# 2.HTML

## 2.1.什么是html

是超文本标记语言，它规定了自己的语法规则，用来表示比“文本”更丰富的意义，比如图片，表格，链接等。浏览器软件知道HTML语言的语法，目前互联网上的绝大部分网页都是使用HTML编写的。

## 2.2.html的结构

文件名：xxx.html或者xxx.htm，但是通常用xxx.html。

<!DOCTYPE HTML>

<html>

<head>

</head>

<body>

</body>

</html>

<!DOCTYPE HTML>：html5的文档声明

<html>：所有的html都要有这个标签，表示文档的开始和结束

<head>：用来存放html中的基本属性信息，比如标题，编码等

<body>：用来存放html中的页面数据，在页面中是可以显示的

<title>：指定网页的标题

<meta charset=UTF-8/>：指定浏览器的解码格式

## 2.3.HTML语法

HTML是一门标记语言, 标记也叫做元素/标签, 标签分为开始标签和结束标签, 如果标签内没有内容要修饰, 可以合并一个自闭标签.

如: <meta/> <br/> <hr/> <input/> <img/>等等

注释: <!--注释内容 -->

如何在网页中做一个空格或者做一个换行?

由于在网页中多个连续的空白字符会被当成一个空格来显示, 所以

如果要做一个空格, 可以用转义字符 &nbsp;来代替;

如果要做一个换行, 可以用<br/>标签来代替;

HTML 和 XML的区别:

xml对语法的要求非常严格, 如区分大小写, 标签不能交叉嵌套,标签一定要关闭等

HTML对语法的要求非常不严格. 如: 标签名可以大小写混用, 标签即使没有结束或者交叉嵌套, 可能也会被浏览器正常解析(最好按照规范来书写!!)

xml中允许用户自己定义标签

HTML中的标签都是预定义好的, 不允许用户自己定义

…

## 2.4.HTML标签

通过不同的标签，HTML文档可以包含不同的内容。比如文本，链接，图片，列表，表格，表单等。

可以通过W3Cschool来学习标签，W3Cschool是一个专业的编程入门学习及技术文档查询应用

### 2.4.1.文本标签<font></font>

文本标签font，有三个属性：

size：大小，范围为1-7

face：字体

color：颜色，有三种形式，英文、16色、RGB（RGB颜色部分浏览器不支持）



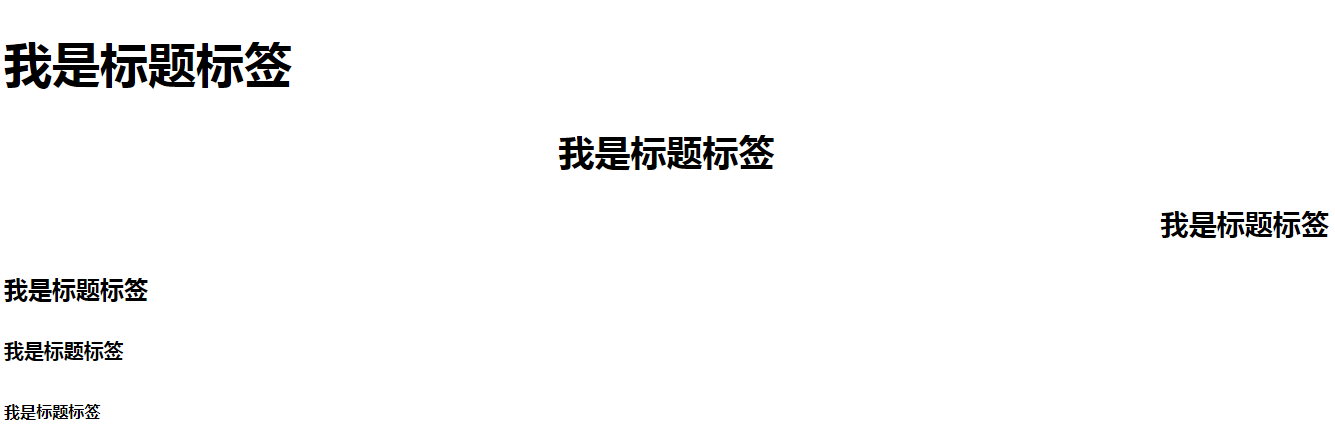
HTML文档内容如下

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <head>  <meta charset=UTF-8/>  <title>我是第一个网页</title>  </head>  <body>  <font size="1" face="楷体" color="red">我是font标签</font><br/>  <font size="5" face="楷体" color="#ff0000">我是font标签</font><br/>  <font size="7" face="楷体" color="rgb(255,0,0)">我是font标签</font><br/>  </body>  </html> |

