

Unit 1

页面文字基本标记

本单元由下列部分组成:

第一节 *html* 概念和基本结构

第二节 网页的文件体与文本标记

在完成本单元学习后, 你将:

- 知道什么是 html 和因特网
- 掌握 html 文档基本结构
- 掌握基本的文字标记
- 能够用记事本或其他文本编辑器编写简单 html 页面

第一课 html 文档和网页文本

在本课里，你将从一个全新的角度来看我们经常使用的因特网，以及浏览的网页，你将学到以下内容：

- 什么是因特网
- 什么是 html
- html 文档的结构
- 网页文本的插入和修改

理论教学

(2 课时)

1.1 什么是因特网

因特网（**Internet**）是当今世界上最大的、开放的、由众多计算机网络互连而成的网络。**Internet** 原译名为互连网络。1997年7月18日，全国科学技术名词审定委员会发布的信息科学领域英文名词的中文译名中，把 **Internet** 统一规范为“因特网”。

Internet 不仅是一个计算机网络，更重要的是它是一个庞大的、实用的、可享受的信息源。我们可以送出或接受电子邮件通信；可以与别人建立联系并互相索取信息；可以在网上发布公告，宣传你的信息；可以参加各种专题小组讨论；可以免费享用大量的信息源和软件资源。

1.1.1 因特网相关概念

◆ **www (world wide web)**

www 称为万维网，它是基于超链接的超文本系统。超文本是收集、存储和浏览离散信息以及建立和表现信息之间关系的技术。超链接 (**hyperlink**) 是指文本中的词、短语、符号、图象、声音剪辑或影视剪辑之间的链接，或者与其他的文件、超文本文件之间的链接，也称为“热链接 (**hotlink**)”，或者“超文本链接 (**hypertextlink**)”。建立互相链接的这些元素不受空间位置的限制，它们可以在同一个文件内也可以在不同的文件之间，通过网络可与世界上的任何一台连网计算机上的文件建立链接关系。

◆ **http 协议 (hypertext transfer protocol)**

http 称为超文本传输协议，是一个属于应用层的面向对象的协议，于 1990 年提出，经由于其简捷、快速的方式，适用于分布式超媒体信息系统。目前在 **www** 中使用的是 **http/1.1**。**http** 协议的主要特点可概括如下：

- ✓ **支持客户/服务器模式；**
- ✓ **简单快速；**
- ✓ **灵活；**
- ✓ **无连接；**
- ✓ **无状态。**

◆ **ftp 协议 (file transfer protocol)**

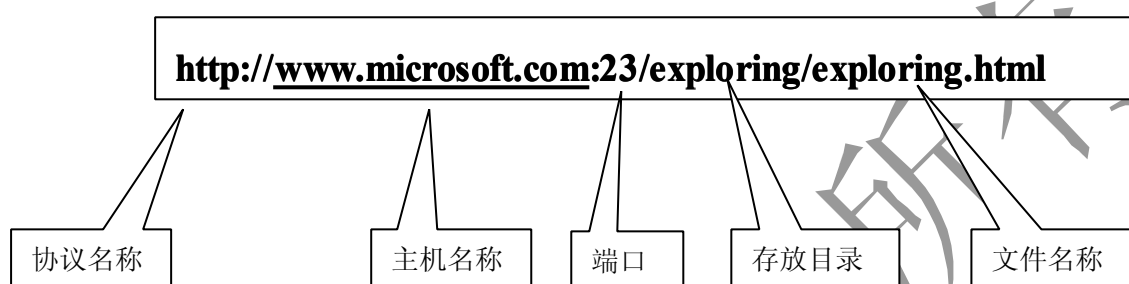
ftp 文件传输协议是 **Internet** 上使用非常广泛的一种通讯协议，它是为 **Internet** 用户进行文件传输(包括文件的上传和下载)而制定的。要想实现 **ftp** 文件传输，必须在相连的两端都装有支持 **ftp** 协议的软件。

◆ url 地址: (uniform resource locator)

url 就是“统一资源定位器”,通俗点说,它是用来指出某一项信息所在位置及存取方式。比如我们要上网访问某个网站,在 IE 或其它浏览器里的地址一栏中所输入的就是 url。url 是 internet 上用来指定一个位置(site)或某一个网页(webpage)的标准方式,它的语法结构如下:

协议名称://主机名称[:端口地址/存放目录/文件名称/

例如:



url 实例介绍

http: www 上的存取服务

`http://www.yahoo.com`

telnet: 使用远端登录的服务

`telnet://bbs.nstd.edu`

ftp: 文件传输协议, 通过互联网传输文件

`ftp://ftp.microsoft.com/`

mailto: 寄一封 e-mail 给他人的服务

`mailto:jt747@ms.hinet.net`

file: 存取硬盘文件的服务

`file://c:/temp/abc.html`

url 可分为两种类型:

◆ 绝对 url 和相对 url

绝对 url 就是指明需要访问的信息或资源的绝对位置。绝对 url 的基本语法格式为:

accessmethod://servername[:port]/directory/name

其中, accessmethod 是指定资源服务器的服务方式, 称为“使用协议”。在 www 系统中, 最常用的就是“http”协议。servername 指服务器域名, 接入到 Internet 中每个可供访问的服务器, 都有一个专用的域名, 用户要访问服务器上的资源, 必须指明服务器的域名。“[:port]”是指明进入一个服务器的端口号, 它是用数字来表示的, 一般可缺省。“directory”

是指明文件所在服务器的目录或路径。“name”是文件名，在缺省的情况下，首先会调出称为“主页”的文件。

相对 url 就是定位需要访问资源的相对路径。所谓“相对路径”，就是所需资源相对于当前位置的路径。例如，当我们已处于服务器中的同一个路径时，如果该路径中有多个文件需要访问，那么我们仅需指明文件名就可以了。

1.2 什么是 html

hypertext markup language 即 **html (超文本标记语言)** 是生成活动文档的代码系统。活动文档就是使你和其他人能够进行交互作用的文档。html 的普遍应用带来的超文本技术使人们可以从一个主题转入另一个主题，而不再是线性地寻找和阅读信息。html 使文档的访问、阅读及使用完成得快捷和有效。

html 4 [html] 是 sgml (standard generalized markup language) 的一个应用，遵从国际标准 iso 8879，被广泛的当作 world wide web 上的标准出版语言。

sgml 是一种描述标记语言，主要用于电子文档交换，文档管理和文档发布的语言。sgml 出现在 80 年代中期，一直保持得很稳定。稳定原因是它有丰富的特征并具有灵活性。但是，灵活性也带来一定程度的复杂性，限制了它在多种环境下的适应性，包括 world wide web。

1.2.1 什么是 web 标准

web 标准是一系列标准的集合。网页主要由三部分组成：**结构 (structure)、表现 (presentation) 和行为 (behavior)**。对应的标准也分三方面：结构化标准语言主要包括 xhtml 和 xml，表现标准语言主要包括 css，行为标准主要包括对象模型（如 w3c dom）、ecmascript 等。这些标准大部分由 **w3c** 起草和发布，也有一些是其他标准组织制订的标准。

◆ 结构标准语言

xml (the extensible markup language) 是可扩展标记语言。目前推荐遵循的是 w3c 于 2000 年 10 月 6 日发布的 xml1.0，参考 (www.w3.org/tr/2000/rec-xml-20001006)。和 html 一样，xml 同样来源于 sgml，但 xml 是一种能定义其他语言的语言。xml 最初设计的目的是弥补 html 的不足，以强大的扩展性满足网络信息发布的需要，后来逐渐

用于网络数据的转换和描述。

xhtml (extensible hypertext markup language) 是可扩展的超文本标记语言。它是一种在 html 4.0 基础上优化和改进的新语言, 目前遵循的是 w3c 于 2000 年 1 月 26 日推荐的 xhtml1.0 (参考 <http://www.w3.org/tr/xhtml1>)。简单的说, 建立 xhtml 的目的就是实现 html 向 xml 的过渡。

◆ 表现标准语言

css 是 cascading style sheets 层叠样式表的缩写。目前遵循的是 w3c 于 1998 年 5 月 12 日推荐的 css2。w3c 创建 css 标准的目的是以 css 取代 html 表格式布局、帧和其他表现的语言。纯 css 布局与结构式 xhtml 相结合能帮助设计师分离外观与结构, 使站点的访问及维护更加容易。

◆ 行为标准

dom 是 document object model 文档对象模型的缩写。根据 w3c dom 规范, dom 是一种与浏览器、平台和语言的接口, 使得你可以访问页面其他的标准组件。dom 解决了 netscape 的 javascript 和 microsoft 的 jscript 之间的冲突, 给予 web 设计师和开发者一个标准的方法, 让他们来访问他们站点中的数据、脚本和表现层对象。

ecmascript 是 ecma(european computer manufacturers association)制定的标准脚本语言 (javascript)。

1.2.2 为什么要使用标准

我们大部分人都有深刻体验, 每当主流浏览器版本的升级, 我们刚建立的网站就可能变得过时, 我们就需要升级或者重新建造一遍网站。例如 1996-1999 年典型的“浏览器大战”, 为了兼容 netscape 和 ie, 网站不得不为这两种浏览器写不同的代码。同样的, 每当新的网络技术和交互设备的出现, 我们也需要制作一个新版本来支持这种新技术或新设备, 例如支持手机上网的 wap 技术。类似的问题不胜举: 网站代码臃肿、繁杂浪费了我们大量的带宽; 针对某种浏览器的 dhtml 特效, 屏蔽了部分潜在的客户; 不易用的代码, 残障人士无法浏览网站等等。这是一种恶性循环, 是一种巨大的浪费。

于是在 w3c (w3c.org) 的组织下, 网站标准开始被建立 (1998 年 2 月 10 日发布 xml1.0 为标志), 并在网站标准组织 (webstandards.org) 的督促下推广执行。

网站标准的目的是：

- ◆ 提供最多利益给最多的网站用户
- ◆ 确保任何网站文档都能够长期有效
- ◆ 简化代码、降低建设成本
- ◆ 让网站更容易使用，能适应更多不同用户和更多网路设备
- ◆ 当浏览器版本更新，或者出现新的网络交互设备时，确保所有应用能够继续正确执行。

对于网站设计和开发人员来说，遵循网站标准就是使用标准；对于你的网站用户来说，网站标准就是最佳体验。

1.2.3 web 设计工具

- ◆ 记事本：最快捷的编辑工具；
- ◆ editplus：功能优于记事本，具有彩色分辨代码的功能；
- ◆ macromedia dreamweaver：综合型编辑工具，功能强大，有可视化的界面。
- ◆ microsoft frontpage：office 组件。

1.3 html 文档的结构

html 文件是标准的 Ascii 文件，它加入了许多被称为标记(tag)的特殊字符串的普遍文本文件。每一个 html 文档结构上都由文档头部(head)和文档体组成 (body)，即头元素<head>..</head>和体元素<body>...</body>和一些注释组成。也就是说，一个 html 文件应具有下面的结构：

文件范例：1-1.htm

```
01: <html>
02: <head>
03: <title>我的第一个网页</title>
04: </head>
05: <body background="background.gif">
06: <h2>这是我的第一个网页！</h2>
07: </body>
08: </html>
```

文件说明：

第 01、02、04、05、07、08 行就是 html 文档的基本结构。

显示效果：



1.3.1 html 文档的元素

绝大多数元素是“容器”，即它有起始标记和结尾标记。在起始标记和结尾标记中间的部分是元素体。每一个元素都有名称和可选择的属性，元素的名称和属性都在起始标记内标明。

一般来讲，html 的元素有下列三种表示方法：

1. <元素名>文件或超文本</元素名>
2. <元素名 属性名=“属性值”……>文本或超文本</元素名>
3. <元素名>

html 文件中，有些元素只能出现在头元素中，绝大多数元素只能出现在体元素中。在头元素中的元素表示的是该 html 文件的一般信息，比如文件名称，是否可检索等等。这些元素书写的次序是无关紧要的，它只表明该 html 有还是没有该属性。与此相反，出现在体元素中的元素是次序敏感的，改变元素在 html 文件中的次序会改变该 html 文件的输出形式。

1.3.2 html 文档的头部

html 头部用<head>标识符来标识头部的开始，用 </head>标识符来标识头部的结束。<head>之间包括文档的头部信息，如文档总标题等，若不需头部信息则可省略此标记。

在头部区域中的主要标识符元素：

title 标记

title 元素是文件头中唯一一个必须出现的元素，它也只能出现在文

件头中。title 标明该 html 文件的题目，是对文件内容的概括，以窗口的名称显示出来。一个好的题目应该能使读者从中判断出该文件的大概内容。一般情况下它的长度不应超过 64 个字符。

基本语法:

```
<title>.....</title>
```

语法解释:

标记内部就是标题的内容。

在范例 1-1.htm 中第 3 行就是页面的标题。

meta 标记

meta 是用来在 html 文档中模拟 http 协议的响应头报文。meta 标记的用处很多，比较常用的有以下几个：

- ◆ 定义编辑工具：告诉搜索引擎你的站点的制作工具。

基本语法:

```
<meta name="generator" content="">
```

语法解释:

content 中填写编辑工具的名称如 microsoft frontpage, dreamweaver 等。

- ◆ 设定关键字：向搜索引擎说明你的网页的关键词。

基本语法:

```
<meta name="keywords" content="">
```

语法解释:

content 中填写关键字，多个关键字用逗号隔开。

- ◆ 设定描述：告诉搜索引擎你的站点的主要内容。

基本语法:

```
<meta name="description" content="">
```

语法解释:

content 中填写描述信息。

- ◆ 设定作者：告诉搜索引擎你的站点的制作的作者。

基本语法:

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

语法解释:

content 中填写作者姓名。

- ◆ 设定字符集：用以说明主页制作所使用的文字以及语言。

基本语法:

```
<meta http-equiv="content-type" content="text/html ; charset=value">
```

语法解释:

charset 的 value 可取值有: 简体中文 gb2312, 英文 iso-8859-1, 还繁体中文 big5 等。

◆ 设定自动跳转: 定时让网页在指定的时间内跳转到某页面。

基本语法:

```
<meta http-equiv="refresh" content="n;url=http://yourlink">
```

语法解释:

n 为跳转时间, 单位是秒; http://yourlink 表示需跳转到页面, 可以使绝对路径, 也可以是相对路径。

如果 content 中只有 n, 则为 n 秒后刷新页面。

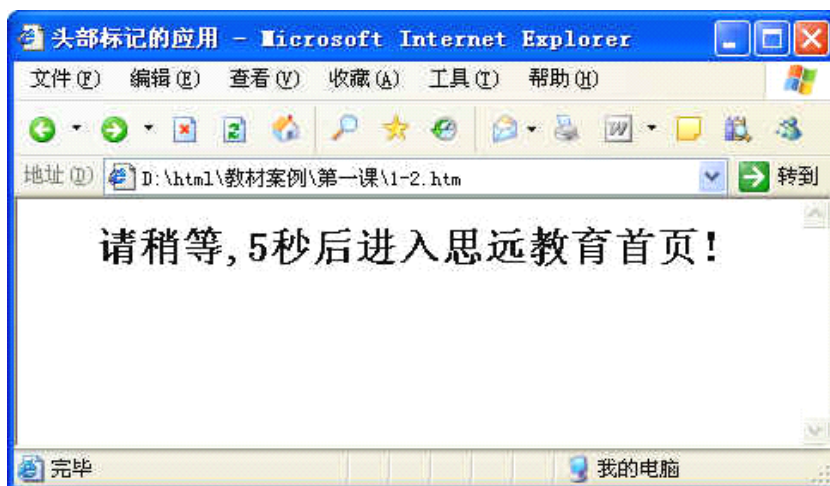
文件范例: 1-2.htm

```
01: <html>
02: <head>
03: <meta name="generator" content="notepad">
04: <meta name="keywords" content="头部标记,应用">
05: <meta name="description" content="这是关于头部标记应用的
    测试页。">
06: <meta name="author" content="思远教育">
07: <meta http-equiv="content-type" content="text/html;charset
    =gb2312">
08: <meta http-equiv="refresh" content="5;url=http://www.think-
    bank.com.cn">
09: <title>头部标记的应用</title>
10: </head>
11: <body>
12: <h2 align="center">请稍等,5 秒后进入思远教育首页!</h2>
13: </body>
14: </html>
```

文件说明:

第 03 行定义编辑工具为记事本, 第 04 行设定关键字为“头部标记”和“应用”两个, 第 05 行设定描述为“这是关于头部标记应用的测试页”, 第 06 行设定作者为“思远教育”, 第 07 行设定字符集为简体中文, 第 08 行设定 5 秒后自动跳转到 http://www.think-bank.com.cn, 第 09 行设定标题为“头部标记的应用”。

显示效果:



1.3.3 html 文档的主体

<body>标识符标识文档体的开始, </body>标识符标识文档体的结束。文档体中包括由段落、列表、表格和其他元素组成的 html 文档内容。

基本语法:

```
<body background="url" text="#rrggbb" leftmargin=""  
topmargin=""" >  
</body>
```

语法解释:

- ◆ background: 指明作为背景图象的 url 地址
- ◆ bgcolor: 指明文档背景颜色
- ◆ text: 指明页面默认文字的颜色
- ◆ leftmargin: 设定页面的左边距
- ◆ topmargin: 设定页面的上边距

1.4 标记和属性

html 不区分大小写, 属性值用单引号、双引号括起来, 也可以不要引号。格式良好的 html 文档应该拥有以下特点:

- ◆ 标记和属性名称必须为小写。
- ◆ 嵌套必须合理。

html 文档的内容应该拥有以下特点:

- ◆ 把精力集中在文档内容本身。
- ◆ 注释: 注释不允许嵌套。

所有 html 文档都应当遵守 html sgml dtd, 定义 html 标准的正式的

文档类型定义 (document type definition dtd)。dtd 定义用来创建 html 文档的标记和语法。

1.5 文本的插入和修改

文本是网页中最基本、最常用的元素。

1.5.1 标题字标记<h>

所谓标题文字就是以某几种固定的字号去显示文字, 共有 6 种, 分别为 h1, h2,...h6。标题号越小, 字体越大。

基本语法:

```
<h1 align=value>...</h1>
```

语法解释:

value 的取值: left 标题居左, center 标题居中, right 标题居右

文件范例: 见范例 1-2.htm 第 14 行

1.5.2 设置字体、字号和颜色属性

◆ 设置字体

确定页面文字显示的字体。

基本语法:

```
<font face=value>...</font>
```

语法解释:

value 的取值为“宋体”、“黑体”等。

◆ 字体大小

html 有七种字号, 1 号最小, 7 号最大。缺省字号为 3, 可以用 <basefontsize=字号> 设置缺省字号。

基本语法:

```
<font size=value>...</font>
```

语法解释:

value 的取值有两种:

1、绝对字号: 1~7

2、相对字号: +1~+7, -1~-7。“+”号表示字体变大, “-”号表示字体变小。

◆ 字体颜色

基本语法:

```
<font color=value>...</font>
```

语法解释:

value 的取值:

- 1、6 位 16 进制数，分别指定红、绿、蓝的值。
- 2、代表颜色的英文单词，如：black, olive, teal 等（详见附录四）。

1.5.3 常用字符级标记

常用字符级标记分为物理风格和逻辑风格。

- ◆ 物理风格直接指定字体。有黑体，<i>斜体，<u>下划线，<tt>打字机体。
- ◆ 逻辑风格指定文本的作用。有强调，特别强调，<code>源代码，<var>变量，<small>较小，<big>较大，<sup>上标，<sub>下标

1.5.4 特殊字符与注释文字

◆ 特殊字符

在 html 中，一个特殊字符有两种表达方式:

1、数字参考

就是用数字来表示文档中的特殊字符，通常由前缀“&#”，加上数值，再加上后缀“;”而组成，其表达方式如下:

&#d; 其中 d 是一个十进制数值

下面是一些特殊字符的数字参考示例:

© 对应于特殊字符“©” ® 对应于特殊字符“®”

2、实体参考

是用有意义的名称来表示特殊字符，通常由前缀“&”，加上字符对应的名称，再加上后缀“;”而组成，其表达方式如下:

&name; 其中 name 是一个用于表示字符的名称，它是区分大小写的。

下面是一些特殊字符的实体参考示例:

 半角空格

© 对应于特殊字符“©”

® 对应于特殊字符“®”

不是所有的浏览器都能够正确认出采用实体参考方式的特殊字符，

◆ 注释文字

基本语法:

语法解释:

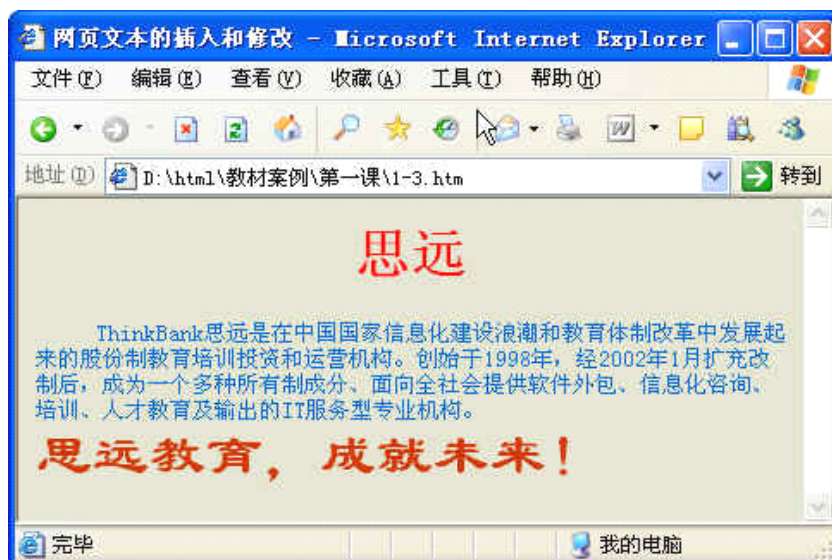
文字标记综合范例

13: </html>

文件说明:

第 06 行设定了页面的背景色为 “#d9d9d9”, 页面文字颜色为红色, 第 07、09 行为注释, 第 08 行为 1 号标题, 并且居中显示, 第 10、11 行是 font 的应用, 在第 10 行中插入了几个特殊字符 “空格”。

显示效果:



课后自测

请完成课后自测题：

课后自测是对学员学习进度的检查，帮助学员找出对本课知识点理解的不足。请学员在理论授课完成后立即完成下列习题，对于不能准确回答或有疑问的问题，请即刻向授课教员提出。

一、填空题

1. 因特网是_____。
2. html 的全称是_____。
3. html 文件是纯文本文件，它采用_____协议在网络上传输。
4. html 文档由_____、_____和_____组成。
5. meta 标记是 _____，其 name 属性的取值有_____。
6. 使文字加粗可以使用_____标记。
7. 上标标记是_____，下标标记是_____。

二、问答题

1. 阐述使用 web 标准的好处。
2. body 中可以加入那些属性？这些属性分别起什么作用？

上机指南

(2 课时)

上机任务书

任务背景:

手工编写如下网页



任务目标:

- 1、需要有完整的 head，需要包括作者信息、关键字和标题等。
- 2、使用标题标记以及 font 标记实现文字的显示。
- 3、尝试对文字进行分段并插入图片。

上机指导书

1. 在桌面建立文件夹 test;
2. 进入 test 并新建文本文件，将其重命名为 index.htm，注意要将扩展名要从 txt 改为 html;
3. 在 test 下建立文件夹 img，并将图片 harry.jpg 拷贝到里面;
4. 用记事本将刚建立的文件 index.htm 打开，并在其中输入代码。
5. 以下是关键代码：

```
<head>
<meta http-equiv="content-type" content="text/html; charset=gb2312">
<meta name="author" content="作者的名字">
<meta name="keywords" content="哈利.波特">
<title> 哈利.波特</title>
</head>
<body>
<font face="黑体" color="#ff0000">
<h1 align="center">第一章 猫头鹰的信</h1>
</font>
.....

.....

</body>
```

课后学习

课后练习

尝试完成如下的网页:

创建优秀网页的6个好习惯

【日期】2005-1-13 【人气】176 【作者】『一分流水』 【来源】『一分流水』

1、不要将链接全部放在图像中

以往的网页中的链接都是文字,而且用下划线以资识别。但不久人们就讨厌了这些文字链接,于是都采用了图像链接。后来,更加嫌一个图像一个链接太麻烦了,于是将所有的链接都放进了一个图像中去。但并不是所有的用户都打开图像载入,如果你把所有的链接都放入图像中去,而没有任何同样功能的文字链接的话,用户也许会不知所措。

2、慎用自动滚动

曾经到过内容“丰富”的网页,它使用了很多常见的网页制作技巧:现在的时间、访问时的时间等等。不过有一个用得不是很好,就是窗口的自动滚动。我只是想发个邮件给网页的主人,但我却无法使滚动条停下来让我点击链接,于是我花了很久才“命中”窗口中的E-mail链接——真后悔军训的时候没有练好射击!

