



# HTML基础

# ■ HTML概述

HTML (Hyper Text Markup Language)

HTML不区分大小写

“超文本标记语言”，是制作网页的标准语言，它不是一种编程语言，而是一种标记语言 (markup language)。标记语言是一套标记标签 (markup tag)。

**超文本，有 2 层含义：**

1. 它可以加入图片、声音、动画、多媒体等内容（超越了文本限制）。
2. 它还可以从一个文件跳转到另一个文件，与世界各地主机的文件连接（超级链接文本）。

# ■ HTML概述

## 标签、元素

由尖括号包围，比如 `<div><ul><li>`

通常是成对出现的。

```
<div class="nav">
  <ul>
    <li><a href="#">新闻</a> </li>
    <li><a href="#">hao123</a></li>
  </ul>
</div>
```

**单标签：** 图片标签`<img />`，换行标签`<br />`，普通分隔线标签`<hr />`

# ■ HTML概述

## 标签嵌套

要有正确的嵌套关系，不可以交叉。

`<html><title></title></html>`

错误： `<html><body></html></body>`

# ■ HTML概述

## 注意缩进

外层：父元素

内层：子元素

同级：兄弟元素

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta httpequiv ="Content-Type"
content="text/html; charset=UTF-8" />
    <link href="style.css" rel
="stylesheet" />
    <title>百度一下, 你就知道 </title>
  </head>
  <body>
    <div id="top">
      <div class="nav">
        <ul>
          <li><a href="#">新闻</a></li>
          <li><a href ="#">hao123</a></li>
        </ul>
      </div>
    </div>
  </body>
</html>
```

# ■ HTML概述

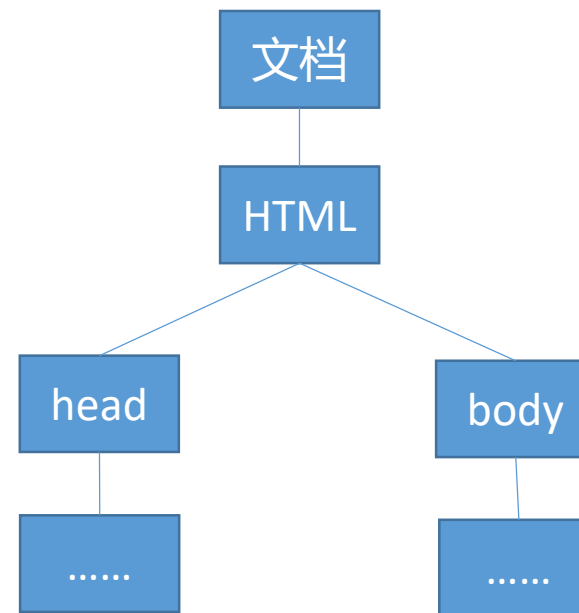
## 注意缩进

外层：父元素

内层：子元素

同级：兄弟元素

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta httpequiv = "Content-Type"
content="text/html; charset=UTF-8" />
    <link href="style.css" rel
="stylesheet" />
    <title>百度一下, 你就知道 </title>
  </head>
  <body>
    <div id="top">
      <div class="nav">
        <ul>
          <li><a href="#">新闻</a></li>
          <li><a href="#">hao123</a></li>
        </ul>
      </div>
    </div>
  </body>
</html>
```



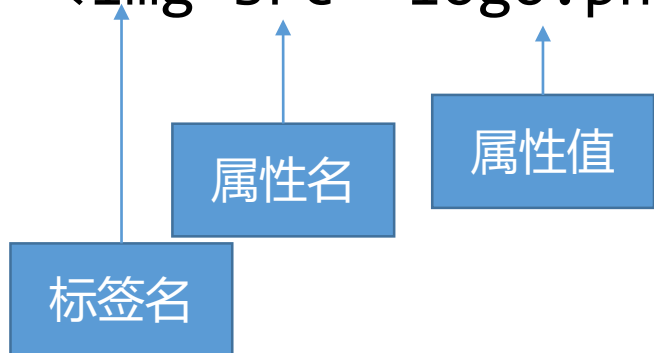
## HTML DOM 树

DOM : Document Object Model (文档对象模型)

# ■ HTML概述

```

```



一个标签可以有多个属性，而且属性之间没有先后顺序

# ■ HTML文件结构

```
<!DOCTYPE html>
```

文档类型

浏览器将按照HTML5标准解析这个网页文件

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8" />
```

```
<title>朱子家训</title>
```

```
</head>
```

头部信息

提供网页的基本信息给浏览器和搜索引擎

```
<body>
```

```
宜未雨而绸缪，毋临渴而掘井。
```

主体信息

```
</body>
```

```
</html>
```



# ■ HTML文件结构

## 字符集与编码

字符 (Character) : 文字、符号;

字符集 (Charset) : 字符的集合, 比如英文字符集、汉字字符集、日文汉字字符集

编码: 将字符和二进制码对应起来



# ■ HTML标签

标题标签: h1~h6

段内换行: br

段内分组: span

段落标签: p

预留格式: pre

水平线: hr

# HTML标签: h1~h6

网易首页 > 新闻中心 > 新闻 > 正文

## 时习之 网络强国 | 共建网上美好精神家园 习近平推动网络文明建设

2022-08-26 19:18:12 来源: 人民网-中国共产党新闻网 北京

举报



514

分享至



(原标题: 时习之 网络强国 | 共建网上美好精神家园 习近平推动网络文明建设)

网络文明是新形势下社会文明的重要内容,是建设网络强国的重要领域。党的十八大以来,以习近平同志为核心的党中央不断推进互联网内容建设,不断培育积极健康、向上向善的网络文化,倡导用社会主义核心价值观和人类优秀文明成果滋养人心、滋养社会。



网易新闻  
iOS Android

### 专题推荐

永远跟党走

数博2022全球传播行动

京彩e品 百年初心映党旗

我的奥运

百年回响

古都风貌 时代风貌

世界环境日主题活动

百年奋进 京彩启航

给你更好的爱

百年京张,筑梦冬奥

```
<h1>卢橘夏熟</h1>
<h2>黄甘橙榛</h2>
<h3>枇杷燃柿</h3>
<h4>亭奈厚朴</h4>
<h5>栲枣杨梅</h5>
<h6>樱桃蒲陶</h6>
```

卢橘夏熟

黄甘橙榛

枇杷燃柿

亭奈厚朴

栲枣杨梅

樱桃蒲陶

一个页面建议只有一个h1

# ■ HTML标签: 段落p

## <p> 段落内容 </p>

