

第2章 HTML和XHTML

2.1 HTML和XHTML概述

2.2 HTML和XHTML的格式与标记

2.3 使用表格

2.4 使用框架

2.5 使用超链接

2.6 制作表单

2.7 CCS样式设计

2.1 HTML和XHTML概述

2.1.1 什么是XHTML

XHTML是EXtensible HyperText Markup Language的英文缩写，即可扩展的超文本标记语言，它不需要编译，可以直接由浏览器执行（属于浏览器解释型语言）：

- XHTML是用来代替HTML的，是2000年W3C（万维网联盟）公布发行的。
- XHTML是一种**增强了的HTML**，它的可扩展性和灵活性将适应未来网络应用更多的需求。
- XHTML是基于XML的应用。
- XHTML更简洁更严紧。
- XHTML也可以说就是HTML4.0的一个升级版本（W3C描述它为HTML 4.01）。

除此之外XHTML和HTML基本相同。

2.1.2 HTML与XHTML的区别

(1) 大小写

HTML不区分大小写，HTML元素和属性名称可以是大写、小写或大小写混合。但是XHTML中大小是敏感的，例如，XHTML中，`<body>`和`<BODY>`是两个完全不同的标记。因此XHTML文档要求所有元素和属性名称必须小写，而属性值则大小写均可。

(2) 标记嵌套

在HTML中，即使使用了不正确的嵌套，一样可以在很多浏览器中使用；而XHTML则要求嵌套必须完全正确。例如，`<i>姓名<i>`不能写成`<i>姓名</i>`。

(3) 有否结束标记

在HTML中，有些标记是可以没有结束标记的，而XHTML要求所有标记都必须有结束标记，例如，HTML中的
在XHTML中必须写成
</br>或者简单地写成
。注意：
中的斜杠前有一个空格。

(4) 引号

HTML中的属性值可以用引号引起来，也可以不使用引号，但XHTML中要求所有属性值都必须加引号，即使是数字也需要加引号。例如：

```

```

除此之外，XHTML还要求属性值不能省略。

(5) id和name

HTML中每个元素都可以定义name属性，也引入了id属性，这两个属性都可以标识某一个元素。而在XHTML中，每个元素只有一个的标识属性id。

(6) 样式的使用

在不使用样式表的情况下，HTML中每一个样式都可以直接使用“属性名=属性值”的方法设置样式。例如：

```
<ul type=square>  
  <li>足球</li>  
  <li>排球</li>  
  <li>篮球</li>  
</ul>
```

其中，type属性设置了无序列表中的项目符号的样式为小方块。但在XHTML中，如果不使用样式表，只能通过style属性来设置样式。在XHTML中上述代码需修改为：

```
<ul style=:list-style-type:square">  
  <li>足球</li>  
  <li>排球</li>  
  <li>篮球</li>  
</ul>
```

2.2 HTML和XHTML的格式与标记

2.2.1 基本格式

一个典型的网页结构如下：

```
<html>
```

```
  <head>
```

```
    <title> 网页的标题 </title>
```

```
  </head>
```

```
  <body>
```

```
    网页的内容
```

```
  </body>
```

```
</html>
```

XHTML文档结构

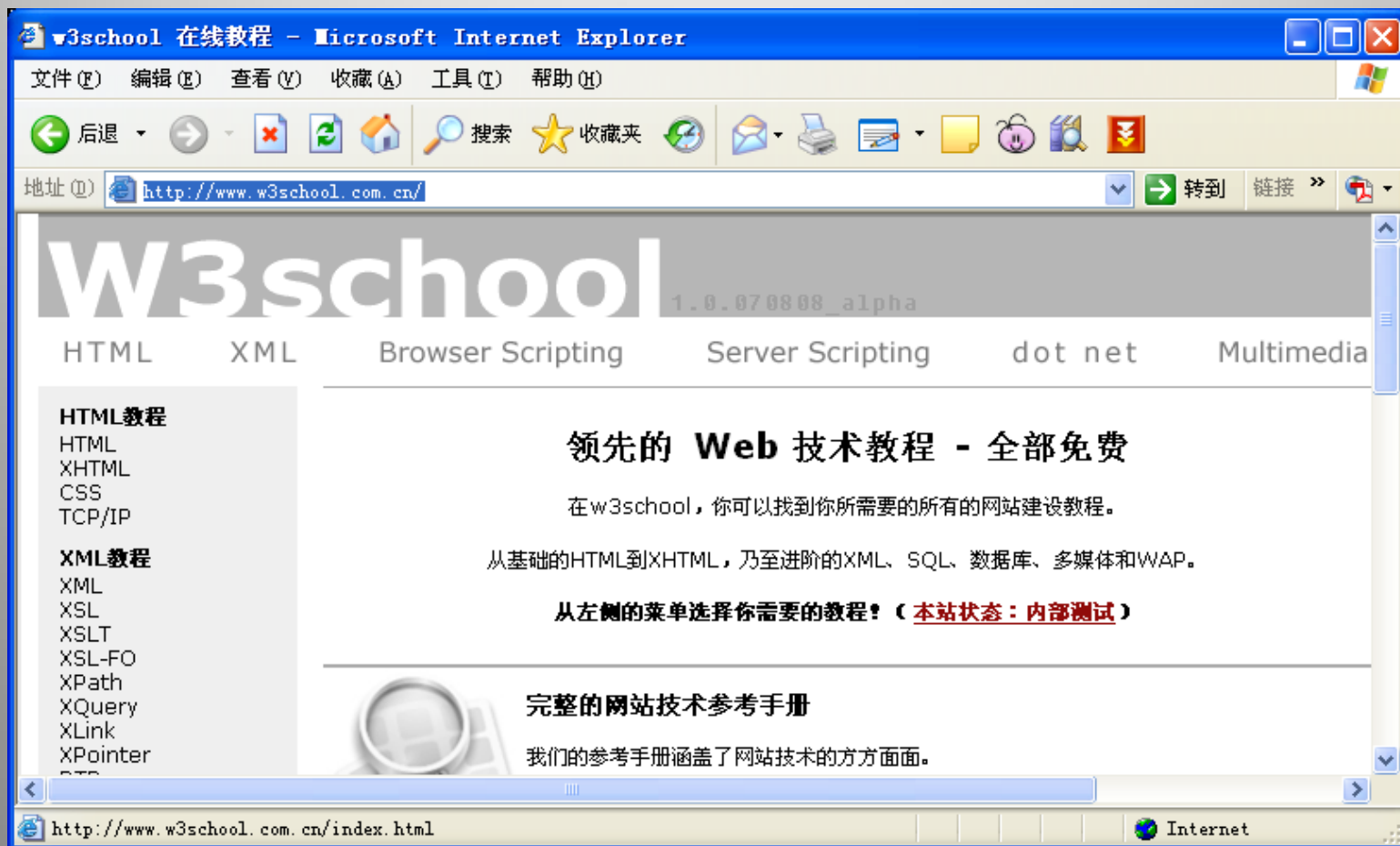
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
  <head>  
    <title>网页标题</title>  
  </head>  
  <body>  
    ...  
  </body>  
</html>
```

① XHTML文档中第一句是文档类型声明<!DOCTYPE>，用于指定网页的语法规则。

② <html>标记中用xmlns属性指定了xml的命名空间。

参考资料

- 参考资料: <http://www.w3school.com.cn/>



2.2.2 HTML的标记

常用的标记如表2.1所示。

表2.1 HTML其他常用的标记

| 标记 | 说明 |
|---------------------------|------------------------------|
| | 字体标记，通过其属性可以设置文字的字体、字号和颜色 |
| <!--注释内容--> | 注释标记 |
| | 强制文本换行标记，行与行之间不空行 |
| <p> | 段落标记，段与段之间空一行 |
| <pre></pre> | 预先排好格式标记 |
| <h n ></h n > | 标题文字标记， $n=1, 2, 3, 4, 5, 6$ |
| 或 | 文字粗体标记 |
| <i></i> | 文字斜体标记 |
| <u></u> | 文字加下划线标记 |

| | |
|---------------------|-----------|
| <a> | 超文本链接标记 |
| <center></center> | 文字居中标记 |
| <div></div> | 文字块标记 |
| <marquee></marquee> | 移动文字标记 |
| | 有序列表标记 |
| | 无序列表标记 |
| | 列表项标记 |
| <dl></dl> | 自定义列表标记 |
| <dt></dt> | 自定义列表项标记 |
| <dd></dd> | 自定义列表子项标记 |
| <table></table> | 表格标记 |
| <caption></caption> | 定义表标题标记 |
| <tr></tr> | 定义表格行标记 |
| <td></td> | 定义表格单元格标记 |
| <th></th> | 定义列标题标记 |
| <form></form> | 表单标记 |
| <frameset></frame> | 框架集标记 |
| <frame> | 框架标记 |
| <iframe></iframe> | 框架标记 |
| | 图片标记 |
| <hr> | 水平线标记 |
| <embed> | 多媒体标记 |

2.2.3 HTML的长度单位

HTML的长度单位主要分绝对单位和相对单位两种：

■绝对单位：cm（厘米）、mm（毫米）、px（像素）、in（英寸）、pt（点， $1\text{pt}=1/72$ 英寸）、pc（ $1\text{pc}=12$ 点）。

■相对单位：%（百分比）、em（1em相当于当前浏览器字体中大写字母M的宽度）、ex（1ex相当于当前浏览器默认字体中小写字母x的高度）。

常用的

- 标题文字标记
 - `<h1 align=left|center|right>`
标题文字 `</h1>`
- 文字样式标记``
 - “color”
 - “face”
 - “size”
- 文字对齐标记
 - `<center> ... </center>`
 - `<left> ... </left>`
 - `<right> ... </right>`
- `<p>`分段标记
- `
`换行标记
- `<hr>`水平线标记
- `<pre>`预格式化标记

e-01.HTML

显示2号标题字

显示蓝色隶书字体

显示删除划线

aaaa@buu.com.cn(显示地址)

显示上标: x^3

显示下标: x_2

预格式化标记应用