3、不要用延时效果的JavaScript

现在的网页制作者为了使网页更加具有影响力,都疯狂地加入JavaScript。以现在的“猫”奔跑的速度而言,这些脚本也算不了什么,用户不会介意。但是,如果这些脚本可能导致延时的话,用户的态度就不同了。

某个国人网页的首页在载入时窗口的底色——窗口里面没有任何内容——由白色逐渐变成黑色,整个过程耗费了几秒,嘿!别小看这几秒,我能检查全部的邮箱了!

4、不要依赖浏览器

太多的网站首页中加入了“请在800×600分辨率下使用IE浏览器浏览本网站”等字样的文字。我是安装了IE、NC和Opera,因为我要制作网页,需要它们来检验网页在这些浏览器下的可读性。但一般用户不会安装那么多的浏览器,更加不会牢记着哪个网站需要用哪个浏览器浏览。

也许不可能让网页在使用所有的浏览器浏览都一样的效果,但尽量地去达到该目的,会让网页在这个浏览器下有这样的特色、在另一个浏览器下又有不同的特色,这是网页制作者的工作,不是吗?

5、不要忘记版权

现在,网上侵犯知识产权的问题非常猖獗,某免费个人网页提供商上的网页尤其严重——甚至有盗版光盘贩卖呢!为此一些新的免费个人网页提供商就要求用户在申请时提交详细的、真实的个人资料,以防备不可预料的事情发生。

所以,如果你的网页需要引用别人的东西的话,要么你就和他联系一下,征求意见;要么就在明显的位置注明其版权归属。这并非多余,这是你的网页“长寿”的秘方!

当然,如果是你自己制作的图像或自己撰写的文章,你绝对有权声明版权归你自己所有!

6、不要懒于更新

某个个人网页聚居地上有很多的网页都是很久前建立而至今没有更新过的,姑且不说这是浪费公用资源的行为,访问过这些网页的人看到网页主人如此懒惰,你说他还会再去访问吗?

谁能确定下次访问不会又是这般模样呢?因特网上的资源如此浩瀚,谁会去访问一个从不更新的网页呢?

【评论】【打印】

相关评论

· 暂时没有评论

[查看更多评论或参与评论](#)

[:: 本站荣誉](#) [:: 联系站长](#) [:: 友情链接](#) [:: 版权声明](#) [:: 网站纪事](#) [:: 关于我们](#) ::

附: 文字素材见“free93“文件夹

小结

今天我们初次接触 html，主要学了以下的内容：

1. 什么是因特网
2. 什么是 html，其有哪些标准，我们如何编制 html 文档
3. html 的文档结构及标记和属性的使用方式
4. 文本的插入与格式的设置

以上内容中需要我们重点去理解的就是 html 的文档结构及标记和属性的使用方式，因为当大家对这种编码方式有了一个很深刻的认识，后面内容的自学就会变得相当的简单了。如何认识呢，就是多看一些网站的源代码，通过教材尽量去理解它，看多了你自然会明白 html 原来并不难。需要强调的是在我们这门课程中源代码都需要大家用记事本输入，从而使大家对标记、属性代码能够更熟练的掌握并使用。

段落、列表、图片、 超链接标记

本单元由下列部分组成：

第一节 段落和列表

第二节 图片与超链接

在完成本单元学习后，你将：

- 掌握段落标记的使用
- 掌握列表标记及其嵌套的使用
- 掌握在页面中插入图片的方法
- 掌握在页面中使用超链接的方法

第二课 段落、列表，图片和超链接

在本课里，我们将一起来看看如何用 `html` 代码实现页面中的段落列表以及图片，并要学到超文本最大的一个特点——超链接，本课可以学到以下内容：

- 常用的段落级标记
- 什么是列表标记
- 列表标记的分类以及嵌套
- 网页中可以使用的图片格式
- 如何插入图片以及图片属性的设置
- 什么是超链接
- 超链接的类型
- 如何正确有效的使用超链接

理论教学

(2 课时)

2.1 段落级标记

在页面中我们通过段落级标记实现段落以将文字进行一种有规则的美观的排列。

2.1.1 段落标记<p>

所谓段落，就是一段格式上统一的文本，使用<p>标记实现。

基本语法：

```
<p>...</p>
...<p>
```

语法解释：

可以使用成对的<p>标记来包含段落，也可以单独使用<p>标记来划分段落。

段落标记对齐属性 align

段落对齐方式，指的是段落相对于文档窗口在水平位置的对齐方式。可实现水平方向上的左、中、右对齐。

基本语法：

```
<p align=value>...</p>
```

语法解释：

默认的段落对齐属性为居左。value 的值如表：

值	描述
left	居左对齐
center	居中对齐
right	居右对齐

文件范例：2-1.htm

01: <html>

02: <head>

03: <title> 段落文字</title>

04: </head>

05: <body>

06: <p>这是一本专业的 html 书籍</p>

07: <p align="center">如有问题请 e-mail:jan126@126.com</p>

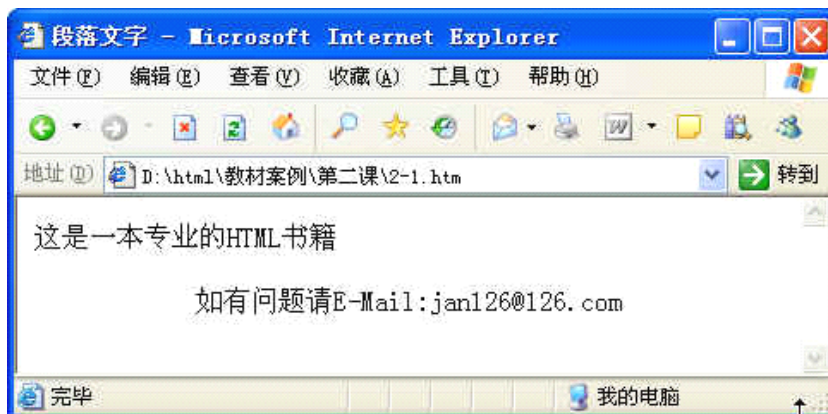
08: <body>

09: </html>

文件说明:

第 6 行设定了段落标记。第 7 行改变了段落标记的对齐方式。

显示结果:



2.1.2 换行标记

使用 br 元素在一行的任意地方实现换行，甚至在一个单词的中间。它没有结束标记。

基本语法:

```
...<br>
```

语法解释:

一个换行使用一个
，多个换行可连续使用多个
标记。

2.2 水平线标记<hr>

水平线用于段落间的分隔，使文档结构更加清晰。hr 元素和 br 元素一样易于使用。

基本语法:

```
<hr>
```

2.2.1 水平线宽度属性 width

默认情况下，水平线的宽度为 100%，我们可调整水平线的宽度。

基本语法:

```
<hr width=value>
```

语法解释:

水平线的宽度可用绝对的像素为单位，也可用相对的百分比为单

位。

2.2.2 水平线高度属性 size

基本语法:

```
<hr size=value>
```

语法解释:

水平线的高度只能用绝对的像素为单位。

2.2.3 水平线颜色属性 color

基本语法:

```
<hr color=value>
```

语法解释:

value 为颜色的英文名称或者十六进制值。

2.2.4 水平线排列属性 align

在水平方向上，可以设置水平线的居左、居中和居右对齐。

基本语法:

```
<hr align=value>
```

语法解释:

默认的水平线对齐属性为居中对齐。value 的值如表：

值	描述
left	居左对齐
center	居中对齐
right	居右对齐

水平线标记综合范例

文件范例：2-2.htm

01: <html>

02: <head>

03: <title> 水平线的使用</title>

04: </head>

05: <body>

06: <h2>文化元素</h2>

07: `<hr width=90% size=2 color=red align=left>`

08: `<p>`三心论：一般员工要有责任心；中层以上除责任心外，还要有上进心；高层要有事业心，要把企业的事业当成自己的事业。`</p>`

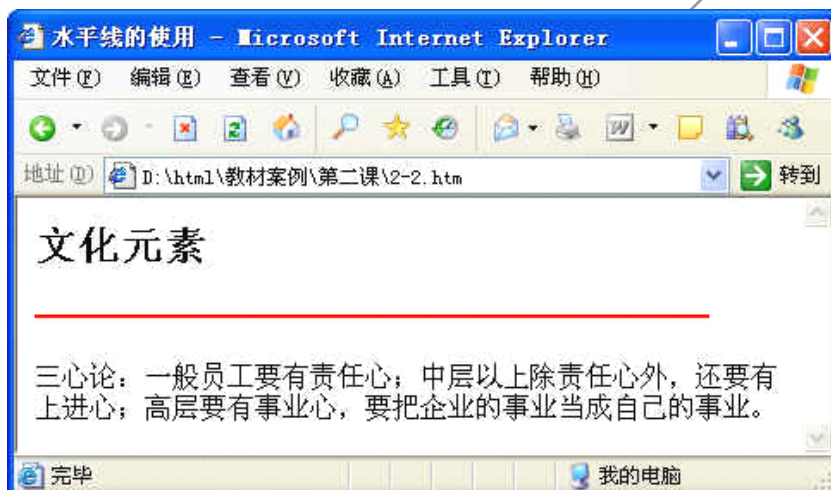
09: `</body>`

10: `</html>`

文件说明：

第 7 行设定了水平线的宽度、高度、颜色及排列方式。

显示结果：



2.3 居中标记<center>

这是一个独立使用的居中标记，可以使我们的文档居中显示。

基本语法：

```
<center>...</center>
```

语法解释：

上述标记间的文字会自动居中对齐。

2.4 列表标记

在 html 页面中，列表可以起到提纲挈领的作用。

列表分为两种类型：无序列表和有序列表。前者使用项目符号来标记无序的项目，后者则使用编号来记录项目的顺序。

列表项

不论你要创建带加重号的列表还是编号列表，都必须在列表中使用 li 元素。结束标记可选。

2.5 无序列表

所谓无序，是指以●、○、■等开头的、没有顺序的列表项目。在无序列表中，各个列表项之间没有顺序级别之分，它通常使用一个项目符号作为每条列表项目的前缀。常用无序列表主要使用、两个标记和 type 属性。

基本语法:

```
<ul>
  <li>项目一
  ...
</ul>
```

语法解释:

无序列表中，使用作为无序的声明，使用作为每个项目的起始。

2.5.1 无序列表的类型属性 type

基本语法:

```
<ul type=value>
  <li>项目一
</ul>
```

语法解释:

默认的无序列表类型为●。value 的值为：

值	描述
disc	●（实心圆）
circle	○（空心圆）
square	■（实心方形）

2.6 有序列表

有序列表是指按照数字或字母等顺序排列的列表项目。有序列表使用编号，而不是项目符号来编排项目，列表中的项目采用数字或英文字母开头，通常各列表项之间有先后的顺序性。有序列表主要使用、两个标记和 type、start 两个属性。

基本语法:

```

<ol>
<li>项目一
...
</ol>

```

语法解释:

有序列表中，使用作为有序的声明，使用作为每个项目的起始。

2.6.1 有序列表的类型属性 type

基本语法:

```

<ol type=value>
<li>项目一
</ol>

```

语法解释:

默认的有序列表类型为数字序号。value 的值为：

值	描述
1	数字 1、2、3...
a	大写字母 a、b、c...
α	小写字母 a、b、c...
i	大写罗马数字 I、II、III...
ii	小写罗马数字 i、ii、iii...

2.6.2 有序列表的类型属性 start

基本语法:

```

<ol start=value>
<li>项目一
</ol>

```

语法解释:

start: 告诉浏览器使用哪个数作为编号的开始计数，默认情况下为1。不论列表编号的类型，其取值都为数字。

2.7 列表嵌套

列表是一种可以进行嵌套的标记，下面我们就介绍列表的嵌套。

2.7.1 无序列表和有序列表的嵌套

这种嵌套类型是最常见的列表嵌套，重复使用和标记可以组合出嵌套列表。

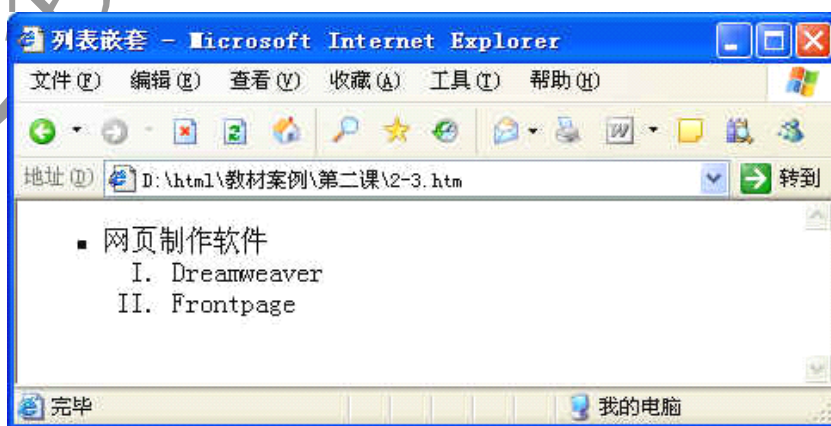
文件范例：2-3.htm

```
01: <html>
02: <head>
03: <title> 列表嵌套</title>
04: </head>
05: <body>
06: <ul type=square>
07: <li>网页制作软件
08: <ol type=i>
09: <li>dreamweaver
10: <li>frontpage
11: </ol>
12: </ul>
13: </body>
14: </html>
```

文件说明：

第6行定义了列表的第1级，第7行定义了无序列表的内容，第8-11行定义了二级列表。

显示结果：



2.8 网页常用图片格式

图像文件的格式有很多种。网页中常用的有两种格式，即 jpeg(图形

图像联合专家组)和 gif(图形接口格式)。

jpeg (joint photographic experts group)

jpeg 是专门为照片图片设计的文件格式。jpeg 支持数百万种色彩，是质量有损耗的格式。

gif (graphics interface format)

gif 是网页图像中很流行的格式，虽然它仅包括 256 种色彩，但 gif 提供了出色的、几乎没有丢失的图片压缩。并且，gif 可以包含透明区域和多帧动画。

2.8.1 图像标记的属性

使用 img 元素在页面上添加图像是很简单的事。img 元素在开始标记中包含了你所需要的一切，它没有内容和结束标记。

基本语法：

```

```

2.8.2 图片的源文件属性 src

配合 src 属性指定图片源文件所在的路径，就可以完成图片的插入了。

基本语法：

```

```

语法解释：

通过 src 属性指定路径，file_name 为要插入图片的路径。

2.8.3 图片的提示文字属性 alt

提示文字有两个作用：

如果图片下载完成，鼠标放在该图片上，鼠标旁边会出现提示文字。

如果图片没有被下载，在图片的位置上就会显示提示文字。

基本语法：

```

```

语法解释：

说明提示文字的内容中英文均可。

2.8.4 图片的宽度、高度属性 width、height

默认情况下，页面中图片的显示大小就是图片默认的宽度高度。

基本语法：

```

```

语法解释：

图片的 width 宽度和 height 高度的单位可以是像素，也可以是百分比。因为网页主要是通过屏幕显示，所以建议使用像素作为单位。

2.8.5 图片的边框属性 border

默认的图片是没有边框的，通过 border 属性可以为图片添加边框线。

基本语法：

```

```

语法解释：

value 为边框线的宽度，单位为像素。

2.8.6 图片的垂直间距属性 vspace

vspace 属性用来调整图像和文字之间的上下距离，其默认单位为像素。

基本语法：

```

```

语法解释：

value 为图片垂直方向上和文字的距离，单位为像素。

2.8.7 图片的水平间距属性 hspace

hspace 属性用来调整图像和文字之间的左右距离，其默认单位为像素。

基本语法：

```

```

语法解释：

value 为图片水平方向上和文字的距离，单位为像素。

2.8.8 图片的排列属性 align

图片和文字之间的排列通过 align 属性来设定。图片的绝对对齐方式和相对文字对齐方式是不一样的，绝对对齐方式的效果和文字的对齐一样，只有 3 种：居左、居右、居中；而相对文字的对齐方式是指图片与一行文字的相对位置的对齐。

- ◆ “基线”（baseline）、“底部”（bottom）、“绝对底部”（absolute bottom）效果一样，是指图片底端与文字底端对齐。
- ◆ “顶端”（top）、“文本上方”（texttop）方式是指图片顶端与文字最高字符的顶端对齐。
- ◆ “中间”（middle）方式是指图片的中间线与文字底端对齐。
- ◆ “绝对中间”（absolute middle）是指图片的中间线与文字的中间线对齐。“绝对中间”对齐方式用处比较多，如文字前有小图标，那么图标应该使用“绝对中间”对齐方式。

基本语法：

```

```

语法解释：

默认的排列方式为图片在文字的左侧。常用 value 的值为：

属性值	描述
top	文字的中间线位于图片上方
middle	文字的中间线位于图片中间
bottom	文字的中间线位于图片底部
left	图片在文字的左侧
right	图片在文字的右侧
absbottom	文字的底线位于图片底部
absmiddle	文字的底线位于图片中间
baseline	英文文字基准线对齐
texttop	英文文字上边线对齐

图片标记综合范例

文件范例：2-4.htm

```
01: <html>
02: <head>
03: <title> 图片的标记使用</title>
04: </head>
```

05: <body>

06:

07: 高级人才的六个标准：一是共同信念和价值观标准；二是忠诚与牺牲精神；三是审时度势，独当一面的指挥能力；四是搭班子、建队伍的管理能力；五是团结多数，使集体成员通力合作的协调能力；六是孜孜不倦、吐故纳新的学习能力。<p>

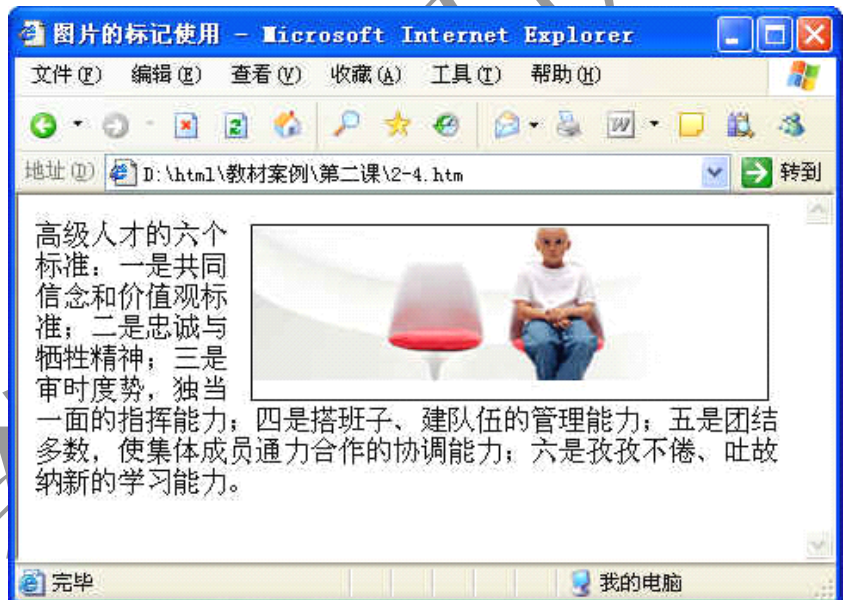
08: </body>

09: </html>

文件说明：

第6行设定了图片的来源、宽度和高度、垂直间距和水平间距、边框以及图片和文字间的排列方式。

显示结果：



2.9 什么是超链接

超链接（hyperlink）又称超文本链接，是指通过使用超文本标记语言编辑包含标记指令的文本文件，在两个不同的文档或同一文档的不同部分建立联系，从而使访问者可以通过一个网址访问不同网址的文件或通过一个特定的栏目访问同一站点上的其他栏目。

2.10 链接的标记<a>

<a>称链接标记，由 <a> 与 所围的文字、图片等等可以成为

一个链接。

基本语法:

```
<a href="url" name="value" target="value">
.....
</a>
```

语法解释:

href: 链接目标的 url, 这属性不能与另一属性 name 同时使用。

name: 为文件埋下锚点, 不会被显示。

target: 设定链接被按了之后所要显示的窗口。属性值为:

- ◆ **_blank** 将链接的内容打开在新的浏览器窗口中。
- ◆ **_parent** 将链接的内容当成文件的上一个画面。
- ◆ **_self** 将链接的内容显示在当前窗口中。(默认值)
- ◆ **_top** 将框架中链接的画面内容, 显示在没有框架的视窗中。(即除去了框架)
- ◆ **框架名称** 运用于框架网页中, 若设定则链接结果将显示在"框架名称"的框窗中, 框窗名称是事先由框架标记命名。

2.11 链接的类型

下面我们就几种最常用的来进一步分析。

2.11.1 内部链接

内部链接是指网站内部文件之间的链接, 通常有下面两种情况:

- 1、链接到相同目录中的网页
- 2、链接到其他目录中网页

要实现内部链接就是在 href 的值中写入需要链接文件的相对路径即可。

2.11.2 外部链接

外部链接是指与其它网站文件之间的链接。这个时候 href 的值中写入的则是其它网站文件的绝对路径。

基本语法:

```
<a href="http://www.thinkbank.com.cn/default.aspx">思远</a>
```

语法解释:

<http://www.thinkbank.com.cn/default.aspx> 就是思远网站主页的绝对路径。

在现在大多数浏览器地址栏中，实际上并不需要输入协议信息 ([http://](#))，简单地输入 [www.thinkbank.com.cn](#) 就可以在显示器上得到思远的主页。这是为什么呢？首先，浏览器假定你正在使用 [http](#) 协议。其次，如果你没有告诉服务器你希望得到目录中的哪个页面，服务器会缺省地传送 [index.htm](#) 或 [default.htm](#) 等，这是由网站服务器设置的。你可以把任意页面设置为服务器缺省传送的页面。

但是超链接的绝对路径，你必须给出协议名。所以，下面这个例子将不能工作：

```
<a href=" www.thinkbank.com.cn" >思远主页</a>
```

2.11.3 邮件链接

基本语法：

```
<a href="mailto:info@mymail.com">联系我们</a>
```

语法解释：

[info@mymail.com](#) 可以替换为任意一个电子邮件地址。

2.11.4 书签链接

当一个页面很长，客户又需要浏览特定内容，我们则应该建立书签链接。

建立书签链接分两个步骤：

1、创建锚点

基本语法：

```
<a name="chapter1">第一章</a>
```

语法解释：

在[第一章](#)处加入了一个名为 [chapter1](#) 的锚点。它不会改变[第一章](#)三个字的样式。

2、链接到锚点

基本语法：

```
<a href="#chapter1">第一章</a>
```

语法解释：

现在 [href](#) 中写的值不再是路径，而是 <#> 加上锚点的名称，这样就形成了一个书签链接。

在浏览器中我们看不到已命名的锚点，但如果单击超链接“[第一](#)

章”，就可以跳转到设置锚点的第一章位置。

超链接综合范例

文件范例：

在同一文件夹下创建如图的目录结构：



content 目录下面创建文件 4-1-1.htm、4-1-2.htm。

4-1.htm:

```
01: <html>
02: <head>
03: <title> 思远教育</title>
04: </head>
05: <body>
06: <h1 align="center"> 思远教育</h1>
07: <h3><a href="#sk">千人工程</a></h3>
08: <h3><a href="#2n">双 n 计划</a></h3>
09: <h3><a href="#2030">2030</a></h3>
10: <h4><a name="sk">千人工程</a></h4>
11: <font size="2" color="#cc00ff"> “千人工程” 旨在构建一个大型、
    高质量的集软件产业政策研究 …… [<a href="content/4-1-1.htm"
    target="_blank">详细</a>]</font>
12: <h4><a name="2n">双 n 计划</a></h4>
13: <font size="2" color="#cc00ff"> “双 n 计划” 按照国际先进的职业
    教育模式，培养掌握系统的网 …… [<a href="content/4-1-2.htm"
    target="_blank">详细</a>]</font>
14: <h4><a name="2030">2030</a></h4>
```

