

HTML

HTML基本结构

```
<html>
  <head>
    <meta charset="GBK">
    <title>网页标题</title>
  </head>
  <body>
    内容
  </body>
</html>
```

img标签

img标签的作用

告诉浏览器我们需要显示一张图片

img标签的格式

```
<img src="">
```

注意点

- 和H系列标签/p标签还有Hr标签不一样, img标签不会独占一行
- 如果我们手动指定了img标签显示的图片的宽度和高度, 有可能导致图片变形, 那么如果又想指定宽度和高度, 又不想让图片变形. 我们可以只指定宽度和高度其中的一个值即可
- 只要指定了高度, 系统会自动根据高度计算出宽度, 只要指定了宽度, 系统会自动根据宽度计算出高度, 并且都是等比拉伸的, 也就是说不会变形

img标签中的其他属性

- img标签中width/height这两个属性的作用, 就是用来告诉img标签将来需要显示的图片有多宽有多高

```

```

- 如果img标签没有指定需要显示的图片的宽高, 那么系统会按照图片默认的宽高来显示
- 如果img标签指定的宽高, 那么系统会按照指定的宽高来显示
- title: 用于告诉浏览器, 当鼠标悬停在图片上时, 需要弹出的描述框中显示什么内容
- alt其实是英文alternate的缩写, 它的作用就是用于告诉浏览器, 当需要显示的图片找不到时显示什么内容

```

```

```

```

br标签

br标签作用: 换行

br标签注意点

- 多个br标签可以连续使用, 使用了多少个br标签就会换多少行
- 由于HTML的作用就是用来给文本添加语义, 而br标签的语义是不另起一个段落换行, 而在企业开发中一般情况下需要换行都是因为需要另起一个段落, 所以在企业开发中很少使用br标签

路径问题

想给src属性赋值有两种方式

相对路径

相对路径就是每次都从.html文件所在的文件夹开始查找, 我们称之为相对路径

- 同级
 - 同级就是"图片"和".html文件"存储在同一个文件夹中
 - 格式: src="Code.jpg"
 - 含义: 在.html文件所在的文件夹中查找名称叫做Code.jpg的图片
- 下级
 - 下级就是"存储图片的文件夹"和".html文件"在同一个文件夹中
 - 格式: src="images/Code.jpg"
 - 含义: 在.html文件所在的文件夹中找到名称叫做images的文件夹, 然后再在images文件夹中找到名称叫做QRCode.jpg的图片
- 上级
 - 上级就是"存储图片的位置"和"存储代码的文件夹"在同一个文件夹中
 - 格式: src="../../QRCode.jpg"
 - 含义: 在.html文件所在的文件夹中找到这个文件夹的上一级文件夹, 然后再在上一级文件夹中找到名称叫做Code.jpg其中../代表找到当前文件夹的上一级文件夹

绝对路径

绝对路径就是每次都从指定的盘符开始查找

格式: src="C:\Users\Peng\Pictures\img.png"

a标签基本使用

a标签的作用: 就是用于控制页面与页面之间跳转的

a标签的格式:

需要展现给用户查看的内容

target属性: 这个属性的作用就是专门用于控制如何跳转

_self: 用于在当前选项卡中跳转, 也就是不新建界面跳转. 默认就是_self

_blank: 用于在新的选项卡中跳转, 也就是新建界面跳转

a标签中还有一个属性, 叫做title. a标签中的title和img标签中的title一样, 都是用来控制鼠标悬停时显示的提示文本的

注意点:

- a标签不仅可以让文字可以点击, 还可以让图片也能够被点击
- 一个a标签必须有一个href属性, 否则a标签不知道要跳转到什么地方
- 如果通过a标签的href属性指定一个URL地址, 那么必须在地址前面加上http://或https://
- a标签的href属性除了可以指定一个网络地址以外, 还可以指定一个本地地址

```
<a href="https://www.nuomi.com/?cid=002540">糯米</a>
```

```
<a href="test/demo/路径练习.html">路径练习.html</a>
```

base标签

base标签就是专门用来统一的指定当前网页中所有的超链接(a标签)需要如何打开

注意点

- base标签必须写在head标签的开始标签和结束标签之间
- 如果既在base中指定了target又在a标签中指定了target,那么浏览器会按照a标签中指定的来执行

```
<head>
  <meta charset="UTF-8">
  <title>base标签</title>
  <base target="_blank">
</head>
```