```
*  
* *  
*
```

```
<html>  
<head>  
  <title>页面修饰标记应用</title>  
</head>  
<body bgcolor=#f0f0f0>  
<h2 align="center" >显示2号标题字</h2>  
<center>  
<p>  
<font color="blue" face="隶书" size=5>显示蓝色隶书字体  
</font>  
<br><s>显示删除划线</s>  
<address>aaaa@buu.com.cn(显示地址)</address>  
<hr color="red">  
显示上标: x<sup>3</sup><br>  
显示下标: x<sub>2</sub>  
</center>  
<hr color=#00FFFF size=4>  
<pre>  
  
预格式化标记应用  
*  
* *  
*  
  
</pre>  
</body>  
</html>
```

2.3 使用表格

表格的一般样式：

表标题

| | 列标题1 | 列标题2 | ... | 列标题 n |
|---------|----------|----------|-----|----------|
| 第1行 | 单元格11 | 单元格12 | ... | 单元格1 n |
| ... | ... | ... | ... | ... |
| 第 m 行 | 单元格 $m1$ | 单元格 $m2$ | ... | 单元格 mn |

2.3.1 建立表格

用法:

`<table 属性="值">...</table>`

其功能是创建一个空的表格，并指定表格的整体外观，此时在浏览器中看不到该表格，只有添加行后才能看到。表格标记常用的属性如下：

- `border="size"`: 设置表格边框大小。
- `width="size"`: 设置表格的宽度（像素或百分比）。
- `height="size"`: 设置表格的高度（像素或百分比）。
- `cellspacing="size"`: 设置单元格间距。
- `cellpadding="size"`: 设置单元格的填充距。
- `background="URL"`: 设置表格背景图像。
- `bgcolor="colorvalue"`: 设置表格背景色。
- `align="alignstyle"`: 设置对齐方式，`alignstyle`可取值`left`（左对齐）、`center`（居中）和`right`（右对齐）值之一。
- `cols="size"`: 设置表格的列数。

2.3.2 定制表格

使用<table>只是定义空表格，还需要定义表标题、行、列标题和单元格。

1. 定义表标题

用法：

`<caption 属性="值">...</caption>`

其功能是定义表格的表标题，其属性较少使用。

2. 定义行

用法：

`<tr 属性="值">...</tr>`

其功能是定义表格的一行。对于每一行，可以定义行属性，常用的行属性如下：

■`bgcolor="colorvalue"`：设置行背景颜色。

■`background="URL"`：设置表格行的背景图像。

■`align="alignstyle"`：设置行对齐方式，`alignstyle`可取值`left`（左对齐）、`center`（居中）和`right`（右对齐）值之一。

■`valign="valignstyle"`：设置行中单元格垂直对齐方式，`valignstyle`可取值`top`（顶端对齐）、`middle`（居中）、`bottom`（底端对齐）和`baseline`（基线对齐）值之一。

3. 定义列标题

用法:

`<th 属性="值">...</th>`

其功能是定义一个列标题。对于每一行，可以定义行属性，常用的行属性如下：

■ `bgcolor="colorvalue"`: 设置列标题背景颜色。

■ `background="URL"`: 设置列标题背景图像。

■ `align="alignstyle"`: 设置列标题对齐方式，`alignstyle`可取值`left`（左对齐）、`center`（居中）和`right`（右对齐）值之一。

■ `valign="valignstyle"`: 设置列标题垂直对齐方式，`valignstyle`可取值`top`（顶端对齐）、`middle`（居中）、`bottom`（底端对齐）和`baseline`（基线对齐）值之一。

4. 定义单元格

用法:

```
<td 属性="值">...</td>
```

在每一行中可以定义若干单元格。单元格的常用属性如下:

- **bgcolor="colorvalue"**: 设置单元格背景颜色。
- **background="URL"**: 设置表格单元格的背景图像。
- **rowspan="num"**: 设置单元格所占的行数。
- **colspan="num"**: 设置单元格所占的列数。
- **align="alignstyle"**: 设置对齐方式。
- **valign="valignstyle"**: 设置单元格垂直对齐方式。
- **width="size"**: 设置单元格宽度。
- **height="size"**: 设置单元格高度。

【例2.1】 设计一个含有表格的网页窗体WForm2-1。



设计界面

WForm2-1网页源视图代码：