15: “2030” 就是欧美软件外包专才培养计划。学员在软件外包企业的……[详细]

16: </body>

17: </html>

4-1-1.htm:

01: <html>

02: <head>

03: <title>千人工程::思远教育</title>

04: </head>

05: <body>

06: <h1 align="center">千人工程</h1>

07: 千人工程旨在构建一个大型、高质量的集软件产业政策研究、人才培养、企业实习、输送就业于一体的综合性软件人才培育平台。

08: <h3>双 n 计划</h3>

09: <h3>2030</h3>

10: </body>

11: </html>

4-1-2.htm:

01: <html>

02: <head>

03: <title>双 n 计划::思远教育</title>

04: </head>

05: <body>

06: <h1 align="center">双 n 计划</h1>

07: 双 n 计划按照国际先进的职业教育模式，培养掌握系统的网络技术的实用型、工程化网络人才。

08: <h3>千人工程</h3>

09: <h3>2030</h3>

10: </body>

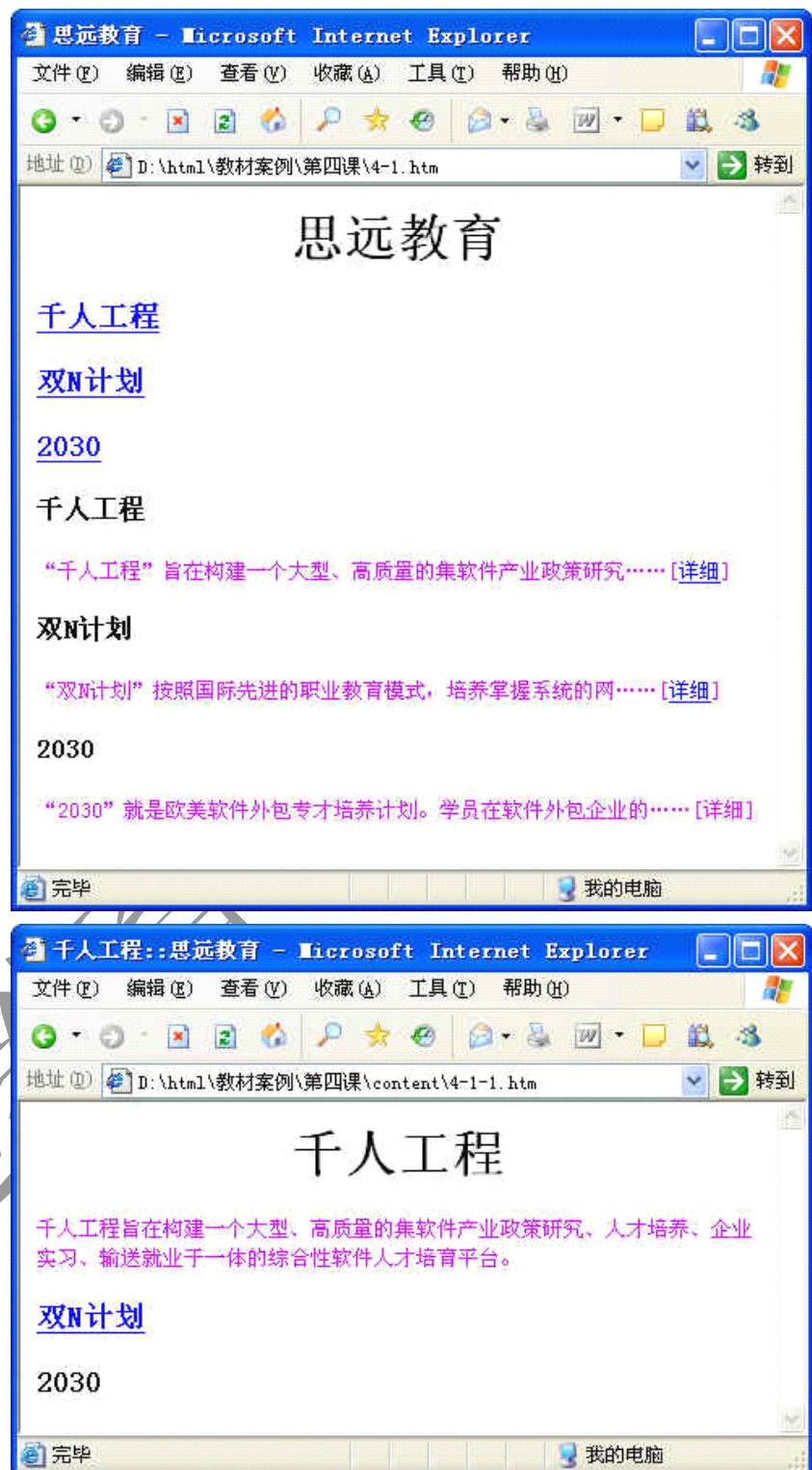
11: </html>

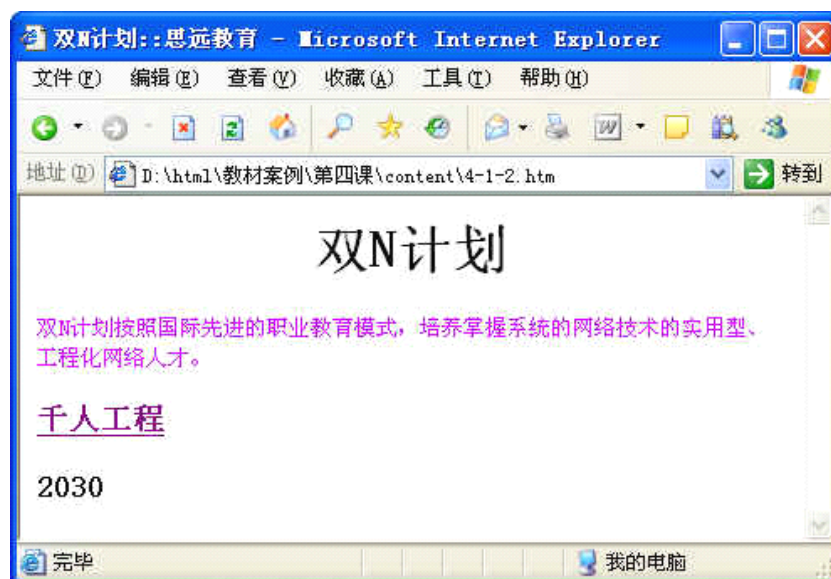
文件说明:

文件 4-1.htm 中第 10、12、14 行建立了三个书签，第 07、08、09 行添加了三个书签链接，第 11、13 行添加了两个内部链接并在新窗口打开。

文件 4-1-1.htm 和 4-1-2.htm 中第 08 行分别添加了一个在本窗口打开的内部链接。

显示效果:





课后自测

一、填空题

1. 段落标记是_____，换行标记是_____。
2. 网页中的列表分为_____和_____。
3. 网页中可以使用的图片格式有_____、_____，在网页中使用_____标记插入图片。
4. 无序列表的列表项的 type 属性值有_____、_____、_____。
5. 有序列表的列表项的 type 属性值有_____、_____、_____。
6. 超链接是_____。
7. 超链接的类型有_____、_____、_____。
8. 内部链接常用_____路径，外部链接常用_____路径。

二、问答题

1. 采用列表嵌套把第三课的内容结构显示出来。
2. 如何创建一个书签链接？

上机指南

(2 课时)

思远教育

上机任务书

任务背景：

现在需要在上一个单元的上机中完成的页面里调整段落格式，用列表实现文章的结构图并用超链接使其具有导航功能。效果图如下：



任务目标：

1. 文章的结构是一个列表的嵌套；
2. 在列表上加入相应的链接。

上机指导书

1. 在桌面建立文件夹 test;
2. 进入 test 并新建文本文件，将其重命名为 index.htm，注意要将扩展名要从 TXT 改为 HTM;
3. 在 test 下建立文件夹 images，并将图片 harry.jpg 拷贝到里面;
4. 用记事本将刚建立的文件 index.htm 打开，并在其中输入代码。
5. 以下是关键代码:

.....

```
<hr>
```

```
<ul type="disc">
```

```
  <li><a href="001.htm">第一章 猫头鹰的信</a>
```

```
    <ol type="A">
```

```
      <li><a href="#A">哈利。波特是一个在很多方面都非常不寻常的男孩...</a>
```

```
      <li><a href="#B">哈利那时恰好在旁边，他听到罗恩的声音时简直呆住了...</a>
```

```
      <li><a href="#C">从窗口飞进来的是三只猫头鹰，其中两只是托着第三只的，它看上去受  
      伤了...</a>
```

```
    </ol>
```

```
  <li><a href="002.htm" target="_blank">第二章 玛各姨妈闯的祸</a>
```

```
  <li><a href="003.htm" target="_blank">第三章 爵士巴士</a>
```

```
  <li><a href="004.htm" target="_blank">第四章 在例格歌注里</a>
```

```
</ul>
```

```
<hr>
```

.....

课后学习

课后练习

在上一个单元的课后练习中加入部分元素：

- i. 嵌入背景音乐
- ii. 将“创建优秀网页的6个好习惯”制作成滚动字幕
- iii. 用列表建立内容的结构

创建优秀网页的6个好习惯

【日期】2005-1-13 【人气】176 【作者】『一分流水』 【来源】『一分流水』

1、不要将链接全部放在图像中

以往的网页中的链接都是文字，而且用下划线以资识别。但不久人们就讨厌了这些文字链接，于是都采用了图像链接。后来，更加嫌一个图像一个链接太麻烦了，于是把所有的链接都放进了一个图像中去。但并不是所有的用户都打开图像载入，如果你把所有的链接都放入图像中去，而没有任何同样功能的文字链接的话，用户也许会不知所措。

2、慎用自动滚动

曾经到过内容“丰富”的网页，它使用了很多常见的网页制作技巧：现在的时间、访问时的时间等等。不过有一个用得不是很好，就是窗口的自动滚动。我只是想发个邮件给网页的主人，但我却无法使滚动条停下来让我点击链接，于是我花了很久才“命中”窗口中的E-mail链接——真后悔军训的时候没有练好射击！

3、不要用延时效果的JavaScript

现在的网页制作者为了使网页更加具有影响力，都疯狂地加入JavaScript。以现在的“猫”奔跑的速度而言，这些脚本也算不了什么，用户不会介意。但是，如果这些脚本可能导致延时的话，用户的态度就不同了。

某个国人网页的首页在载入时窗口的底色——窗口里面没有任何内容——由白色逐渐变成黑色，整个过程耗费了几秒，嘿！别小看这几秒，我能检查全部的邮箱了！

4、不要依赖浏览器

太多的网站首页中加入了“请在800×600分辨率下使用IE浏览器浏览本网站”等字样的文字。我是安装了IE、NC和Opera，因为我要制作网页，需要它们来检验网页在这些浏览器下的可读性。但一般用户不会安装那么多的浏览器，更加不会牢记着哪个网站需要用哪个浏览器浏览。

也许不可能让网页在使用所有的浏览器浏览都一样的效果，但尽量地去达到该目的，会让网页在这个浏览器下有这样的特色、在另一个浏览器下又有不同的特色，这是网页制作者的工作，不是吗？

5、不要忘记版权

现在，网上侵犯知识产权的问题非常猖獗，某免费个人网页提供商上的网页尤其严重——甚至有盗版光盘贩卖呢！为此一些新的免费个人网页提供商就要求用户在申请时提交详细的、真实的个人资料，以防备不可预料的事情发生。

所以，如果你的网页需要引用别人的东西的话，要么你就和他联系一下，征求意见；要么就在明显的位置注明其版权归属。这并非多余，这是你的网页“长寿”的秘方！

当然，如果是你自己制作的图像或自己撰写的文章，你绝对有权声明版权归你自己所有！

6、不要懒于更新

某个个人网页聚居地上有很多的网页都是很久前建立而至今没有更新过的，姑且不说这是浪费公用资源的行为，访问过这些网页的人看到网页主人如此懒惰，你说他还会再去访问吗？

谁能确定下次访问不会又是这般模样呢？因特网上的资源如此浩瀚，谁会去访问一个从不更新的网页呢？

【评论】 【打印】

相关评论

· 暂时没有评论

[查看更多评论或参与评论](#)

:: 本站荣誉 :: 联系站长 :: 友情链接 :: 版权声明 :: 网站纪事 :: 关于我们 ::

小结

在这个单元里面我们学了以下的内容：

1. 段落标记
2. 列表标记
3. 图片的使用
4. 超链接的使用

这里面重点要掌握的是段落标记，图片的使用和滚动字幕的使用。因为这些内容在后面会经常的使用到，譬如<P>、
标记就是会经常使用的，而图片又是一个网站所不可或缺的。对于超链接，需要重点掌握的就是其设置，因为超链接可以说是网站的灵魂，其设置合理与否直接影响到浏览者对网站的评价，一个好的网站的会让人觉得总能很轻松的找到自己所需要的资源，而不是在你的网站里面走迷宫。

Unit

3

软件开发过程

本单元由下列部分组成：

第一节 软件工程瀑布模型

在完成本单元学习后，你将学会：

- 什么是软件工程
- 什么是瀑布模型
- 如何利用瀑布模型开发网站

第三课 软件开发过程

在本课里，我们将一起来看看现在一款软件的开发过程，你将学到以下内容：

- 什么是软件危机
- 什么是软件工程
- 什么是瀑布模型
- 如何使用瀑布模型进行开发

理论教学

(2 课时)

3.1 软件的发展和软件危机

自 20 世纪 40 年代中出现了世界上第一台计算机以后，就有了程序的概念。其后经历了几十年的发展，计算机软件经历了三个发展阶段：

- ◆ 程序设计阶段，约为 50 至 60 年代
- ◆ 程序系统阶段，约为 60 至 70 年代
- ◆ 软件工程阶段，约为 70 年代以后

几十年来最根本的变化体现在：

1、人们改变了对软件的看法。50 年代到 60 年代时，程序设计曾经被看做是一种任人发挥创造才能的技术领域。当时人们认为，写出的程序只要能在计算机上得出正确的结果，程序的写法可以不受任何约束。随着计算机的广泛使用，人们要求这些程序容易看懂、容易使用，并且容易修改和扩充。于是，程序便从个人按自己意图创造的“艺术品”转变为能被广大用户接受的工程化产品。

2、软件的需求是软件发展的动力。早期的程序开发者只是为了满足自己的需要，这种自给自足的生产方式仍然是其低级阶段的表现。进入软件工程阶段以后，软件开发的成果具有社会属性，它要在市场中流通以满足广大用户的需要。

3、软件工作的范围从只考虑程序的编写扩展到涉及整个软件生存周期。

在软件技术发展的第二阶段，随着计算机硬件技术的进步，要求软件能与之相适应。然而软件技术的进步一直未能满足形势发展提出的要求。致使问题积累起来，形成了日益尖锐的矛盾。这就导致了软件危机。

软件危机：指在计算机软件的开发和维护过程中所遇到的一系列严重问题。这些问题表现在以下几个方面：

- ◆ 用户对开发出的软件很难满意。
- ◆ 软件产品的质量往往靠不住。
- ◆ 一般软件很难维护。
- ◆ 软件生产效率很低。
- ◆ 软件开发成本越来越大。
- ◆ 软件成本与开发进度难以估计。
- ◆ 软件技术的发展远远满足不了计算机应用的普及与深入的需要。

3.2 软件工程

许多计算机和软件科学家尝试，把其它工程领域中行之有效的工程学知识运用到软件开发工作中来。经过不断实践和总结，最后得出一个结论：按工程化的原则和方法组织软件开发工作是有用的，是摆脱软件危机的一个主要出路。

3.2.1 软件工程的定义

Fritz Bauer 曾经为软件工程下了定义：“软件工程是为了经济地获得能够在实际机器上有效运行的可靠软件而建立和使用的一系列完善的工程化原则。”

1983 年 IEEE (The Institute of Electrical and Electronics Engineers, 美国电气与电子工程师学会) 给出的定义为：“软件工程是开发、运行、维护和修复软件的系统方法”，其中，“软件”的定义为：计算机程序、方法、规则、相关的文档资料以及在计算机上运行时所必需的数据。

后来尽管又有一些人提出了许多更为完善的定义，但主要思想都是强调在软件开发过程中需要应用工程化原则的重要性。

3.2.2 软件工程的要素

软件工程包括三个要素：方法、工具和过程。

软件工程方法：为软件开发提供了“如何做”的技术。它包括了多方面的任务，如项目计划与估算、软件系统需求分析、数据结构、系统总体结构的设计、算法过程的设计、编码、测试以及维护等。

软件工具：为软件工程方法提供了自动的或半自动的软件支撑环境。目前，已经推出了许多软件工具，这些软件工具集成起来，建立起称之为计算机辅助软件工程(CASE)的软件开发支撑系统。CASE 将各种软件工具、开发机器和一个存放开发过程信息的工程数据库组合起来形成一个软件工程环境。

软件工程的过程：是将软件工程的方法和工具综合起来以达到合理、及时地进行计算机软件开发的目的。过程定义了方法使用的顺序、要求交付的文档资料、为保证质量和协调变化所需要的管理、及软件开发各个阶段完成的里程碑。

3.2.3 软件工程的目标与主要环节

软件工程的目标：提高软件的质量与生产率，最终实现软件的工业化生产。

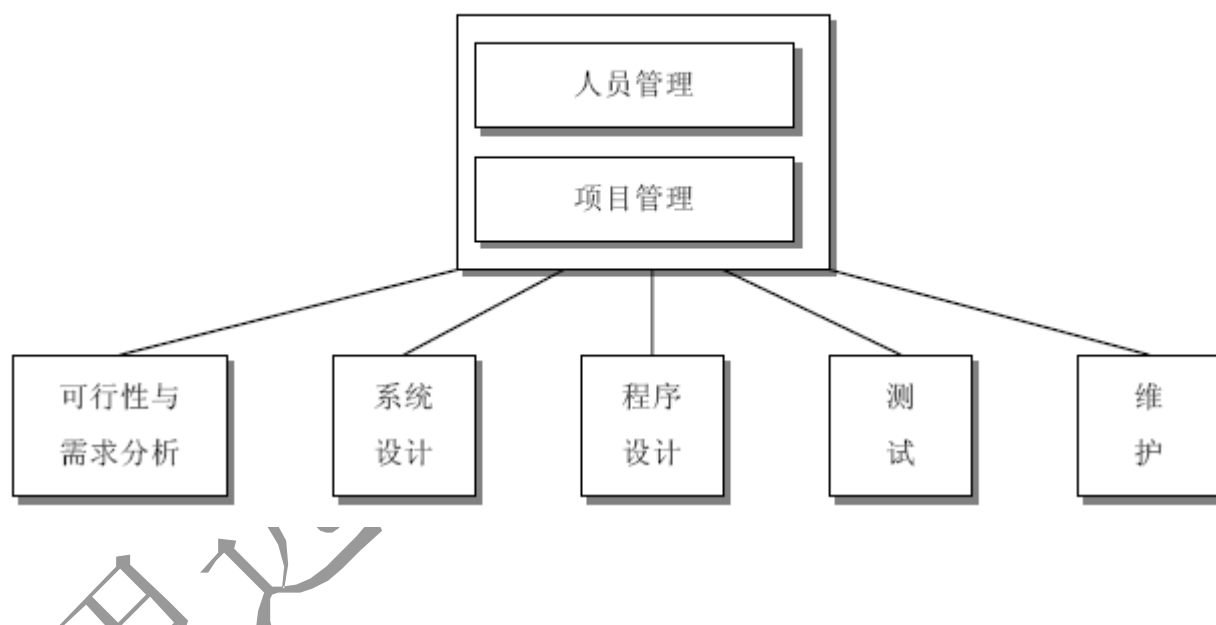
对开发人员而言，如果非得在质量与生产率之间分个主次不可，那么应该是质量第一，生产率第二。这是因为：

1、质量直接体现在软件的每段程序中，高质量自然是开发人员的技术追求，也是职业道德的要求。

2、高质量对所有的用户都有价值，而高生产率只对开发方有意义。

3、如果一开始就追求高生产率，容易使人急功近利，留下隐患。宁可进度慢些，也要保证每个环节的质量，以图长远利益。

软件工程的主要环节：人员管理、项目管理、可行性与需求分析、系统设计、程序设计、测试、维护等，如图所示。



3.3 软件工程过程和软件生存周期

3.3.1 软件工程过程 (Software Engineering Process)

软件工程过程：为获得软件产品，在软件工具支持下由软件工程师完成的一系列软件工程活动。

软件工程过程通常包含四种基本的过程活动：

- ◆ **P (Plan)** ：软件规格说明。规定软件的功能及其运行的限制；
- ◆ **D (Do)** ：软件开发。产生满足规格说明的软件；

- ◆ **C (Check)**：软件确认。确认软件能够完成客户提出的要求；
- ◆ **A (Action)**：软件演进。为满足客户的变更要求，软件必须在使用使用过程中演进。

事实上，软件工程过程是一个软件开发机构针对某一类软件产品为自己规定的工作步骤，它应当是科学的、合理的，否则必将影响到软件产品的质量。

3.3.2 软件生存周期 (life cycle)

软件生存周期：从软件定义、开发、使用、维护到淘汰的全过程。

软件生存周期的六个步骤：

- ◆ **制定计划**：确定要开发软件系统的总目标，给出它的功能、性能、可靠性以及接口等方面的要求；研究完成该项软件任务的可行性，探讨解决问题的可能方案；制定完成开发任务的实施计划，连同可行性研究报告，提交管理部门审查。
- ◆ **需求分析**：对待开发软件提出的需求进行分析并给出详细的定义。编写出软件需求说明书及初步的用户手册，提交管理机构评审。
- ◆ **软件设计**：把已确定了的各项需求转换成一个相应的体系结构。进而对每个模块要完成的工作进行具体的描述。编写设计说明书，提交评审。
- ◆ **程序编写**：把软件设计转换成计算机可以接受的程序代码。
- ◆ **软件测试**：在设计测试用例的基础上检验软件的各个组成部分。
- ◆ **运行 / 维护**：已交付的软件投入正式使用，并在运行过程中进行适当的维护。

3.3.3 软件生存周期模型

软件生存周期模型是从软件项目需求定义直至软件经使用后废弃为止，跨越整个生存周期的系统开发、运作和维护所实施的全部过程、活动和任务的结构框架。主要有瀑布型、渐增型和变换型。

1、瀑布型：按照软件生存周期的划分依次实施，每一个阶段有明确规定的任务。它的特点：

- ◆ 各个阶段的顺序性和依赖性；
- ◆ 划分逻辑设计与物理设计，尽可能推迟程序的物理实现；
- ◆ 每个阶段必须完成规定的文档，对其中问题通过复审及早发

现，及早解决。

2、渐增型方法及特点：

- ◆ 从部分需求出发，先建立一个不完全的系统，通过测试运行该系统取得经验和信息反馈，加深对软件需求的理解，进一步使系统扩充和完善。如此反复，直至软件人员和用户对所设计完成的软件系统满意为止。
- ◆ 在渐增型开发下的软件是随软件开发的过程而逐渐形成的。
- ◆ 渐增型开发方法适合于知识型软件的开发，设计系统时对用户需求的认识开始不是很清楚的，需要在开发过程中不断认识、不断获得新的知识去丰富和完善系统。多数研究性质的试验软件，一般采用此方法。