### 2.4.2.标题标签<h1>--<h6>

标题标签，从h1-h6依次变小，只有一个属性

align：文本的位置,left居左（默认），center居中，right居右



HTML文档内容如下

|  |
| --- |
| <h1 align="left">我是标题标签</h1>  <h2 align="center">我是标题标签</h2>  <h3 align="right">我是标题标签</h3>  <h4>我是标题标签</h4>  <h5>我是标题标签</h5>  <h6>我是标题标签</h6> |

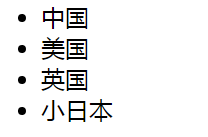
### 2.4.3.列表标签<ul>/<li>

列表标签:ul：定义一个无序标签，无序指列表前无编号，属性有：

type：表示列表前面的形状，取值有disc（默认）：实心圆，circle：空心圆，square：方块

li：定义列表中的项





HTML文档内容如下

|  |
| --- |
| <ul type="disc">  <!-- -->  <li>中国</li>  <li>美国</li>  <li>英国</li>  <li>小日本</li>  </ul> |

### 2.4.4.图片标签<img …/>

图像标签img：属性

src:图片网址或者路径

width:宽度，可以是像素，也可以是百分比

hight：高度

alt：图像的代替文本

border：边框的宽度 px



HTML文档内容如下

|  |
| --- |
| <img src="C:\Users\Public\Pictures\Sample Pictures\Chrysanthemum.jpg" width="500px" height="40%" alt="菊花"/> |

### 2.4.5.超链接标签<a …></a>

超链接标签a,属性

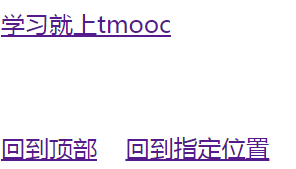
href:用来指定跳转网址

title：鼠标移到改标签时所显示内容

target：在何处打开目标url，常用有\_self:当前页面打开，\_blank:新页面打开，\_top:回到顶部

如果需要回到指定位置，需要在指定位置加入<a name="place"/>





HTML文档内容如下

|  |
| --- |
| <a href="http://www.tmooc.cn" title="一分钟帮你搞定java" target="\_blank">学习就上tmooc</a>  <br/><br/>  <a href="" target="\_top">回到顶部</a>&nbsp;&nbsp;&nbsp;  <a href="#place">回到指定位置</a> |

### 2.4.6.表格标签<table>/<tr>/<td>

table – 用来定义一个表格

tr – 用来定义表格中的行

td – 用来定义表格中的单元格

th – 用来定义表头

table属性:

border：边框

cellspacing：单元格之间的距离

cellpadding：边框和内容之间的距离

bgcolor：背景颜色

width：定义表格的宽度

align：定义表格的位置

tr属性:

bgcolor：背景颜色

align：定义表格的位置

td属性:

bgcolor：背景颜色

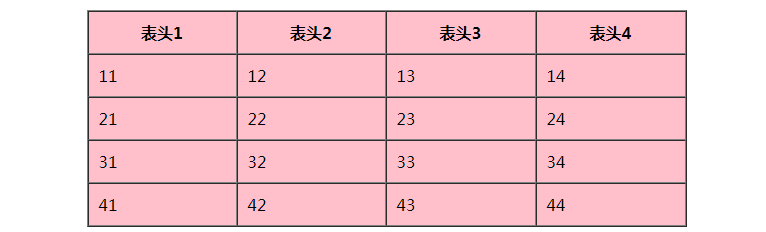
align：定义表格的位置

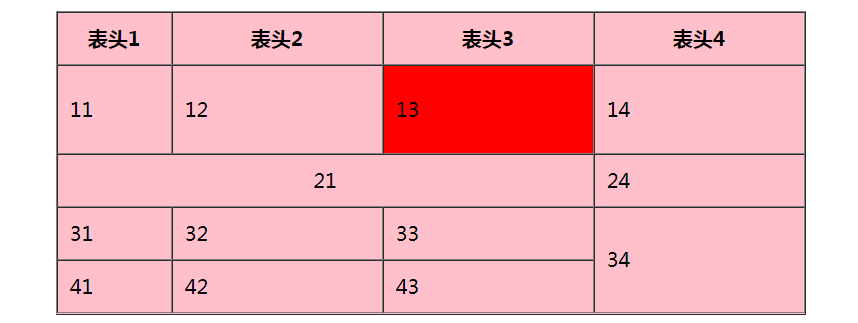
width：定义单元格的宽度

height：定义单元格的高度

colspan：定义单元格横跨的列数

rowspan：定义单元格竖跨的行数





HTML文档规则表格内容如下

|  |
| --- |
| <h3 align="center">我是一个表格</h3>  <table border="1" cellspacing="0" cellpadding="10" bgcolor="pink" width="600px" align="center">  <!--<caption>我是表格</caption>-->  <tr>  <th>表头1</th>  <th>表头2</th>  <th>表头3</th>  <th>表头4</th>  </tr>  <tr>  <td>11</td>  <td>12</td>  <td>13</td>  <td>14</td>  </tr>  <tr>  <td>21</td>  <td>22</td>  <td>23</td>  <td>24</td>  </tr>  <tr>  <td>31</td>  <td>32</td>  <td>33</td>  <td>34</td>  </tr>  <tr>  <td>41</td>  <td>42</td>  <td>43</td>  <td>44</td>  </tr>  </table> |

HTML文档不规则表格内容如下

|  |
| --- |
| <h3 align="center">我是一个表格</h3>  <table border="1" cellspacing="0" cellpadding="10" bgcolor="pink" width="600px" align="center">  <!--<caption>我是表格</caption>-->  <tr>  <th>表头1</th>  <th>表头2</th>  <th>表头3</th>  <th>表头4</th>  </tr>  <tr>  <td width="70" height="50">11</td>  <td>12</td>  <td bgcolor="red">13</td>  <td>14</td>  </tr>  <tr>  <td colspan="3" align="center">21</td>  <td>24</td>  </tr>  <tr>  <td>31</td>  <td>32</td>  <td>33</td>  <td rowspan="2">34</td>  </tr>  <tr>  <td>41</td>  <td>42</td>  <td>43</td>  </tr>  </table> |

### 2.4.7.表单标签<form>

1.浏览器向服务器发送数据的两种方式

(1) 通过超链接向服务器发送数据

https://www.baidu.com/?user=zhangsan&password=123&like=lanqiu&like=zuqiu

其实就是在超链接后面通过?拼接参数将数据带给服务器

参数和参数值之间用等号分隔, 参数可以有多个, 多个参数之间用&分隔, 并且参数的名字可以重复

(2) 通过表单向服务器发送数据

其实里通过表单及表单项标签, 用户可以通过表单项输入数据, 通过提交表单向服务器发送数据.

2.form标签

<form action=”http://www.baidu.com” method=”GET”></form>

action 必须存在的属性, 用来指定表单提交的目的地地址

method 可选属性, 用来指定以何种方式来提交表单, 如果不指定, 默认是GET提交

HTTP协议中规定了7种提交方式, 其中5种都不常用, 只用GET和POST.

只有使用表单, 并且明确的指定了提交方式为POST时, 才是POST提交, 其他方式都是GET提交

GET和POST提交的区别: 主要区别在于请求参数传输过程的不相同

GET提交:

通过在地址栏拼接参数将数据发送给服务器

数据显示在地址栏, 非常不安全

通过地址栏发送数据, 数据量不能太大, 不能超过1kb或者4kb

POST提交:

通过底层的流将数据发送给服务器

没有将数据显示在地址栏, 相对更安全

没有通过地址栏发送数据, 数据量理论上没有限制

3.表单中的项

表单中可以有多个输入项，输入项必须有name属性才可以被提交，如果输入项没有name属性，则表单在提交时会忽略它

(1)<input> 输入框

type属性: type属性的值可以有多个, 当type属性的值不同时, input标签的功能效果也不同, 例如:

type=”text” 文本输入框

type=”password” 密码框

通过size属性设置框的宽度

type=”radio” 单选框 多个单选框的name属性值必须一致才可以单选, 并且需要通过value属性指定被提交时的值

type=”checkbox” 复选框(多选框) 需要通过value属性指定被提交时的值

type=”submit” 提交按钮 用来提交表单

type=”reset” 重置按钮, 将表单项重置到初始时的状态

type=”button” 普通按钮

type=”file” 文件上传项, 可以选择文件进行上传

type=”hidden” 隐藏域，一般在信息修改的时候用来隐藏用户id

通用的属性:

readonly=”readonly” 设置表单项为只读, 只能读不能写, 但是可以被提交

disabled=”disabled” 设置表单项为禁用, 不能被提交.