```
1 <%@ Page Language="VB" AutoEventWireup="false" CodeFile="WForm2-1.aspx.vb" Inherits="ch2_WForm2_1" %>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" >
4 <head runat="server">
5 <title>无标题页</title>
6 </head>
7 <body>
8 <form id="form1" runat="server">
9 <div>
10 <span style="font-size: 16pt; font-family: 隶书">诗人表<br />
11 </span>
12 <table border="1" style="font-size: 16pt; font-family: 隶书">
13 <tr>
14 <td style="width: 71px; text-align: center">
15 <span style="font-family: 黑体">编号</span></td>
16 <td style="width: 59px; text-align: center">
17 <span style="font-family: 楷体_GB2312">101</span></td>
18 <td style="width: 61px; text-align: center">
19 <span style="font-family: 黑体">性别</span></td>
20 <td style="width: 50px; text-align: center">
21 <span style="font-family: 楷体_GB2312">男</span></td>
22 <td rowspan="3" style="width: 100px">
23 </td>
24 </tr>
25 <tr>
26 <td style="width: 71px; text-align: center">
27 <span style="font-family: 黑体">姓名</span></td>
28 <td colspan="3">
29 &nbsp; &nbsp; <span style="font-family: 楷体_GB2312">李白</span></td>
30 </tr>
31 <tr>
32 <td style="width: 71px; height: 13px; text-align: center">
33 <span style="font-family: 黑体">特长</span></td>
34 <td colspan="3" style="height: 13px">
35 <span style="font-family: 楷体_GB2312">浪漫主义诗人</span></td>
36 </tr>
37 </table>
38 </div>
39 </form>
40 </body>
41 </html>
```

滚动字幕 “插入” -“标签”

- **<marquee>**标记常用属性
 - align属性，设定活动字幕的位置，取值可以是left、center、right、top或bottom。
 - bgcolor属性，设定活动字幕的背景颜色，一般是十六进制数。
 - direction属性，设定活动字幕的滚动方向，取值可以是left、right、up或down。
 - behavior属性，设定滚动的方式，主要有三种方式：behavior="scroll"表示由一端滚动到另一端；behavior="slide"表示由一端快速滑动到另一端，且不再重复；behavior="alternate"表示在两端之间来回滚动。
 - height和width属性，设定滚动字幕的高度和宽度。
 - hspace和vspace属性，设定滚动字幕的左右边框和上下边框的宽度。
 - scrollamount属性，设定活动字幕的滚动距离。
 - scrolldelay属性，用于设定滚动两次之间的延迟时间。
 - loop属性，用于设定滚动的次数，当loop=-1表示一直滚动下去，直到页面更新。
- 例：<marquee onmouseover=this.stop() onmouseout=this.start() behavior="alternate" direction="left" bgcolor="#3399FF" width="100%" height="15"> 超链接和多媒体示例</marquee>

超链接标记

- `链接文本或图像`
- **href**指出转向的URL。**target**属性指出该超链接指向的HTML文档在指定目标窗口中打开，**target**属性取值及其说明如下表所示。

| 名称 | 说明 |
|----------------|--------------------|
| _blank | 将链接的内容显示在新的浏览器窗口中。 |
| _self | 将链接的内容显示在当前窗口中。 |
| _parent | 将链接的内容显示在父窗口中。 |
| _top | 将链接的内容显示在浏览器主窗口中。 |
| 自定义URL | 在自定义的窗口中显示 |

超链接标记的使用

- `上一页`
- ` `
- `电子信箱地址`
- `播放音乐文件`
- `播放影像文件`

页面多媒体技术

- 图像标记
 - width、 height

```

```
- 背景音乐标记（放在头部head）
 - < bgsound src ="音乐文件的URL" loop=播放次数>
 - Loop=-1表示循环播放

页面多媒体技术-续

- 音乐和影像文件
 - `< embed src ="音乐或影像文件的URL"`
 `width=宽度` `height=高度` `hidden= 是否隐藏面板`
 `autostart=是否自动播放` `loop=是否重复`>
 - `<embed src="bg.mid" width="307"`
 `height="198" autostart="true" loop="true">`

2.4 使用框架

框架是网局的重要工具，它与表格的不同之处在于表格是把网页分割成小的单元格，而框架是把浏览器的窗口分割成一个个小的子窗口，这些子窗口称为框架，每一个框架都相当于一个浏览器窗口，这样就使一个浏览器窗口可以显示多个网页。

通常使用<iframe>标记建立框架，它与<frame>不同的是，<iframe>可以嵌在网页中的任意部分。使用<iframe>标记建立的框架称为**iframe**框架。

2.4.1 建立iframe框架

用法:

```
<iframe 属性="值">...</iframe>
```

其功能是建立一个**iframe**框架，通常用**id**属性指定框架名称。

2.4.2 iframe框架的属性

iframe框架的常用属性如下：

- `src="URL"`：设置要链接到该子窗口的URL。
- `width="size"`：设置iframe框架的宽度。
- `height="size"`：设置iframe框架的高度。
- `frameborder="size"`：指定iframe框架是否有边框，size可取yes、no、1和0值之一。
- `marginwidth="size"`：用来控制显示内容和窗口左右边界的距离，默认为1。
- `marginheight="size"`：用来控制显示内容和窗口上下边界的距离，默认为1。
- `scrolling="scrollingstyle"`：指定子窗口是否使用滚动条，scrollingstyle可取yes/no/auto这3个值之一，默认为auto，即根据窗口内容决定是否有滚动条。

注意：

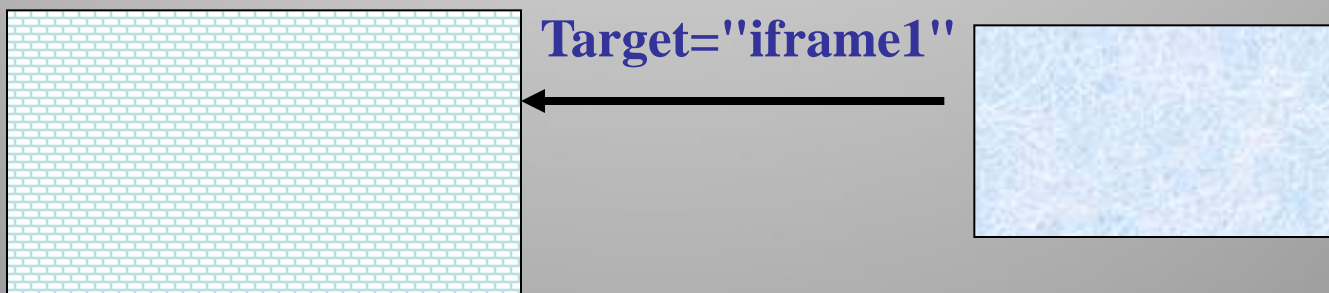
在Visual Studio.NET 2005中，**iframe**框架只能在视图中用直接输入代码的方式创建。

iframe框架中显示的内容用src属性指定，而<iframe>和</iframe>标记之间的内容只有在不支持iframe框架的浏览器中才显示。

另外，当一个**iframe**框架**iframe1**已建立后，可以暂时不指定src属性，以后将一个网页（aaa.aspx）在其中显示时，需指定该框架的name属性为"iframe1"，并将网页aaa.aspx的Target属性为"iframe1"，这样便可以在iframe1框架中显示aaa.aspx网页的内容。

iframe框架， name="iframe1"

网页aaa.aspx

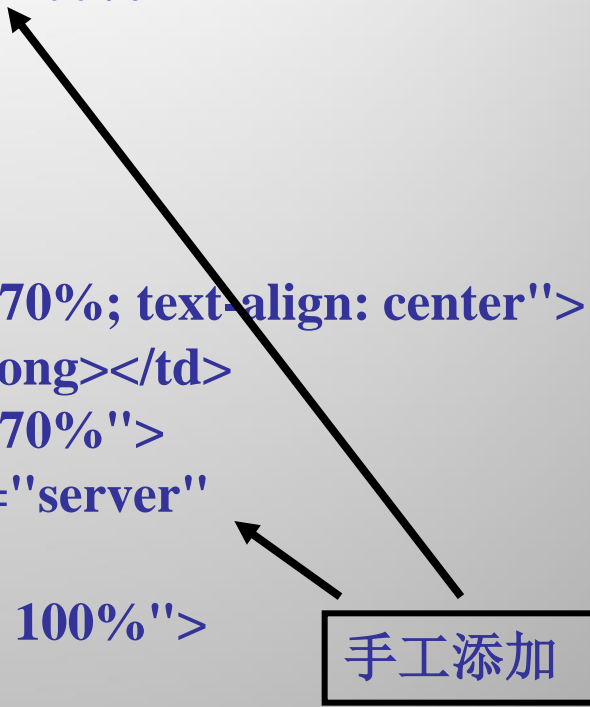


【例2.2】 设计一个含有iframe框架的网页窗体WForm2-2。

源视图代码如下：