```
<p>大汕岛位于德胜河上，坐落在大良与容桂之间，隶属容桂小黄圃，是西江顺德段的第一大岛。</p>  
<p>大汕岛原名“大伞岛”，因岛屿形状而得名，也有一说是一条叫“大汕”的河涌横穿此中，因而取名大汕岛。</p>  
<p>段落内部文字忽略连续 空格，  
也不会显示空行，且不会换行。</p>
```

大汕岛位于德胜河上，坐落在大良与容桂之间，隶属容桂小黄圃，是西江顺德段的第一大岛。

大汕岛原名“大伞岛”，因岛屿形状而得名，也有一说是一条叫“大汕”的河涌横穿此中，因而取名大汕岛。

段落内部文字忽略连续 空格，也不会显示空行，且不会换行。

## ■ HTML标签: 段内换行br

**<br /> 单独出现的标签，直接结束**

```
<p>大汕岛原名“大伞岛”，因岛屿形状而得名，也有一说是一条叫“大汕”的河涌横穿此中。  
<br />因而取名大汕岛。</p>
```

大汕岛原名“大伞岛”，因岛屿形状而得名，也有一说是一条叫“大汕”的河涌横穿此中。  
因而取名大汕岛。

## ■ HTML标签: 空格标签&nbsp;

注意以&开头, ;结束

```
<p>  
    段落内部文字忽略连续 &nbsp;&nbsp;&nbsp;空格, 也不会显示空行, <br />  
    <br />且不会换行。  
</p>
```

段落内部文字忽略连续 空格, 也不会显示空行,  
且不会换行。

# ■ HTML标签: 特殊字符标签

在 HTML 页面中，一些特殊的符号很难或者不方便直接使用，此时我们就可以使用下面的字符来替代。

特殊字符	描述	字符的代码
	空格符	&nbsp;
<	小于号	&lt;
>	大于号	&gt;
&	和号	&amp;
¥	人民币	&yen;
©	版权	&copy;
®	注册商标	&reg;
°	摄氏度	&deg;
±	正负号	&plusmn;
×	乘号	&times;
÷	除号	&divide;
²	平方2（上标2）	&sup2;
³	立方3（上标3）	&sup3;

重点记住：空格、大于号、小于号 这三个，其余的使用很少，如果需要回头查阅即可。

## ■ HTML标签: 预留格式 pre

定义预格式化的文本，文本中的空格和换行符会被保留

```
<pre>你好    我是Natalie</pre>
<pre>
#include <stdio.h>
int main(){
    int a=10;
    int b=1;
    printf("a+b=%d", (a+b));
    return 0;
}
</pre>
```

---

```
你好    我是Natalie

#include <stdio.h>
int main() {
    int a=10;
    int b=1;
    printf("a+b=%d", (a+b));
    return 0;
}
```



## ■ HTML标签: 行内组合span

组合行内元素，以便通过CSS样式来格式化;

<span> 标签提供了一种将文本的一部分或者文档的一部分独立出来的方式。

```
<p>这只小狗有个 <span style="color:red">大大</span> 的耳朵。</p>
```

---

这只小狗有个 大大 的耳朵。

## ■ HTML标签: 水平线hr

<hr /> 单标签

```
<p>哈士奇</p>  
<hr />  
<p>太能拆家了! </p>
```

哈士奇

---

太能拆家了!

# ■ HTML标签: 注释内容

<!--此处为注释内容-->

快捷键: ctrl + /

```
<body>
  <!--这是一个注释, 注释在浏览器中不会显示-->
  <p>这是一个段落</p>
</body>
```

---

这是一个段落

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5   </head>
6   <body>
7     <!--这是一个注释, 注释在浏览器中不会显示-->
8     <p>这是一个段落</p>
9   </body>
10 </html>
11
```

# ■ HTML标签: <a>标签

## 超链接标签: <a>标签

在 HTML 标签中, <a> 标签用于定义超链接, 作用是从一个页面链接到另一个页面。

1.链接的语法格式

2.链接的分类

```
<a href="跳转目标" target="目标窗口的弹出方式"> 文本或图像 </a>
```

单词 anchor的缩写, 意为: 锚。

属性	作用
href	用于指定链接目标的url地址, (必须属性) 当为标签应用href属性时, 它就具有了超链接的功能
target	用于指定链接页面的打开方式, 其中_self为默认值, __blank为在新窗口中打开方式。

# ■ HTML标签: <a> 标签

- 1.外部链接: 例如 `< a href="http:// www.baidu.com "> 百度</a >`。
- 2.内部链接: 网站内部页面之间的相互链接。直接链接内部页面名称即可, 例如 `< a href="index.html"> 首页 </a >`。
- 3.空链接: 如果当时没有确定链接目标时, `< a href="#"> 首页 </a >`。
- 4.下载链接: 如果 href 里面地址是一个文件或者压缩包, 会下载这个文件。
- 5.网页元素链接: 在网页中的各种网页元素, 如文本、图像、表格、音频、视频等都可以添加超链接。
- 6.锚点链接: 当我们点击链接, 可以快速定位到页面中的某个位置。
  - 在链接文本的 href 属性中, 设置属性值为 #名字 的形式, 如 `<a href="#two"> 第2集 </a>`
  - 找到目标位置标签, 里面添加一个 id 属性 = 刚才的名字, 如: `<h3 id="two">第2集介绍</h3>`