假链接

就是点击之后不会跳转的链接我们称之为假链接

假链接的格式

- #
- JavaScript:

两者的区别

#的假链接会自动回到网页的顶部, 而javascript:的假链接不会自动回到网页顶部

锚点

要想通过a标签跳转到指定的位置, 那么必须告诉a标签一个独一无二的身份证号码id, 这样a标签才能在当前界面中找到需要跳转到的目标位置

要想实现通过a标签跳转到指定的位置分为两步

- 给目标位置的标签添加一个id属性, 然后指定一个独一无二的值
- 告诉a标签你需要跳转到的目标标签对应的独一无二的身份证号码是多少

格式:

```
<a href="#center">跳转到中部</a>
<h2 id="center">我是中部</h2>
```

注意点:

- 通过我们的a标签跳转到指定的位置, 是没有过度动画的, 是直接一下子就跳转到了指定位置
- a标签除了可以跳转到当前界面的指定位置以外, 还可以在跳转到其它界面的时候直接跳转到其它界面的指定位置

格式:

```
<a href="锚点测试界面.html#bottom" target="_blank">跳转到锚点测试界面</a>
<h2 id="bottom">我是锚点测试界面</h2>
```

无序列表

什么是列表标签

列表标签的作用:

给一堆数据添加列表语义, 也就是告诉搜索引擎告诉浏览器这一堆数据是一个整体

HTML列表标签的分类

- 无序列表(最多)(unordered list)
- 有序列表(最少)(ordered list)
- 定义列表(其次)(definition list)

无序列表的作用

给一堆数据添加列表语义, 并且这一堆数据中所有的数据都没有先后之分

无序列表的格式

```
<ul>
  <li>需要显示的条目内容</li>
</ul>
```

注意点

- 一定要记住ul标签是用来给一堆数据添加列表语义的, 而不是用来给他们添加小圆点的
- ul标签和li标签是一个整体, 是一个组合. 所以一般情况下ul标签和li标签都是一起出现, 不会单个出现. 也就是说不会单独使用一个ul标签或者单独使用一个li 标签, 都是结合在一起使用
- 由于ul标签和li标签是一个组合, 所以ul标签中不推荐包含其它标签, 也就是说以后你在ul标签中只会看到li标签, li里边可以包含其他标签

```
<ul>
  <li>广州</li>
  <li>北京</li>
  <li>上海</li>
  <li>武汉</li>
</ul>
```

有序列表

有序列表的作用：

给一堆数据添加列表语义，并且这一堆数据中所有的数据都有先后之分

有序列表的格式

```
<ol>
  <li></li>
</ol>
```

其它用法和ul都差不多，也就是应用场景/注意点都和ul差不多

定义列表

先通过dt标签定义列表中的所有标题，然后再通过dd标签给每个标题添加描述信息

定义列表的格式

```
<dl>
  <dt></dt>
  <dd></dd>
  <dt></dt>
  <dd></dd>
</dl>
```

dt是英文definition title的缩写，所以dt的含义就是用来定义列表中的标题

dd是英文definition description的缩写，所以dd的含义就是用来定义标题对应的描述的

应用场景

- 做网站尾部的相关信息
- 做图文混排

```
<dl>
  <dt>北京</dt>
  <dd>中国的首都</dd>
  <dd>雾都，雾霾比较严重</dd>
  <dt>上海</dt>
  <dd>富人集中地</dd>
</dl>
```