```
<%@ Page Language="VB" AutoEventWireup="false"
    CodeFile="WForm2-2.aspx.vb" Inherits="ch2_WForm2_2" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
    <head runat="server">
        <title>无标题页</title>
    </head>
    <body>
        <form id="form1" runat="server">
            <div>
                <table style="width: 700px; height: 350px">
                    <tr>
                        <td style="width: 30%; height: 40%; text-align: center">
                            <strong>WForm1-1网页</strong></td>
```

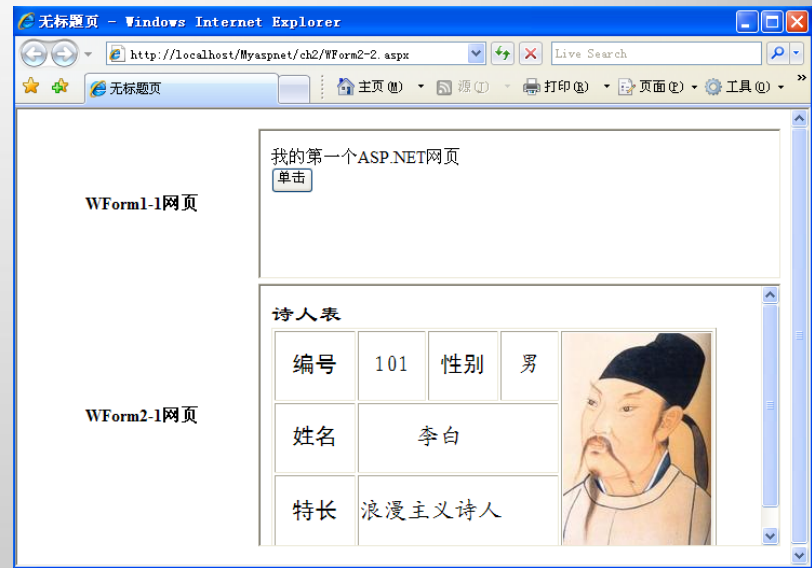
```
<td style="width: 70%; height: 40%">
  <iframe id="Iframe1" src="../ch1/WForm1-1.aspx"
    style="width: 100%; height: 100%">
  </iframe>
</td>
</tr>
<tr>
  <td style="width: 30%; height: 70%; text-align: center">
    <strong>WForm2-1网页</strong></td>
  <td style="width: 70%; height: 70%">
    <iframe id="Iframe2" runat="server"
      src="WForm2-1.aspx"
      style="width: 100%; height: 100%">
    </iframe>
  </td>
</tr>
</table>
</div>
</form>
</body>
</html>
```



A diagram consisting of a rectangular box with a black border containing the text "手工添加" (Manually Added) in blue. Two black arrows originate from the top corners of this box. One arrow points diagonally upwards and to the left, terminating at the opening tag of the first <iframe> element in the code block. The second arrow points diagonally upwards and to the right, terminating at the opening tag of the second <iframe> element in the code block.

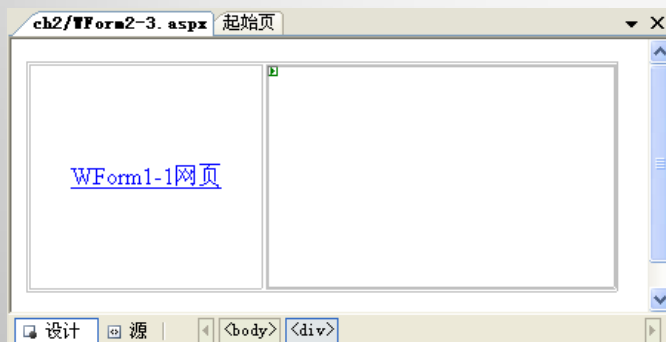


设计界面



运行界面

【例2.3】 设计一个含有超链接的网页窗体WForm2-3。



源视图代码如下：

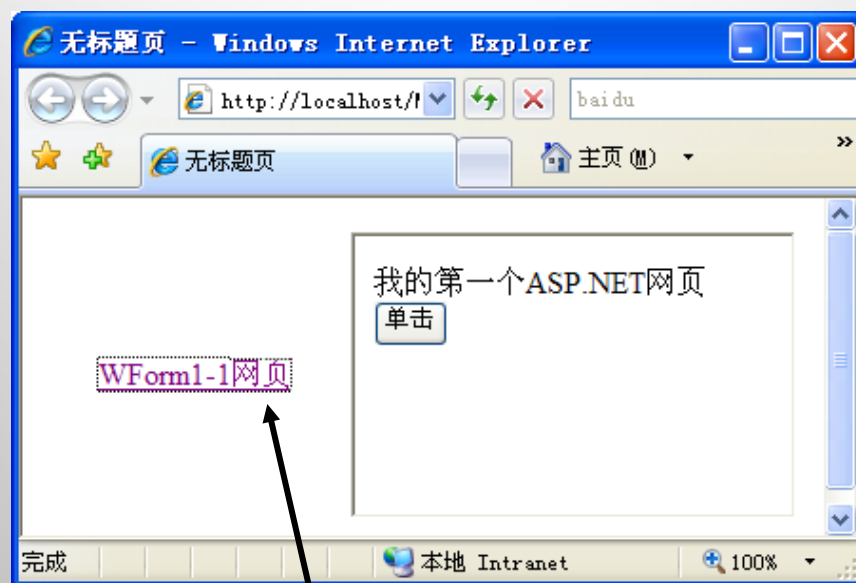
```
<%@ Page Language="VB" AutoEventWireup="false"
    CodeFile="WForm2-3.aspx.vb" Inherits="ch2_WForm2_3" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
    <head runat="server">
        <title>无标题页</title>
    </head>
```



