计表单,可用于检索信息、订购产品等。
- 在文档中直接包含电子表格、视频剪辑、声音剪辑以及其他的一些应用。

因为 HTML 4.0 是新近推出的规范,要正确地理解和应用它,最好的方法就是与以前的规范做比较,因此,本章的内容从介绍 HTML 4.0 的新特点开始,另外介绍了 HTML 4.0 的基本数据类型和使用 HTML 4.0 编写文档时应注意的一些原则。在本书的附录中,对 HTML 4.0 与 HTML3.2 之间的差别进行了详细的比较。

1.1 HTML 4.0 的新特点

HTML 4.0 扩展了 HTML 的风格页,脚本程序,图文框,嵌入对象,并支持从右向左的文本排版及混合方向的文本排版,使表格的格式更加丰富,增强了表单的功能。同时,HTML 4.0 也考虑了残疾人的因素,在文档的可操作性方面作出了很大的努力。

1.1.1 国际化

许多有经验的专家一直从事 HTML 国际化方面的工作,HTML 4.0 中吸收了他们许多好的建议,使 HTML 4.0 在国际化方面较以往的版本有了长足的进步。这样,就可以用任一国家的语言来书写 HTML 文档,而且很容易在全世界范围内发布传输。

在国际化方面,HTML 4.0 迈出的重要一步是将 ISO/IEC:10646 标准作为 HTML 文档的字符集。这一标准包含了世界各国印刷业中使用到的几乎所有字符、文本方向、标点符号等。

HTML 4.0 支持在一个文档中使用多种语言,这有助于为搜索引擎提供更加有效的索引,使特定语言文本有更高的排版质量,更好地将文本转换为语音信息,更适当地使用连字符等等。

1.1.2 可操作性

随着环球网的发展越来越壮大,使用人群的组成也越来越复杂,他们的能力和技能有很大的差别,有的甚至有生理缺陷。这就要求 HTML 技术都要照顾到尽可能多的细节。可操作性成为 HTML 的一个重要指标,它对于环球网的发展是至关重要的。HTML 4.0 的可操作性主要包括以下几个方面的内容:

- 将文档的结构与文档的风格很好地分开,这就表示要鼓励使用文档风格页来代替文档

中的风格元素和属性。

- 创建易于交互的表单,表单的控制项要提供快捷键来进入,并能够将表单的控制项正确分组,将多个选择集也逻辑正确分组,并且提供活动的标签。

- 对文档中用 OBJECT 元素包含进来的每一个对象,都能够对它设置可替换的文本说明。

- 提供新的客户端的图像映像机制,使图像与文本链接很好地结合起来。

- 对由 IMG 元素包含进来的图像,需要设置替换显示的文本。

- 所有的元素都包含 title 和 lang 属性。

- 支持 ABBR 和 ACRONYM 元素。

- 利用风格页,使文档支持多种媒体。

- 可以设计更好的表格,如支持表格标题,支持表格的按列分组,以及非可见方式的表格显示。

- 支持对表格、图像、图文框的长说明文档等等。

创建可操作性强的文档,不但受到网上用户的欢迎,而且还会在将来的技术革新中获得益处。比如一篇文档结构与文档风格分开的 HTML 文档,若将来的风格格式发生了变化,只要在风格页中稍做改动就可再利用此文档,减少了重复劳动。

1.1.3 表格

HTML 4.0 的表格增加了一些属性与相关元素,使表格的结构与布局更易于控制(比如按列分组)。通过对表格列宽的定义,可以预先显示表格的框架,在表格数据传输的过程中将数据填写在表格中,而不是等到整个表格数据传输完才开始显示。

1.1.4 混合文档

HTML 4.0 提供了一种在 HTML 文档中嵌入原始媒体对象与应用程序的标准机制(有别于 IMG 与 APPLET 元素),这就是新元素 OBJECT,该元素与 IMG 元素、APPLET 元素一起,都用于在文档中嵌入对象,如:图像,视频,声音,数学公式,特殊的应用程序以及其他的一般对象。对于那些不支持特定对象的用户浏览器,可以在文档中定义打开对象的应用程序。