(2)<textarea> 文本域

<textarea cols=”30” rows=”5”>请输入个人信息…</textarea>

属性:

cols 设置列数, 即输入框的宽度

rows 设置行数, 即输入框的高度

(3)<select> <option>标签

select 定义一个下拉选框

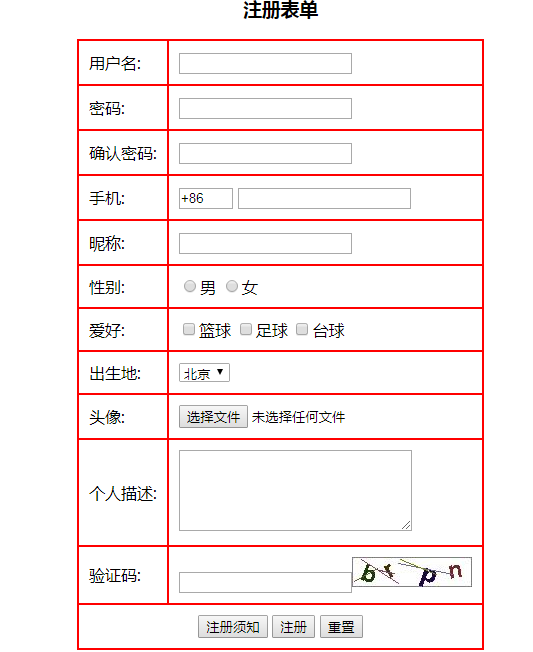
option 用来定义下拉选框中的选项

select属性:

name属性 -- name指定该项提交时参数的名字

size属性 –- 指定可见选项的个数

multiple –- 指定支持多选



4.HTML中内容

|  |
| --- |
| <!--表单标签-->  <h3 align="center">注册表单</h3>  <form action="http://www.baidu.com" method="post">  <table border="1" bordercolor="red" cellspacing="0" cellpadding="10" align="center">  <tr>  <td>用户名:</td>  <td>  <input type="text" name="username"/>  </td>  </tr>  <tr>  <td>密码:</td>  <td>  <input type="password" name="password"/>  </td>  </tr>  <tr>  <td>确认密码:</td>  <td>  <input type="password" name="password2"/>  </td>  </tr>  <tr>  <td>手机:</td>  <td>  <input type="text" value="+86" readonly="readonly" size="3"/>  <input type="text" name="phone"/>  </td>  </tr>  <tr>  <td>昵称:</td>  <td>  <input type="text" name="nickname"/>  </td>  </tr>  <tr>  <td>性别:</td>  <td>  <input type="radio" name="sex" value="man"/>男  <input type="radio" name="sex" value="women"/>女  </td>  </tr>  <tr>  <td>爱好:</td>  <td>  <input type="checkbox" name="like" value="lanqiu"/>篮球  <input type="checkbox" name="like" value="zuqiu"/>足球  <input type="checkbox" name="like" value="taiqiu"/>台球  </td>  </tr>  <tr>  <td>出生地:</td>  <td>  <select name="city">  <option value="beijing">北京</option>  <option value="shanghai">上海</option>  <option value="guangzhou">广州</option>  <option value="shenzhen">深圳</option>  </select>  </td>  </tr>  <tr>  <td>头像:</td>  <td>  <input type="file" name="photo"/>  </td>  </tr>  <tr>  <td>个人描述:</td>  <td>  <textarea cols="30" rows="5" name="description"></textarea>  </td>  </tr>  <tr>  <td>验证码:</td>  <td>  <input type="text" name="valistr"/><img src="E:\web\html\resouce\vc.jpg" alt="验证码" />  </td>  </tr>  <tr>  <td colspan="2" align="center">  <input type="button" onclick="alert('这里是注册须知')" value="注册须知"/>  <input type="submit" value="注册"/>  <input type="reset"/>  </td>  </tr>  </table>  </form> |

### 2.4.8. div、span、p标签

这三个标签都可以看作是一个容器, 可以用来包裹其他的 html内容, 被包裹起来的内容形成了一个组, 可以通过css样式对这一组的内容统一设置样式.

div: 独占一行，标签可以把文档分割为独立的、不同的部分。

span：并非独占一行，多个span会并排排列，直到一行排满才会换行，标签被用来组合文档中的行内元素。

p：独占一行，但是元素会自动在其前后创建一些空白，是段落标签。

元素的类型:

块级元素: 默认情况下独占一行的元素, 如: div p h1~h6 br hr

行内元素: 默认情况下, 多个行内元素可以处在同一行。如：span input font…

## 2.5.作业练习

1.post提交和get提交的区别

答：GET和POST提交的区别: 主要区别在于请求参数传输过程的不相同

GET提交:

通过在地址栏拼接参数将数据发送给服务器

数据显示在地址栏, 非常不安全

通过地址栏发送数据, 数据量不能太大, 不能超过1kb或者4kb

POST提交:

通过底层的流将数据发送给服务器

没有将数据显示在地址栏, 相对更安全

没有通过地址栏发送数据, 数据量理论上没有限制