```
<body>
  <form id="form1" runat="server">
    <div>
      <table id="TABLE1" style="width: 400px; height: 150px">
        <tr>
          <td style="width: 40%; height: 100%; text-align: center">
            <a href=" ../ch1/WForm1-1.aspx"
              target="Iframe1">WForm1-1网页</a>
          </td>
          <td style="width: 60%; height: 100%">
            <iframe id="Iframe1" runat="server" frameborder="1"
              name="Iframe1" style="width: 100%;
              height: 100%"></iframe>
          </td>
        </tr>
      </table>
    </div>
  </form>
</body>
</html>
```

手工输入





单击

2.6 制作表单

HTML提供的表单（form）是用来将用户输入的数据从浏览器传递给Web服务器的。

2.6.1 创建表单

用法：

`<form 属性=值>.....</form>`

`<form>`标记的主要属性如下：

■**action**：用来指出当这个表单提交后需要执行的驻留在Web服务器上的程序名（包括路径）是什么。一旦Internet网络用户提交输入信息后服务器便激活这个程序，完成某种任务。

例如：`<form action = "login.aspx" method = "POST" >...</form>`，当用户点击本表单的提交按钮以后，执行Web服务器上的login.asp文件将接收用户输入的信息，以登记用户信息。

■**method**: 用来说明从客户端浏览器将Internet网络用户输入的信息传送给Web服务器时所使用的方**式**。

有两种方式: **POST**和**GET** (默认方式), 区别是: 在使用**POST**时, 表单中所有的变量及其值都按一定的规律放入报文中, 而不是附加在**action**所设定的URL之后。在使用**GET**时将**form**的输入信息作为字符串附加在**action**所设定的URL的后面, 中间用“?” 隔开, 即在客户端浏览器的地址栏中可以直接看见这些内容。

■**enctype**: 用于确定提交的信息的类型。

■**target**: 用于确定服务器程序将结果返回时在哪一个框架窗口中显示, 默认值为“_self”。

2.6.2 表单控件

在<form>与</form>之间可以嵌入各种控件，也称为表单域标记。它们的通用格式为：

<input type="输入控件类型" id="id值">

其中，type属性设置该控件的类型，id确定该控件在整个文档中id值。在Visual Studio.NET 2005中，工具箱的“HTML”选项卡上提供下列HTML控件：

- ✓Input（Button）控件：INPUT type="button"元素。
- ✓Input（Submit）控件：INPUT type="submit"元素。
- ✓Input（Reset）控件：INPUT type="reset"元素。
- ✓Input（Checkbox）控件：INPUT type="checkbox"元素。
- ✓Input（Radio）控件：INPUT type="radio"元素。
- ✓Input（Hidden）控件：INPUT type="hidden"元素。
- ✓Input（Text）控件：INPUT type="text"元素。
- ✓Input（Password）控件：INPUT type="password"元素。
- ✓Select控件：多重选择框元素。

1. Input（Button）控件

用于建立按钮，其用法如下：

```
<input id="id值" type="button" name="域名称" value="值">
```

其中，type设置为button表示这个控件为按钮。value值既是它被按下时传给服务程序的值，又是它的标题，即写在按钮上的文字。

有两个特殊的控件，它们实质上也是按钮，但其type不是“button”，而分别是“submit”（提交按钮）和“reset”（重置按钮）。其用法分别为：

```
<input id="id值" type="submit" name="域名称" value="值">
```

```
<input id="id值" type="reset" name="域名称" value="值">
```

按下提交按钮后，表单就把当前所获得的信息以method指定的方式全部传给action指定的程序。按下重置按钮后，则表单中的所有控件都被重置，恢复初始状态。

按钮的常用事件为OnClick。

2. Input (Checkbox) 控件

用于建立复选框，其用法如下：

```
<input id="id值" type="checkbox" name="域名称" value="值" checked>
```

其中，type设置为checkbox表示这个控件为复选框。value设置当这个复选框被选中后，发送给action指定处理程序的值。checked表示预置该复选框被选中，如果有这一项，该复选框初始值为被选中。

复选框的常用事件有OnClick和OnFocus。

3. Input (Radio) 控件

用于建立单选框（选项按钮），其用法如下：

```
<input type="radio" name= "域名称" value="值" checked>
```

各属性意义与复选框的基本相同。

注意：在设置一组单选框时，各选项必须同名（具有相同的name），取值不同（value不相同），并且几个选项中必须有且只能有一个预置为选中。如果没有预置选中项，缺省预置第一个选择项被选中。

单选框的常用事件有OnClick。

4. Input（Hidden）控件

用于建立隐藏项，其用法如下：

```
<input id="id值" type="hidden" name="域名称" value="值">
```

该控件的内容在表单中被隐藏起来，并不在网页中显示。通常用来以隐藏方式向服务器传送有关信息。

5. Input (Text) 控件

用于建立单行文本输入框，其用法如下：

```
<input id="id值" type="text" name="域名称" value="默认值"  
maxlength=值 size="值">
```

其中，**value**属性确定该文本框预置的文字。**maxlength**属性确定在这个文本框中所能容纳的字符串最大长度，该项可以不设。**size**属性确定这个文本框的显示宽度，以能显示多少个字符来衡量。

单行文本框的常用事件有**OnClick**、**OnFocus**和**OnChange**。

有一种特殊的单行文本输入框专门用于输入密码（password），不同之处在于它对键盘输入的回显字符为“*”，即它把用户的输入隐藏了。其用法为：

```
<input type="password" name="域名称" value="默认值"  
    maxlength=值 size=值>
```

6. Select控件

用于建立多重选择框，其用法如下：

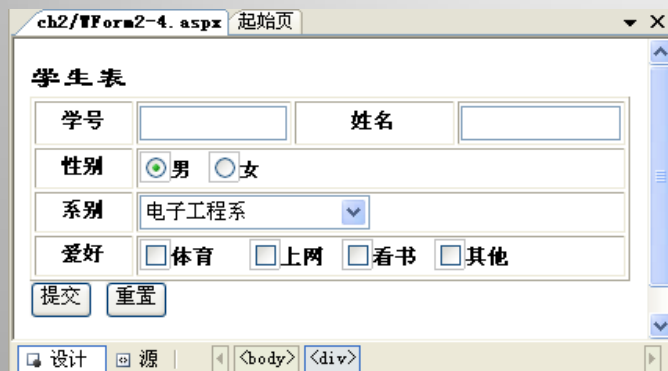
```
<select id="id值" name="域名称" size="值" multiple>.....</select>
```

其中，size指定选择框可以显示的行数（默认值为1）；若指定multiple则该选择框为多重选择框。在<select>与</select>之间加入选择项，选择项形式为：