# ■ HTML标签: <img> 标签

## 图像格式

### 不同格式的图片



jpg

照片

色彩丰富



png

颜色少

支持透明度



webp

体积更小

兼容性一般



gif

动画

体积大

# ■ HTML标签: <img> 标签

## 插入图像

在 HTML 标签中，<img> 标签用于定义 HTML 页面中的图像。

```

```

src 是<img>标签的必须属性，用于指定图像文件的路径和文件名。

所谓属性：简单理解就是属于这个图像标签的特性。

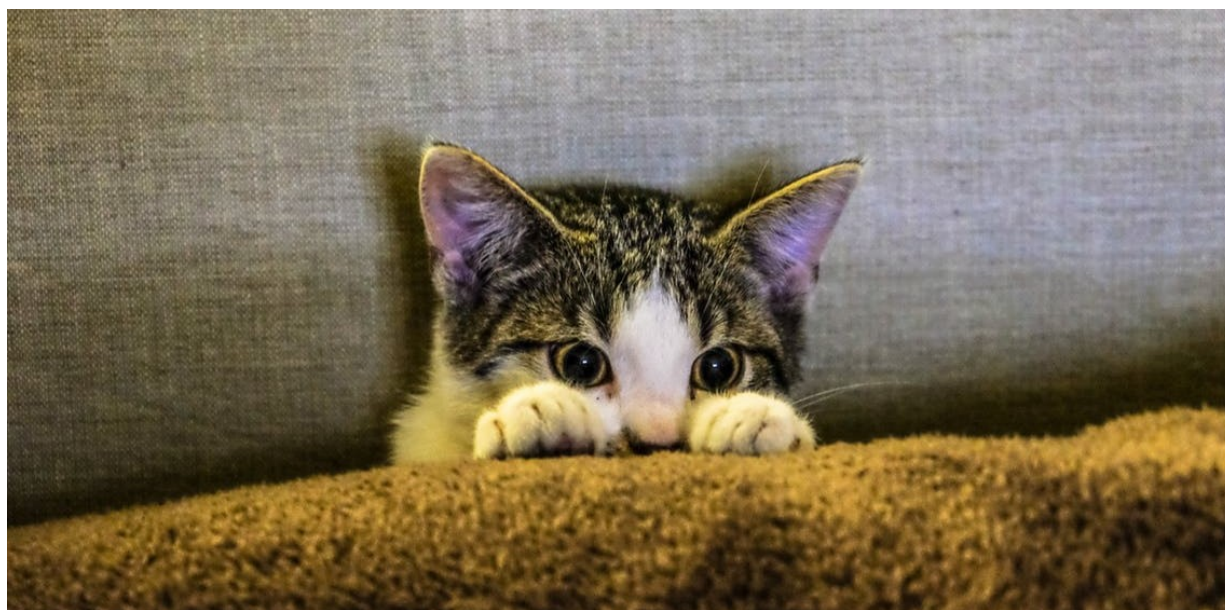
属性	属性值	说明
src	图片路径	必须属性
alt	文本	替换文本。图像不能显示的文字
title	文本	提示文本。鼠标放到图像上，显示的文字
width	像素	设置图像的宽度
height	像素	设置图像的高度
border	像素	设置图像的边框粗细

## ■ HTML标签: <img> 标签

插入图像

```

```





# ■ HTML标签: <img> 标签

## 插入图像

指定宽度和高度

```

```

src属性: 路径+文件名



# ■ HTML标签: <img> 标签

## 绝对路径与相对路径

页面中的图片会非常多，通常我们会新建一个文件夹来存放这些图像文件（images），这时再查找图像，就需要采用“路径”的方式来指定图像文件的位置。

路径可以分为：绝对路径和相对路径

**绝对路径：**是指目录下的绝对位置，直接到达目标位置，通常是从盘符开始的路径。

例如，“D:\web\img\logo.gif” 或完整的网络地址

“http://www.xxxxx.xxx/images/logo.gif” 。

# ■ HTML标签: <img> 标签

## 绝对路径与相对路径

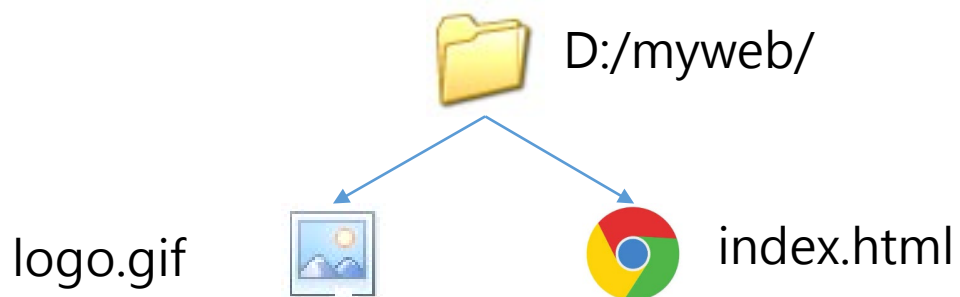
**相对路径：**以引用文件所在位置为参考基础，而建立出的目录路径。  
简单来说，就是图片相对于当前文档的位置。

相对路径分类	符号	说明
同一级路径		图像文件位于HTML文件同一级 如
下一级路径	/	图像文件位于HTML文件下一级 如。
上一级路径	../	图像文件位于HTML文件上一级 如。

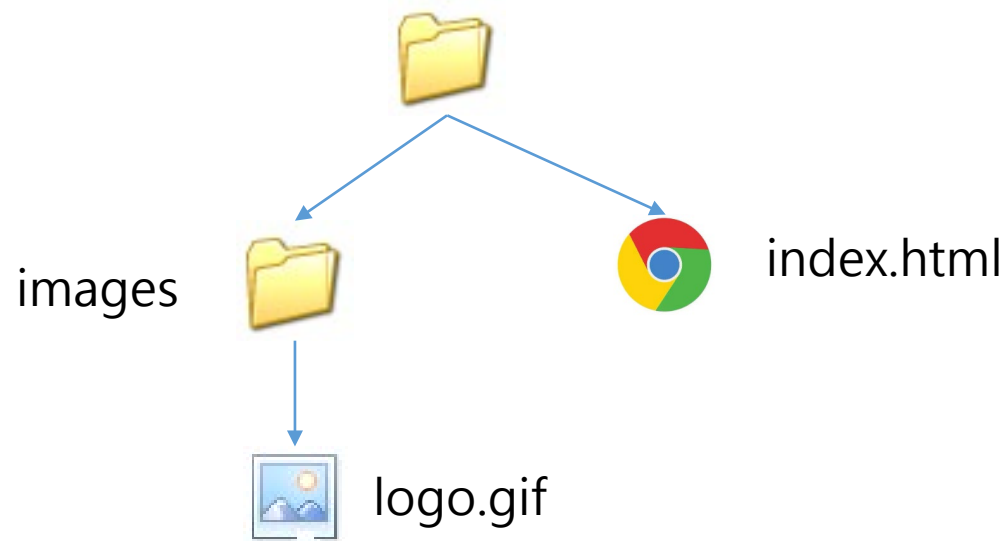
相对路径是从代码所在的这个文件出发，去寻找目标文件的，而我们这里所说的上一级、下一级和同一级就是 **图片相对于文档所在的位置**。

# ■ HTML标签: <img> 标签

## 绝对路径与相对路径



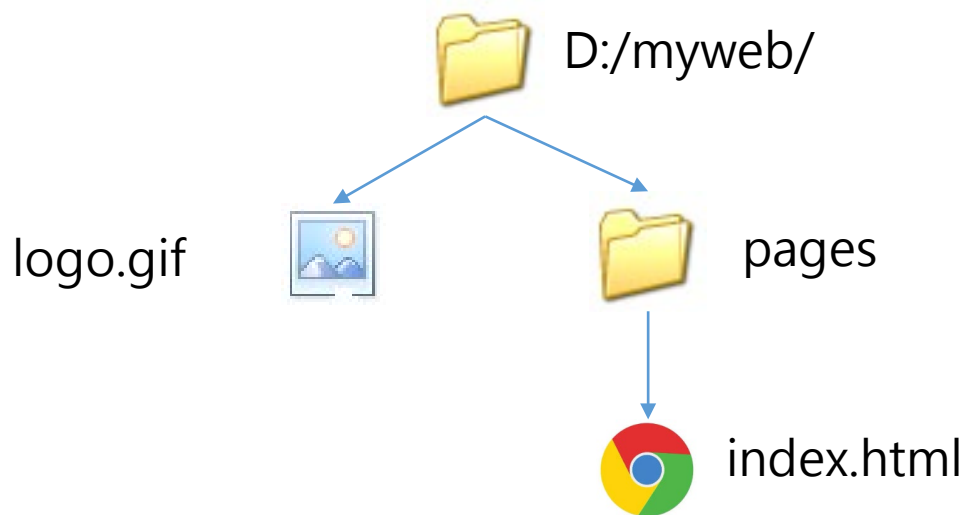
<img src = "logo.gif" />



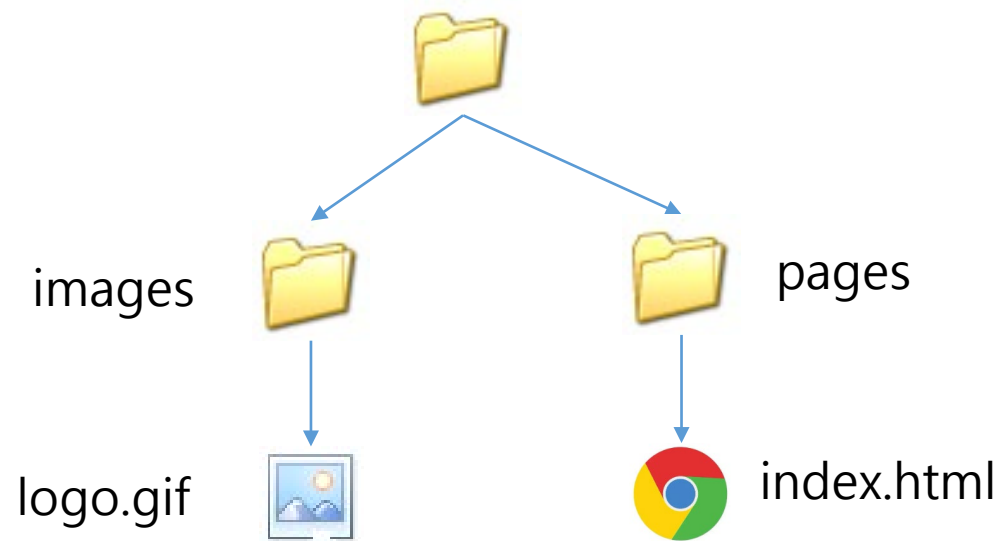
<img src = "images/logo.gif" />

# ■ HTML标签: <img> 标签

## 绝对路径与相对路径



<img src = **"../logo.gif"** />



<img src = **"../images/logo.gif"** />

# ■ HTML标签: <div> 标签

<div> 标签用来布局，一行只能放一个<div>。是个大盒子，用来定义文档中的分区或节。



## ■ HTML标签: <div> 标签

```
<div align="center">  
  <h1>web 前端开发</h1>  
  <p>HTML</p>  
  <p>CSS</p>  
  <p>JavaScript</p>  
</div>
```

确定一个单独的区域,  
比如页面的一个组成部分, 一个栏目版块

# ■ HTML标签: 列表标签

列表就是用来布局的

列表最大的特点就是整齐、整洁、有序，它作为布局会更加自由和方便。

根据使用情景不同，列表可以分为三大类：无序列表、有序列表和自定义列表。

## 购物清单

- 锅
- 碗
- 瓢
- 盆

无序列表

## 世界电影票房排行榜

1. 阿凡达
2. 泰坦尼克号
3. 复仇者联盟

有序列表

## 霸王别姬

导演：  
陈凯歌  
主演：  
张国荣  
张丰毅  
巩俐  
上映日期：  
**1993-01-01**

自定义列表



# ■ HTML标签: 列表标签

## 无序列表

```
<h2>购物清单 </h2>
```

```
<ul>
```

```
  <li>锅</li>
```

```
  <li>碗</li>
```

```
  <li>瓢</li>
```

```
  <li>盆</li>
```

```
</ul>
```

购物清单

- 锅
- 碗
- 瓢
- 盆

# ■ HTML标签: 列表标签

## 有序列表

```
<h2>世界电影票房排行</h2>
```

```
<ol start="1">
```

```
  <li>阿凡达 </li>
```

```
  <li>泰坦尼克号 </li>
```

```
  <li>复仇者联盟 </li>
```

```
</ol>
```

## 世界电影票房排行榜

1. 阿凡达
2. 泰坦尼克号
3. 复仇者联盟

# ■ HTML标签: 列表标签

## 自定义列表

自定义列表常用于对术语或名词进行解释和描述，定义列表的列表项前没有任何项目符号。

```
<h2>霸王别姬</h2>
<dl>
  <dt>导演: </dt>
  <dd>陈凯歌</dd>
  <dt>主演: </dt>
  <dd>张国荣</dd>
  <dd>张丰毅</dd>
  <dd>巩俐</dd>
  <dt>上映日期: </dt>
  <dd>1993-01-01</dd>
</dl>
```

## 霸王别姬

导演:  
陈凯歌

主演:  
张国荣  
张丰毅  
巩俐

上映日期:  
**1993-01-01**

# ■ HTML标签: 列表标签

在 HTML 标签中, `<dl>` 标签用于定义描述列表 (或定义列表), 该标签会与 `<dt>` (定义项目/名字) 和 `<dd>` (描述每一个项目/名字) 一起使用。

其基本语法如下:

```
<dl>
  <dt>名词1</dt>
  <dd>名词1解释1</dd>
  <dd>名词1解释2</dd>
</dl>
```

- `<dl>` `</dl>` 里面只能包含 `<dt>` 和 `<dd>`。
- `<dt>` 和 `<dd>` 个数没有限制, 经常是一个 `<dt>` 对应多个 `<dd>`。

## ■ HTML标签: 列表标签

标签名	定义	说明
<code>&lt;ul&gt;&lt;/ul&gt;</code>	<b>无序标签</b>	里面只能包含li 没有顺序, 使用较多。li里面可以包含任何标签
<code>&lt;ol&gt;&lt;/ol&gt;</code>	有序标签	里面只能包含li 有顺序, 使用相对较少。li里面可以包含任何标签
<code>&lt;dl&gt;&lt;/dl&gt;</code>	自定义列表	里面只能包含dt 和 dd 。 dt和dd里面可以放任何标签

# ■ HTML标签: 表格标签table tr td

表格是实际开发中非常常用的标签。

表格主要用于显示、展示数据，因为它可以让数据显示的非常的规整，可读性非常好。特别是后台展示数据 的时候，能够熟练运用表格就显得很重要。一个清爽简约的表格能够把繁杂的数据表现的很有条理。

总结: 表格不是用来布局页面的，而是用来展示数据的。

支出项目	单价 (元)	数目	金额 (元)
垃圾箱	30	12	360
垃圾牌	100	12	1200
宣传单	0.08	400	32
宣传小册子	3	250	750
合计	—	—	2342

## ■ HTML标签: 表格标签table tr td

The diagram illustrates the structure of an HTML table. A blue box labeled 'table' points to the entire table structure. A blue box labeled 'thead' points to the first row (header). A blue box labeled 'tbody' points to the remaining three rows (body). A blue box labeled 'tr' points to the first row of the body (the first data row). A blue box labeled 'th' points to the first column header '客户id'. A blue box labeled 'td' points to the cell containing '客户B' in the second column of the third row. The table itself is as follows:

客户id	客户名称	所属城市	账户余额
123	客户A	北京	3424.32
124	客户B	上海	2544.00
125	客户C	深圳	433.09

# ■ HTML标签: 表格标签table tr td

## 表格的基本语法

```
<table>
  <tr>
    <td>单元格内的文字</td>
    ...
  </tr>
  ...
</table>
```

1. <table> </table> 是用于定义表格的标签。
2. <tr> </tr> 标签用于定义表格中的行，必须嵌套在 <table> </table> 标签中。
3. <td> </td> 用于定义表格中的单元格，必须嵌套在 <tr> </tr> 标签中。
4. 字母 td 指表格数据 (table data) ，即数据单元格的内容。



# ■ HTML标签: 表格标签table tr td

<th> 标签表示 HTML 表格的表头部分(table head 的缩写), 表头单元格里面的文本内容加粗居中显示。

客户id	客户名称	所属城市	账户余额
<b>123</b>	客户 <b>A</b>	北京	<b>3424.32</b>
<b>124</b>	客户 <b>B</b>	上海	<b>2544.00</b>
<b>125</b>	客户 <b>C</b>	深圳	<b>433.09</b>

```
<table>
  <thead>
    <tr>
      <th>客户id</th>
      <th>客户名称</th>
      <th>所属城市</th>
      <th>账户余额</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>123</td>
      <td>客户A</td>
      <td>北京</td>
      <td>3424.32</td>
    </tr>
    <tr>
      <td>124</td>
      <td>客户B</td>
      <td>上海</td>
      <td>2544.00</td>
    </tr>
    <tr>
      <td>125</td>
      <td>客户C</td>
      <td>深圳</td>
      <td>433.09</td>
    </tr>
  </tbody>
</table>
```

# ■ HTML标签: 表格结构标签thead tbody

使用场景:因为表格可能很长,为了更好的表示表格的语义,可以将表格分割成**表格头部**和**表格主体**两大部分。

在表格标签中,分别用: **<thead>标签: 表格的头部区域**、**<tbody>标签: 表格的主体区域**。这样可以更好的分清表格结构。

The screenshot displays a web page with a stock table. The table has a header row with columns: 推荐股票, 当前价, 最高目标价, and 目标空间. Below the header, there are several rows of stock data. A red arrow points from the **<tbody>** tag in the developer tools to the main body of the table. Another red arrow points from the **<thead>** tag in the developer tools to the header row of the table. The developer tools show the HTML structure of the table, with the **<thead>** and **<tbody>** tags highlighted.

推荐股票	当前价	最高目标价	目标空间
安控科技	16.87	62.00	268.61%
光一科技	43.06	150.00	249.57%
领先科技	30.70	100.00	225.73%
用友软件	38.00	120.00	219.06%
南都电源	15.67	47.50	202.74%
深圳机场	8.50	25.00	196.21%
易华录	34.69	100.00	188.27%
万达信息	37.06	100.00	180.03%

Elements Network Sources Timeline Profiles Resources Audits Console

```
<div class="tabcon">
  <table width="100%" cellspacing="0" cellpadding="0" border="0" class="t
    <thead>...</thead>
    <tbody>...</tbody>
  </table>
</div>
```

# ■ HTML标签: 合并单元格

合并单元格：跨行和跨列

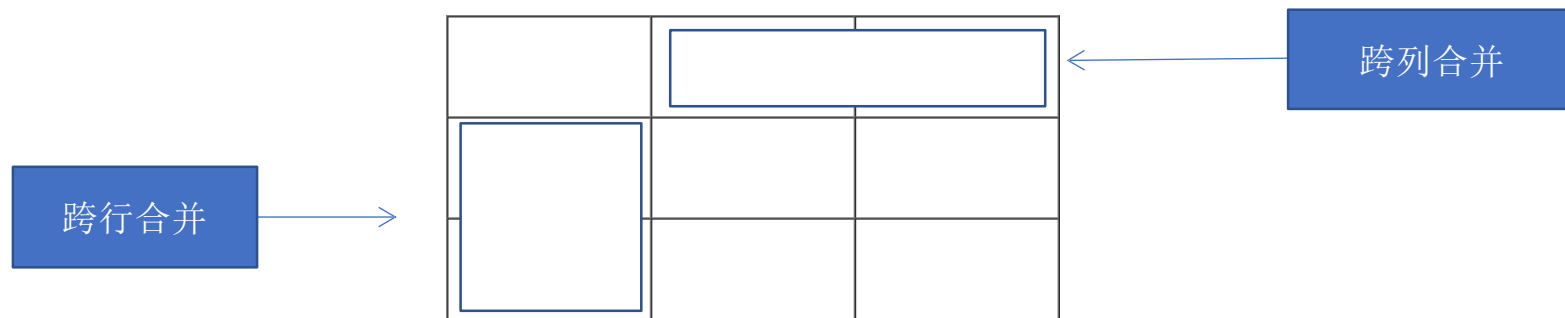
分类	技能	分数
编程语言	JavaScript	8
	C++	7
	Swift	9
产品设计	Photoshop	6
	Axure	5
销售	社交媒体	3
	SEO	5

```
<table border="1" align="center" cellspacing="0">
  <thead>
    <tr>
      <th>分类</th>
      <th>技能</th>
      <th>分数</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td rowspan="3">编程语言</td>
      <td>JavaScript</td>
      <td>8</td>
    </tr>
    <tr>
      <td>C++</td>
      <td>7</td>
    </tr>
    <tr>
      <td>Swift</td>
      <td>9</td>
    </tr>
    <tr>
      <td rowspan="2">产品设计</td>
      <td>Photoshop</td>
      <td>6</td>
    </tr>
    <tr>
      <td>Axure</td>
      <td>5</td>
    </tr>
    <tr>
      <td rowspan="2">销售</td>
      <td>社交媒体</td>
      <td>3</td>
    </tr>
    <tr>
      <td>SEO</td>
      <td>5</td>
    </tr>
  </tbody>
</table>
```