北京

中国的首都

雾都, 雾霾比较严重

上海

富人集中地

表格标签

表格标签作用：
用来给一堆数据添加表格语义

表格标签的格式：

```
<table>
  <tr>
    <td>需要显示的内容</td>
  </tr>
</table>
```

- tr标签代表整个表格中的一行数据
- td标签代表表格中一行中的一个单元格

注意点：

- 表格标签有一个边框属性, 这个属性决定了边框的宽度. 默认情况下这个属性的值是0, 所以看不到边框
- 表格标签和列表标签一样, 它是一个组合标签, 所以table/tr/td要么一起出现, 要么一起不出现, 不会单个出现

```
<table border="1">
  <tr>
    <td>1.1</td>
    <td>1.2</td>
    <td>1.3</td>
  </tr>
  <tr>
    <td>2.1</td>
    <td>2.2</td>
    <td>2.3</td>
  </tr>
</table>
```

表格的标题

在表格标签中提供了一个标签专门用来设置表格的标题, 这个标签叫做caption. 只要将标题写在caption标签中, 那么标题就会自动相对于表格的宽度居中

caption注意点:

- caption一定要写在table标签中, 否则无效
- caption一定要紧跟在table标签后面

单元格标题标签

在表格标签中提供了一个标签专门用来存储每一列的标题, 这个标签叫做**th**标签, 只要将当前列的标题存储在这个标签中就会自动居中+加粗文字

```
<table bgcolor="black" cellspacing="1px" width="800px" align="center">
  <caption>
    <h2>今日小说排行榜</h2>
  </caption>
  <tr bgcolor="#a9a9a9">
    <th>排名</th>
    <th>关键词</th>
    <th>趋势</th>
    <th>今日搜索</th>
    <th>最近七日</th>
    <th>相关链接</th>
  </tr>
  <tr bgcolor="white" align="center">
    <td>1</td>
    <td align="left">暴走大事件</td>
    <td>
      
    </td>
    <td>623557</td>
    <td>4088311</td>
    <td>
      <a href="#">贴吧</a>
      <a href="#">图片</a>
      <a href="#">百科</a>
    </td>
  </tr>
</table>
```

表格的完整结构

```
<table>
  <caption>表格的标题</caption>
  <thead>
    <tr>
      <th>每一列的标题</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>数据</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>数据</td>
    </tr>
  </tfoot>
</table>
```

```
</tfoot>
</table>
```

caption作用：指定表格的标题
thead作用：指定表格的表头信息
tbody作用：指定表格的主体信息
tfoot作用：指定表格的附加信息

注意点

- 如果我们没有编写tbody, 系统会系统给我们添加tbody
- 如果指定了thead和tfoot, 那么在修改整个表格的高度时, thead和tfoot有自己默认的高度, 不会随着表格的高度变化而变化

单元格的合并

水平方向上的单元格合并

可以给td标签添加一个colspan属性, 来指定把某一个单元格当做多个单元格来看待(水平方向)
例如:
`<td colspan="2"></td>`
含义: 把当前单元格当做两个单元格来看待

注意点

- 由于把某一个单元格当做了多个单元格来看到, 所以就会多出一些单元格, 所以需要删掉一些单元格才能正常显示
- 一定要记住单元格合并永远都是向后或者向下合并, 而不能向前或者向上合并

垂直方向上的单元格合并

可以给td标签设置一个rowspan属性, 来指定把某一个单元格当做多个单元格来看待(垂直方向)
例如:
`<td rowspan="2"></td>`
含义: 把当前单元格当做两个单元格来看待

```
<table bgcolor="black" width="500px" height="300px" align="center">
  <tr bgcolor="white">
    <td rowspan="2"></td>
    <td></td>
    <td></td>
  </tr>
  <tr bgcolor="white">
    <!--<td></td>-->
    <td></td>
    <td></td>
  </tr>
  <tr bgcolor="white">
    <td></td>
    <td></td>
    <td></td>
  </tr>
</table>
```


