爱创课堂前端培训

# html

第2天课堂笔记（本课程共2天）

班级：北京前端训练营12期

讲师：冯楠娜

日期：2018年1月5日

爱创课堂官网 ：[www.icketang.com](http://www.icketang.com)

# 目录

[html](#_Toc29949)

[目录](#_Toc12921)

[复习](#_Toc32281)

[一、 标签](#_Toc3567)

[1.1 a标签](#_Toc18808)

[1.2 div和span](#_Toc19502)

[1.3 div+css](#_Toc205)

[二、 列表](#_Toc22172)

[2.1 无序列表](#_Toc15453)

[2.2 有序列表](#_Toc20219)

[2.3 自定义列表](#_Toc7983)

[三、 表格](#_Toc23388)

[3.1 基础表格](#_Toc4519)

[3.2 单元格的合并](#_Toc6852)

[3.3 三个划分区域的标签](#_Toc30240)

# 复习

Html骨架：

|  |
| --- |
| 1. <html> 2. <head> 3. <title>网页标题</title> 4. </head> 5. <body> 6. 网页主体部分 7. </body> 8. </html> |

DTD:文档类型定义。

Html语法：

标签之间对空格，缩进，换行不敏感，对嵌套敏感。

文本内容空白折叠现象。

标签语法：

标签名必须书写在一对尖括号中。

双标签成对出现。

结束标签必须书写关闭符号/。

标签的分类：

容器级标签：h1

文本级标签：p

标签属性：k = “v”

标签：

h1-h6.

P:段落

Img:

Src:路径。

Alt:图片加载失败提示文本

Width:图片的宽度。

|  |
| --- |
| 1. <img src = “images/1.jpg” alt=”图片加载失败” width=”300” height=”400px” /> |

a :超级链接标签

Href:要跳转的路径。

Target:新窗口打开方式。

“\_blank”

Title:鼠标悬停时的提示文本。

# 标签

## a标签

①Name:属性，可以使用name属性实现页面内锚点的跳转。

跳转亦可使用其他部分的id来实现，这种方法常用。

|  |
| --- |
| 1. <a href="#yanyi">演艺经历</a> 2. <h3><a href="#" name="yanyi">演艺经历</a></h3> |

②跨页面锚点的跳转

更改href属性值，写法：页面名称+#name/id名

页面名称即为其路径。

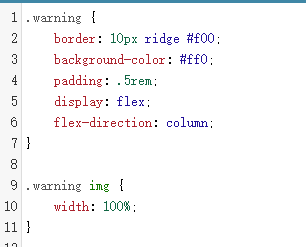
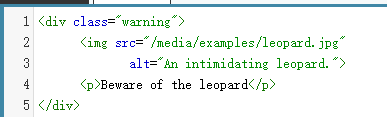
|  |
| --- |
| 1. <a href="05\_跨页面锚点的跳转2.html#yanyi">演艺经历</a> |

## div和span

Div:标签 表示大范围，大区域。独占一行，没有特殊默认样式。

容器级标签：将一些内容相近或者是功能相同的标签包裹在div。

有class属性，与CSS配合使用。



Span:小范围，小跨度，小区域。

文本级标签，并排显示不独占一行。

Div>p>span

## div+css

之前网页布局用表格布局（table），没有进行结构和样式分离。书写繁琐不便于更改样式。已经被舍弃。

现在使用div+css布局。结构和样式进行了分离，书写简单，更改方便。

# 列表

**三类：**无序列表，有序列表，自定义列表

## 2.1 无序列表

作用：给文本添加无序列表的语义。

|  |
| --- |
| 1. <ul> 2. <li>羽绒服</li> 3. <li>保暖</li> 4. <li>毛衣</li> 5. </ul> |



ul:unordered list 无序列表

li:list item 列表项（每一项之间没有先后顺序之分）

Ul和li是我们学习的第一组对标签。必须同时出现。ul>li

Ul标签只能嵌套li标签。

|  |
| --- |
| 1. ~~<ul>~~ 2. ~~<h3>冬季热卖服装</h3>~~ 3. ~~<li>羽绒服</li>~~ 4. ~~<li>保暖</li>~~ 5. ~~<li>毛衣</li>~~ 6. ~~</ul>~~ |

正确书写：

|  |
| --- |
| 1. <h3>冬季热卖服装</h3> 2. <ul> 3. <li>羽绒服</li> 4. <li>保暖</li> 5. <li>毛衣</li> 6. </ul> |

Ul里面可以书写一个li或者多个li.