1.1.5 风格页

风格页简化了 HTML 标记,解除了 HTML 在文档表达形式方面的责任,方便了对文档风格的控制。文档的设计者以及用户都可以用风格页来控制文档的外观,比如:文本的字体,段落的对齐方式,字符的颜色等等。

风格信息可以为单独的一个元素设置,也可以为一组元素设置。风格信息可包含在 HTML 文档中,也可以包含到单独的风格页中。

要注意的是,将一个风格页与一个 HTML 文档相关联的机制,与具体所使用的风格页语言是不相关的。

在风格页出现之前,只能在浏览文档中控制文档的风格。HTML3.2 中包含许多用于控制对齐方式,字符字体大小,文本颜色的元素与属性。有些设计者也经常用图像与表格来格式化文档。所有的这些格式化文档的方法,都要花费作者许多宝贵的时间。风格页出现之后,这一切有了很大程度的改观。由于风格页可以与文档分开,所以可为同一结构的文档设计许多不同的风格,而不需修改原先的文档,仅修改风格页即可。由于风格页的这些优越性,HTML 4.0

不建议使用文档中的风格元素与属性,而用相应的风格页属性来代替。

1.1.6 脚本程序

通过脚本程序,可以设计出动态的 Web 页(比如可以自动校验的表单)以及基于 HTML 的网络应用程序。在 HTML 中包含的脚本程序是独立于脚本语言的,它不依赖于任何一种特定的脚本语言。

1.1.7 打印

有时,用户在打印文档,希望不仅仅打印当前的文档,还想打印与本文档相关的文档。在 HTML 4.0 中,当一个文档是一系列文档中一个时,可以用 LINK 元素描述它与其他文档的关系。打印时,也可将其他相关的文档一并打印出来。

1.2 HTML 4.0 基本数据类型

本章主要介绍在 HTML 4.0 文档中可能出现于元素中的内容以及属性值的基本数据类型。

1.2.1 大小写敏感性

每个属性的定义中都包含大小写敏感性信息。定义大小写敏感性的关键字的具体含义如下:

CS

表示属性的值对大小写敏感(Case-Sensitive),也就是说浏览器将字母“a”与“A”解释为不同的值。

CI

表示属性的值对大小写不敏感(Case-Insensitive),也就是说浏览器将字母“a”与“A”解释为相同的值。

CN

表示属性的值与大小写无关(Case-Neural),这是因为属性的值是文档的字符集中的一个数字或者一个字符。

CA

表示元素或属性的定义中自己给出大小写敏感性。

CT

表示大小写敏感性需参考数据类型的具体定义。

如果属性的值是多个值的列表,则列表中的每一个值都作了大小写敏感性说明,或指明它们的共同大小写敏感性。

1.2.2 SGML 基本数据类型

在 DTD 文档中用 SGML 的基本数据类型对 HTML 元素的内容和属性的值作了定义,下面简单介绍这些基本的数据类型。

- CDATA:这表示一个由文档字符集中任意字符组成的字符序列。其中也可以包含实体

字符,浏览器对这个字符序列作如下处理:

- 用字符代替序列中的实体编码;
- 忽略序列中的换行符;
- 用单个空格代替序列中的回车,多个空格或制表符。

浏览器往往忽略 CDATA 类型属性值中的前导与后续空格,比如“ myval ”会被浏览器解释为“myval”。作者在为属性赋值应尽量避免含有前导与后续空格。

有一些在 DTD 文档中设置为 CDATA 类型值的属性,它的值还应作具体的定义和限制,这在 DTD 文档中无法体现出来,在本书的属性定义中都有较为详细的描述。

在 STYLE 与 SCRIPT 元素中,它们的内容的数据类型也定义为 CDATA,但浏览器处理这些数据时,与其他 CDATA 数据不太一样。此时内容中的标记和实体字符必须被作为原始的文本来处理,并且被传送到具体的应用程序。当在元素的内容中出现“</”字符(即元素的结束标记)时,标志元素的内容已经结束。这与在其他情况,“</”作为元素的结束标记是很类似的。

● ID 和 NAME:这两个数据类型,经常被用于在 HTML 文档中定义 id 属性和 title 属性。它要求字符串必须以一个英文字符开头(A~Z,a~z),在后续的字符中,可以包含任意数目的英文字符,数字(0~9),连字符(-),下划线(_),冒号(:),以及点号(·)。