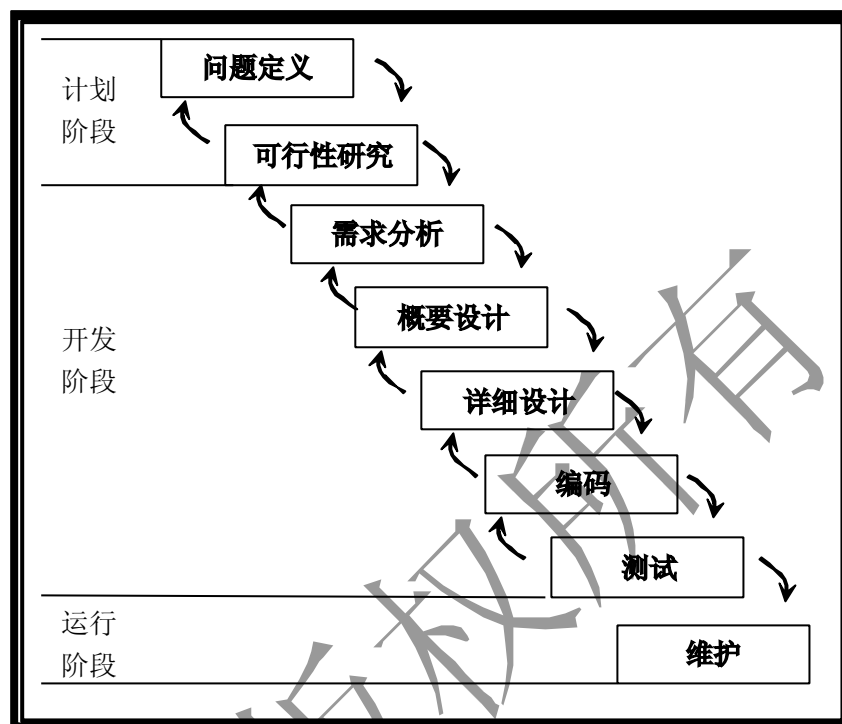
3、变换型方法及特点：

- ◆ 从软件需求的形式化规格说明出发，经过一系列的程序变换，得到最终的程序系统。
- ◆ 该方法必须有严格的数学理论和形式化技术的支持。

3.4 瀑布模型

瀑布模型是最早出现的软件工程模型又称线性模型。“线性”是人们最容易掌握并能熟练应用的思想方法。一个软件系统的整体可能是复杂的，而单个子程序总是简单的，可以用线性的方式来实现，否则干活就太累了。线性是一种简洁，简洁就是美。当我们领会了线性的精神，就不要再呆板地套用线性模型的外表，而应该用活它。

3.4.1 瀑布模型的流程图



软件开发的实践表明，上述各项活动之间并非完全是自上而下，呈线性图式。实际情况是，每项开发活动均处于一个质量环（输入-处理-输出-评审）中。只有当其工作得到确认，才能继续进行下一项活动，在图中用向下的箭头表示；否则返工，在图中由向上的箭头表示。

3.4.2 计划阶段

计划阶段的主要任务是分析用户的基本需求，分析新系统应设定的目标，按设定的目标要求进行问题定义并分析开发该系统的可行性。

问题定义

问题定义的任务：将用户提出的要求具体化、量化；确定研制系统的范围，明确研制的边界。

问题定义阶段的工作：

- ◆ 通过调查研究，了解系统需求；
- ◆ 确定系统的功能需求、性能需求、可靠性需求、安全及保密性、资源、开发费用及开发进度等的需求；
- ◆ 问题定义阶段的产品--系统目标与范围说明书。

可行性研究

可行性研究目的：确定在问题定义中所提出的问题是否值得去解，在限制条件下，问题能否解决。

可行性研究的任务：

- ◆ 进一步分析和澄清问题的定义，在澄清问题的基础上，导出系统的逻辑模型；
- ◆ 从系统逻辑模型中，选择问题的若干种主要解法，研究每一种解法的可行性，为以后的行动提出建议；
- ◆ 如果问题没有可行的解，建议停止系统开发；如果问题有可行的解，应该推荐一个较好的解决方案，并为工程制定一个初步的计划。

可行性研究包括：

- ◆ 技术可行性：现有技术能否实现本系统，现有技术人员能否胜任，开发系统的资源能否满足；
- ◆ 经济可行性：经济效益是否超出开发成本；
- ◆ 操作可行性：系统操作在用户内部行得通吗？
- ◆ 法律可行性：新系统开发是否会侵犯他人、集体或国家利益，是否违反国家法律。

可行性研究的步骤：

- ◆ 复查系统的规模和目标；
- ◆ 研究目前正在使用的系统，总结现有系统的优劣，提出新系统的雏形；
- ◆ 导出新系统的高层逻辑模型；
- ◆ 推荐建议方案；
- ◆ 推荐行动方针；
- ◆ 书写计划任务书(可行性报告)；
- ◆ 提交审查。

可行性研究报告的主要内容：

- ◆ 系统概述：说明开发的系统名称,提出单位和开发单位。
- ◆ 可行性研究的前提:系统目标；要求；约束和限制；可行性研究的基本准则等。
- ◆ 对现有系统的分析:处理流程，图示说明现有系统的处理流程和数据流程；现有系统存在的问题。
- ◆ 系统需求:主要功能；主要性能及其要求；操作要求；信息要求；限制性要求。
- ◆ 建议系统:系统目标；处理流程；系统结构,功能,性能；系统技

术可行性；投资和效益分析；操作可行性；法律可行性。

- ◆ 其它可选方案:与国内外同类型方案的比较；提出一两个可行性方案供论证和探讨。
- ◆ 制定下一阶段的预算。
- ◆ 结论性意见：由用户方、设计方和投资方共同签署意见。

3.4.3 开发阶段

经过计划阶段，确定了需要开发软件后，在这个阶段主要完成软件的开发。

需求分析

需求分析的基本任务：准确定义未来系统的目标，确定为了满足用户的需要系统必须做什么。

需求分析的工作，可以分成以下四个方面：

◆ 问题识别：

首先系统分析人员要确定对目标系统的综合要求，即软件的需求。并提出这些需求实现条件，以及需求应达到的标准。

这些需求包括功能需求、性能需求、环境需求、可靠性需求、安全保密要求、用户界面需求、资源使用需求、软件成本消耗与开发进度需求，并预先估计以后系统可能达到的目标。

此外，还需要注意其它非功能性的需求。如针对采用某种开发模式，确定质量控制标准、里程碑和评审、验收标准、各种质量要求的优先级等，以及可维护性方面的需求。

◆ 分析与综合

分析员必须从信息流和信息结构出发，逐步细化所有的软件功能，找出系统各元素之间的联系、接口特性和设计上的限制，判断是否存在因片面性或短期行为而导致的不合理的用户要求，是否有用户尚未提出的真正有价值的潜在要求。剔除其不合理的部分，增加其需要部分。最终综合成系统的解决方案，给出目标系统的详细逻辑模型。

◆ 编制需求分析阶段的文档

已经确定下来的需求应当得到清晰准确的描述。通常我们把描述需求的文档叫做软件需求说明书。同时，为了确切表达用户对软件的输入输出要求，还需要制定数据要求说明书及编写初步的用户手册。

◆ 需求分析评审

作为需求分析阶段工作的复查手段，应该对功能的正确性、文档的

一致性、完备性、准确性和清晰性，以及其它需求给予评价。为保证软件需求定义的质量，评审应以专门指定的人员负责，并按规程严格进行。评审结束应有评审负责人的结论意见及签字。除分析员之外，用户 / 需求者，开发部门的管理者，软件设计、实现、测试的人员都应当参加评审工作。

总体设计

到这个时候，系统需要做什么已经很清楚，目前的任务是解决系统如何做，即系统的功能如何实现。

总体设计的主要任务：完成软件结构的设计,确定系统的模块及其模块之间的关系。

模块：数据说明、可执行语句等程序对象的集合，可以单独命名且可通过名字来访问。模块具有输入和输出(参数传递)、功能、内部数据结构(局部变量)和程序代码四个特性。概要设计主要考虑输入、输出(参数传递)和功能两个特性。

任务具体分为两个阶段：

系统设计：确定系统的具体实现方案

结构设计：确定软件结构

从开发规模和开发成本上考虑，有三种可能的方案：

- ◆ 低成本的解决方案，系统只能完成最必要的工作。
- ◆ 中等成本的解决方案，系统不仅能够很好地完成预定的任务，使用起来很方便，而且可能还具有用户没有具体指定的某些功能和特点。虽然用户没有提出这些具体要求，但是系统分析员根据自己的知识和经验断定，这些附加是应该有的，或是不能少的。
- ◆ 高成本解决方案，系统具有用户可能希望有的所有功能和特点。

详细设计

详细设计的目的：为软件结构图中的每一个模块确定采用的算法和块内数据结构,用某种选定的表达工具给出清晰的描述。

详细设计的主要任务：

- ◆ 为每一个模块确定采用的算法，选择某种适当的工具表达算法的过程,写出模块的详细过程描述。
- ◆ 确定每一模块使用的数据结构。

- ◆ 确定模块结构的细节,包括对系统外部的接口和用户界面,对系统内部其它模块的接口,以及关于模块输入数据、输出数据及局部数据的全部细节。
- ◆ 为每一个模块设计出一组测试用例,以便在编码阶段对模块代码(即程序)进行预定的测试。

编码

编码的任务：使用选定的程序设计语言，把模块的过程性描述翻译为用语言书写的源程序(源代码)。

源程序要求：正确可靠、简明清晰、效率高。

- ◆ 源程序的正确性是对程序质量的最基本要求；
- ◆ 源程序的简明清晰，便于验证源代码和模块规格说明的一致性，容易进行测试和维护；
- ◆ 对于大多数模块，编码时应该把简明清晰放在第一位；
- ◆ 除了编码阶段产生源代码外，在测试阶段也需要编写一些测试程序，用于对软件的测试。

选择程序设计语言需要考虑的因素：

- ◆ 选择用户熟悉、便于用户维护的语言。
- ◆ 选择目标系统的环境中可以提供的编译程序所能选用的语言。
- ◆ 选择可以得到的软件工具,能支持程序开发中可以利用的语言。
- ◆ 根据工程规模的大小、目标系统应用范围。
- ◆ 选择程序员熟悉的语言。
- ◆ 选择标准化程度高、程序可移植性好的语言。
- ◆ 根据算法与计算的复杂性、数据结构的复杂性选择。
- ◆ 根据实时要求系统需要的响应速度和效率选择相应的语言。

测试

软件测试的基本任务：按照特定的规则，发现软件错误的过程。

- ◆ 好的测试方案是尽可能发现迄今尚未发现错误的测试
- ◆ 成功的测试方案是发现迄今尚未发现错误的测试

软件测试的方法：

◆ 黑盒测试：

黑盒测试也称为功能测试,它着眼于程序的外部特征,而不考虑程序的内部逻辑结构。测试者把被测程序看成一个黑盒,不用关心程序的内部结构。黑盒测试是在程序接口处进行测试,它只检查程序功能是否能按

照规格说明书的规定正常使用，程序是否能适当地接收输入数据产生正确的输出信息，并且保持外部信息(如数据库或文件)的完整性。

◆ **白盒测试：**

白盒测试又称代码级测试。测试者了解被测程序的内部结构和处理过程，对程序的所有逻辑路径进行测试，不同点检查程序状态，确定实际状态与预期状态是否一致。

软件测试的一般步骤：单元测试、子系统测试、系统测试、验收测试、平行测试。

3.4.4 运行阶段

运行阶段主要进行软件的维护。因为软件的开发过程中，一般很难检测到所有的错误，其次软件在应用过程中需要随用户新的要求或运行环境的变化而进行软件的修改或完成功能的增删等，为了提高软件的应用水平和使用寿命，软件的维护是不可避免的。

软件的维护的分类：

- ◆ 校正性维护：满足用户对已开发产品的性能与运行环境不断提高的要求，进而达到延长软件寿命的目的。
- ◆ 适应性维护：对程序使用期间发现的程序错误进行诊断和改正的过程，配合变化了的环境进行修改软件的活动；
- ◆ 完善性维护：满足用户在使用过程中提出增加新的功能或修改已有功能的建议而进行的工作；
- ◆ 预防性维护：为了改善未来的可维护性或可靠性而修改软件的工作。

影响软件维护的因素：

开发方法：

- ◆ 采用模块化详细设计文档有助于理解软件的结构、界面功能和内部流程；
- ◆ 开发过程中严格而科学的管理规划及清晰可靠的文档资料对发生错误后的理解与纠错是至关重要的；
- ◆ 开发过程中模块的独立程度越高，对软件修改越容易，对软件的改进和移植越方便。

开发条件：

- ◆ 软件开发及维护人员的水平决定了软件开发的质量和维护的效率；
- ◆ 开发过程中使用标准的程序设计语言和标准的操作系统接口，可以大大提高软件的可维护性；

- ◆ 在测试过程中用例的有效性，可极大地减少软件存在的错误；其次使用规范化的文档资料可为维护提供更好的依据。

决定软件可维护性的因素：

- ◆ 软件的可理解性、可测试性、可修改性；
- ◆ 文档描述符合要求、用户文档简洁明确、系统文档完整并且标准。

思远教育版权所有

课后自测

请完成课后自测题：

课后自测是对学员学习进度的检查，帮助学员找出对本课知识点理解的不足。请学员在理论授课完成后立即完成下列习题，对于不能准确回答或有疑问的问题，请即刻向授课教员提出。

一、 填空题

1. 软件危机是_____。
2. 软件工程的要素为_____、_____和_____。
3. 软件工程过程包含的四个基本过程：_____、_____、_____、_____。
4. 软件生存周期的步骤有：_____、_____、_____、_____、_____。
5. 瀑布模型是：_____。

二、 问答题

1. 简述瀑布模型的流程。
2. 简述计划阶段在软件开发过程中的意义。

上机指南

(2 课时)

思远教育

上机任务书

任务背景：

现在要进行一个网站的建设，需要有一份建站策划书来指导建站。

任务目标：

按照瀑布模型，完成建站策划书，需要包括以下内容：

- 1、问题定义：确定网站的类型等。
- 2、可行性研究：技术、成本等。
- 3、需求分析：确定网站具体内容等。
- 4、总体设计：确定网站的栏目及整体结构等。
- 5、详细设计：确定每个页面的结构及配色方案等。

上机指导书

1. 按照瀑布模型的流程，组内逐一讨论各项；
2. 创建 word 文档，编制网站策划书。

思远教育版权所有

课后学习

扩展阅读:

Software Engineering

A decades-long goal has been to find repeatable, predictable processes or methodologies that improve productivity and quality. Some try to systematize or formalize the seemingly unruly task of writing software. Others apply project management techniques to writing software. Without project management, software projects can easily be delivered late or over budget. With large numbers of software projects not meeting their expectations in terms of functionality, cost, or delivery schedule, effective project management is proving difficult.

Process steps

Software engineering requires performing many tasks, notably the following, some of which may not seem to directly produce software.

Requirements Analysis

Extracting the requirements of a desired software product is the first task in creating it. While customers probably believe they know what the software is to do, it may require skill and experience in software engineering to recognize incomplete, ambiguous or contradictory requirements.

Specification

Specification is the task of precisely describing the software to be written, in a mathematically rigorous way. In reality, most successful specifications are written to understand and fine-tune applications that were already well-developed. Specifications are most important for external interfaces, that must remain stable.

Design and Architecture

Design and architecture refer to determining how software is to function in a general way without being involved in details. Usually this phase is divided into two sub-phases.

Coding

Reducing a design to code may be the most obvious part of the software engineering job, but it is not necessarily the largest portion.

Testing

Testing of parts of software, especially where code by two different engineers must work together, falls to the software engineer.

Documentation

An important (and often overlooked) task is documenting the internal design of software for the purpose of future maintenance and enhancement. Documentation is most important for external interfaces.

Maintenance

Maintaining and enhancing software to cope with newly discovered problems or new requirements can take far more time than the initial development of the software. Not only may it be necessary to add code that does not fit the original design but just determining how software works at some point after it is completed may require significant effort by a software engineer. About 2/3 of all software engineering work is maintenance, but this statistic can be misleading. A small part of that is fixing bugs. Most maintenance is extending systems to do new things, which in many ways can be considered new work. Similarly, about 2/3 of all civil engineering,

architecture, and construction work is maintenance in a similar way.

思远教育版权所有

小结

在这个单元里面我们学了以下的内容：

1. 软件危机产生
2. 软件工程
3. 瀑布模型

这里面重点要掌握的是瀑布模型。现在的软件开发已经不再是呈个人英雄主义的时候了，现在需要的是具有很强开发能力的团队，而软件的开发也是由团队来完成的，因此团队里每个人都起着至关重要的作用。作为一个团队来开发一款软件就需要按工程化的原则和方法组织软件开发工作才能有效，这就是软件工程，所以我们必须对软件工程有一个更深的认识，并将这种方法应用的以后的软件开发中。在本课中只是着重介绍了瀑布模型这一种开发模型，对于其它的模型现在有多有应用，大家需要在课余去了解这些模型，这样更有利于大家对现在的软件开发有一个深刻的认识。

Unit 4

HTML 表格

本单元由下列部分组成：

第一节 表格相关的标记

第二节 使用表格嵌套

积土而为山，积水而为海。

——《荀子·儒效》

第四课 html 表格

在完成本单元学习后，你将：

- 知道什么是表格
- 理解表格布局的规范、含义
- 掌握表格相关的标记设置表格的基本属性
- 能够设置单元格属性
- 能够熟练使用表格嵌套的各种技巧

理论教学

(2 课时)

4.1 前言

网页的布局设计变得越来越重要。虽然内容很重要，但只有当网页布局和网页内容成功接合时，这种网页或者说站点才是受人喜欢的。

4.2 网页布局的基本概念

◆ 页面尺寸

页面尺寸和显示器大小及分辨率有关系，分辨率越高页面尺寸越大。

浏览器的工具栏也是影响页面尺寸的原因。一般目前的浏览器的工具栏都可以取消或者增加，那么当你显示全部的工具栏时，和关闭全部工具栏时，页面的尺寸是不一样的。

◆ 整体造型

造型就是创造出来的物体形象，这里是指页面的整体形象，这种形象应该是一个整体，图形与文本的接合应该是层叠有序。

对于不同的形状，它们所代表的意义是不同的。但目前的网页制作多数是接合多个图形加以设计，在这其中某种图形的构图比例可能占的多一些。

◆ 页头

页头又可称之为页眉，页眉的作用是定义页面的主题。页头是整个页面设计的关键，它将牵涉到下面的更多设计和整个页面的协调性。页头常放置站点名字的图片和公司标志以及旗帜广告（logo）。

◆ 文本

文本在页面中出现都以行或者块（段落）出现，它们的摆放位置决定着整个页面布局的可视性。

◆ 页脚

页脚和页头相呼应。页头是放置站点主题的地方，而页脚是放置制作者或者公司信息的地方。你能看到，许多制作信息都是放置在页脚的。

◆ 图片

图片和文本是网页的两大构成元素，缺一不可。

◆ 多媒体

除了文本和图片，还有声音，动画，视频等等其它媒体。

4.3 制作表格

表格是网页排版的灵魂。在html 的语法中，表格通过 3 个标记来构成，表格标记、行标记、单元格标记。

表格的标记：

标记	描述
<table> ...</table>	表格标记
<tr>...</tr>	行标记
<td>...</td>	单元格标记

基本语法：

```
<table>
  <tr>
    <td>...</td>
  </tr>
  ...
</table>
```

语法解释：

<table> 标记代表表格的开始，<tr>标记代表行的开始，而<td>和</td>之间的就是单元格的内容。

4.3.1 表格标记属性

◆ 表格的宽度、高度属性 width、height

基本语法：

```
<table width="value" height="value">
```

语法解释：

value 单位为像素或百分比。

◆ 表格的边框属性 border

基本语法：

```
<table border="value">
```

语法解释：

默认情况下，表格的边框为 0。value 单位为像素。

◆ 表格的背景颜色属性 bgcolor

基本语法：

```
<table bgcolor="color_value">
```

语法解释：

color_value 可以使用英文颜色名称或十六进制颜色值表现。

◆ 表格的水平对齐属性 align

基本语法：

```
<table align="value">
```

语法解释：

默认的表格对齐属性为居左对齐。value 的值如表：

值	描述
left	居左对齐
center	居中对齐
right	居右对齐

◆ 单元格间距属性 cellpadding

基本语法：

```
<table cellpadding="value">
```

语法解释：

设定表格的单元格与单元格之间的间距，value 单位为像素。

◆ 单元格边距属性 cellspacing

基本语法：

```
<table cellspacing="value">
```

语法解释：

设定单元格内容和边框之间的距离，value 单位为像素。

4.4 表格标题与表头

在 html 语言中，可以自动通过标记为表格添加标题，另外，表格的第一行称为表头，也可以通过 html 标记来实现。

4.4.1 表格标题<caption>

基本语法：

```
<caption>...</caption>
```

语法解释：

<caption> 标记之间就是标题的内容，它使用在<table>标记中。

◆ 表格标题的水平对齐属性 align

基本语法:

```
<caption align="value">...</caption>
```

语法解释:

默认情况下, 表格的标题为水平居中对齐。value 的值如表:

值	描述
left	居左对齐
center	居中对齐
right	居右对齐

◆ 表格标题的垂直对齐属性 valign

基本语法:

```
<caption valign="value">...</caption>
```

语法解释:

value 的值如表:

值	描述
top	居上对齐
bottom	居下对齐

4.4.2 表格的表头<th>

表头是指表格的第一行, 其中的文字可以实现居中并加粗显示。

基本语法:

```
<table>
  <tr>
    <th>...</th>
  </tr>
  ...
</table>
```

语法解释:

使用<th>标记替代<td>标记。唯一不同就是标记中的内容居中并加粗显示。

4.5 行标记<tr>属性

用于设定表格中的某一行的属性。常用属性如表:

属性	描述
align	行内容的水平对齐
valign	行内容的垂直对齐
bgcolor	行的背景颜色
background	行的背景图像
bordercolor	行的边框颜色

4.6 单元格标记<td>、<th>属性

从表格的整体外观上看，单元格是表格的基本构成单元，一张表格至少要有有一个单元格。单元格由 `<tr><td></td></tr>` 共同组成，但在的一组 `<tr></tr>` 里，可以拥有多组 `<td></td>`。

用于设定表格中某一单元格的属性。常用属性如表：

属性	描述
align	单元格内容的水平对齐
valign	单元格内容的垂直对齐
bgcolor	单元格的背景颜色
background	单元格的背景图像
bordercolor	单元格的边框颜色
width	单元格的宽度
height	单元格的高度

表格基本标记综合范例

文件范例：3-1.htm

01: `<html>`

02: `<head>`

03: `<title> 表格的使用</title>`

04: `</head>`

05: `<body>`

06: `<table border="1" align="center"`

`width="600" cellspacing="0" cellpadding="5">`

07: `<caption>记录表</caption>`

08: `<tr>`

09: `<th>日期</th><th>内容</th>`

10: `</tr>`

11: <tr>

12: <td>2005 年</td><td>html 课程</td>

13: </tr>

14: </table>

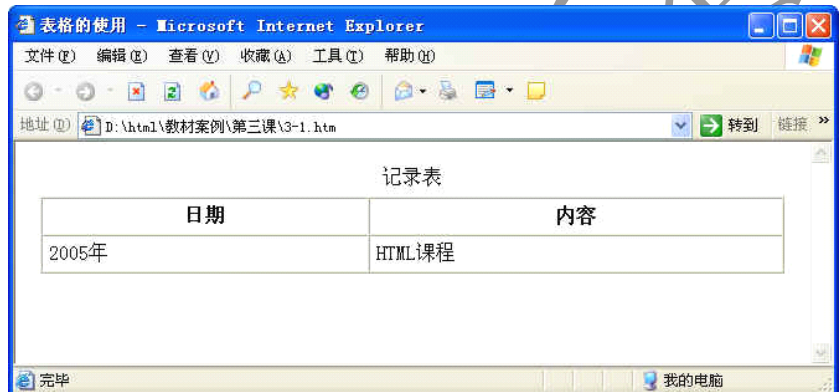
15: </body>

16: </html>

文件说明:

第 6 行定义了表格的相关属性, 第 7 行定义了表格标题, 第 8-13 行定义了表格的内容。

显示结果:



4.6.1 单元格跨行属性 rowspan

在复杂的表格结构中, 有的单元格在水平方向上是跨多个单元格的, 这是需要使用跨行属性 rowspan。

基本语法:

```
<td rowspan="value">...</td>
```

语法解释:

value 代表单元格跨的行数。

4.6.2 单元格跨列属性 colspan

在复杂的表格结构中, 有的单元格在垂直方向上是跨多个单元格的, 这是需要使用跨列属性 colspan。

基本语法:

```
<td colspan="value">...</td>
```

语法解释:

value 代表单元格跨的列数。

单元格跨行、跨列属性运用案例

文件范例：3-2.htm

```

01: <html>
02: <head>
03: <title> 表格的跨行跨列使用</title>
04: </head>
05: <body>
06: <table border="1" width="450" height="200" cellpadding="0"
    cellspacing="0" align="center">
07: <tr align="center">
08: <td colspan="2">主导航菜单</td>
09: </tr>
10: <tr>
11: <td width="100">公司简介</td><td rowspan="2">内容</td>
12: </tr>
13: <tr>
14: <td width="100">公司产品</td>
15: </tr>
16: </table>
17: </body>
18: </html>

```

文件说明：

第 8 行定义了单元格跨 2 列，第 11 行定义了单元格跨 2 行。

显示结果：



4.7 表格嵌套

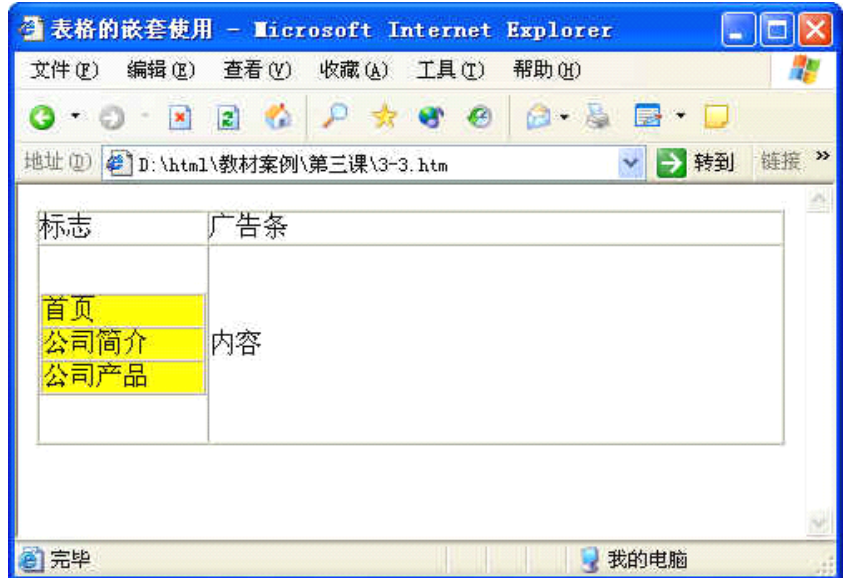
作为容器，表格可以往里放置可以放置的元素。表格也是一个元素，故而表格放在表格里是合理的，因此就有了表格的嵌套。

文件范例：3-3.htm

```
01: <html>
02: <head>
03: <title> 表格的嵌套使用</title>
04: </head>
05: <body>
06: <table border="1" width="450" align="center" cellpadding="0"
cellspacing="0">
07: <tr>
08:   <td width="100">标志</td><td>广告条</td>
09: </tr>
10: <tr>
11:   <td height="120">
12:     <table border="1" width="100%" bgcolor="yellow"
cellpadding="0" cellspacing="0">
13:       <tr>
14:         <td>首页</td>
15:       </tr>
16:       <tr>
17:         <td>公司简介</td>
18:       </tr>
19:       <tr>
20:         <td>公司产品</td>
21:       </tr>
22:     </table>
23:   </td>
24:   <td>
25:     内容
26:   </td>
27: </tr>
28: </table>
29: </body>
30: </html>
```

文件说明：

第 11-23 行在表格第二行的第一个单元格内，嵌套了三行一列的一个表格。

显示结果：

课后自测

请完成课后自测题：

课后自测是对学员学习进度的检查，帮助学员找出对本课知识点理解的不足。请学员在理论授课完成后立即完成下列习题，对于不能准确回答或有疑问的问题，请即刻向授课教员提出。

一. 填空题

1. 表格的基本标记是_____。
2. `valign` 规定的是_____方向的对齐方式，`valign` 的取值方法主要有_____、_____、_____。
3. 表格的基本属性有_____、_____、_____、_____、_____。
4. `<caption>...</caption>` 是用来定义表格的_____。

二. 判断题

1. 单元格由`<tr><td></td></tr>`共同组成，但在一组 `tr` 里，只能拥有一组 `td`。 ☐
2. `align` 的取值方法主要有：`left`、`middle`、`top`、`bottom`。 ☐
3. 作为容器，表格可以往里放置可以放置的元素。 ☐
4. 网页整体布局只能依靠表格来布局。 ☐
5. `<td></td>`标记里是没有 `valign` 的属性的。 ☐

三. 问答题

a) 阐述页面布局中使用表格布局的优点、缺点。

b) 请用最少的html代码实现以下表格嵌套。

上机指南

(2 课时)

思远教育

上机任务书

任务背景：



任务目标:

为一个网站制作手机分页。

上机指导书

1. 在桌面建立文件夹 test;
2. 进入 test 并新建文本文件, 将其重命名为 index.htm, 注意要将扩展名要从 TXT 改为 HTM;
3. 在 test 下建立文件夹 images, 并将图片拷贝到里面;
4. 用记事本将刚建立的文件 index.htm 打开, 并在其中输入代码。
5. 以下是关键代码:

<!-- 左边单元格嵌套表格-->

```
<table border="0" width="145" cellspacing="8" cellpadding="0">  
  <tr><td height="35"> 诺基亚</td></tr>  
  <tr><td height="35">CDMA 手机</td></tr>  
</table>
```

<!-- 右边单元格嵌套表格-->

```
<table border="0" width="615" height="415" cellspacing="0" cellpadding="0">  
  <tr>  
    <td colspan="2" align="center" valign="top"></td></tr>  
  <tr>  
    <td align="center" valign="top">  
      </td>  
    <td align="center" valign="top" width="100">7 系列</td>  
  </tr>  
</table>
```

课后学习

课后练习

尝试完成如下的网页：



附：素材见 table-1 文件夹

拓展学习

网站设计的原则

01 明确内容

如果你想成为一个网站设计者，并正想建一个网站的话，首先应该考虑网站的内容，包括网站功能和你的用户需要什么。你的整个设计都应该围绕这些方面来进行。

02 抓住用户

如果用户不能够迅速地进入你的网站，或操作不便捷，网站设计就是失败的。不要让用户失望而转向你的对手的网站。

03 优化内容

内容是核心。大约在两年以前，企业网站就像一本广告册子，更糟糕的是，网站使用了大量的图片，似乎要几个世纪才能下载完。

04 快速下载

没有什么比要花很长时间下载页面更糟糕的了。作为一条经验，一个标准的网页应不大于 60K，通过 56K 调制解调器加载花 30 秒的时间。有的设计者说网页加载应在 15 秒内。

05 网站升级

时刻注意网站的运行状况。性能很好的主机随着访问人数的增加，可能会运行缓慢。但是，如果你不想失去访问者的话，一定要仔细计划好你的升级计划。

06 学习 HTML

用 HTML 设计网站，可以控制设计的整个过程。但是，如果你仅仅是网站设计的新手，你应该寻找一个允许修改 HTML 的软件包。HomeSite4 是一个很好的 Web 设计工具。在设计过程中，HomeSite4 能帮助你学习 HTML。它还允许你切换到所见即所得的模式，以便你在把网站发送到 Web 之前，预览你的网站。

07 用笔画一个网站的框架

圣人云：笔比剑更强大。在用计算机之前，用笔画一个网站的框架，显示出所有网页的相互关系。计划好你的用户如何以最少的时间浏览你的网站。

08 网站地图

许多设计者把他们的网站地图放在网站上，这种做法，却是弊大于利。绝大部分的访问者上网是寻找一些特别的信息，他们对于你的网站是如何工作的，并没有兴趣。如果你觉得你的网站需要地图，那很可能是需要改进你的导航和工具条。

09 点击规则

听说过 3 次点击规则吗？对于小型网站，在你的主页上，没有任何一条信息，需要点击次数超过 3 次的。对于大型网站，使用导航和工具条来改善操作。

10 特殊字体的应用

虽然你可以在你的 HTML 中使用特殊的字体，但是，你不可能预测你的访问者在他们的计算机上将看到什么。在你的计算机里看起来相当好的页面，在另一个不同的平台上看起来可能非常糟糕。一些网站设计员喜欢使用来定义特性，这虽然允许你使用特殊的字体，但是仍需要一些变通的方法，以免你所选择的字体在访问者的计算机上不能显示。级联风格表 CSS 有助于解决这些问题，但是只有最新版的浏览器才支持 CSS。

11 检查错别字

好的拼写是人们一生中重要的技能。但是遗憾的是，许多设计者都缺少这种技能。确保你拼写正确，并且格外注意平常容易误写的错别字。

12 避免长文本页面

在一个站点上有许多只有文本的页面，是令人乏味的，且也浪费 Web 的潜力。如果你有大量的基于文本的文档，应当以 Adobe Acrobat 格式的文件形式来放置，以便你的访问者能离线阅读。

13 不要使用卷滚条

人们厌恶在网上使用卷滚条。Trouble 网站 (www.Trouble.co.uk) 是一个典型的设计很差的网站。它基于一个浮动的架构，为了阅读所有的文本，浏览者不得不使用卷滚条。

14 网站介绍

你应当有一个很清晰的网站介绍，告诉访问者你的网站能够提供些什么，以便访问者能找到想要的东西。但是，许多设计者都没有这样做。有效的导航条和搜索工具使人们很容易找到有用的信息，这对访问者很重要。告诉访问者你所提供的正是他们想要的信息。

15 闪烁让人头痛

通过使用标识可以吸引访问者对你的主页特殊部分的注意，但这也让你的访问者头痛。如果你想使访问者再次光顾你的网站，就少用此方法。

16 背景颜色

背景颜色也会产生一些问题，可能会使网页难于阅读。你应当坚持使用白色的背景和黑色的文本，另外还应当坚持使用通用字体。

17 向前和向后按钮

应当避免强迫用户使用向前和向后按钮。你的设计应当使用户能够很快地找到他们所要的东西。绝大多数好的站点在每一页同样的位置上都有相同的导航条，使浏览者能够从每一页上访问网站的任何部分。

18 点击计数器

不要轻易考虑在你的网站上放置一个醒目的点击计数器。你设计网站是为了给访问者提供服务，而不是推销你自己认为重要的东西。大多数浏览者认为计数器毫无意义，它们很容易被做假，浏览者也不想看广告。如果你显示你的网站是多么受欢迎，你最好提供一个链接，显示访问日志。

19 不要用框架

与计数器一样，框架在网页上越来越流行。在大多数网站上，在屏幕的左边有一个框架。但是设计者立刻就发现，在使用框架时产生了许多问题。使用框架时如果没有 17 英寸的显示屏几乎不可能显示整个网站。框架也使得网站内个人主页不能够成为书签。也许更重要的是，搜索引擎常常被框架混淆，从而不能列出你的网站。

20 去掉图像

在浏览器中即使去掉了图像功能，也要保证访问者能够在你的网站上获得满意的效果。对于那些使用 ISDN 连接并且关掉了图像功能的访问者，还能获得好的网页加载性能。可以通过在网页底部提供另外的链接和使用替代文字，而不是图像来满足访问者的需要。

21 重复使用图像

一些网站由于使用大量不重复的图像而错过了使用更好的技巧的机会。在创建商标时，在网页上多次使用同样的图像是一个好的方法，并且一旦它们被装入，以后重新载入就会很快。

22 避免使用过大的图像

不要使用横跨整个屏幕的图像。避免访问者向右滚动屏幕。占 75% 的屏幕宽度是一个好的建议。

小结

在本单元中我们接触到了 HTML 中使用很广泛的表格。主要学习了以下内容：

1. 表格的标记及其属性
2. 单元格的属性设置
3. 表格的嵌套

可以说这个单元的内容都是重点，不过对于表格的结构来说是一个难点，更大的一个难点就是表格的嵌套。如果要掌握得更深入，就需要大家下去多看一些网站的结构，先自己分析它是如何进行表格嵌套的，然后看其源代码具体是如何实现的嵌套，找出不同点，分析哪种方案会更加的合理。

思远教育版权所有

拓展学习

策划网页配色方案时的判断标准

网页设计师策划一个网站需要经过反复多次的思考，而在决定网页配色方案时同样需要经过再三的思量。一个大型站点是由几层甚至数十层的链接和上百上千种不同风格的网页所构成的，在需要时还应该绘制一个合理的层级图。

色彩意象的设计应用

当我们看到色彩时，除了会感觉其物理方面的影响，心理也会立即产生感觉，这种感觉我们一般难以用言语形容，我们称之为印象，也就是色彩意象。

- A. 红色的色彩意象：选取基调为红色的页面设计通常会带来果断与热情强烈的印象，极富表现力。
- B. 橙色的色彩意象：橙色明视度高，在工业安全用色中，橙色是警戒色，如登山服装、救生衣的设计大都含有橙色。由于橙色非常明亮刺眼，有时会使设计发生负面低俗的意象。所以在运用橙色时，要注意选择搭配的色彩和表现方式，才能把橙色明亮活泼的特性发挥出来。由于符合大自然创造的色彩搭配模式，橙色与具有一定灰度的绿色搭配的效果非常好。
- C. 黄色的色彩意象：高亮度的黄色在页面设计中很少出现，多半是由于颜色本身过于刺眼，造成浏览者视觉上的不愉快。但使用黄色与黑色的搭配，会强烈地突出设计主体，在细节部分使用这种搭配收效很好。
- D. 绿色的色彩意象：在页面设计中，绿色所传达的清爽、理想、希望、生长的意象，符合了人们接近自然的生理要求。一般的医疗机构场所，常采用绿色来作空间色彩规划。使用绿色基调可以避免长时间用眼所造成的疲劳。绿色是舒适惬意的象征，是年轻所特有的颜色。但在设计中应注意添加变化。偏黄的绿色有种醒目的感觉，充满生机和跳跃感。
- E. 蓝色的色彩意象：由于蓝色沉稳的特性，具有理智、准确的意象。在商业设计中，强调科技、效率的商品或企业形象，大多选用蓝色作为标准色，如电脑、汽车、摄影器材等，另外蓝色也代表忧郁，这是受了西方文化的影响，这个意象通常运用在感性诉求的商业设计中。使用变化的蓝色系作为页面基调，能营造出平稳深邃的意境，更能展示博大的胸怀。但要注意颜色过深会导致色彩趋向过冷，应该用适当的紫色的色彩意象。
- F. 紫色的色彩意象：由于具有强烈的女性化性格，在商业设计用色中，紫色也受到限制，除了和女性有关的商品或企业形象之外，其他类的设计不常采用紫色为主色。深紫色会造成恐怖的气氛；明度很低的紫色有清澈爽朗的感觉；而具有一定灰度的浅紫色会有纤巧雅致的感觉，适合

表现与女性有关的题材。

- G. 褐色的色彩意象：在商业设计用色中，褐色通常用来表现原始材料的质感，如麻、木材、软木等，或用来传达某些饮品原料的色泽及味感，如咖啡、茶等，或强调格调古典优雅的企业或商品形象。
- H. 白色的色彩意象：白色具有高级、科技的意象，通常需和其他的色彩搭配使用，纯白色会带给别人寒冷、严峻的感觉。所以在使用白色时，都会掺一些其他的色彩，如象牙白、米白、乳白等。白色是永远流行的主要色，是可以和任何颜色作搭配的万能颜色。初学的设计者应尽量选用白色作为主体背景，毕竟白色的设计占据了网站设计作品的绝大部分。浏览者也最熟悉这种设计方式。
- I. 黑色的色彩意象：在商业设计中，黑色具有高贵、稳重、科技的意象，许多科技产品的用色，如电视、跑车、音响的色彩大多采用黑色方案。黑色庄严的意象常用于一些特殊氛围的页面设计。生活用品和服饰设计大多利用黑色来塑造高贵的形象，也是一种永远流行的主要颜色，适合与许多色彩搭配。
- J. 灰色的色彩意象：在商业设计中，灰色具有柔和、高雅的意象，而且属于中间性格，男女皆能接受，所以灰色也是永远的流行的主要颜色。许多高科技产品，尤其是和金属材料有关的，几乎都采用灰色来传达高级、科技的形象。使用灰色时，大多利用不同的层次变化组合或搭配其他色彩，才不会由于色彩过于素造成沉闷而又呆板、僵硬的感觉。以灰色作为设计基调是非常流行的做法，在缓解过度对比造成的强烈反差时，灰色显示了其杰出的调和能力。

CSS 样式表

本单元由下列部分组成：

第一节 *css* 样式表

在完成本单元学习后，你将：

- 了解 dhtml 概念
- 了解 *css* 的概念及版本
- 使用 *css* 对页面进行简单的布局控制

第五课 **css** 层叠样式表

在本课里，你将：

- 了解 dhtml 的概念
- 掌握 **css** 的概念及版本
- 掌握 **css** 的基本语法
- 掌握 **css** 的基本写法
- 了解 **css** 的基本属性

理论教学

(2 课时)

5.1 什么是 dhtml

dhtml 的全称为 **dynamic html**，也就是我们常说的动态 html。它的目的在于加强网页的交互性；对用户的操作在本地就可做实时处理，从而得到更快的用户响应；使网页的界面更丰富多变，使页面设计者可以随心所欲地表达自己的构思。

dhtml 的内容组成：

1. html4.0

组成了 dhtml 的基础，虽然在 4.0 版本浏览器里并没有完全支持 html4.0 标准，但在 w3c 的白皮书里却明确规定了 html4.0 的标准。

2. css

样式表，是 dhtml 编程的基础。在风格表的基础上加上脚本程序，就能建立出非常 cool 的动态效果了。

3. dom(文档对象模型)

用于定义一个与 html 文档连接的独立编程接口平台，用该接口可以控制文档的内容、结构和样式，web 开发人员可以使用 dom 使他的网页具有动态和交互的能力，而且不需要 web 服务器的介入。换句话说，当你提交诸如留言本之类的页面表单，再不需要后台 cgi 的支持。

4. css1 (客户端脚本语言)

最后的问题当然要归结到编程上。这里的脚本程序是指 4.0 版本浏览器里的 script 程序。

5.2 css 的概念及版本

css (**casading style sheets 层叠样式表**)，跟 html 一样也是一种标记语言。

万维网标准的委员会 w3c 制定了层叠样式表 (css) 的标准，现在已经推出了两个版本：css 1.0 和 css 2.0

css 1.0：此版本引入层叠样式表概念，用于格式化文本

css 2.0：此版本入了诸如定位框和分层的功能

5.3 css 的样式规则

css 的定义是由三个部分构成：选择符 (selector)，属性 (properties) 和属性的取值 (value)，选择符是样式名称 (如 tr 或 p)。

基本写法:

选择符 { 属性 1:值 1;属性 2:值 2 }

如果有多个属性, 需要用分号隔开。

5.4 css 基本语法

常用的 css 基本语法有以下三种:

1、选择符

任何 html 元素都可以是一个 css 的选择符。选择符仅仅是指向特别样式的元素。

基本语法:

```
p{text-align: center;color: black}
```

语法解释:

这里将 p 标记中文字的样式都设置成黑色并居中显示。

选择符也可以成组使用, 只要把相同属性和值的选择符组合起来书写, 并用逗号将选择符分开即可。

基本语法:

```
h1,h2,h3,h4,h5,h6 {color:red;font-family:黑体 }
```

语法解释:

将 h1~h6 六种标题的文字全部设置成了颜色为红色, 字体为黑体。

2、类选择符

单一个选择符能有不同的 class(类), 因而允许同一元素有不同样式。我们可以通过两种方式来实现:

◆ 选择符后加点再加类的名称

使用这种方式定义的类选择符只对此种标记有效。

基本语法:

```
p.right {text-align:right}
```

```
<p class="right">    </p>
```

语法解释:

第一行定义了一个类, 第二行引用了这个类。但是这里定义的这个类只对 p 标记有效。

◆ 类的名称前加点

使用这种方式定义的类选择符可以对所有具有该属性的标记有效。

基本语法:

```
.center {text-align:center}
```

```
<p class="center">    </p>
```

```
<h2 class="center">    </h2>
```

语法解释:

在第一行定义了一个类，在第二、三行分别在不同的标记里面进行了引用。

注: 类的名称可以是任意英文单词或以英文开头与数字的组合，一般以其功能和效果简要命名。

3、伪类选择符

这是一种特殊类型的样式，常用的有以下 4 种：

- ◆ **a:link:** 设定正常状态下链接文字的样式
- ◆ **a:visited:** 设定访问过的链接文字的样式
- ◆ **a:hover:** 设定鼠标放置在链接文字上时文字的样式
- ◆ **a:active:** 设定鼠标单击时链接文字的样式

基本语法:

a:link	{text-decoration:none;font-size:9pt;color:#000000}
a:visited	{text-decoration:none;font-size:9pt;color:#ff00ff}
a:hover	{text-decoration:none;font-size:9pt;color:#ff0000}
a:active	{text-decoration:none;font-size:9pt;color:#0000ff}

语法解释:

这里将链接文字设定为在所有状态下均没有下划线且文字大小为 9 磅，同时正常链接时为黑色，鼠标放置时为红色，点击时为蓝色，访问后为紫色。

5.5 css 的基本写法

1. 在 head 内实现

在 head 内实现即将 css 代码使用 `<style>...</style>` 标记直接插入每个页面的 html 的 `<head>` 区。

基本语法:

```
<style type="text/css">
<!--
h2 {font-size: 12pt;color: #ff0000}
p {font-size: 9pt;color: #0000ff}
-->
</style>
```

语法解释:

在头部定义了 h2、p 连个标记的样式。

文件范例: 5-1.htm

01: <html>

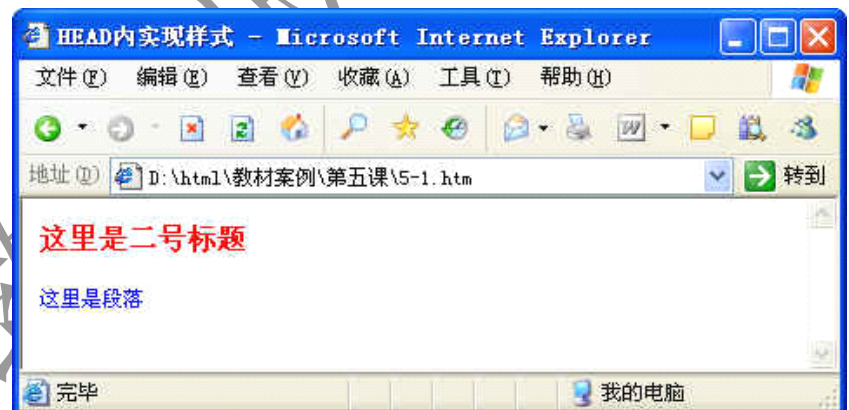
```

02: <head>
03: <title>head 内实现样式</title>
04: <style type="text/css">
05: <!--
06: h2 {font-size: 12pt;color: #ff0000}
07: p {font-size: 9pt;color: #0000ff}
09: -->
10: </style>
11: </head>
12: <body>
13: <h2>这里是二号标题</h2>
14: <p>这里是段落</p>
15: </body>
16: </html>

```

文件说明:

第 04~10 行进行了样式表的定义。

显示效果:**2. 在 body 内实现**

采用 html 标记的“style”属性，可以在标记内直接定义样式表，它的特点是“定义某一个标记的式样表风格”，只对所定义的标记起作用，并非对整个页面起作用。

基本语法:

```
<p style="font-size: 14pt ; color: red">层叠式样表</p>
```

语法解释:

将这—个段落文字定义为了红色 14 磅大小。

使用这样的样式表，你必须为每个标记指定样式规则，否则下一个标记将使用页面的缺省设置，但是有时候这种方式却非常有效。

文件范例: 5-2.htm

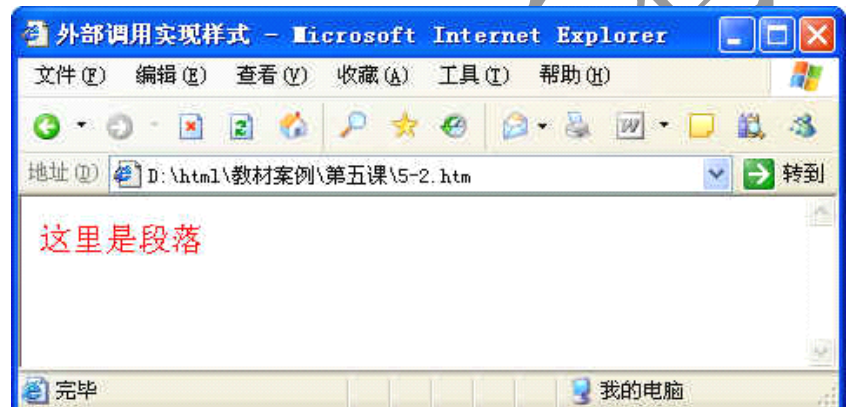
```
01: <html>
```

```
02: <head>
03: <title>body 内实现样式</title>
04: </head>
05: <body>
06: <p style="font-size: 14pt ; color: red">这里是段落</p>
07: </body>
08: </html>
```