```
<option selected value="域值">显示文本</option>
```

若为一个选择项指定selected，则该选择项预置为选中。

【例2.4】 设计一个含有超链接的网页窗体WForm2-4。

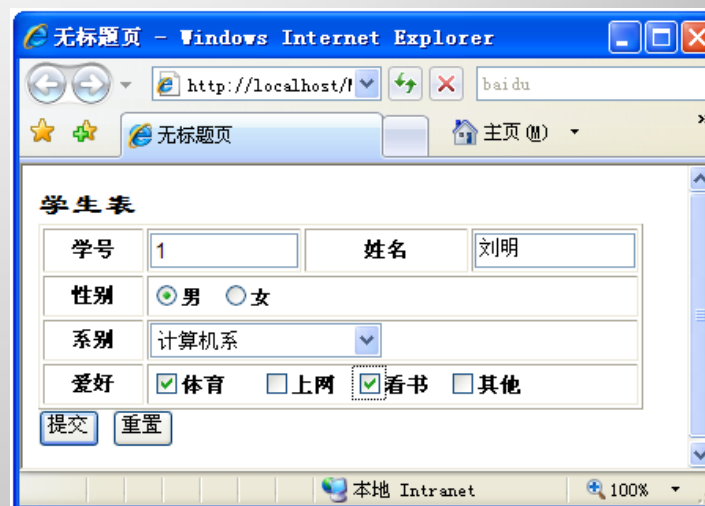
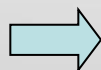


The image shows the design view of a web form titled "ch2/WForm2-4.aspx" with a "起始页" (Start Page) tab. The form is titled "学生表" (Student Table). It contains the following fields and controls:

- 学号** (Student ID): A text input field.
- 姓名** (Name): A text input field.
- 性别** (Gender): Radio buttons for ☒ 男 (Male) and ☐ 女 (Female).
- 系别** (Department): A dropdown menu with "电子工程系" (Department of Electronic Engineering) selected.
- 爱好** (Hobbies): Checkboxes for ☐ 体育 (Sports), ☐ 上网 (Internet), ☐ 看书 (Reading), and ☐ 其他 (Other).
- 提交** (Submit) and **重置** (Reset) buttons.

The bottom status bar shows "设计" (Design) and "源" (Source) tabs, with the current view being "设计".

设计界面



The image shows the run view of the web form in a "Windows Internet Explorer" window titled "无标题页 - Windows Internet Explorer". The address bar shows "http://localhost/" and the search bar contains "baidu". The form is titled "学生表" (Student Table) and contains the following fields and controls:

- 学号** (Student ID): A text input field with the value "1".
- 姓名** (Name): A text input field with the value "刘明" (Liu Ming).
- 性别** (Gender): Radio buttons for ☒ 男 (Male) and ☐ 女 (Female).
- 系别** (Department): A dropdown menu with "计算机系" (Department of Computer Science) selected.
- 爱好** (Hobbies): Checkboxes for ☒ 体育 (Sports), ☐ 上网 (Internet), ☒ 看书 (Reading), and ☐ 其他 (Other).
- 提交** (Submit) and **重置** (Reset) buttons.

The bottom status bar shows "本地 Intranet" and a zoom level of "100%".

运行界面

例： e-04.html和e-04.aspx的运行效果

- 单击“e-04.html”的“提交”按钮后，将该网页提交给“e-04.aspx”处理。
- `<form id="form1" name="form1" method="post" action="e-04.aspx">`

| | |
|--|---|
|  | <p>用户名 <input type="text" value="小王"/></p> <p>密码: <input type="password" value="..."/></p> <p><input type="button" value="提交"/> <input type="button" value="重置"/></p> |
| <p>新浪网</p> | <p>22:20:41</p> |

您输入的姓名是：小王，密码是：123

2.7 CCS样式设计

2.7.1样式和样式表

CSS（Cascading Style Sheets）是W3C定义和维护的**标准**，是一种用来为结构化文档（如HTML文档或XML文档）添加**样式**（字体、间距和颜色等）的**计算机语言**。

1. 样式

样式是指每一个网页元素呈现在浏览器中的风格，比如字体的大小、颜色、页面的背景色、背景图等。定义样式的基本格式如下：

样式属性1:值1; 样式属性2:值2; ...

样式属性与值之间用冒号“:”分隔，如果一个样式中有多个样式属性，各样式属性之间要用分号“;”隔开。

(1) 内联式样式设置

通过style属性指定所修饰的元素使用的“私有样式”，用法如下：

style="属性1:值1;属性2:值2;...属性 n :值 n "

例如：

<h1 style="font-size:40px;color:Red;">中华人民共和国</h1>

<h1 style="font-size:30px;color:Blue;">教育部</h1>

(2) 直接嵌入式样式设置

使用<style>和</style>标记建立一个或多个样式，置于<head></head>之间，在<body></body>内直接使用这些样式。<style>标记有一个属性type，它指出样式的类别，如果使用CSS样式，则type取值为“text/css”。

例如：

```
<html>
  <head>
    <title>样式引用示例</title>
    <style type="text/css">
      h1{font-size:40px; color:Red;}
      h2{font-size:30px; color:Blue;}
    </style>
  </head>
  <body>
    <h1>中华人民共和国</h1>
    <h2>教育部</h2>
  </body>
</html>
```

2. 样式表文件

所谓样式表文件就是将网页元素的样式定义设计为一个独立的文件。凡是在网页的<head></head>部分与该样式表文件建立链接的HTML文件，其页面元素的样式就会按照样式表文件中的定义显示。样式表文件的扩展名为“**.css**”

样式表文件中样式定义的一般格式如下：

样式定义选择符 { 样式属性1:值1; 样式属性2:值2;}

例如，样式表文件StyleSheet1.css的内容为：

```
body { background-color:#33bb66;}
```

```
h1 { font-size:40pt; color:Blue;}
```

```
h2 { font-size:30pt; color:White;}
```

在网页文件中引用该样式表文件只需要在网页的<head>与</head>之间添加如下代码：

```
<link href="StyleSheet1.css " type="text/css" rel="Stylesheet" />
```

其中，**rel**规定了被链接文件的关系，取值永远是“Stylesheet”，**type**属性规定了链接文件的类型；**href**属性则指定了要链接的样式表文件的URL。

CSS优先级

- ID的样式优先级高于class的样式
- 后面的样式覆盖前面的样式
- 指定的高于继承的样式
- 行内样式高于内部或外部样式

总结：单一的(id)高于共用的(class)，有指定的用指定的，无指定则继承离它最近的

注意：不同位置CSS样式规则的优先级是内联样式最高，其次是页面中的CSS样式，最后是外部样式表。

样式表文件中设计每个样式的“样式定义选择符”指样式定义的对象，可选项有：HTML标记、用户自定义的类、用户自定义的ID和虚类等。

(1) HTML标记

HTML标记是最典型的选择符类型。如果有多个不同的标记要使用相同的样式，则可以采用编组的方法简化定义。例如：