表单标签

表单就是专门用来收集用户信息的

表单格式

```
<form>
  <表单元素>
</form>
```

常见的表单元素

input标签，**input**标签有一个**type**属性，这个属性有很多类型的取值，取值的不同就决定了**input**标签的功能和外观不同

表单元素一定要写在表单中

```
<form>
  <!-- 明文输入框 -->
  账号:<input type="text"><br>
  <!-- 暗文输入框 -->
  密码:<input type="password"><br>
  <!-- 给输入框设置默认值 -->
  账号:<input type="text" value="lnj"><br>
  密码:<input type="password" value="123"><br>

  <!--
  单选框
  注意点:
  1. 默认情况下单选框不会互斥, 要想单选框互斥那么必须给每一个单选框标签都设置一个name属性, 然后name属性还必须设置相同的值
  2. 要想让单选框默认选中某一个框子, 那么可以给input标签添加一个checked属性
  3. 在HTML中如果属性的取值和属性的名称一样, 可以只写一个. 但是在XHTML中必须写上取值, 所以在企业开发中我们推荐大家不要省略取值
  -->
  性别:
  <input type="radio" name="xx" checked>男
  <input type="radio" name="xx">女
  <input type="radio" name="xx" >保密<br>

  <!-- 多选框 -->
  爱好:
  <input type="checkbox">篮球
  <input type="checkbox">足球
  <input type="checkbox" checked="checked">棒球
  <input type="checkbox" checked="checked">足浴
</form>
```

```
<form action="http://www.520it.com">
  <!-- 明文输入框 -->
  账号:<input type="text" name="aa"><br>
  <!-- 暗文输入框 -->
  密码:<input type="password" name="bb"><br>
```

```

<!--
定义普通按钮
可以通过value属性来给按钮指定标题
作用：配合JS完成一些操作
-->
<input type="button" value="我是按钮">
<!--
图片按钮
作用：配合JS完成一些操作
-->
<input type="image" src="images/register.jpg">
<!--
重置按钮
作用：用于清空表单中已经填写好的数据
注意点：
如果想改重置按钮默认的按钮标题可以通过value属性来修改
-->
<input type="reset" value="清空">
<!--
提交按钮
作用：将表单中已经填写好的数据，提交到远程服务器
注意点：
要想把表单中填写好的数据提交到远程服务器，必须具备两个条件
1.需要给form表单添加一个action的属性，通过这个action属性指定需要提交到的服务器地址
2.需要给需要提交到服务器的表单元素添加一个name属性
-->
<input type="submit">

<!--
隐藏域
作用：配合提交按钮将一些数据默默的悄悄咪咪的提交到服务器
Ajax
-->
<input type="hidden" name="cc" value="kukuku">
</form>

```

Label标签

默认情况下文字和输入框是没有关联关系的，也就是说点击文字输入框不会聚焦，如果想点击文字时让对应的输入框聚焦，那么就需要让文字和输入框进行绑定，也就是点击文字后鼠标聚焦到输入框
要想让输入框和文字绑定在一起，那么我们可以使用Label标签

绑定的格式

- 将文字利用label标签包裹起来
- 给输入框添加一个id属性
- 在label标签中通过for属性和输入框中的id进行绑定即可

```
<label for="account">账号:</label><input type="text" id="account">
```

```

<form action="">
  <label for="account">账号:</label><input type="text" id="account"><br>
  <label for="pwd">密码:</label><input type="password" id="pwd"><br>
</form>

```

Datalist标签

作用：
给输入框绑定待选项

datalist格式：

```
<datalist>
  <option>待选项内容</option>
</datalist>
```

如何给输入框绑定待选列表

- 搞一个输入框
- 搞一个datalist列表
- 给datalist列表标签添加一个id
- 给输入框添加一个list属性,将datalist的id对应的值赋值给list属性即可

请输入你的车型：<input type="text" list="cars">

```
<datalist id="cars">
  <option>奔驰</option>
  <option>宝马</option>
  <option>奥迪</option>
  <option>路虎</option>
  <option>宾利</option>
</datalist>
```

表单标签H5特性

```
<form>
  <!--
  可以自动校验输入的内容是否符合邮箱的格式
  -->
  邮箱:<input type="email"><br>
  <!--
  可以自动校验输入的内容是否是URL地址
  -->
  域名:<input type="url"><br>
  <!--
  输入框中只能输入数字
  -->
  电话:<input type="number"><br>
  <!--
  可以通过日历来选择时间
  -->
  时间:<input type="date"><br>

  <!--
  可以通过取色板来选择颜色
  -->
  颜色: <input type="color"><br>

  <input type="submit">
```

```
</form>
```