文件说明:

第 06 行直接在 p 标记中应用了样式表。

显示效果:



3. 在文件外调用

在文件外调用即你可以将多个页面的排版风格都用一个样式表文件控制。这个外部的样式表文件（一个扩展名是css的文本文件）将设定你所有网页的规则。如果你改变样式表文件中的某行，所有页面风格用这个css文件定义的页面都会随之改变。如果你的站点文件非常多，则这项功能就非常方便了。

基本语法:

在<head>区内使用<link>标记（注意：不再是<style>标记了）:

```
<link rel="stylesheet" href="url" type="text/css">
```

语法解释:

href 的值为样式表文件的路径。

文件范例:

style.css

```
01: p{font-size:16pt;color:red;}
02: .blue20pt{font-size:20pt;color:blue;}
```

5-3.htm

```
01: <html>
02: <head>
03: <title> 外部调用实现样式</title>
```

04: `<link rel="stylesheet" href="style.css" type="text/css">`

05: `</head>`

06: `<body>`

07: `<p>这里是段落</p>`

08: `<p class="blue20pt">这里是 blue20pt 类的样式</p>`

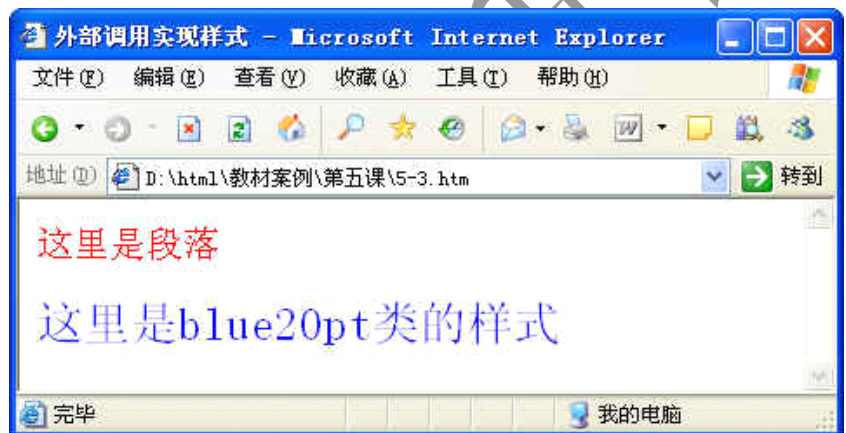
09: `</body>`

10: `</html>`

文件说明:

5-3.htm 第 04 行引用了样式表文件 **style.css**，在第 07、08 行应用了 **style.css** 中定义的样式。

显示效果:



样式表可以灵活运用，具体的样式表属性可以参看附录。

课后自测

请完成课后自测题：

课后自测是对学员学习进度的检查，帮助学员找出对本课知识点理解的不足。请学员在理论授课完成后立即完成下列习题，对于不能准确回答或有疑问的问题，请即刻向授课教员提出。

一、 填空题

1. DHTML 是_____，包括_____三部分。
2. CSS 全称是_____。
3. CSS 的基本写法有_____、_____。

二、 判断题

1. CSS 样式表用来控制网页内容的外观。 ☐
2. 可以使用 DOM 使网页具有交互的能力，需要 Web 服务器的介入。 ☐
3. CSS 的样式规则由 2 部分组成：选择器和声明。 ☐
4. 为了减少样式表的重复声明，组合的选择符声明是不允许的。 ☐
5. 在 BODY 内实现 CSS,对所定义的标记和对整个页面起作用。 ☐

三、 问答题

1. 阐述 CSS 运用在 HTML 页面中的几种方式？各自的优点、缺点。
2. CSS 样式实现方式冲突时，该对象应该遵循 CSS 实现的规则是什么？

上机指南

(2 课时)

思远教育

上机任务书

任务背景：



任务目标：

将当前的手机页面通过一个外部 CSS 样式表控制。

上机指导书

本部分介绍了如何将页面的元素通过 CSS 样式表控制。

CSS 样式表是控制网页元素的重要手段之一。如何创建一个外部样式表，请按以下步骤操作：

1. 创建一个名为 CSS 的文件夹。
2. 双击该文件夹，在内部创建一个名为 style.txt 的文本文件。
3. 选中 style.txt，点右键的重命名，将其设置成名为 style.css 的 CSS 文档。
4. 文件重要代码
 - 1) 在名为 index.html 源代码的 HEAD 中加入以下代码：
<link rel=stylesheet href="css/style.css" type="text/css">
 - 2) 名为 style.css 的代码：

```
a:link{font-size:12px;color:black;text-decoration:none;}
```

```
a:visited{font-size:12px;color:orange;text-decoration:none;}
```

```
.left{text-align:center;width:145;height:60;line-height:2;font-size:14px;}
```

```
.right{vertical-align:top; height:80; width:100; font-size:14px;}
```

```
.right1{vertical-align:top; height:89;}
```

```
.picsize1{width:205; height:89;}
```

```
.picsize2{width:145; height:80;}
```

课后学习

课后练习

尝试用样式表完成如下的网页的控制：



拓展学习

CSS 文件的格式样例代码：

```
<style type="text/css">
<!--
p { text-indent: 2em; }
body { font-family: "宋体"; font-size: 9pt; color: #000000; margin-top: 0px; margin-right: 0px; margin-
bottom: 0px; margin-left: 0px}
table { font-family: "宋体"; font-size: 9pt; line-height: 20px; color: #000000}
a:link { font-size: 9pt; color: #FFFFFF; text-decoration: none}
a:visited { font-size: 9pt; color: #99FFFF; text-decoration: none}
a:hover { font-size: 9pt; color: #FF9900; text-decoration: none}
a:active { font-size: 9pt; color: #FF9900; text-decoration: none}
a.l:link { font-size: 9pt; color: #3366cc; text-decoration: none}
a.l:visited { font-size: 9pt; color: #3366cc; text-decoration: none}
a.l:hover { font-size: 9pt; color: #FF9900; text-decoration: none}
a.l:active { font-size: 9pt; color: #FF9922; text-decoration: none}
.blue { font-family: "宋体"; font-size: 10.5pt; line-height: 20px; color: #0099FF; letter-spacing: 5em}
-->
</style>
```

这里尤其要注意的是 a:link a:visited a:hover a:active 的排列顺序一定要严格照上面的样例代码，否则或多或少会出问题。另外我们规定重定义的最先，伪类其次，自定义最后，便于自己和他人阅读！

为了保证不同浏览器上字号保持一致，字号建议用点数 pt 和像素 px 来定义，pt 一般使用中文宋体的 9pt 和 11pt，px 一般使用中文宋体 12px 和 14.7px 这是经过优化的字号，黑体字或者宋体字加粗时，一般选用 11pt 和 14.7px 的字号比较合适。

小结

在这个单元里面我们接触到了一种新的设置属性的方式，CSS 样式表。我们主要学习了以下的内容：

1. DHTML 的概念
2. CSS 的概念及规则
3. CSS 的基本语法与写法

在这些内容里面重点和难点都是掌握 CSS 的基本语法与写法，因为样式表现在是普遍使用的一种控制格式的方式。现在网站讲究一个整体的风格，而要实现整体的风格必然要制定很多格式的标准，如果将这些标准逐一应用到每个标记中，将是一种相当繁琐的事情。如果使用了样式表，便可以轻松实现，并且要改变整个网站的风格，简单到只需要将样式表更换便可以实现。那么大家在发现某个网站的整体效果不错的时候就可以把它的样式表好好研究一下，研究多了你自然也会增长很多的经验，从而设计出更加优秀的网站。

选修内容

在这个部分，你可以学到以下内容：

- 如何在网页中使用多媒体文件
- 框架的设置
- 浮动框架的应用
- 框架中链接的使用
- 什么是表单
- input 标记的属性
- 列表、文字域的使用
- 如何正确有效的使用表单

多媒体标记

现在网络上大量使用了多媒体信息，这些信息是如何加到网页中的呢？下面我们就简单介绍几种可以嵌入多媒体信息的标记。

背景音乐标记<bgsound>

在页面中可以嵌入背景音乐，这种音乐以 MIDI 音乐为主。

MIDI 是英文 Musical Instrument Digital Interface（乐器的数字化接口）的缩写。MIDI 技术的一大优点是它送到和存储在电脑里的数据量相当小，1 分钟的 MIDI 音乐文件只有约 8 千字节，所以 MIDI 文件非常适合于因特网的传输。

MIDI 只是一种电子音乐，可是我们往往希望我们的页面的不背景音乐是真人真唱的歌曲，这该如何实现呢？其实背景音乐也是支持 MP3、WMA 等多种格式的，这些音乐文件虽然比 MIDI 大，但是现在的网速都提升了，完全可以很好的播放这中高品质的音乐。

基本语法：

```
<bgsound src="url" loop="value">
```

语法解释：

- ◆ **Src:** 声音的 URL
- ◆ **Loop:** 背景音乐的循环次数

嵌入多媒体内容

有些多媒体文件不需要特殊插件就可以在浏览器中得到支持，其他多媒体文件则需要插件。

在页面中可放置如 MP3 音乐、视频、SWF 动画等多媒体内容。

作为网页设计者，你要记住，为了使访问者能够正确地加载网页可能需要特殊的软件，你应该在网页中包含指向网页所需插件的链接。

基本语法：

```
<embed src="url" width="value" height="value">...</embed>
```

语法解释：

- ◆ **Src:** 多媒体文件的 URL
- ◆ **Width:** 多媒体文件的宽度
- ◆ **Height:** 多媒体文件的高度

滚动文字标记<marquee>

在 HTML 页面中，可以实现如字幕一般的滚动文字效果。页面中实现滚动的文字，也可以用于实现滚动的图片。

基本语法：

```
<marquee>...</marquee>
```

滚动方向属性 direction

设定文字的滚动方向分别为向上、向下、向左、向右。

基本语法：

```
< marquee direction="value">滚动文字</marquee>
```

语法解释：

常用 Value 的值为：

属性值	描述
Up	滚动文字向上
Down	滚动文字向下
Left	滚动文字向左
Right	滚动文字向右

滚动方式属性 behavior

设定文字的滚动方向分别为向上、向下、向左、向右。

基本语法：

```
< marquee behavior="value">滚动文字</marquee>
```

语法解释：

常用 Value 的值为：

属性值	描述
Scroll	循环往复
Slide	只进行一次滚动
Alternate	交替进行滚动

滚动速度属性 scrollamount

基本语法：

```
< marquee scrollamount="value">滚动文字</marquee>
```

语法解释：

这个属性调整文字滚动的速度，数值越大速度越快。

滚动延迟属性 **scrolldelay**

基本语法：

```
< marquee scrolldelay="value">滚动文字</marquee>
```

语法解释：

滚动延迟速度属性，通过这个属性，可以在每一次滚动时产生一段时间的延迟，数值越大延迟的时间越长，基数为 0.001 秒。

滚动范围属性 **width、height**

基本语法：

```
< marquee width="value" height="value">滚动文字</marquee>
```

语法解释：

对于各种方式的滚动，可以设定滚动文字的区域。

滚动背景颜色属性 **bgcolor**

基本语法：

```
< marquee bgcolor="color_value">滚动文字</marquee>
```

语法解释：

在滚动文字后面可以添加背景颜色。

滚动文字标记综合范例

文件范例：2-5.htm

01: <html>

02: <head>

03: <title> 滚动标记的使用</title>

04: </head>

05: <body>

06: <marquee direction="right" behavior="slide"

scrollamount="10" width="600" height="80" bgcolor="yellow">

高级人才的六个标准</marquee>

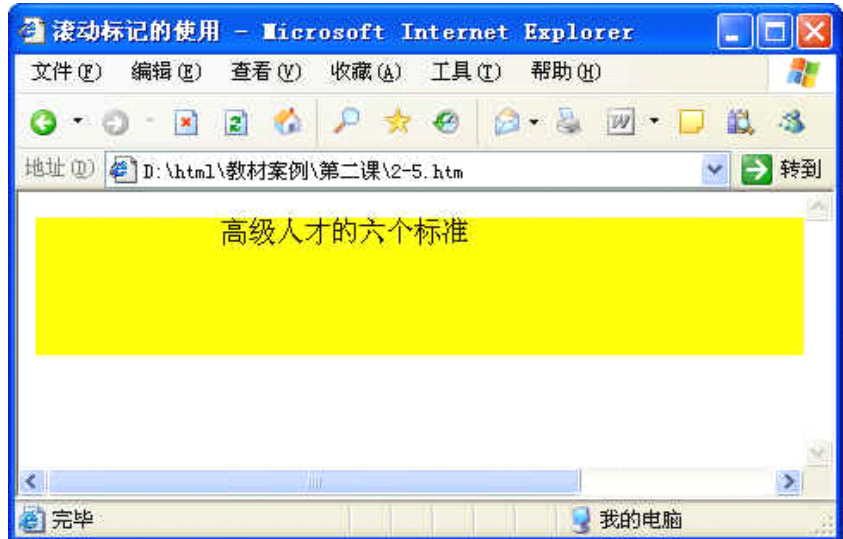
07: </body>

08: </html>

文件说明:

第 6 行设定了滚动文字的方向、方式、速度、宽度和高度以及背景颜色。

显示结果:



框架结构

框架可以用来划分浏览器窗口以使窗口中能够容纳多个逻辑页面。当你创建一个具有框架的站点时，主框架网页有一个 URL，而网页内的每个框架也有自己的 URL。因为一个带框架的站点中的每个框架都有它自己的 URL，所以你可以将其他网页装入浏览器的某一部分，而剩下的框架将保持不变。

因此，框架可以用来提高网页导航性能，并能够更灵活地控制站点的布局。下面将介绍框架是如何工作的及如何充分利用框架等内容。

框架标记

浏览器是怎么知道它需要装入多个 URL 呢?这些 URL 将放到什么地方呢?你需要创建一个主框架文档(master frame document)。

主框架文档是一个标准 HTML 文档，但其中没有 BODY 元素，而是有一个 HEAD 元素及一个 FRAMESET 元素。在 FRAMESET 元素中又可以有 FRAME 元素和 NOFRAMES 元素。

框架集标记<frameset>

创建框架集的标记。

基本语法:

```
<frameset rows="" cols="" framespacing="" frameborder=""
bordercolor="">
</frameset>
```

语法解释:

内容: FRAMESET 元素、FRAME 元素和 NOFRAMES 元素

属性:

rows: 用于指定各行从上至下的宽度; 缺省值为 100%; 你也可以以像素、占浏览器窗口的百分比或相对宽度为单位定义宽度

cols: 用于指定各列从左至右的宽度; 缺省值为 100%; 你也可以以像素、占浏览器窗口的百分比或相对宽度为单位定义宽度

framespacing: 以像素为单位设定框架集边框的宽度

frameborder: 告诉浏览器在所有框架周围划一个边框(1, 缺省值)或不划边框(0)

bordercolor: 设定框架集边框的颜色

框架标记<frame>

在每个框架中装入什么内容是使用 FRAME 元素可以完成的。

基本语法:

```
<frame name="" src="" frameborder="" marginheight=""
marginwidth="" scrolling="" noresize>
```

语法解释:

属性:

name: 用于定义与该框架相关联的名字(目标名)

src: 用于定义该框架中最初装入的内容

frameborder: 告诉浏览器在该框架周围划一个边框(1, 缺省值)或不划边框(0); 如果相邻的框架有一个框架边框, 它将出现在无边框框架和有边框框架之间

marginheight: 以像素为单位定义框架内容与上下边界之间空白的宽度

marginwidth: 以像素为单位定义框架内容与两侧边界之间空白的

宽度

noresize: 告诉浏览器这个框架不能由访问者重新定义大小

scrolling: 该属性有三个值: **yes**, **no** 和 **auto**。

不支持框架标记<noframes>

如果访问者的浏览器不支持框架,或者是浏览器虽然能够支持框架,但经过配置却不能显示框架,在上述情况下,都可以使用 NOFRAMES 元素包含你希望访问者能得到的网页。

基本语法:

```
<noframes>.....</noframes>
```

语法解释:

标记的内容可以是文字,也可以是其它 HTML 元素,该标记应该放在 frameset 标记中。

两种常见的框架分割

在同一文件夹下建立文件 4-2.htm, 4-3.htm, main.htm, left.htm, top.htm。

◆ 只有行

文件范例: 4-2.htm

01: <html>

02: <head>

03: <title> 上下分割框架</title>

04: </head>

05: <frameset rows="20%,80%">

06: <frame name="top" src="top.htm">

07: <frame name="main" src="main.htm">

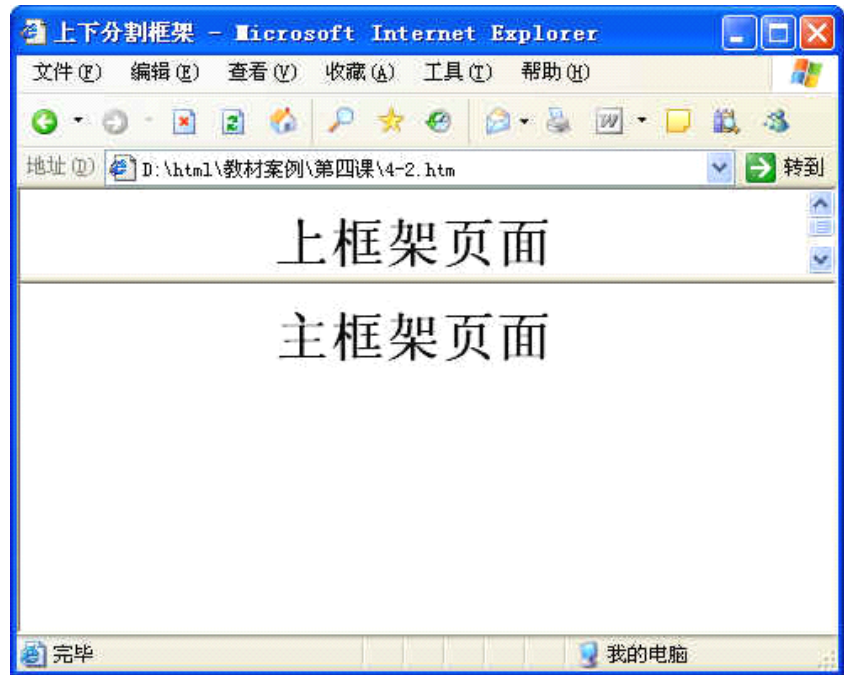
08: </frameset>

09: </html>

文件说明:

第 05~08 行建立了框架集,第 05 行中用百分比分割了框架。

显示效果:



◆ 只有列

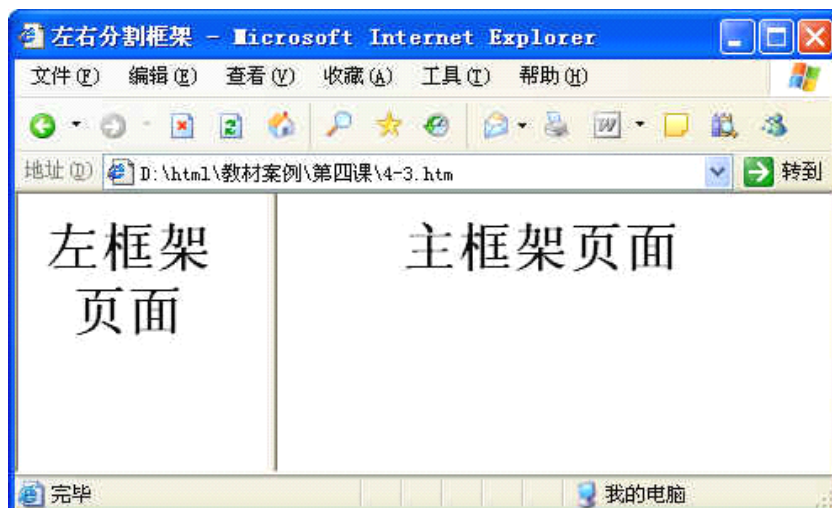
文件范例: 4-3.htm

```
01: <html>
02: <head>
03: <title>上下分割框架</title>
04: </head>
05: <frameset cols="150,*">
06:     <frame name="left" src="left.htm">
07:     <frame name="main" src="main.htm">
08: </frameset>
09: </html>
```

文件说明:

第 05~08 行建立了框架集, 第 05 行中用绝对像素分割了框架, *表示将剩余部分分给右边的框架。

显示效果:



嵌套 frameset

我们可以利用框架集的嵌套来形成更加复杂的框架结构。

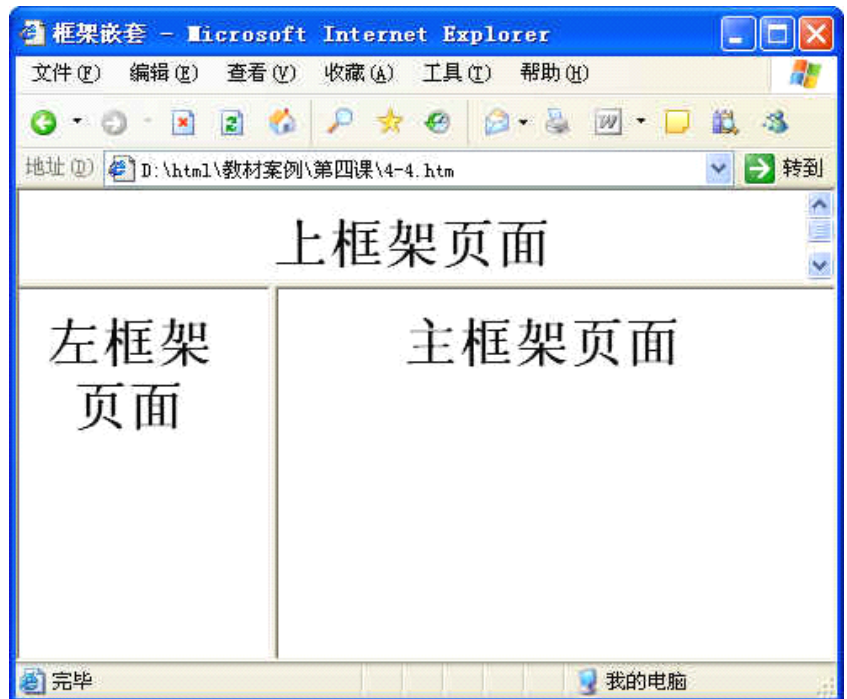
文件范例：

```
01: <html>
02: <head>
03: <title>框架嵌套</title>
04: </head>
05: <frameset rows="20%,80%">
06:     <frame name="top" src="top.htm">
07:     <frameset cols="150,*">
08:         <frame name="left" src="left.htm">
09:         <frame name="main" src="main.htm">
10:     </frameset>
11: </frameset>
12: </html>
```