Ul ,li都是容器级标签。Li可以盛放ul>li。

|  |
| --- |
| 1. <h3>服装分类</h3> 2. <ul> 3. <li> 4. <h4>男装</h4> 5. <ul> 6. <li>夹克</li> 7. <li>西服</li> 8. <li>牛仔裤</li> 9. </ul> 10. </li> 11. <li> 12. <h4>女装</h4> 13. <ul> 14. <li>裙子</li> 15. <li>羽绒服</li> 16. </ul> 17. </li> 18. <li>儿童装</li> 19. </ul> |

ul前面的小圆点不是ul控制的，是样式控制。

## 2.2 有序列表

作用：给文本添加有序列表语义，每一项之间有顺序之分。

ol>li

Ol:ordered list 有序列表

Li:list item 列表项

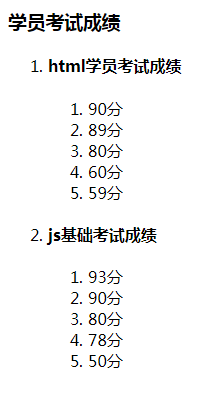
|  |
| --- |
| 1. <ol> 2. <li>html考试</li> 3. <li>js基础考试</li> 4. <li>js进阶考试</li> 5. <li>nodeJs考试</li> 6. </ol> |

Ol,li也是一组对标签，必须同时出现。

Ol里面只能书写li，不能书写任何其他标签。

Ol,li都是容器级标签，li可以书写ol>li(ul>li)

|  |
| --- |
| 1. <ol> 2. <li> 3. <h4>html学员考试成绩</h4> 4. <ol> 5. <li>90分</li> 6. <li>89分</li> 7. <li>80分</li> 8. <li>60分</li> 9. <li>59分</li> 10. </ol> 11. </li> 12. <li> 13. <h4>js基础考试成绩</h4> 14. <ol> 15. <li>93分</li> 16. <li>90分</li> 17. <li>80分</li> 18. <li>78分</li> 19. <li>50分</li> 20. </ol> 21. </li> 22. </ol> |



Ol里面可以书写一个或者是多个li标签。

Ol前面的数字也是css样式控制的。

## 2.3 自定义列表

作用：给文本添加自定义列表的语义, 同级用加号连接。

涉及3个标签:dl > (dt + dd)

|  |
| --- |
| 1. <dl> 2. <dt></dt> 3. <dd></dd> 4. </dl> |

Dl: definition list 自定义列表

Dt: definition title 列表的标题

Dd: definition description 描述（是对列表标题的解释说明）

|  |
| --- |
| 1. <dl> 2. <dt>小米</dt> 3. <dd>小米公司是国内的公司，产品种类很多</dd> 4. <dt>华为</dt> 5. <dd>华为公司位于深圳，是比较有企业理念的公司</dd> 6. </dl> |