表单标签之下拉框

select标签

作用：

用于定义下拉列表

格式

```
<select>
  <optgroup label="分组名称">
    <option>列表数据</option>
  </optgroup>
</select>
```

注意点

- 下拉列表不能输入内容,但是可以直接在列表中选择内容
- 可以通过给option标签添加一个selected属性来指定列表的默认值
- 可以通过给option标签包裹一层optgroup标签来给下拉列表中的数据分类

```
<select>
  <option>北京</option>
  <option>上海</option>
  <option>广州</option>
  <option>广西</option>
  <option selected="selected">武汉</option>
</select>
```

```
<select>
  <optgroup label="北京">
    <option>朝阳区</option>
    <option>昌平区</option>
    <option>通州区</option>
  </optgroup>
  <optgroup label="广州">
    <option>天河区</option>
    <option>越秀区</option>
    <option>黄浦区</option>
  </optgroup>
</select>
```

表单标签之textarea标签

作用：

定义一个多行输入框

格式

```
<textarea></textarea>
```

```
<textarea cols="20" rows="5"></textarea>
```

注意点

- 默认情况下输入框可以无限换行
- 默认情况下输入框有自己的宽度和高度
- 可以通过cols和rows来指定输入框的宽度和高度, 但是虽然指定了列数和行数, 但是还是可以无限往下输入
- 默认情况下输入框是可以手动拉伸的

武汉 ▼ 朝阳区 ▼

练习

```
<form action="http://www.it.com">
  <!--
  fieldset标签: 可以给表单添加一个边框
  legend标签: 可以给边框指定一个标题
  -->
  <fieldset>
    <legend>注册界面</legend>
    <p>
      账号: <input type="text" name="account">
    </p>
    <p>
      密码: <input type="password" name="pwd">
    </p>
    <p>
      性别:
      <input type="radio" name="gender" value="male">男
      <input type="radio" name="gender" value="female">女
      <input type="radio" name="gender" checked="checked" value="yao">保密
    </p>
    <p>
      <!--注意点:
      单选框和多选框的name都需要指定相同的name值
      -->
      爱好:
      <input type="checkbox" name="sport" value="basketball">篮球
      <input type="checkbox" name="sport" value="football">足球
      <input type="checkbox" name="sport" checked="checked"
value="crazy">足浴
    </p>
  </p>
```



```
<video>
  <source src="" type=""></source>
  <source src="" type=""></source>
</video>
```

video标签的第二种格式存在的意义就是为了解决浏览器适配问题

video 元素支持三种视频格式，我们可以把这三种格式都通过**source**标签指定给**video**标签，那么以后当浏览器播放视频时它就会从这三种中选择一种自己支持的格式来播放

audio标签

作用：播放音频

格式：

```
<audio src="">
</audio>

<audio>
  <source src="" type="">
</audio>
```

注意点

audio标签的使用和**video**标签的使用基本一样，**video**中能够使用的属性在**audio**标签中大部分都能够使用，并且功能都一样
只不过有3个属性不能用，**height/width/poster**

```
<audio autoplay="autoplay" controls="controls">
  <source src="images/yinyue.mp3" type="audio/mp3">
</audio>
```

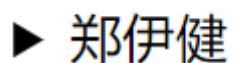
详情和概要标签

作用：

利用**summary**标签来描述概要信息，利用**details**标签来描述详情信息默认情况下是折叠展示，想看见详情必须点击

格式：

```
<details>
  <summary>概要信息</summary>
  详情信息
</details>
```



▼ 郑伊健

简介：郑伊健，1967年10月4日出生
告而入行，拜罗文为师。1991年...

字符实体

在HTML中对空格/回车/tab不敏感，会把多个空格/回车/tab当做一个空格来处理

 空格，一个 就是一个空格，有多少个 就有多少个空格
< 小于符号 <
(less than)
> 大于符号 >
(greater than)
© 版权符号