● IDREF 和 IDREFS:这两个数据类型表示属性的值是引用文档中的其他属性中所定义 id 的值,这里有引用的含义。IDREF 是指单个引用,IDREFS 是表示由空格分隔的多个引用。

● NUMBER:此数据类型表示属性的值至少包含一个数字(0~9)。

1.2.3 文本(text)

此数据类型表示元素的内容或属性的值是具有可读性的文本,在 DTD 文档中,以参数实体 %Text 定义。

1.2.4 统一资源标识符(URI)

在新的 HTML 规范中,一律用 URI(统一资源标识符)代替 URL(统一资源定位器),而 URL 是 URI 的一个子集。

在 HTML 文档中,有些属性的值是一个 URI,这个值的数据类型就定义为 URI。在 DTD 文档中,用参数实体“%URI;”定义。一般的 URI 类型的数据都是大小写敏感的(有些不敏感,如主机名部分。但用户应该总是认为 URI 是大小写敏感的,这是出于安全性的考虑,以减少资源定位错误)。在 URI 类型数据中还可以包括非 ASCII 字符,如一些实体编码字符。

1.2.5 颜色(Color)

对于那些与颜色有关的属性,比如 bgcolor,vlink 等,它们的数据类型为 color。在 DTD 文档中,以 %Color 定义。color 类型的数据可以用十六进制的 RGB 源色表示,也可以用颜色的英文名称表示,并且这些颜色的名称是对大小写不敏感的。表 1-1 列出了常用的颜色的名称与源色表示法。

表 1-1 常用颜色的名称与源色表示法

颜色名	英文名表示	源色表示
黑色	Black	"#000000"

续表

颜色名	英文名表示	源色表示
绿色	Green	"#008000"
银灰	Silver	"#C0C0C0"
浅绿色	Lime	"#00FF00"
灰色	Gray	"#808080"
橄榄色	Olive	"#808000"
白色	White	"#FFFFFF"
黄色	Yellow	"#FFFF00"
栗色	Maroon	"#800000"
深蓝色	Navy	"#000080"
红色	Red	"#FF0000"
蓝色	Blue	"#0000FF"
紫色	Purple	"#800080"
灰绿色	Teal	"#008080"
浅紫色	Fuchsia	"#FF00FF"
浅蓝色	Aqua	"#00FFFF"

由上表中可以看出,Black 与"#000000"都可以将属性值设置为黑色。

尽管颜色在文档中的应用,可以使文档丰富多彩,吸引用户的注意力,但在设计时,需要注意以下几个方面的问题:

- 在 HTML 4.0 中,用于设置颜色的元素和属性都可以用文档风格页来替换,这样,可以方便地修改文档的风格。
- 在使用丰富多彩的颜色时,也应当考虑到那些有色彩障碍(色盲)的人,为他们设计可替换的文档。另外,过于艳丽的文档,即使正常视觉的读者也很容易疲劳。
- 当使用背景图像或背景颜色时,注意文本颜色的设置,要有较强的对比度,一般背景选较淡较柔和的颜色,文本选较深的颜色。
- 在文档中设置的文本或背景的颜色,在不同的平台,如工作站、个人电脑、苹果机、笔记本等,将显示为不完全相同的颜色,这对文档的外观有些影响。最好在发布 Web 页之前,首先在不同的平台上测试一下。
- 在实践中,采用那些已被广泛接收的配色方案,将加快文档的设计,并最大限度减少用户的适应时间。

1.2.6 长度(lengths)

HTML 中有关长度类型的数据有以下三种定义方法:

1. 像素法(Pixels):这种定义长度的方法,是给出一个整数,它表示文档中的对象在浏览器窗口中占有的像素数。比如对于属性值"50",它表示 50 个像素点。在 DTD 文档中的定义为"%Pixels;"。

2. 百分比长度法:这种定义长度的方法是给出一个百分比值,它表示当前对象在水平方向或垂直方向的长度占整个可用空间(例如浏览器窗口、打印机的纸张等)的百分比。例如一个水平长度的设置值为 50%,则它表示在宽为 800 个像素的窗口中占 400 个像素。在 DTD 文