文件说明：

第 05~11 行建立了框架集，第 05 行先将页面分割成上下两个部分，第 06 行建立上框架，第 07~10 行插入一个新的左右分割框架集替代了原来的下框架。

显示效果：



浮动框架<iframe>

浮动框架具有框架的许多优点，但它却没有在多个框架中导航的缺点。浮动框架看上去与框架完全不同。

基本语法:

```
<iframe name="" src="" height="" width="" align=""></iframe>
```

语法解释:

iframe 没有内容，通常可以放在一个页面<body>中的任何地方，也可以放到单元格中。

属性:

name: 该元素的名字;

src: 你准备插入浮动框架中的网页的 URL

height: 以像素为单位定义浮动框架的高度

width: 以像素为单位定义浮动框架的宽度

align: 用于定义网页上的对齐方式

frameborder: 缺省值为“1”; 如果不想有，边框则设置为“0”

marginwidth: 以像素为单位定义左右两侧空白的宽度

marginheight: 以像素为单位定义上下两端空白的高度

scrolling: 其值为 yes, no 或 auto: 只有在设置后高度和宽度比 URL 中内容的尺寸小的情况下，才需要考虑该属性

noresize: 告诉浏览器这个框架不能由访问者重新定义大小

bordercolor: 定义边框的颜色

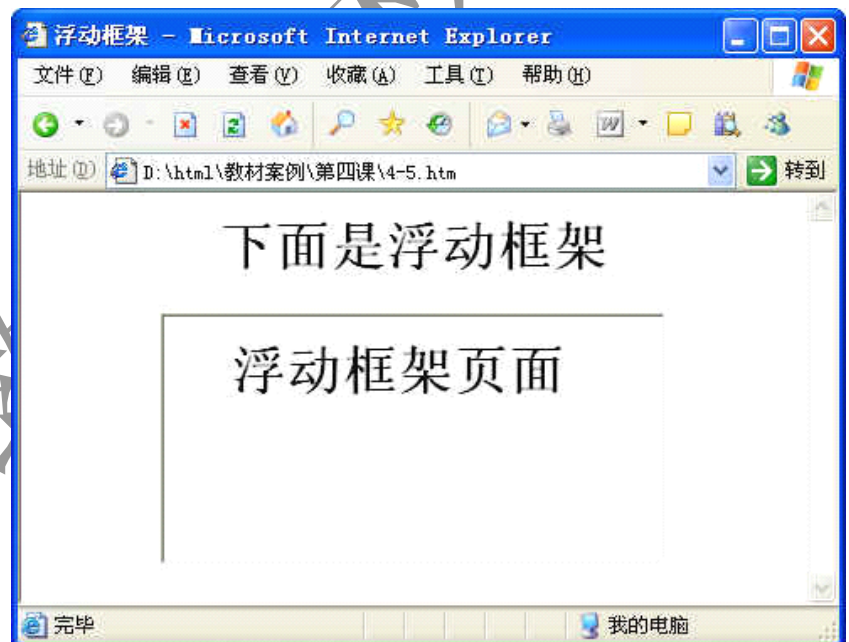
文件范例:

```
01: <html>
02: <head>
03: <title>浮动框架</title>
04: </head>
05: <body>
06: <h1 align="center">下面是浮动框架</h1>
07: <center><iframe width="300" height="150" name="myiframe"
08:   src="myiframe.htm"></iframe></center>
09: </body>
10: </html>
```

文件说明:

第 07 行建立了一个宽 300 像素，高 150 像素的浮动框架。

显示效果:



框架与链接

使用框架的目的就是为了实现一个很好的导航与布局，因此就离不开超链接。

frame 元素和 iframe 元素都有一个 name 属性。使用 name 属性可以为框架定义一个名字，这个名字就是框架的目标名。你可以将其他链接指向这个目标。即通过在 a 元素上定义 target 属性使那些链接的目的地装入该框架中。name 名必须以字母开始。

表单

表单是一种从站点访问者处收集信息的方法。根据站点的不同以及应用程序的不同从表单中收集的信息、收集信息的方法以及处理信息的方式都会有所不同。

表单的工作过程是这样的：

1. 在网页上放置一些数据输入域，它们可以是文本域、单选按钮、复选框、文本域(用于接受大量文本的输入)以及其他按钮(如提交按钮和重新设置按钮)。
2. 网站访问者在自己的计算机上填写上述输入域，并按某个按钮。
3. 访问者输入的信息通过 Internet 发送到你的服务器上。
4. 你的服务器处理这些数据。
5. 理想情况下，你应该向访问者提供一个屏幕确认信息，告诉他数据已经接收到并将对这些数据进行处理。

你还可以做其他工作，如通过 E-mail 发送确认信息、当别人在使用你的表单时给自己发送一个通知信息，另外还可以把访问者提交的数据送入数据库。

表单标记<form>

表单由两部分组成：

- ◆ FORM 元素
- ◆ 接受数据的域或控件

如果没有 FORM 元素，你的表单在测试时看起来并不会有什么不对的地方，但它实际上却绝对不能工作。

基本语法：

```
<form name="" action="" method="" >
.....
</form>
```

语法解释：

表单的内容可以接受数据的域或控件和 html 元素的结合。

属性：

name: 表单的名称

action: 如果处理工作只是要发送数据的话，该属性表示用来处理数据的脚本所在的 URL 或者是一个电子邮箱地址

method: 数据传送到脚本的方法，两种选择：

- ◆ **get:** 将表单内容附加在 URL 地址后面，所以对提交信息的长

度进行了限制，最多不可以超过 8192 个字符，如果信息太长，将被截去，从而导致意想不到的处理结果。同时 Get 方法不具有保密性，而且不能传送非 ASCII 码的字符。(默认值)

- ◆ **post:** 将用户在表单中填写的数据包含在表单的主体中，一起传送到服务器上的处理程序中，该方法没有字符的限制，它包含了 ISO10646 的字符集，是一种打包传送的方式，在浏览器的地址栏不显示提交的信息。

输入标记<input>

最常用的标记是输入标记 INPUT，它可以被用来创建标准文本域、口令域、复选框、单选按钮、提交按钮、重新设置按钮等。

基本语法:

```
<input name="" type="">
```

语法解释:

Name: 域的名称

Type: 域的类型，其取值有:

- ◆ **Text** 文字域
- ◆ **Password** 密码域
- ◆ **File** 文件域
- ◆ **Checkbox** 复选框
- ◆ **Radio** 单选框
- ◆ **Button** 普通按钮
- ◆ **Submit** 提交按钮
- ◆ **Reset** 重置按钮
- ◆ **Image** 图像域（图像提交按钮）

文字域 text

在页面上显示一个单行的文本输入框。

基本语法:

```
<input type="text" name="" maxlength="" size="" value="">
```

语法解释:

size: 当网页显示时，该域在屏幕上的大小。

maxlength: 该域能够接受的最大字符数。

value: 文字域的默认值。

密码域 password

如果希望某个文本输入域中接受的数据不被其他人看到，那么就应该使用密码域，这时你输入的内容都显示为*。

基本语法：

```
<input type="password" name="" maxlength="" size="">
```

语法解释：

Size: 当网页显示时，该域在屏幕上的大小。

Maxlength: 该域能够接受的最大字符数。

复选框 checkbox

我们在表单中，常会遇到需要在若干个选项中做出选择的情况，如果是在若干选项中选择多个项目，则使用复选框。

基本语法：

```
<input type="checkbox" name="" value="" checked >
```

语法解释：

Value: 选中项目后传送到服务器端的值

Checked: 此项默认被选中

单选框 radio

如果是在若干选项中只允许选择一项，则使用单选框。

基本语法：

```
<input type="radio" name="" value="" checked >
```

语法解释：

Value: 选中项目后传送到服务器端的值

Checked: 此项默认被选中

同组单选框的名称必须相同。

确认按钮 submit

实现表单内容的提交。

基本语法：

```
<input type="submit" name="" value="">
```

语法解释：

Value: 按钮上显示的文字（若不设置，默认值由浏览器决定）

重置按钮 reset

可以清除表单已填写的内容，恢复成默认的表单内容设定。

基本语法:

```
<input type="reset" name="" value="">
```

语法解释:

Value: 按钮上显示的文字（若不设置，默认由浏览器决定）

列表元素<select>、<option>

使用 SELECT 元素和 OPTION 元素可以创建标准的下拉式菜单或列表，这样可以避免使用过多的单选或复选框，从而节省网页空间。

基本语法:

```
<select name="" size="" multiple>
  <option value="" selected>选项
  <option value="">选项
</select>
```

语法解释:

select 中的 option 可以不断添加。

select 中的属性:

Name: 列表的名称

Size: 显示项目的数目，如果为 1，则为下拉菜单，默认值为 1。

Multiple: 列表中的项目可以多选

option 中的属性:

Value: 选项值

Selected: 默认选项

文字域元素<textarea>

这个元素可以在网页中实现一个多行的文字域，使用户可以输入更多的文本。

基本语法:

```
<textarea name="" rows="" cols="">
文字域中显示的内容
</textarea>
```

语法解释:**Name:** 文字域的名称**Rows:** 文字域的行数**Clos:** 文字域的列数**表单综合范例****文件范例: 6-1.htm**

```

01: <html>
02: <head>
03: <title> 表单</title>
04: </head>
05: <body>
06: <h1 align="center">注册信息</h1>
07: <form name="reg" action="6-1.htm" method="post">
08: <br>用户名: <input name="username" type="text" size="12"
maxlength="12">
09: <br>密码: <input name="password" type="password"
size="12" maxlength="20">
10: <br>确认密码:<input name="cmfpassword" type="password"
size="12" maxlength="20">
11: <br>性别: <input name="sex" type="radio" value="male"
checked>男<input name="sex" type="radio" value="female">女
12: <br>出生年份: <select name="year">
13:         <option value="1979" selected>1979</option>
14:         <option value="1980">1980</option>
15:         <option value="1981">1981</option>
16:         <option value="1982">1982</option>
17:         <option value="1983">1983</option>
18:         </select>
19:
20: <br>爱好: <input name="music" type="checkbox" value=
"music" checked> 音乐 <input name="movie" type="checkbox"
value="movie"> 电影 <input name="football" type="checkbox"
value="football">足球
21: <br>个性签名: <textarea name="qm" cols="50" rows="5">
</textarea>
22: <input name="submit" type="submit" value="注册">&nbsp;  

```

 <input name="reset" type="reset" value="重填">

23: </form>

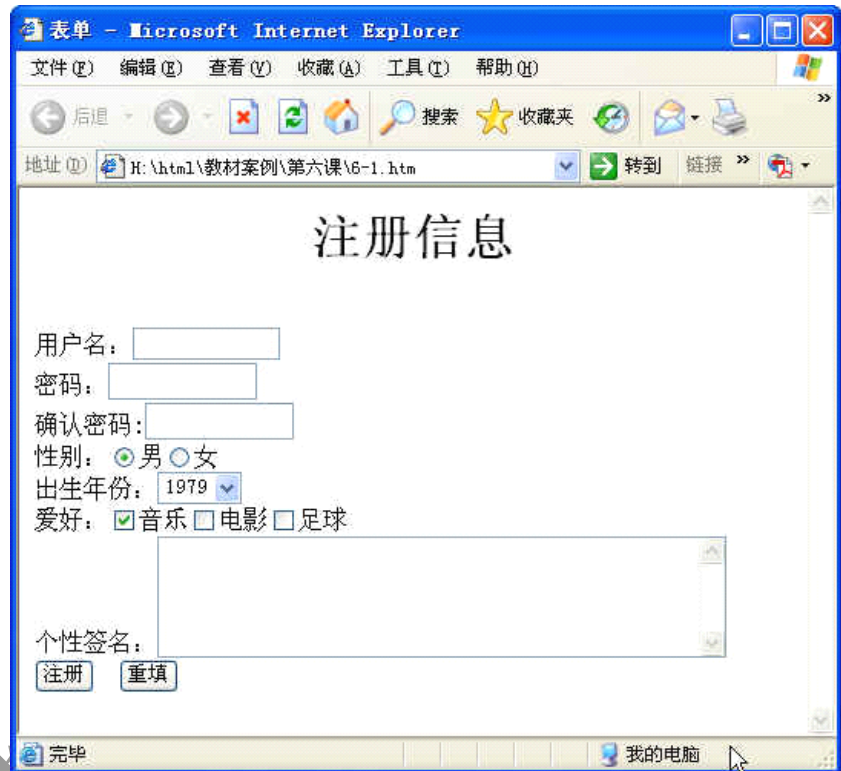
24: </body>

25: </html>

文件说明:

第 07~23 行是表单。

显示效果:



常用 javascript 代码

◆ 插入当前年月

```
<script language=JavaScript>
today=new Date();
function initArray(){
this.length=initArray.arguments.length
for(var i=0;i<this.length;i++)
this[i+1]=initArray.arguments[i] }
var d=new initArray(
"星 期 日",
"星 期 一",
"星 期 二",
```

```

"星 期 三",
"星 期 四",
"星 期 五",
"星 期 六");
document.write(
"<font color=##000000 style='font-size:9pt;font-family: 宋体'> ",
today.getYear(),"年",
today.getMonth()+1,"月",
today.getDate(),"日"," ",
d[today.getDay()+1],
"</font>" );
</script>

```

◆ 跑马灯效果

```

<script Language="JavaScript">
var msg="建议使用 1024X768 的分辨率";
var interval = 100;
var spacelen = 120;
var space10=" ";
var seq=0;
function Scroll() {
    len = msg.length;
    window.status = msg.substring(0, seq+1);
    seq++;
    if ( seq >= len ) {
        seq = spacelen;
        window.setTimeout("Scroll2();", interval );
    }
    else
        window.setTimeout("Scroll();", interval );
}
function Scroll2() {
    var out="";
    for (i=1; i<=spacelen/space10.length; i++) out +=
        space10;    //头部添加空格
    out = out + msg;    //尾部加上消息字符串
    len=out.length;
    window.status=out.substring(seq, len);    //向左移动

```

```
seq++;  
if ( seq >= len ) { seq = 0; };  
window.setTimeout("Scroll2();", interval );  
}  
Scroll();  
</script>
```

思远教育版权所有

课后自测

课后自测是对学员学习进度的检查，帮助学员找出对本课知识点理解的不足。请学员在理论授课完成后立即完成下列习题，对于不能准确回答或有疑问的问题，请即刻向授课教员提出。

一、 填空题

- 1. 列举三种网页中可以使用的声音文件_____、_____、_____。
- 2. 框架 的 标 记 有 _____、_____、_____、_____。
- 3. 表单是_____。
- 4. 表 单 由 两 个 部 分 组 成 ， 一 个 是 _____ ， 另 一 个 是 _____。
- 5. INPUT 元素的 type 属性的取值有_____。
- 6. 单选框的_____属性表示默认选中，列表选项的_____属性表示默认选中。

二、 问答题

- 1. 阐述 FORM 中 method 的两个属性值 post 和 get 的区别。
- 2. <MARQUEE>有哪些属性，这些属性的作用是什么？
- 3. 建立一个如下图的框架结构，并使 menu 框架中的链接在 main 框架中显示。

Top	
menu	Main
Foot	

附录一 html 语法

注：按英文字母顺序排列

卷 标	属 性
<! -- ... -->	批注
<a href target>	指定超级链接的分割窗口
	指定锚名称的超级链接
<a href>	指定超级链接
	被连结点的名称
<address>...</address>	用来显示电子邮箱地址
	粗体字
<base target>	指定超级链接的分割窗口
<basefont size>	更改预设字形大小
<bgsound src>	加入背景音乐
<big>	显示大字体
<blink>	闪烁的文字
<body text link vlink>	设定文字颜色
<body>	显示本文
 	换行
<caption align>	设定表格标题位置
<caption>...</caption>	为表格加上标题
<center>	向中对齐
<cite>...</cite>	用于引经据典的文字
<code>...</code>	用于列出一段程序代码
<comment>...</comment>	加上批注
<dd>	设定定义列表的项目解说
<dfn>...</dfn>	显示“定义”文字
<dir>...</dir>	列表文字卷标
<dl>...</dl>	设定定义列表的卷标
<dt>	设定定义列表的项目
	强调之用
	任意指定所用的字形
	设定字体大小
<form action>	设定户动式窗体的处理方式
<form method>	设定户动式窗体之资料传送方式

<frame marginheight>	设定窗口的上下边界
<frame marginwidth>	设定窗口的左右边界
<frame name>	为分割窗口命名
<frame noresize>	锁住分割窗口的大小
<frame scrolling>	设定分割窗口的滚动条
<frame src>	将 html 文件加入窗口
<frameset cols>	将窗口分割成左右的子窗口
<frameset rows>	将窗口分割成上下的子窗口
<frameset>...</frameset>	划分分割窗口
<h1>~<h6>	设定文字大小
<head>	标示文件信息
<hr>	加上分网格线
<html>	文件的开始与结束
<i>	斜体字
	调整图形影像的位置
	为你的图形影像加注
	加入影片
	插入图片并预设图形大小
	插入图片并预设图形的左右边界
	预载图片功能
	设定图片边界
	插入图片
	插入图片并预设图形的上下边界
<input type name value>	在窗体中加入输入字段
<isindex>	定义查询用窗体
<kbd>...</kbd>	表示使用者输入文字
<li type>...	列表的项目（可指定符号）
<marquee>	跑马灯效果
<menu>...</menu>	条列文字卷标
<meta name="refresh" content url>	自动更新文件内容
<multiple>	可同时选择多项的列表栏
<noframe>	定义不出现分割窗口的文字
...	有序号的列表
<option>	定义窗体中列表栏的项目

<p align>	设定对齐方向
<p>	分段
<person>...</person>	显示人名
<pre>	使用原有排列
<samp>...</samp>	用于引用字
<select>...</select>	在窗体中定义列表栏
<small>	显示小字体
<strike>	文字加横线
	用于加强语气
<sub>	下标字
<sup>	上标字
<table border=n>	调整表格的宽线高度
<table cellpadding>	调整数据域位之边界
<table cellspacing>	调整表格线的宽度
<table height>	调整表格的高度
<table width>	调整表格的宽度
<table>...</table>	产生表格的卷标
<td align>	调整表格字段之左右对齐
<td bgcolor>	设定表格字段之背景颜色
<td colspan rowspan>	表格字段的合并
<td nowrap>	设定表格字段不换行
<td valign>	调整表格字段之上下对齐
<td width>	调整表格字段宽度
<td>...</td>	定义表格的数据域位
<textarea name rows cols>	窗体中加入多少列的文字输入栏
<textarea wrap>	决定文字输入栏是自动否换行
<th>...</th>	定义表格的标头字段
<title>	文件标题
<tr>...</tr>	定义表格美一行
<tt>	打字机字体
<u>	文字加底线
<ul type>...	无序号的列表（可指定符号）
<var>...</var>	用于显示变量

附录二 html 标记快速参考

文件类型 `<html></html>` （放在档案的开头与结尾）
 文件主题 `<title></title>` （必须放在「文头」区块内）
 文头 `<head></head>` （描述性资料，像是「主题」）
 文体 `<body></body>` （文件本体）

结构性定义（由浏览器控制的显示风格）

标题 `<h?></h?>` （从 1 到 6，有六层选择）
 3.0 标题的对齐 `<h? align=left|center|right></h?>`
 3.0 区分 `<div></div>`
 3.0 区分的对齐 `<div align=left|right|center|justify></div>`
 引文区块 `<blockquote></blockquote>` （通常会内缩）
 强调 `` （通常会以斜体显示）
 特别强调 `` （通常会以加粗显示）
 引文 `<cite></cite>` （通常会以斜体显示）
 码 `<code></code>` （显示原始码之用）
 样本 `<samp></samp>`
 键盘输入 `<kbd></kbd>`
 变数 `<var></var>`
 定义 `<dfn></dfn>` （有些浏览器不提供）
 地址 `<address></address>`
 3.0 大字 `<big></big>`
 3.0 小字 `<small></small>`

与外观相关的标记

加粗 ``
 斜体 `<i></i>`
 3.0 底线 `<u></u>` （尚有些浏览器不提供）
 3.0 删除线 `<s></s>` （尚有些浏览器不提供）
 3.0 下标 ``
 3.0 上标 ``
 打字机体 `<tt></tt>` （用单空格字型显示）
 预定格式 `<pre></pre>` （保留文件中空格的大小）
 预定格式的宽度 `<pre width=?></pre>` （以字元计算）
 n1.0+ 向中看齐 `<center></center>` （文字与图片都可以）
 n1.0 闪耀 `<blink></blink>` （有史以来最被嘲弄的标记）
 n1.0 字体大小 `` （从 1 到 7）
 n1.0 改变字体大小 ``
 n1.0 基本字体大小 `<basefont size=?>` （从 1 到 7；内定为 3）
 n2.0 字体颜色 ``

连结与图形

连结 ``

连结到锚点 ``（如果锚点在另一个档案）

``（如果锚点目前的档案）

n2.0 连结到目的视框 ``

设定锚点 ``

图形 ``

图形看齐方式 ``

n1.0 图形看齐方式 ``

取代文字 ``（如果没有办法显示图形则显示此文字）

点选图 ``（需要 cgi 程式）

n2.0 点选图 ``

n2.0 地图 `<map name="***"></map>`（描述地图）

n2.0 段落 `<area shape="rect" coords=",,,," href="url"|nohref>`

3.0 大小 ``（以 pixels 为单位）

n1.0 图形边缘 ``（以 pixels 为单位）

n1.0 图形边缘空间 ``（以 pixels 为单位）

n1.0 低解析度图形 ``

n1.1 用户端拉 `<meta http-equiv="refresh" content="?; url=url">`（使用端自动更新）

n2.0 内嵌物件 `<embed src="url">`（将物件插入页面）

n2.0 内嵌物件大小 `<embed src="url" width="?" height="?">`

分隔

段落 `<p>`（通常是两个 return）

3.0 段落 `<p></p>`（新定义成容器型标记）

3.0 文字看齐方式 `<p align=left|center|right></p>`

换行 `
`（一个 return）

n1.0 文字部份看齐方式 `<br clear=left|right|all>`（与图形合用时）

横线 `<hr>`

n1.0 横线对齐 `<hr align=left|right|center>`

n1.0 横线厚度 `<hr size=?>`（以 pixels 为单位）

n1.0 横线宽度 `<hr width=?>`（以 pixels 为单位）

n1.0 横线比率宽度 `<hr width=%>`（以页宽为 100%）

n1.0 实线 `<hr noshade>`（没有立体效果）

n1.0 不可换行 `<nobr></nobr>`（不换行）

n1.0 可换行处 `<wbr>`（如果需要，可在此断行）

列举（可以巢状列举）

无次序式列举 ``（`` 放在每一项前）

n1.0 公布式列举 <ul type=disc|circle|square> (定义全部的列举项)

<li type=disc|circle|square> (定义这个及其後的列举项)

有次序式列举 (放在每一项前)

n1.0 数标型态 <ol type=a|i|i|1> (定义全部的列举项)

<li type=a|i|i|1> (定义这个及其後的列举项)

n1.0 起始数字 <ol value=?> (定义全部的列举项)

<li value=?> (定义这个及其後的列举项)

定义式列举 <dl><dt><dd></dl> (<dt>项目, <dd>定义)

表单式列举 <menu></menu> (放在每一项前)

目录式列举 <dir></dir> (放在每一项前)

背景与颜色

3.0 重复排列的背景 <body background="url">

n1.1+ 背景颜色 <body bgcolor="#\$\$\$\$\$"> (依序为红、绿、蓝)

n1.1+ 文字颜色 <body text="#\$\$\$\$\$">

n1.1+ 连结颜色 <body link="#\$\$\$\$\$">

n1.1+ 看过的连结 <body vlink="#\$\$\$\$\$">

n1.1 使用中的连结 <body alink="#\$\$\$\$\$">

特殊字元 (以下标记需用小写)

特别符号 &#?; (其中 ? 代表 iso 8859-1 的编码)

< > & & " "

注册商标 tm ®

n1.0+ 注册商标 tm ®

著作权符号 ©

n1.0+ 著作权符号 ©

表单 (通常需要与 cgi 程式配合)

定义表单 <form action="url" method=get|post></form>

n2.0 上传档案 <form enctype="multipart/form-data"></form>

输入栏位

<input type="text|password|checkbox|radio|image|hidden|submit|reset">

栏位名称 <input name="***">

栏位内定值 <input value="***">

已选定 <input checked> (适用於 checkboxes 与 radio boxes)

栏位宽度 <input size=?> (以字元数为单位)

最长字数 <input maxlength=?> (以字元数为单位)

下拉式选单 <select></select>

下拉式选单名称 <select name="***"></select>

选单项目数量 <select size=?></select>

多选式选单 <select multiple> (多选)

选项 <option>

内定选项 <option selected>

文字输入区 <textarea rows=? cols=?></textarea>

输入区名称 <textarea name="***"></textarea>

n2.0 输入区换行方式 <textarea wrap=off|virtual|physical></textarea>

表格

3.0 定义表格 <table></table>

3.0 表格框线 <table border></table> (有或没有)

n1.1 表格框线 <table border=?></table> (可以设定数值)

n1.1 储存格左右留白 <table cellpadding=?>

n1.1 储存格上下留白 <table cellspacing=?>

n1.1 表格宽度 <table width=?> (以 pixels 为单位)

n1.1 宽度比率 <table width=%> (页宽为 100%)

3.0 表格列 <tr></tr>

3.0 表格列内容看齐 <tr align=left|right|center valign=top|middle|bottom>

3.0 储存格 <td></td> (须与列并用)