# ■ HTML标签: 合并单元格

## 合并单元格方式:

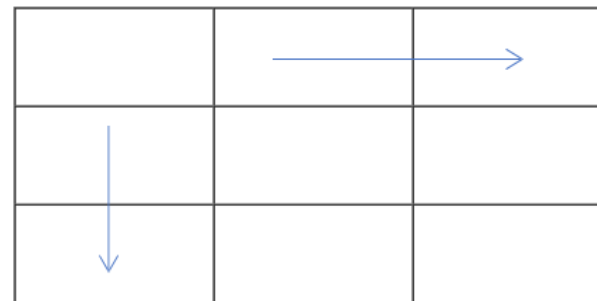
- 跨行合并: `rowspan="合并单元格的个数"`
- 跨列合并: `colspan="合并单元格的个数"`



目标单元格: (写合并代码)

跨行: 最上侧单元格为目标单元格, 写合并代码

跨列: 最左侧单元格为目标单元格, 写合并代码



## ■ HTML标签: 合并单元格

属性名	属性值	描述
align	left、center、right	规定表格相对周围元素的对齐方式。
border	1 或 ""	规定表格单元是否拥有边框，默认为""，表示没有边框
cellpadding	像素值	规定单元边沿与其内容之间的空白，默认1像素。
cellspacing	像素值	规定单元格之间的空白，默认2像素。
width	像素值或百分比	规定表格的宽度。

# ■ HTML标签: 表单与表单元素

```
<input type="" >  
text  
password  
submit  
reset  
radio  
checkbox  
checkbox
```

```
<select>  
  <option>
```

```
<textarea >
```

账户:

密码:

爱好:

性别: 男 ☐ 女 ☐

爱好: ☐ 音乐 ☐ 体育 ☐ 阅读

爱好: 

旅游

看书

旅游

运动

购物

个人简介:

在这里输入内容...

# ■ HTML标签: 表单与表单元素

**type**属性的属性值及其描述:

属性值	描述
button	定义可点击按钮（多数情况下，用来通过JS启动脚本）
checkbox	定义复选框
file	定义输入字段和“浏览”按钮，供文件上传
hidden	定义隐藏的输入字段
image	定义图像形式的提交按钮
password	定义密码字段，该字段中的字符被掩码
radio	定义单选按钮
reset	定义重置按钮，清除表单中所填数据
submit	定义提交按钮，把表单数据发送到服务器
text	定义单行的输入字段，用户可在其中输入文本，默认宽度20个字符

## ■ HTML标签: 表单与表单元素

除type属性外, <input> 标签还有很多其他属性值:

属性	属性值	描述
name	由用户自定义	定义input元素的名称
value	由用户自定义	规定input元素的值
checked	checked	规定此input元素首次加载时应当被选中
maxlength	正整数	规定输入字段中的字符最大长度



## ■ HTML标签: 表单与表单元素

用户名:

密码:

邮箱:

☐ 同意用户协议和隐私政策

**表单**: 是一个区域, 采集用户信息。使用表单目的是为了**收集用户信息**。

**表单元素**: 文本框、按钮、单选、复选、下拉列表、文本域。

在我们网页中, 我们也需要跟用户进行交互, 收集用户资料, 此时就需要表单。

在 HTML 中, 一个完整的表单通常由**表单域**、**表单控件** (也称为表单元素) 和 **提示信息** 3个部分构成。

# ■ HTML标签: 表单与表单元素

## input

用户名:

密 码:

年 龄:

头 像:  未选择任何文件

生 日:

label标签一般用来提升用户体验性，常和input标签联用。  
主要实现的是，点击label标签包裹的内容，相当于点击了input表单

# ■ HTML标签: 表单与表单元素

## 文本框、密码框input

```
<form>  
  <input type="text|password" />  
</form>
```

```
<form action="数据处理网页">  
  账户: <input type="text" name="userName" /><br />  
  密码: <input type="password" name="userPassword" />  
</form>
```

账户:

密码:

# ■ HTML标签: 表单与表单元素

## 提交按钮input--submit

```
<form>  
    姓名: <input type="text" name="userName" value=""/>  
    <input type="submit" name="submitBtn" value="提交"/>  
</form>
```

姓名:

value属性的值: 按钮上显示的文字

# ■ HTML标签: 表单与表单元素

## 重置按钮input--reset

```
<form>  
  课程名称: <input type="text" />  
  <input type="submit" name="submitBtn" value="提交" />  
  <input type="reset" name="resetBtn" value="重置" />  
</form>
```

课程名称:

value属性的值: 按钮上显示的文字

# ■ HTML标签: 表单与表单元素

单选框、复选框input—radio、checkbox

```
<form>  
  <input type="radio|checkbox"  
         value="值" name="名称" checked="checked"/>  
</form>
```

单选框：只能选择一项

多选框：任意选择多项

checked：当设置checked="checked"时，该选项被默认选中

# ■ HTML标签: 表单与表单元素

## 单选框 radio

```
<form>
  <p>请选择您最喜欢的水果 </p>
  <input type="radio" name="fruit" id="i1"> <label for="i1"> 香蕉 </label>
  <input type="radio" name="fruit" id="i2"> <label for="i2">猕猴桃 </label>
  <input type="radio" name="fruit" id="i3"> <label for="i3">葡萄 </label>
  <p>请选择您的衣服尺码 </p>
  ....
</form>
```

请选择您最喜欢的水果

☐香蕉 ☐猕猴桃 ☐葡萄

请选择您的衣服尺码

☐XS ☐S ☐M ☐L ☐XL

# ■ HTML标签: 表单与表单元素

## 多选框 checkbox

```
<form>
  <p>请选择您最喜欢的水果? </p>
  <input type="checkbox" checked= "checked" id="xj"><label for="xj">香蕉 </label>
  <input type="checkbox" id="mht"><label for="mht">猕猴桃 </label>
  <input type="checkbox" id="pt"><label for="pt">葡萄 </label>
</form>
```

请选择您最喜欢的水果?