档中的定义为“%Length;”。

3. 相对长度法:这种定义长度的方法一般用于几个对象的同时定义,它定义了每个对象所占的相对尺寸,定义的格式为“i*”,其中i是一个整数。比如现在有一个属性的值为“m* n* i*”(m,n,i都为整数),若可用空间尺寸为1,则它表示三个对象所占尺寸为

$\frac{m}{m+n+i} \cdot l, \frac{n}{m+n+i} \cdot l, \frac{i}{m+n+i} \cdot l$ 。在 DTD 文档中的定义为“%MultiLength;”。

这几种定义长度的方法可以混合使用。在分配空间时,先分配像素与百分比法定义的对象尺寸。在剩余的空间中,再分配由相对法定义的对象尺寸。例如一宽度属性值为“100 50% 1* 2*”,窗口的宽度为 800 像素点,则第一个对象分配 100 像素点,第二个对象分配 400 像素点,剩下的 300 个像素点,第三个对象分配 100 个,第四个对象分配 200 个。

注意长度值是与大小写无关的。

1.2.7 内容类型(MIME 类型)

内容类型即 MIME(多类 Internet 邮件扩展)数据类型。这些数据类型主要包括:

“text/html”:表示 HTML 格式的文本文件。

“image/png”:表示 png 格式的一幅图像。

“image/gif”:表示 gif 格式的一幅图像。

“video/mpeg”:表示 mpeg 格式的视频文档。

“audio/basic”:表示基本格式的声音文档。

“text/tcl”:表示 tcl 格式的文本文件。

“text/javascript”:表示 javascript 程序代码的文本文件。

“text/vbscript”:表示 vbscript 程序代码的文本文件。

“text/css”:表示含风格语言代码的文本文件。

此类型仅是内容类型的一部分,还有其他的类型,而且这些类型还在不断地增加。

注意:

“媒体类型”(“media type”)指定了一个被链接资源的本质。为了与现有的用法保持一致,现在是采用了术语“内容类型”。该类型在 DTD 文档中表示为“%ContentType;”。内容类型是大小写不敏感的。

1.2.8 语言

语言代码是设置与语言有关的属性的一种数据类型,它一般用两个字母代表一种语言,如“en”代表英语,“fr”代表法语。在 DTD 文档中,语言代码数据类型以“%language Code;”表示。在语言代码中,不允许出现空格字符。语言代码是不区分大小写的。

1.2.9 字符编码方式

字符编码方式是为“charset”属性赋值的一种数据类型,在 DTD 文档中以“%Charset”定义。为 charset 属性设置值时,它必须是在 IANA 中已定义了编码方式,例如“edu-jp”。字符的编码方式是大小写不敏感的。用户浏览往往根据 charset 属性中设置的编码方式来解释相关的内容或下载相关的资源。

1.2.10 单个字符

某些属性需要使用单个字符,这是在文档的字符集中得到的。在 DTD 文档中,这些属性采用“%Character”类型。

单个字符可以由字符引用来指定,例如“&”。

1.2.11 日期和时间

在国际标准中有多种表示日期时间的格式,HTML 中采用了一种格式来表示其文档中的日期与时间类型的数据。在 DTD 文档中,日期和时间数据类型是用“%Datetime”来表示的。

它的格式为:

YYYY-MM-DDThh:mm:ssTZD

其中:

YYYY:以四位数表示年代;

MM:以两位数表示月(从 01~12);

DD:以两位数表示日(从 01~31);

T:固定字符,表示时间的开始,必须大写;

hh:两位数表示时间(从 00~23),并且 AM 和 PM 不允许出现;

mm:两位数表示分(从 00~59);

ss:两位数表示秒(从 00~59);

TZD:表示时间区域。

TZD 有以下几种格式:

Z:表示零时区(UTC),必须大写;

+hh:mm:表示在零时区之前的时区,超过零时区 hh 小时 mm 分钟;

-hh:mm:表示在零时区之后的时区,滞后零时区 hh 小时 mm 分钟。

注意在以上的表示格式中,标点符号如冒号等是不可缺省的。若无法确定秒时,可以用“00”表示。定时与分也类似。

在以上的格式中,大写的 T 表示时间元素的开始。

1.2.12 链接文档类型

链接文档类型是指在 HTML 文档的头部,由 LINK 元素链接的与当前文档相关的文档与当前文档之间的关系。在 DTD 文档中,链接文档类型以“%LinkTypes”定义,它表示了一系列链接类型的列表,其中以空格分隔。在链接文档类型中不能出现空格,这些类型字符串是不区分大小写的,如“Alternate”与“alternate”具有相同的意义。