3.0 储存格内容看齐 <td align=left|right|center valign=top|middle|bottom>

3.0 不换行 <td nowrap>

n3.0 储存格背景颜色 <td bgcolor=#\$\$\$\$\$>

3.0 储存格横向连接 <td colspan=?>

3.0 储存格纵向连接 <td rowspan=?>

n1.1 储存格宽度 <td width=?> (以 pixels 为单位)

n1.1 储存格宽度比率 <td width=%> (页宽为 100%)

3.0 表格标题 <th></th> (跟<td>一样, 不过会对中并加粗)

3.0 表格标题对齐 <th align=left|right|center valign=top|middle|bottom>

3.0 表格标题不换行 <th nowrap>

3.0 表格标题占几栏 <th colspan=?>

3.0 表格标题占几列 <th rowspan=?>

n1.1 表格标题宽度 <th width=?> (以 pixels 为单位)

n1.1 表格标题比率宽度 <th width=%> (页宽为 100%)

3.0 表格抬头 <caption></caption>

3.0 表格抬头看齐 <caption align=top|bottom> (在表格之上/之下)

视框 (定义与控制萤幕上的特定区域)

n2.0 视框格式总定义 <frameset></frameset> (取代<body>)

n2.0 视框行长度分配 <frameset rows=,,,></frameset> (pixels 或 %)

n2.0 视框行长度分配 <frameset rows=*></frameset> (* = 相对大小)

n2.0 视框栏宽度分配 <frameset cols=,,,></frameset> (pixels 或 %)

n2.0 视框栏宽度分配 <frameset cols=*></frameset> (* = 相对大小)

n2.0 定义个别视框 <frame> (定义个别视框)

n2.0 个别视框内容 <frame src="url">

n2.0 个别视框名称 <frame name="***"|_blank|_self|_parent|_top>
 n2.0 边缘宽度 <frame marginwidth=?> (「左」与「右」边界)
 n2.0 边缘高度 <frame marginheight=?> (「天顶」与「地底」边界)
 n2.0 卷动条 <frame scrolling="yes|no|auto">
 n2.0 不可改变大小 <frame noresize>
 n2.0 无视框时的内容 <noframes></noframes> (如果浏览器不提供视框功能的话)

杂项

说明 (浏览器不会显示)

档案型态声明 <!doctype html public "-//ietf//dtd html 2.0//en">

3.0 档案型态声明 <!doctype html public "-//w3o//dtd w3 html 3.0//en">

可搜寻 <isindex> (指示可搜寻的索引项)

n1.0 提示 <isindex prompt="***"> (要求输入的提示文字)

送出搜寻条件 (其中的问号不是数字, 而是「问号」)

基本档案的 url <base href="url"> (必须放在「文头」区段内)

n2.0 基本视窗名称 <base target="***"> (必须放在「文头」区段内)

相关性 <link rev="***" rel="***" href="url"> (必须放在「文头」区段内)

背景资讯 <meta> (必须放在「文头」区段内)

附录三 css 属性快速参考

css font properties 字体属性

font properties 属性	css version 版本	compatibility 兼容性	inherit from parent 继承性	Description 简介
font	css1/css2	ie4+ , ns4+	有	复合属性。设置或检索对象中的文本特性
color	css1	ie4+ , ns4+	有	检索或设置对象的文本颜色
font-family	css1	ie4+ , ns4+	有	设置或检索用于对象中文本的字体名称序列
font-size	css1	ie4+ , ns4+	有	设置或检索对象中的字体尺寸
font-size-adjust	css2	none	有	设置或检索用于对象中文本的字体名称序列是否强制使用同一尺寸
font-stretch	css2	none	有	设置或检索用于对象中文本的文字是否横向的拉伸变形
font-style	css1	ie4+ , ns4+	有	设置或检索对象中的字体样式
font-weight	css1	ie4+ , ns4+	有	设置或检索对象中的文本字体的粗细
text-decoration	css1	ie4+ , ns4+	无	检索或设置对象中的文本的装饰, 如下划线、闪烁等
text-underline-position	ie 专有属性	ie5.5+	有	设置或检索 text-decoration 属性定义的下划线的位置
text-shadow	css2	none	无	设置或检索对象中文本的文字是否有阴影及模糊效果
font-variant	css1	ie4+ , ns6+	有	设置或检索对象中的文本是否为小型的大写字母
text-transform	css1	ie4+ , ns4+	有	检索或设置对象中的文本的大小写
line-height	css1	ie4+ , ns4+	有	检索或设置对象的行高。即字体最底端与字体内部顶端之间的距离
letter-spacing	css1	ie4+ , ns6+	有	检索或设置对象中的文字之间的间隔
word-spacing	css1	ie4+ , ns6+	有	检索或设置对象中的单词之间插入的空格数。对于 ie4+而言仅在 mac 平台上可用

css text properties 文本属性

text properties 属性	css version 版本	Compatibility 兼容性	inherit from parent 继承性	Description 简介
text-indent	css1	ie4+ , ns4+	有	检索或设置对象中的文本的缩进
text-overflow	ie 专有属性	ie6+	无	设置或检索是否使用一个省略标记 (...) 标示对象内文本的溢出
vertical-align	css1/css2	ie4+ , ns4+	无	设置或检索对象内容的垂直对其方式
text-align	css1	ie4+ , ns4+	有	设置或检索对象中文本的对齐方式
layout-flow	ie 专有属性	ie5.5+	有	设置或检索对象内文本的流动和方向
writing-mode	ie 专有属性	ie5.5+	有	设置或检索对象的内容块固有的书写方向
direction	css2	ie5+	无	用于设置文本流的方向
unicode-bidi	css2	ie5+	无	用于同一个页面里存在从不同方向读进的文本显示。与 direction 属性一起使用
word-break	ie 专有属性	ie5+	无	设置或检索对象内文本的字内换行行为。尤其在出现多种语言时
line-break	ie 专有属性	ie5+	无	设置或检索用于日文文本的换行规则
white-space	css1	ie5.5+ , ns4+	有	设置或检索对象内空格的处理方式
word-wrap	ie 专有属性	ie5.5+	有	设置或检索当当前行超过指定容器的边界时是否断开转行
text-autospace	ie 专有属性	ie5+	无	设置或检索对象文本的自动空格和紧缩空格宽度调整的方式
text-kashida-space	ie 专有属性	ie5.5+	有	设置或检索如何拉伸字符来调节文本行排列
text-justify	ie 专有属性	ie5+	无	设置或检索对象内文本的对齐方式

css layout properties 布局属性

layout properties 属性	css version 版本	compatibility 兼容性	inherit from parent 继承性	Description 简介
clear	css1	ie4+ , ns4+	无	该属性的值指出了不允许有浮动对象的边。请参阅 float 属性
float	css1	ie4+ , ns4+	无	该属性的值指出了对象是否及如何浮动。请参阅 clear 属性
clip	css2	ie4+ , ns6+	无	检索或设置对象的可视区域。区域外的部分是透明的
overflow	css2	ie4+ , ns6+	无	检索或设置当对象的内容超过其指定高度及宽度时如何管理内容
display	css1/css2	ie4+ , ns4+	无	设置或检索对象是否及如何显示
visibility	css2	ie4+ , ns6+	无	设置或检索是否显示对象。与 display 属性不同, 此属性为隐藏的对象保留其占据的物理空间

css background properties 背景属性

background properties 属性	css version 版本	compatibility 兼容性	inherit from parent 继承性	Description 简介
background	css1	ie4+ , ns4+	无	复合属性。设置或检索对象的背景特性
background-attachment	css1	ie4+ , ns6+	无	设置或检索背景图像是随对象内容滚动还是固定的
background-color	css1	ie4+ , ns4+	无	设置或检索对象的背景颜色
background-image	css1	ie4+ , ns4+	无	设置或检索对象的背景图像
background-position	css1	ie4+ , ns6+	无	设置或检索对象的背景图像位置
background-positionx	ie 专有属性	ie5.5+	无	设置或检索对象的背景图像横坐标位
background-positiony	ie 专有属性	ie5.5+	无	设置或检索对象的背景图像纵坐标位
background-repeat	css1	ie4+ , ns4+	无	设置或检索对象的背景图像是如何铺排

css borders properties 边框属性

borders properties 属性	css version 版本	compatibility 兼容性	inherit from parent 继承性	Description 简介
border	css1	ie4+ , ns4+	无	复合属性。设置对象边框的特性
border-color	css1	ie4+ , ns6+	无	设置或检索对象边框颜色
border-style	css1	ie4+ , ns4+	无	设置或检索对象边框样式
border-width	css1	ie4+ , ns4+	无	设置或检索对象边框宽度
border-top	css1	ie4+ , ns6+	无	复合属性。设置对象顶边的特性
border-top-color	css2	ie4+ , ns6+	无	设置或检索对象顶边颜色
border-top-style	css2	ie4+ , ns6+	无	设置或检索对象顶边样式
border-top-width	css1	ie4+ , ns4+	无	设置或检索对象顶边宽度
border-right	css1	ie4+ , ns6+	无	复合属性。设置对象右边的特性
border-right-color	css2	ie4+ , ns6+	无	设置或检索对象右边颜色
border-right-style	css2	ie4+ , ns6+	无	设置或检索对象右边样式
border-right-width	css1	ie4+ , ns4+	无	设置或检索对象右边宽度
border-bottom	css1	ie4+ , ns6+	无	复合属性。设置对象底边的特性
border-bottom-color	css2	ie4+ , ns6+	无	设置或检索对象底边颜色
border-bottom-style	css2	ie4+ , ns6+	无	设置或检索对象底边样式
border-bottom-width	css1	ie4+ , ns4+	无	设置或检索对象底边宽度
border-left	css1	ie4+ , ns6+	无	复合属性。设置对象左边的特性
border-left-color	css2	ie4+ , ns6+	无	设置或检索对象左边颜色
border-left-style	css2	ie4+ , ns6+	无	设置或检索对象左边样式
border-left-width	css1	ie4+ , ns4+	无	设置或检索对象左边宽度

css pseudo-classes reference 伪类

pseudo-classes 伪类	css version 版本	Compatibility 兼容性	Description 简介
:link	css1	ie4+ , ns4+	设置 a 对象在未被访问前的样式表属性
:hover	css1/css2	ie4+ , ns4+	设置对象在其鼠标悬停时的样式表属性
:active	css1/css2	ie4+	设置对象在被用户激活（在鼠标点击与释放之间发生的事件）时的样式表属性
:visited	css1	ie4+ , ns4+	设置 a 对象在其链接地址已被访问过时的样式表属性

css table properties 表格属性

table properties 属性	css version 版本	compatibility 兼容性	inherit from parent 继承性	Description 简介
border-collapse	css2	ie5+	有	设置或检索表格的行和单元格的边是合并，还是按照标准的html样式分开
border-spacing	css2	none	有	设置或检索当表格边框独立时，行和单元格的边框在横向和纵向上的间距
caption-side	css2	none	有	设置或检索表格的caption对象是在表格的那一边
empty-cells	css2	none	有	设置或检索当表格的单元格无内容时，是否显示该单元格的边框
table-layout	css2	ie5+	有	设置或检索表格的布局算法
speak-header	css2	none	有	设置或检索表格头与其后的一系列单元格发生多少次关系

附录四 颜色名称及十六进制

颜色中文	十六进制值	英文
浅粉红	#ffb6c1	lightpink
粉红	#ffc0cb	pink
深红/猩红	#dc143c	crimson
淡紫红	#fff0f5	lavenderblush
弱紫罗兰红	#db7093	palevioletred
热情的粉红	#ff69b4	hotpink
深粉红	#ff1493	deeppink
中紫罗兰红	#c71585	mediumvioletred
暗紫色/兰花紫	#da70d6	orchid
洋李色/李子紫	#dda0dd	plum
紫罗兰	#ee82ee	violet
洋红/玫瑰红	#ff00ff	magenta
紫红/灯笼海棠	#ff00ff	fuchsia
深洋红	#8b008b	darkmagenta
紫色	#800080	purple
中兰花紫	#ba55d3	mediumorchid
暗紫罗兰	#9400d3	darkviolet
暗兰花紫	#9932cc	darkorchid
靛青/紫兰色	#4b0082	indigo
蓝紫罗兰	#8a2be2	blueviolet
中紫色	#9370db	mediumpurple
中暗蓝色/中板岩蓝	#7b68ee	mediumslateblue
石蓝色/板岩蓝	#6a5acd	slateblue
暗灰蓝色/暗板岩蓝	#483d8b	darkslateblue
淡紫色/熏衣草淡紫	#e6e6fa	lavender
幽灵白	#f8f8ff	ghostwhite
纯蓝	#0000ff	blue
中蓝色	#0000cd	mediumblue
午夜蓝	#191970	midnightblue
暗蓝色	#00008b	darkblue
海军蓝	#000080	navy
皇家蓝/宝蓝	#4169e1	royalblue
矢车菊蓝	#6495ed	cornflowerblue
亮钢蓝	#b0c4de	lightsteelblue
亮蓝灰/亮石板灰	#778899	lightslategray
灰石色/石板灰	#708090	slategray
闪兰色/道奇蓝	#1e90ff	dodgerblue
钢蓝/铁青	#4682b4	steelblue
亮天蓝色	#87cefa	lightskyblue
天蓝色	#87ceeb	skyblue

深天蓝	#00bfff	deepskyblue
亮蓝	#add8e6	lightblue
粉蓝色/火药青	#b0e0e6	powderblue
军兰色/军服蓝	#5f9ea0	cadetblue
蔚蓝色	#00ffff	azure
淡青色	#e0ffff	lightcyan
弱绿宝石	#afeeee	paleturquoise
青色	#00ffff	cyan
浅绿色/水色	#00ffff	aqua
暗绿宝石	#00ced1	darkturquoise
暗瓦灰色/暗石板灰	#2f4f4f	darkslategray
暗青色	#008b8b	darkcyan
水鸭色	#008080	teal
中绿宝石	#48d1cc	mediumturquoise
浅海洋绿	#20b2aa	lightseagreen
绿宝石	#40e0d0	turquoise
宝石碧绿	#7fffd4	aquamarine
中宝石碧绿	#66cdaa	mediumaquamarine
中春绿色	#00fa9a	mediumspringgreen
薄荷奶油	#f5ffa	mintcream
春绿色	#00ff7f	springgreen
中海洋绿	#3cb371	mediumseagreen
海洋绿	#2e8b57	seagreen
蜜色/蜜瓜色	#ff0000	honeydew
淡绿色	#90ee90	lightgreen
弱绿色	#98fb98	palegreen
暗海洋绿	#8fbc8f	darkseagreen
闪光深绿	#32cd32	limegreen
闪光绿	#00ff00	lime
森林绿	#228b22	forestgreen
纯绿	#008000	green
暗绿色	#006400	darkgreen
黄绿色/查特酒绿	#7ff000	chartreuse
草绿色/草坪绿	#7cfc00	lawngreen
绿黄色	#adff2f	greenyellow
暗橄榄绿	#556b2f	darkolivegreen
黄绿色	#9acd32	yellowgreen
橄榄褐色	#6b8e23	olivedrab
米色/灰棕色	#f5f5dc	beige
象牙色	#ffff00	ivory
浅黄色	#ffffe0	lightyellow
纯黄	#ffff00	yellow
橄榄	#808000	olive

暗黄褐色/深卡叽布	#bdb76b	darkkhaki
柠檬绸	#fffacd	lemonchiffon
黄褐色/卡叽布	#f0e68c	khaki
金色	#ffd700	gold
金菊黄	#daa520	goldenrod
暗金菊黄	#b8860b	darkgoldenrod
花的白色	#fffaf0	floralwhite
浅黄色/小麦色	#f5deb3	wheat
鹿皮色/鹿皮靴	#ffe4b5	moccasin
橙色	#ffa500	orange
番木色/番木瓜	#ffefd5	papayawhip
白杏色	#ffeacd	blanchedalmond
古董白	#faebd7	antiquewhite
茶色	#d2b48c	tan
硬木色	#deb887	burlywood
陶坯黄	#ffe4c4	bisque
深橙色	#ff8c00	darkorange
桃肉色	#ffdab9	peachpuff
沙棕色	#f4a460	sandybrown
巧克力色	#d2691e	chocolate
重褐色/马鞍棕色	#8b4513	saddlebrown
海贝壳	#fff5ee	seashell
珊瑚	#ff7f50	coral
橙红色	#ff4500	orangered
番茄红	#ff6347	tomato
浅玫瑰色/薄雾玫瑰	#ffe4e1	mistyrose
雪白色	#ffaafa	snow
淡珊瑚色	#f08080	lightcoral
玫瑰棕色	#bc8f8f	rosybrown
印度红	#cd5c5c	indianred
纯红	#ff0000	red
棕色	#a52a2a	brown
深红色	#8b0000	darkred
栗色	#800000	maroon
纯白	#ffffff	white
淡灰色	#dcdcdc	gainsboro
浅灰色	#d3d3d3	lightgrey
银灰色	#c0c0c0	silver
深灰色	#a9a9a9	darkgray
灰色	#808080	gray
暗淡灰	#696969	dimgray
纯黑	#000000	black

意见反馈表

您好,请您抽出宝贵的时间填写这份反馈表,并将此页剪下提交您所在项目部的职业指导师。我们会在以后的工作中充分考虑您的意见和建议,以便更好的交流。

书名:《HTML&CSS》

个人资料

姓 名: _____ 性 别: _____ 年 龄: _____

项目部: _____ E-mail: _____ 文化程度: _____

1. 您对本书封面设计的满意程度:

☐ 很满意 ☐ 比较满意 ☐ 一般 ☐ 不满意
☐ 改进建议 _____

2. 您对本书的总体满意程度:

从文字的角度	<input type="checkbox"/> 很满意	<input type="checkbox"/> 比较满意	<input type="checkbox"/> 一般	<input type="checkbox"/> 不满意
从技术角度	<input type="checkbox"/> 很满意	<input type="checkbox"/> 比较满意	<input type="checkbox"/> 一般	<input type="checkbox"/> 不满意
从编写思路角度	<input type="checkbox"/> 很满意	<input type="checkbox"/> 比较满意	<input type="checkbox"/> 一般	<input type="checkbox"/> 不满意
从版式设计角度	<input type="checkbox"/> 很满意	<input type="checkbox"/> 比较满意	<input type="checkbox"/> 一般	<input type="checkbox"/> 不满意

3. 您希望书中程序代码的比例是多少:

☐ 少量的代码辅以大量的文字 ☐ 代码与文字的比例相当
☐ 大量的代码辅以少量的文字

4. 本书最令您满意的是:

a. _____
 b. _____
 c. _____

5. 您使用本书时遇到了哪些困难:

a. _____
 b. _____
 c. _____

6. 您希望本书在哪些方面进行改进:

a. _____
 b. _____

c. _____

7. 您在学习中遇到了什么困难:

a. _____

b. _____

c. _____

8. 本书有无重大错误? 如有, 请详述:

9. 您希望随本书配套提供哪些相关内容?

☐ 教学大纲 ☐ 电子教案 ☐ 习题答案 ☐ 无所谓

☐ 其他 _____

10. 您的其他要求:

11. 您认为本书有哪些错误:

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

章节 _____ 页码 _____ 行 _____ 图号 _____ 错误 _____

应改为 _____

目录

第一课	html 文档和网页文本	3
1.1	什么是因特网	6
1.1.1	因特网相关概念	6
1.2	什么是 html	8
1.2.1	什么是 web 标准	8
1.2.2	为什么要使用标准	9
1.2.3	web 设计工具	10
1.3	html 文档的结构	10
1.3.1	html 文档的元素	11
1.3.2	html 文档的头部	11
1.3.3	html 文档的主体	14
1.4	标记和属性	14
1.5	文本的插入和修改	15
1.5.1	标题字标记<h>	15
1.5.2	设置字体、字号和颜色属性	15
1.5.3	常用字符级标记	16
1.5.4	特殊字符与注释文字	16
	文字标记综合范例	17
	课后自测	19
	上机任务书	23
	任务背景:	23
	任务目标:	23
	上机指导书	24
第二课	段落、列表、图片和超链接	33
2.1	段落级标记	36
2.1.1	段落标记<p>	36
	段落标记对齐属性 align	36
2.1.2	换行标记 	37
2.2	水平线标记<hr>	37
2.2.1	水平线宽度属性 width	37
2.2.2	水平线高度属性 size	38
2.2.3	水平线颜色属性 color	38
2.2.4	水平线排列属性 align	38
	水平线标记综合范例	38
2.3	居中标记<center>	39
2.4	列表标记	39
	列表项	39
2.5	无序列表	40
2.5.1	无序列表的类型属性 type	40

2.6 有序列表.....	40
2.6.1 有序列表的类型属性 type.....	41
2.6.2 有序列表的类型属性 start.....	41
2.7 列表嵌套.....	41
2.7.1 无序列表和有序列表的嵌套.....	42
2.8 网页常用图片格式.....	42
2.8.1 图像标记的属性.....	43
2.8.2 图片的源文件属性 src.....	43
2.8.3 图片的提示文字属性 alt.....	43
2.8.4 图片的宽度、高度属性 width、height.....	44
2.8.5 图片的边框属性 border.....	44
2.8.6 图片的垂直间距属性 vspace.....	44
2.8.7 图片的水平间距属性 hspace.....	44
2.8.8 图片的排列属性 align.....	45
图片标记综合范例.....	45
2.9 什么是超链接.....	46
2.10 链接的标记<a>.....	46
2.11 链接的类型.....	47
2.11.1 内部链接.....	47
2.11.2 外部链接.....	47
2.11.3 邮件链接.....	48
2.11.4 书签链接.....	48
超链接综合范例.....	49
课后自测.....	53
上机任务书.....	57
任务背景:	57
任务目标:	57
上机指导书.....	58
第三课 软件开发过程.....	65
3.1 软件的发展和软件危机.....	68
3.2 软件工程.....	69
3.2.1 软件工程的定义.....	69
3.2.2 软件工程的要素.....	69
3.2.3 软件工程的目标与主要环节.....	70
3.3 软件工程过程和软件生存周期.....	70
3.3.1 软件工程过程 (Software Engineering Process).....	70
3.3.2 软件生存周期 (life cycle).....	71
3.3.3 软件生存周期模型.....	71
3.4 瀑布模型.....	72
3.4.1 瀑布模型的流程图.....	73
3.4.2 计划阶段.....	73
3.4.3 开发阶段.....	75
3.4.4 运行阶段.....	78
课后自测.....	80

上机任务书.....	83
任务背景:	83
任务目标:	83
第四课 html 表格	92
4.1 前言.....	95
4.2 网页布局的基本概念.....	95
4.3 制作表格.....	96
4.3.1 表格标记属性.....	96
4.4 表格标题与表头.....	97
4.4.1 表格标题<caption>.....	97
4.4.2 表格的表头<th>.....	98
4.5 行标记<tr>属性.....	98
4.6 单元格标记<td>、<th>属性.....	99
表格基本标记综合范例.....	99
4.6.1 单元格跨行属性 rowspan.....	100
4.6.2 单元格跨列属性 colspan.....	100
单元格跨行、跨列属性运用案例.....	101
4.7 表格嵌套.....	102
课后自测.....	104
上机任务书.....	107
任务背景:	107
任务目标:	107
上机指导书.....	108
第五课 css 层叠样式表	121
5.1 什么是 dhtml.....	124
5.2 css 的概念及版本.....	124
5.3 css 的样式规则.....	124
5.4 css 基本语法.....	125
5.5 css 的基本写法.....	126
课后自测.....	130
上机任务书.....	133
任务背景:	133
任务目标:	133
上机指导书.....	134
选修内容	141
多媒体标记.....	143
背景音乐标记<bgsound>.....	143
嵌入多媒体内容.....	143
滚动文字标记<marquee>.....	144
滚动方向属性 direction.....	144
滚动方式属性 behavior.....	144
滚动速度属性 scrollamount.....	144
滚动延迟属性 scrolldelay.....	145
滚动范围属性 width、height.....	145

滚动背景颜色属性 bgcolor	145
滚动文字标记综合范例	145
框架结构	146
框架标记	146
浮动框架<iframe>	151
框架与链接	152
表单	153
表单标记<form>	153
输入标记<input>	154
列表元素<select>、<option>	156
文字域元素<textarea>	156
表单综合范例	157
常用 javascript 代码	158
课后自测	161
附录一 html 语法	162
附录二 html 标记快速参考	165
附录三 css 属性快速参考	171
附录四 颜色名称及十六进制	176
意见反馈表	179