☒ 香蕉 ☐ 猕猴桃 ☐ 葡萄



# ■ HTML标签: 表单与表单元素

## 下拉选择框 select option

```
<select name="fruit">  
  <option value="1">苹果 </option>  
  <option value="2">香蕉 </option>  
  <option value="3" selected= "selected" >芒果 </option>  
  <option value="4">菠萝 </option>  
  <option value="5">榴莲 </option>  
  <option value="6">木瓜 </option>  
</select>
```

芒果 ▼

checked= “checked” 时，被选中的一项

# ■ HTML标签: 表单与表单元素

## 多行文本框textarea

```
<textarea rows="行数" cols="列数">文本</textarea>
```

```
<textarea rows="5" cols="30"> </textarea>
```



```
<form>
  个人简介: <br >
  <textarea cols="50" rows="10">
    在这里输入内容...
  </textarea>
  <br />
  <input type="submit" value="确定" />
  <input type="reset" value="重置" />
</form>
```

个人简介:

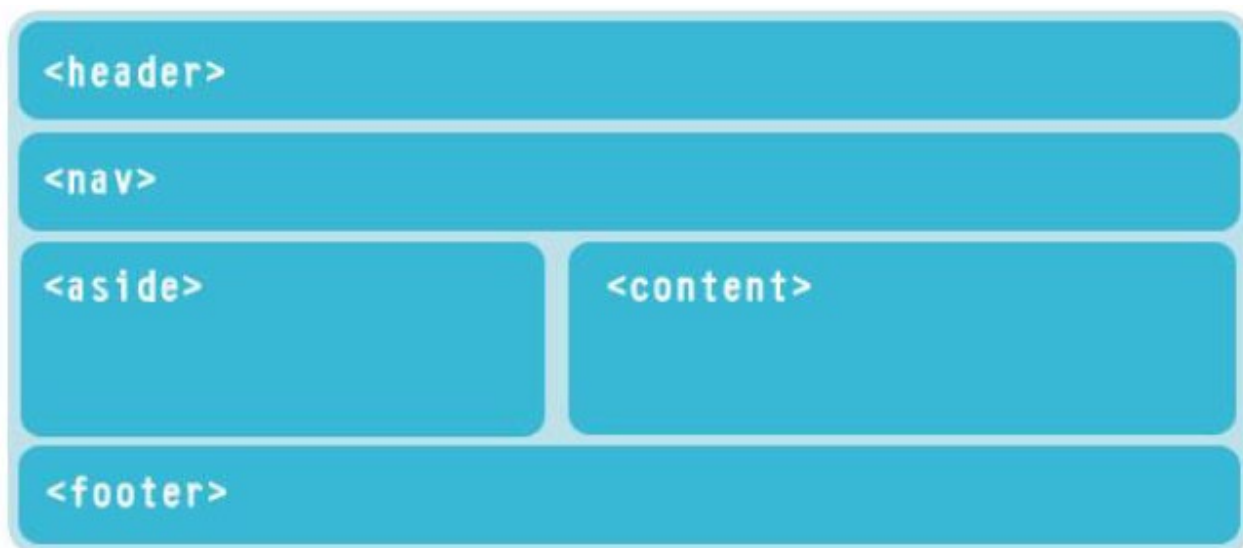
在这里输入内容...


确定

重置

## ■ HTML组织页面内容

- HTML5中， 直接利用对应的标签完成排版布局





```
<header><h1>HTML5结构新标签</h1></header>
```

```
<nav>
```

```
  <ul>
```

```
    <li><a href="#">菜单一 </a></li>
```

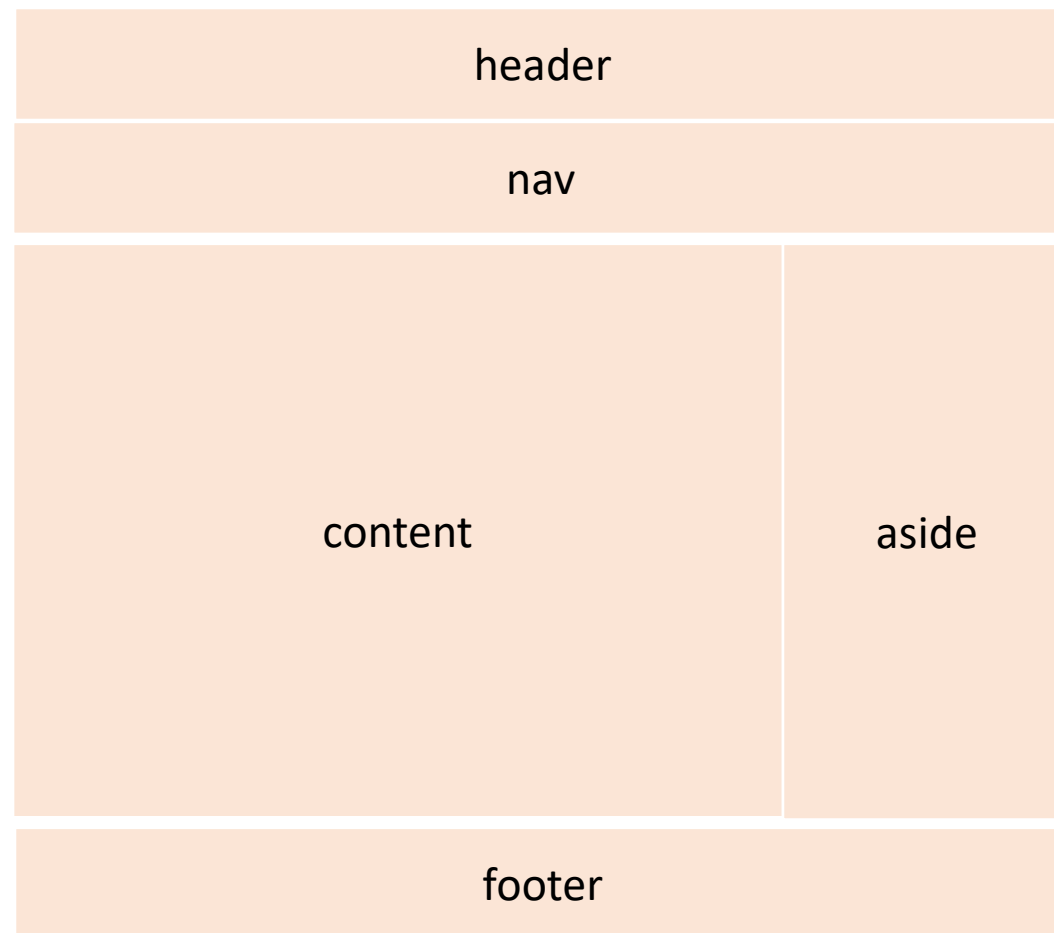
```
    <li><a href="#">菜单二 </a></li>
```

```
    <li><a href="#">菜单三 </a></li>
```

```
    <li><a href="#">菜单四 </a></li>
```

```
  </ul>
```

```
</nav>
```