浏览器、搜索引擎等对于这些链接类型的解释可以采用多种不同的方式,比如说,浏览器可能通过一个导航条来提供对链接文档的进入。

主要的链接文档类型有:

Alternate

表示链接文档是比当前文档更新的文档。当此类型与 lang 属性一起使用时,表示链接文档是当前文档的一个其他语言版本。当与属性 media 一起使用时,表示链接文档是当前文档用于其他媒体的一个版本。

Stylesheet

表示链接文档是当前文档的一个外部风格页。

Start

表示链接文档是一系列文档中的第一个文档,这有助于让搜索引擎确定一系列文档的开始点。

Next

表示链接文档是一系列文档中排在当前文档之后的文档。浏览器可以预先下载此文档,以节省浏览时间。

Prev

表示链接文档是一系列文档中排在当前文档之前的文档。在一些浏览器中“Previous”也表示同样的含义。

Contents

表示链接文档是当前文档内容的目录,在一些浏览器中,“ToC”(Table of Contents)也表示同样的含义。

Index

表示链接文档是当前文档的一个索引。

Glossary

表示链接文档是当前文档的一个词汇表。

Copyright

表示链接文档是当前文档版权声明。

Chapter

表示链接文档是一系列文档中的一个章节。

Section

表示链接文档是作为一系列文档中的一个部分。

Subsection

表示链接文档是一系列文档中的一个组成部分。

Appendix

表示链接文档是一系列文档中的一个附录。

Help

表示链接文档提供了帮助信息。

Bookmark

表示一个书签。这个书签链接到外部文档的一个进入点。可以为这个书签设置 title 属性,此属性可用于书签名的显示。在每一个文档中都可以定义多个书签。

在文档中可以应用以上这些链接文档的类型。如果用户还想要定义在此规范中未定义的其他链接文档类型,他可以使用 HEAD 元素的 profile 属性。详情请参见后面章节的介绍。

1.2.13 媒体说明

媒体说明数据类型是用于说明显示或处理文档中的元素的设备。在 DTD 文档中,媒体说明以“% MediaDesc”定义。主要的媒体类型有以下几种:

screen

代表不分页的计算机显示屏。

tty

表示使用固定宽度字体的媒体设备,如电传电报打字机(teletypes)、终端或者显示能力受限的可移植设备。

tv

类似于电视(TV)的显示设备。这类设备分辨率差,颜色少,卷屏能力有限。

projection

投影仪。

handheld

手持数字助理,这类设备的显示屏较小,甚至是单色的,只能输出点阵图形,带宽也有限。

print

打印机,或者是显示器上模拟的打印预览。这类设备一般分页显示,纸质不透明。

briaille

盲文输出设备。

aural

语音合成器。

all

以上的所有设备。

目前各种浏览器还仅支持以上的几种媒体,将来还会有许多新的媒体类型出现。目前浏览器处理新设备的方法如下:

1. 对 Media 属性中的以逗号分隔的媒体分类。

例如:

```
media = "screen, 3d-glasses, print and resolution > 90dpi"
```

被分类为:

```
"screen"
```

```
"3d-glasses"
```

```
"print and resolution > 90dpi"
```

2. 将每种媒体的名称进行简化。简化的方法是:在第一个不是 US ASCII 字母(a~z, A~Z)、数字(0~9)和连字符("-")之后的内容被截去。在上一个例子中,对媒体设备进行简化后的结果为:

```
"screen"
```

```
"3d-glasses"
```

```
"print"
```

3. 将上述简化后的媒体名与前述的媒体设备名作匹配比较,同时要区分大小写。匹配的媒体类型将被保留。在上一个例子中,媒体类型"3d-glasses"不匹配,因此所剩下的类型是 screen 和 print。

注意,在风格页中可能会包含与媒体有关的变量,如 CSS @media construct。在这种情况下,使用"media = all"是合适的。