dl里面可以书写一组或者是多组dt和dd。（dl里面只能书写dt, dd）

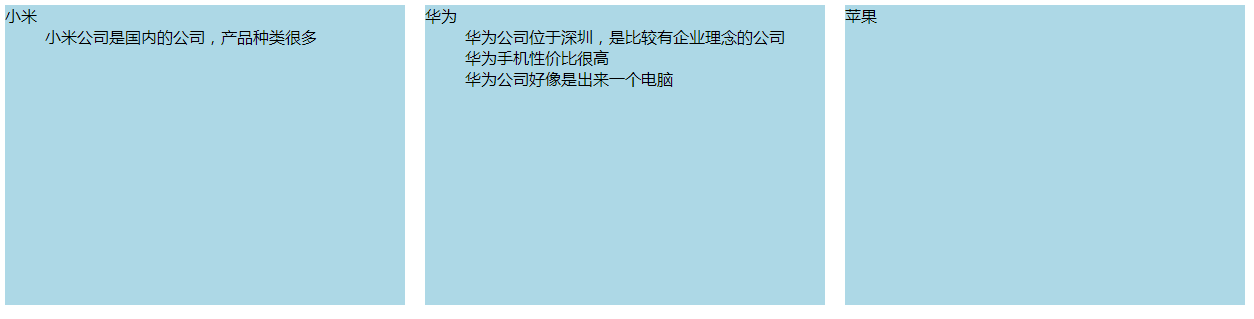
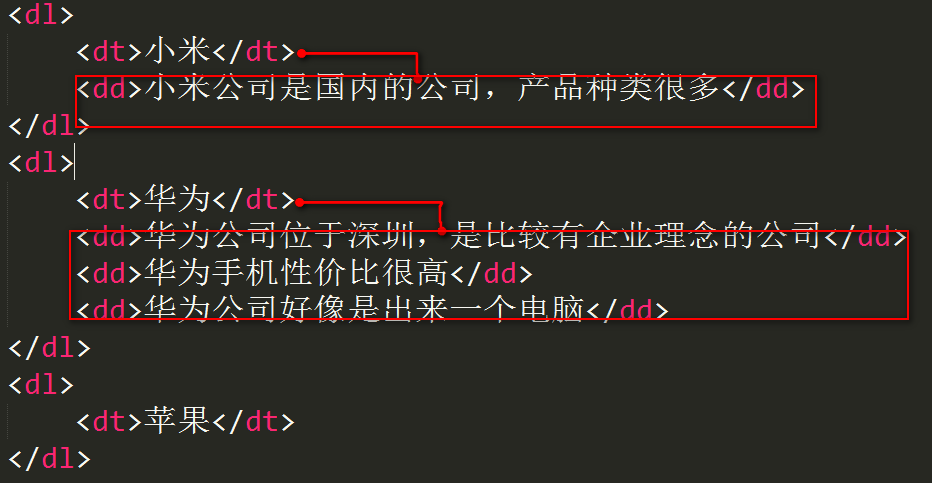
Dd是对dt的解释说明。

Dd可以有一条或者是多条，甚至可以没有dd，表示没有解释说明。

|  |
| --- |
| 1. <dl> 2. <dt>小米</dt> 3. <dd>小米公司是国内的公司，产品种类很多</dd> 4. <dt>华为</dt> 5. <dd>华为公司位于深圳，是比较有企业理念的公司</dd> 6. <dd>华为手机性价比很高</dd> 7. <dd>华为公司好像是出来一个电脑</dd> 8. <dt>苹果</dt> 9. </dl> |

dt和dd是容器级标签。可以放任何内容。

实际工作一般是一个dl嵌套一组dt和dd。



实际工作中，具体使用哪一种列表，根据语义。

# 表格

## 3.1 基础表格

最简单的表格要求每一行单元格数量相同。

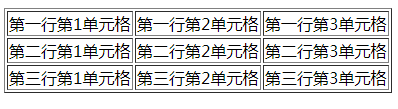
Table > tr > td

Table:表格

Tr: table row 行

Td: table dock 单元格

|  |
| --- |
| 1. <table border="1"> 2. <tr> 3. <td>第一行第1单元格</td> 4. <td>第一行第2单元格</td> 5. <td>第一行第3单元格</td> 6. </tr> 7. <tr> 8. <td>第二行第1单元格</td> 9. <td>第二行第2单元格</td> 10. <td>第二行第3单元格</td> 11. </tr> 12. <tr> 13. <td>第三行第1单元格</td> 14. <td>第三行第2单元格</td> 15. <td>第三行第3单元格</td> 16. </tr> 17. </table> |