<content>

<section>

<header>内容标题 </header>

<article>文章内容 </article>

<footer>文章版权内容 </footer>

</section>

<section>

<header>内容标题 </header>

<article>文章内容 </article>

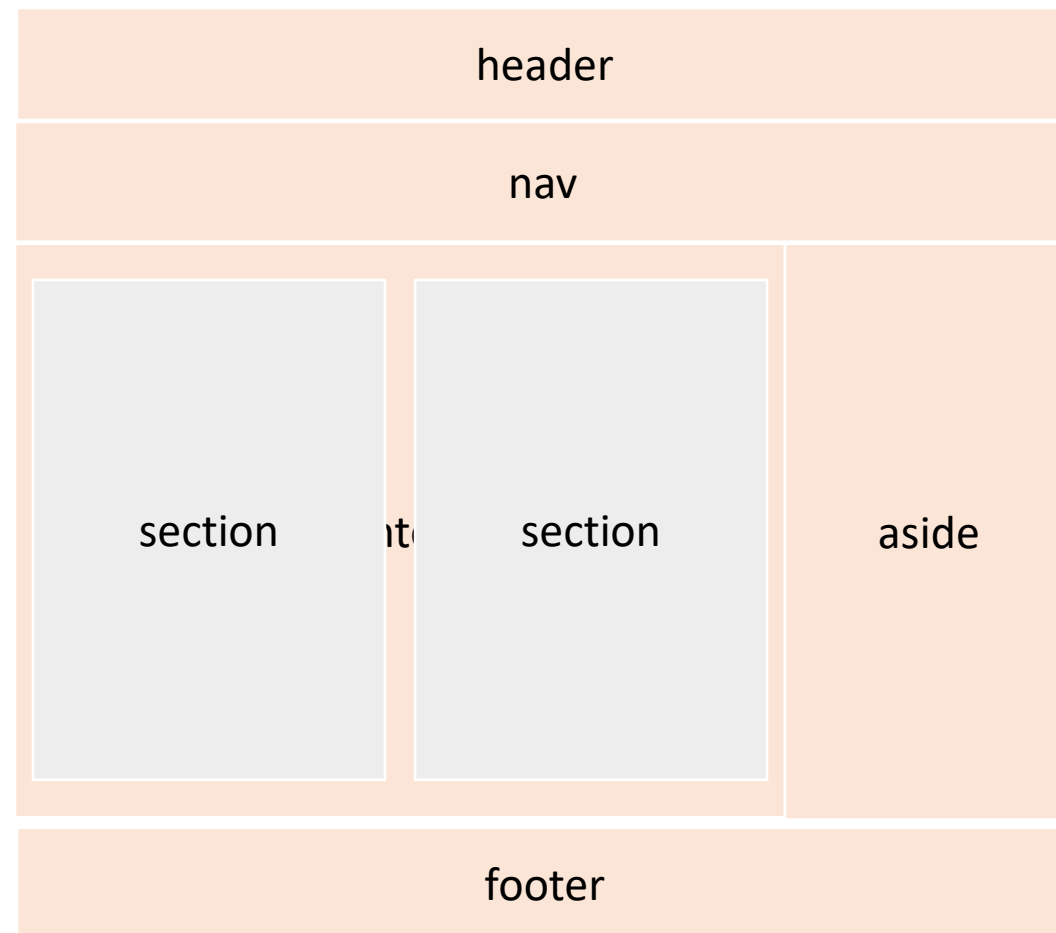
<footer>文章版权内容 </footer>

</section>

</content>

<aside>相关内容 </aside>

<footer>页脚、页面版权内容 </footer>



# ■ HTML语义化

**语义化是什么？--可以不通过了解HTML的内容，就可以知道这个部分所代表的意义。**

- ✓ HTML中的元素、属性及属性值都拥有某些含义
- ✓ 开发者应该遵循语义来编写HTML, 根据网页中内容的结构, 选择适合的HTML标签进行编写
  - 有序列表用ol;
  - 无序列表用ul;
  - lang属性表示内容所使用的语言



让页面具有良好的结构与含义, 从而让人和机器都能快速理解网页内容

## **语义化的好处**

1. 结构清晰, 利于团队的开发、维护
2. 有利于搜索引擎理解
3. SEO (Search Engine Optimization) 搜索引擎优化
4. 容易兼容不同设备

# ■ HTML的可扩展性

- 自定义属性

```
<ul>  
  <li data-id="1">苹果</li>  
  <li data-id="2">葡萄</li>  
  <li data-id="3">橙子</li>  
</ul>
```

# ■ HTML的可扩展性

- meta

```
<!-- 编码 -->
```

```
<meta charset="UTF-8" />
```

```
<!-- 指定HTTP Header-->
```

```
<meta http-equiv="Content-Security-Policy" content="script-src 'self'">
```

```
<!-- SEO搜索引擎优化 -->
```

```
<meta name="description" content="京东JD.COM-专业的综合网上购物商城，满足各种购物需求。" />
```

```
<meta name="Keywords" content="网上购物, 网上商城, 家电, 手机, 电脑, 服装, 居家, 母婴, 美妆, 个护, 食品" />
```

```
<!-- 移动设备viewport -->
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

```
<!-- 360浏览器指定内核 -->
```

```
<meta name="renderer" content="webkit" />
```



# ■ HTML的可扩展性

- link

<!-- 引入CSS样式 -->

```
<link rel="stylesheet" href="index.css">
```

<!-- 指定favicon -->指定页面的图标

```
<link rel="icon" href="favicon.ico">
```

<!-- 提供RSS -->给页面提供一个对应的RSS的访问地址

```
<link rel="alternate" href="rss.xml" type="application/rss+xml">
```

<!-- 预先加载资源 -->性能优化, 可以对一些资源、图片进行预加载

```
<link rel="prefetch" href="images.png">
```

# ■ HTML的可扩展性

- microdata
- 对HTML进行属性级的扩展，属于轻量级的标注方法，用于创建嵌套的名称值对数据组

```
<div>
  <h1>梦的化石</h1>
  
  <dl>
    <dt>作者:</dt>
    <dd>
      <span>[日]</span>
      <span>今敏</span>
    </dd>
    <dt>ISBN:</dt>
    <dd>9787559625984</dd>
  </dl>
  <p>本书是今敏漫画故事短篇全集，收录 ... </p>
</div>
```

# ■ HTML的可扩展性

- microdata

```
<div itemscope itemtype="http://schema.org/Book">
  <h1 itemprop="name">梦的化石</h1>
  
  <dl>
    <dt>作者:</dt>
    <dd itemprop="author" itemscope itemtype="http://schema.org/Person">
      <span itemprop="nationality">[日]</span>
      <span itemprop="name">今敏</span>
    </dd>
    <dt>ISBN:</dt>
    <dd itemprop="isbn">9787559625984</dd>
  </dl>
  <p itemprop="description">本书是今敏漫画故事短篇全集，收录 ... </p>
</div>
```