```
H1,H2,H3 {color:red}
```

(2) 类 (class)

用户自定义的类是用来为某一个HTML标记创建多个样式，或者为多个标记创建同一种样式。类的定义格式如下：

样式定义选择符.类名{样式属性1:值1; 样式属性2:值2;.....}

例如，H1.first和H1.second的样式代码分别如下：

H1.first {color:Red;font-size:40px}

H1.second{color:Blue;font-size:30px}

这样，可以通过以下方式使用它们：

<h1 class="first">中华人民共和国</h1>

<h1 class="second">教育部</h1>

(3) 自定义ID

自定义ID以“#”为标志，依靠这个唯一的标志可以在定义一套样式，称为“私有命名样式”。

私有命名样式的定义方法为：

#idname {属性1:值1;属性2:值2;...属性n:值n}

其中，id标志前的“#”符号一定不能省略。例如：

#customId1 {color:Red}

在网页中引用该样式的标记符内使用id属性即可，例如：

<p id="customId1">本段落文字为红色</p>

ID与类主要区别是，类可以在同一个网页的多个标记中重复使用；而ID则在同一网页中只能使用**一次**。

(4) 虚类

虚类是专用于<a>标记的选择符，使用虚类可以设置不同类型超链接的显示方式：

A:link: 未被访问过的超链接。

A:visited: 已被访问过的超链接。

A:active: 当超链接处于选中状态。

A:hover: 当鼠标指针移动到超链接上。

定义虚类的语法格式如下：

a:状态 {属性:值;.....}

例如：

a:visited,a:link { color:blue }

a:hover { color:yellow; text-decoration:none}

2.7.2 样式生成器

直接使用代码设计样式是十分麻烦的，为此Visual Studio.NET 2005提供了专门的样式生成器可以可视化地设计样式。

【例2.5】 设计一个显示若干文字的网页窗体WForm2-5，其中这些文字引用相应的样式进行输出。

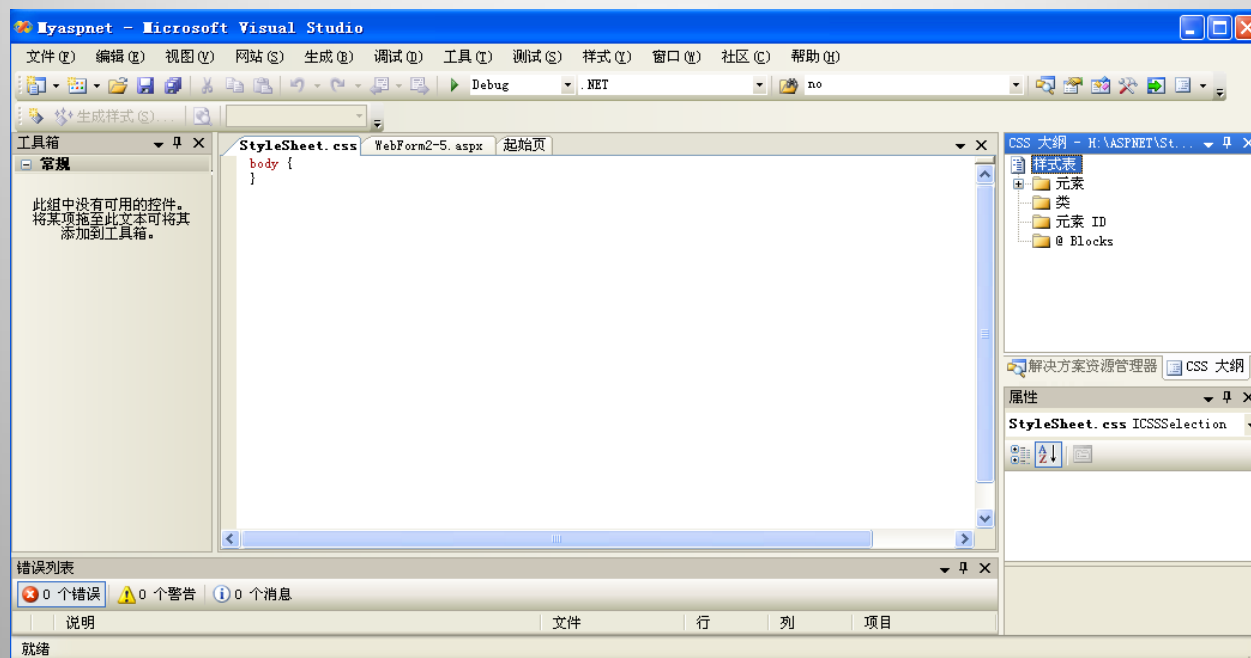
其设计步骤如下：

(1) 在Myaspnet网站的ch2文件夹中添加一个名称为WForm2-5的空网页。

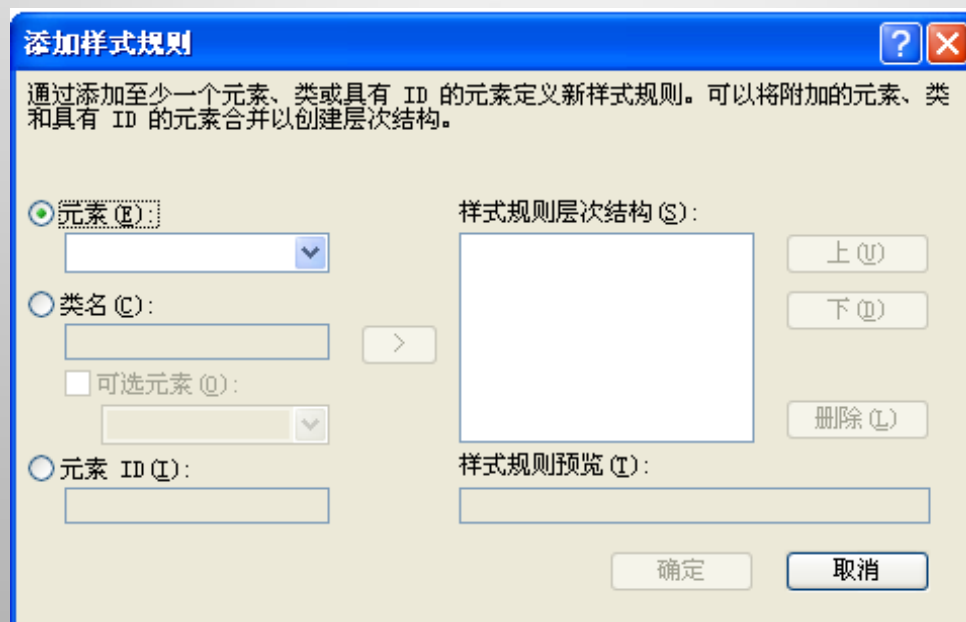
(2) 选择“网站”|“添加新项”菜单命令，出现“添加新项”对话框，在“模板”列表中选择“样式表”，保持默认文件名StyleSheet.css不变，如下图所示，单击“添加”按钮。



(3) 出现如下图所示的样式设计对话框，如果右边不是“CSS大纲”窗口，单击工具栏中的下三角箭头，在出现的菜单中选择“文档大纲”命令。



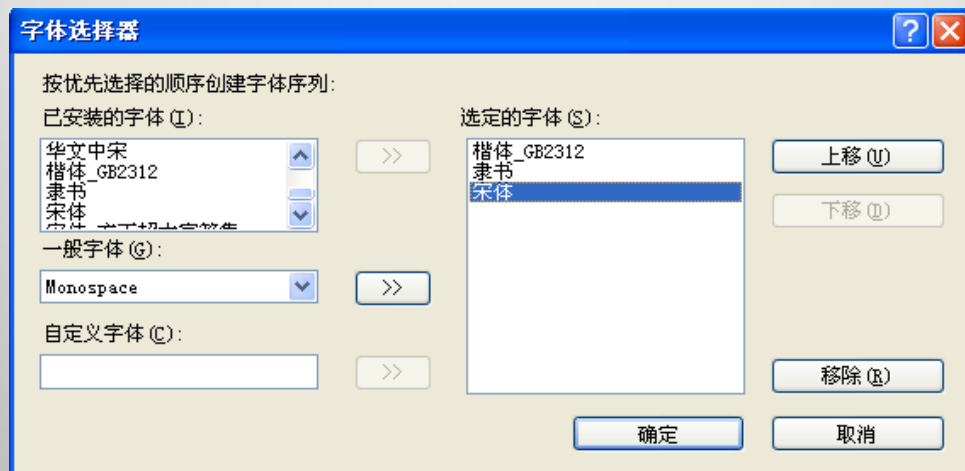
(4) 选择“样式|添加样式规则”菜单命令，出现如下图所示的“添加样式规则”对话框。



(5) 从“元素”组合框中选中H1标记或直接输入H1，在“样式规则预览”文本框中出现“H1”，该项是不可修改的，单击“确定”按钮。此时样式表中自动添加H1{}的样式定义代码框架，并在“CSS大纲”窗口中的“元素”节点下自动添加一个H1标记。

(6) 在H1的定义体内，或者在“CSS大纲”窗口中的“元素”项的H1标记上，单击鼠标右键，从出现的快捷菜单中选择“生成样式”命令，出现“样式生成器-H1”对话框，从颜色组合框中选择“Red”，选中大小下“绝对”单选框，从后面的组合框中选中“大”，选中粗体下“绝对”单选框，从后面的组合框中选中“粗体”。

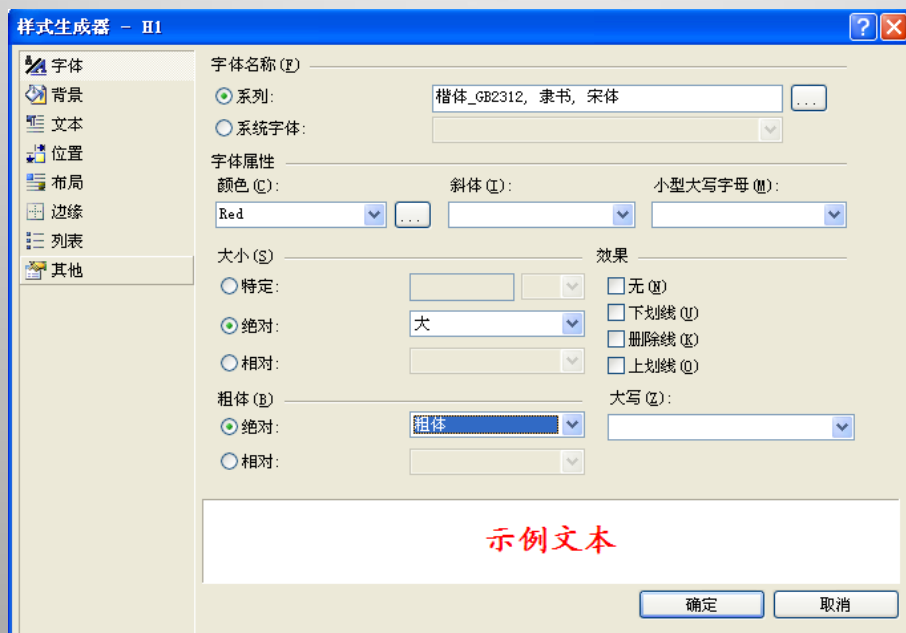
(7) 单击“系列”后面的“...”按钮，出现“字体选择器”对话框，从“已安装的字体”列表中选择3种字体到“选定的字体”列表中，如下图所示：



单击“确定”按钮。最后设置好后的“样式生成器-H1”对话框如下图所示。单击“确定”按钮，返回时看到H1样式的代码变为：

H1

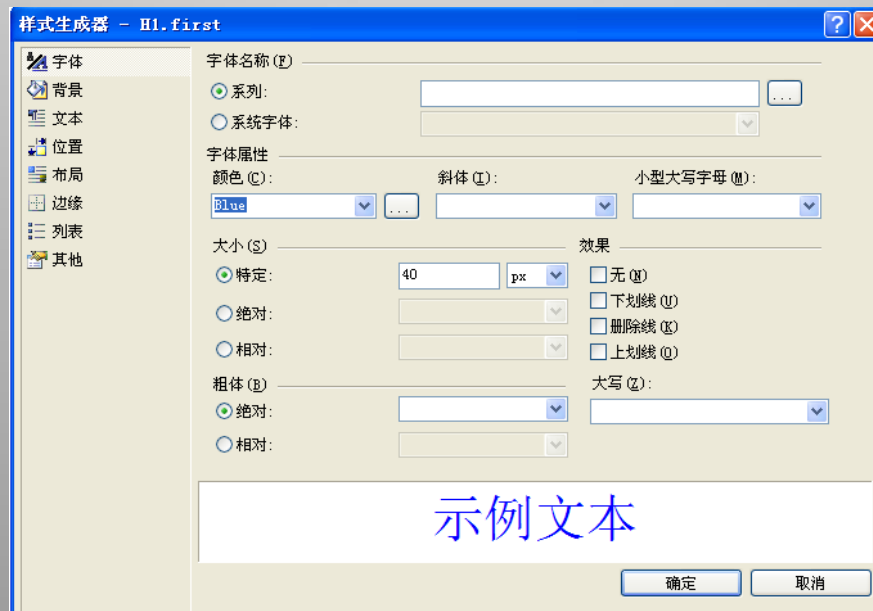
```
{ font-weight: bold; font-size: large; color: red;  
font-family: 楷体_GB2312, 隶书, 宋体; }
```



(8) 再次选择“样式|添加样式规则”菜单命令，出现“添加样式规则”对话框。选中“类名”单选框，输入自定义类名first，对话框下方的“样式规则预览”列表框中出现“.first”（点号是类名的起始标识）。

选中“可选元素”复选框，使其下方的组合框变为活动的，从中选出H1标记，这时“样式规则预览”列表框中的内容变为“H1.first”，单击“确定”按钮，此时样式表中自动添加H1.first{}的样式定义代码框架，并在“CSS大纲”窗口中的“类”节点下自动添加一个H1标记。通过“样式生成器-H1.first”对话框设置其属性如下图所示。对应的代码为：

```
H1.first { font-size: 40px; color: blue; }
```



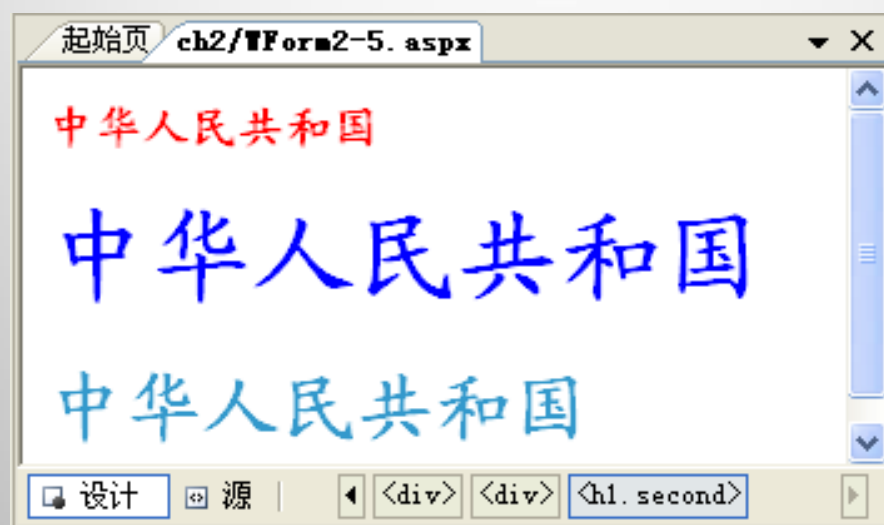
(9) 采用同样的方法设计H1.second类的代码为:

```
H1.second { font-size: 30px; color: #3399cc; }
```

(10) 保存样式表文件, 返回到WForm2-5网页, 单击“设计”选项卡出现空的设计界面。输入3行“中华人民共和国”, 切换到视图, 修改代码如下(只增加黑体部分, 其余不变):

```
<%@ Page Language="VB" AutoEventWireup="false"  
    CodeFile="WForm2-5.aspx.vb" Inherits="ch2_WForm2_5" %>  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml" >
```

```
<title>无标题页</title>
  <link href="../StyleSheet.css" rel="stylesheet" type="text/css" />
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <div><h1>中华人民共和国</h1>
      <h1 class="first"> 中华人民共和国</h1>
      <h1 class="second">中华人民共和国</h1>
    </div>
  </form>
</body>
</html>
```



运行界面