如果表格有表头的概念，只需要将td改为th (table head)， th有默认加粗。

|  |
| --- |
| 1. <table border="1"> 2. <tr> 3. <th>姓名</th> 4. <th>性别</th> 5. <th>年龄</th> 6. </tr> 7. <tr> 8. <td>小红</td> 9. <td>女</td> 10. <td>18岁</td> 11. </tr> 12. <tr> 13. <td>小明</td> 14. <td>男</td> 15. <td>20岁</td> 16. </tr> 17. <tr> 18. <td>小刚</td> 19. <td>男</td> 20. <td>22岁</td> 21. </tr> 22. </table> |



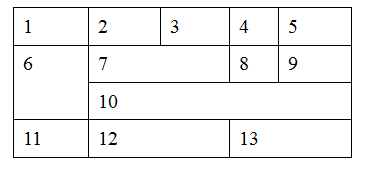
## 3.2 单元格的合并

Td (或th)有两个属性可以设置单元格的合并。

colspan: 跨列合并， 几列合为一列。

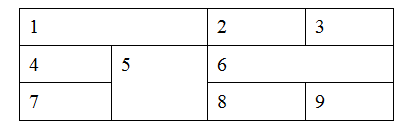
rowspan :跨行合并。

属性值是数字，跨几行或者跨几列就书写几。没有单位



|  |
| --- |
| 1. <table border="1"> 2. <tr> 3. <td>1</td> 4. <td>2</td> 5. <td>3</td> 6. <td>4</td> 7. <td>5</td> 8. </tr> 9. <tr> 10. <td rowspan="2">6</td> 11. <td colspan="2">7</td> 12. <td>8</td> 13. <td>9</td> 14. </tr> 15. <tr> 16. <td colspan="4">10</td> 17. </tr> 18. <tr> 19. <td>11</td> 20. <td colspan="2">12</td> 21. <td colspan="2">13</td> 22. </tr> 23. </table> |

练习：



|  |
| --- |
| 1. <table border="1"> 2. <tr> 3. <td colspan="2">1</td> 4. <td>2</td> 5. <td>3</td> 6. </tr> 7. <tr> 8. <td>4</td> 9. <td rowspan="2">5</td> 10. <td colspan="2">6</td> 11. </tr> 12. <tr> 13. <td>7</td> 14. <td>8</td> 15. <td>9</td> 16. </tr> 17. </table> |

## 3.3 三个划分区域的标签

Table标签内部可以嵌套三个划分区域的标签。

Caption:表格标题

Thead:表头（thead>tr>th）

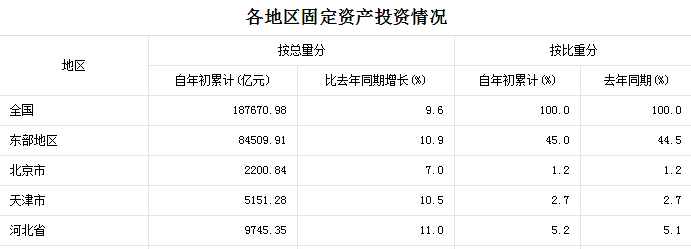
Tbody:表格主体(tbody>tr>td)

Table > caption +（thead>tr>th）+ (tbody > tr >td)

|  |
| --- |
| 1. <table border="1"> 2. <!-- 表格标题 --> 3. <caption>学员信息统计表</caption> 4. <!-- 表头 --> 5. <thead> 6. <tr> 7. <th>姓名</th> 8. <th>年龄</th> 9. <th>性别</th> 10. </tr> 11. </thead> 12. <!-- 表格主体 --> 13. <tbody> 14. <tr> 15. <td>小红</td> 16. <td>18岁</td> 17. <td>女</td> 18. </tr> 19. <tr> 20. <td>小红</td> 21. <td>18岁</td> 22. <td>女</td> 23. </tr> 24. <tr> 25. <td>小红</td> 26. <td>18岁</td> 27. <td>女</td> 28. </tr> 29. <tr> 30. <td>小红</td> 31. <td>18岁</td> 32. <td>女</td> 33. </tr> 34. </tbody> 35. </table> |



练习：



|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |