

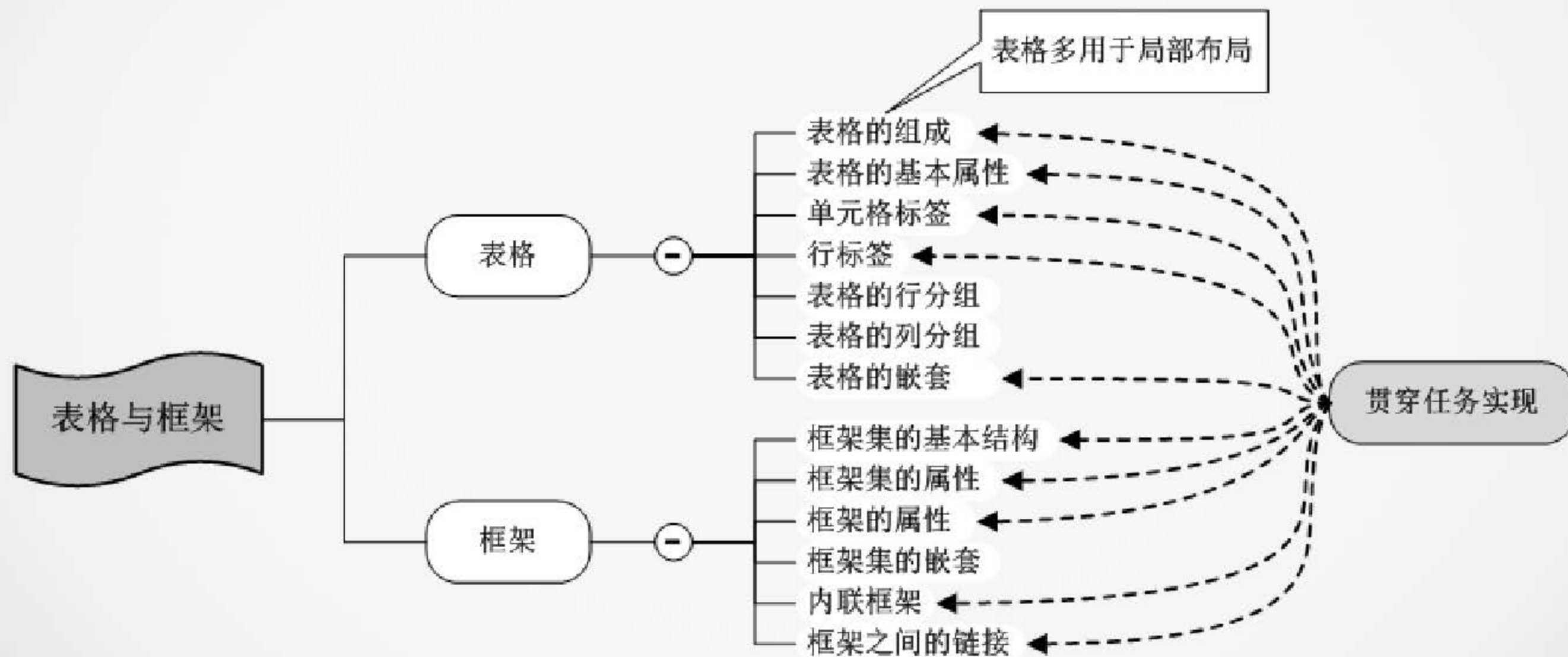
第二章 表格与框架

— Web前端设计与开发 —

- 本章任务是完成“漫步时尚广场”的购物模块首页和后台管理界面：

【任务2-1】 实现Q-Walking E&S购物模块首页的设计。

【任务2-2】 实现Q-Walking E&S后台管理页面。



| 知识点 | 听 | 懂 | 做 | 复习 | 精通 |
|-----------|---|---|---|----|----|
| 表格的基本元素 | ★ | ★ | ★ | ★ | |
| 表格及子元素的属性 | ★ | ★ | ★ | ★ | ★ |
| 表格的分组 | ★ | ★ | ★ | | |
| 框架集的基本结构 | ★ | ★ | ★ | | |
| 框架集的嵌套 | ★ | ★ | | | |
| 内联框架 | ★ | ★ | ★ | ★ | |
| 框架之间的链接 | ★ | ★ | ★ | | |

- 表格可以将数据有效的组织在一起，并以网格的形式进行显示
- 表格除了用于组织数据外，还可以实现页面或局部页面的排版布局
- 表格使用行与列的方式组织信息
- 表格元素主要由行、列、表头单元格、正文单元格、标题、表头行、正文行、表尾行等构成。
- 在HTML中，通过<table>标签创建表格元素

2.1.1 表格的组成

```
<table>
  <!-- 一行可以包含多个单元格 -->
  <tr>
    <td>单元格内容</td>
    <td>单元格内容</td>
    <!--更多单元格-->
    .....
  </tr>
  <!--更多行-->
  .....
</table>
```

● 其中：

- 表格的各组成部分均包含在<table>标签之内
- 单元格是表格的基本元素，使用<td>标签表示
- 行是表格的水平元素，使用<tr>标签表示
- 一行可以由一个或多个单元格构成，而一个表格可以由一行或多行构成

2.1.1 表格的组成

```
<body>
  <table>
    <tr>
      <td>姓名</td>
      <td>部门</td>
      <td>邮箱</td>
      <td>电话</td>
    </tr>
    <tr>
      <td>张叁</td>
      <td>市场部</td>
      <td>zhangs@itshixu.com</td>
      <td>18966882233</td>
    </tr>
    <tr>
      <td>李斯</td>
      <td>研发部</td>
      <td>lis@itshixu.com</td>
      <td>13688995566</td>
    </tr>
  </table>
</body>
```

一个三行四列的表格，第一行是表格的头部信息，后面两行是员工的基本信息；默认情况下，表格的边框是隐藏的



| 姓名 | 部门 | 邮箱 | 电话 |
|----|-----|--------------------|-------------|
| 张叁 | 市场部 | zhangs@itshixu.com | 18966882233 |
| 李斯 | 研发部 | lis@itshixu.com | 13688995566 |

2.1.2 表格的基本属性

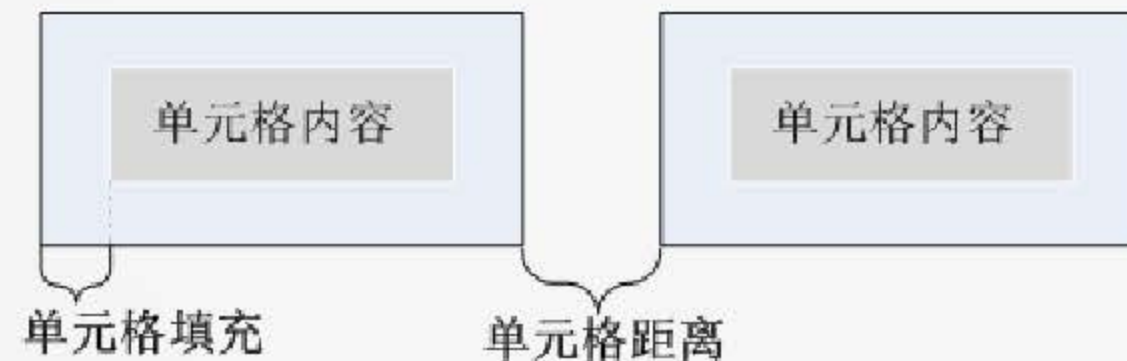
- 表格的常用属性有对齐方式、背景颜色、边框、高度、宽度等。

| 属性 | 取值 | 描述 |
|-------------|---|----------------------|
| align | left、center、right | 设置表格相对周围元素的对齐方式 |
| bgcolor | rgb(x,x,x)、#xxxxxx、colorName | 设置表格的背景颜色 |
| border | 像素 | 设置表格边框的宽度 |
| cellpadding | 像素或百分比 | 设置单元格与其内容之间的距离 |
| cellspacing | 像素或百分比 | 设置单元格之间的距离 |
| height | 像素或百分比 | 设置表格的高度 |
| width | 像素或百分比 | 设置表格的宽度 |
| rules | none、groups、rows、cols、all | 设置表格中的表格线显示方式，默认是all |
| frame | void、above、below、hsides、vsides、lhs、rhs、box、border | 设置表格的外部边框的显示方式 |

2.1.2 表格的基本属性

- 属性cellpadding表示单元格边界与单元格内容之间的距离
- cellspacing表示单元格与单元格之间的距离

| 属性 | 描述 |
|------------|-----------------------------|
| void | 不显示边框 |
| above | 只显示顶部边框 |
| below | 只显示底部边框 |
| hsides | 显示顶部和底部边框 |
| vsides | 只显示左右两侧边框 |
| lhs | 只显示左侧边框 |
| rhs | 只显示右侧边框 |
| box或border | 显示表格的所有边框（不指定frame属性时的默认边框） |



2.1.2 表格的基本属性

```
<table border="2" cellpadding="5" cellspacing="5" bgcolor="rgb(ff,ff,66)"  
      align="right" width="400" height="100">
```

```
<tr>  
  <td>姓名</td>  
  <td>部门</td>  
  <td>邮箱</td>  
  <td>电话</td>  
</tr>
```

<!--此处代码省略-->

</table>

<table>标签的align属性使表格在窗口中靠右对齐，且表格后面的文本采用跟随环绕的方式显示

移动通信录是一种利用互联网或移动互联网实现通信录信息同步更新和备份的应用/服务。你可以在个人电脑、掌上电脑、移动电话等任何联网设备上录访问员工的姓名、部门、有限、电话等通.....



2.1.3 单元格标签

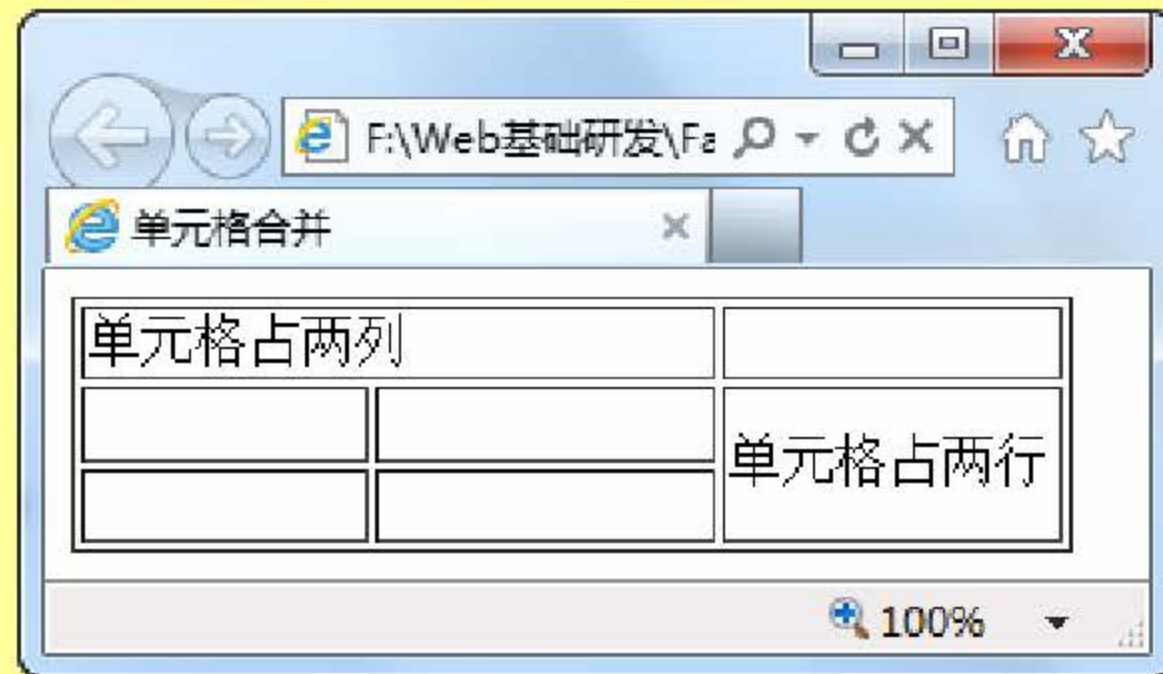
- 单元格是表的基本元素，包括<td>和<th>标签
- <td>标签多用来包含表格中的数据部分，而<th>标签用来包含表格的标题部分
- 单元格常用的属性有水平对齐、垂直对齐、水平跨度、垂直跨度、宽度、高度、背景颜色等

| 属性 | 描述 |
|---------|---|
| align | 设置单元格内容的水平对齐方式：left、center、right、justify |
| valign | 设置单元格内容的垂直对齐方式：top、middle、bottom、baseline |
| rowspan | 设置单元格跨越的行数 |
| colspan | 设置单元格跨越的列数 |
| scope | 定义将表头数据与单元数据相关联的方法 |
| width | 设置单元格的宽度 |
| height | 设置单元格的高度 |
| bgcolor | 设置单元格的背景颜色 |

水平跨度colspan是指表格内的某个单元格在水平方向上跨越单元格的列数；
垂直跨度rowspan是单元格在垂直方向上所跨的行数。

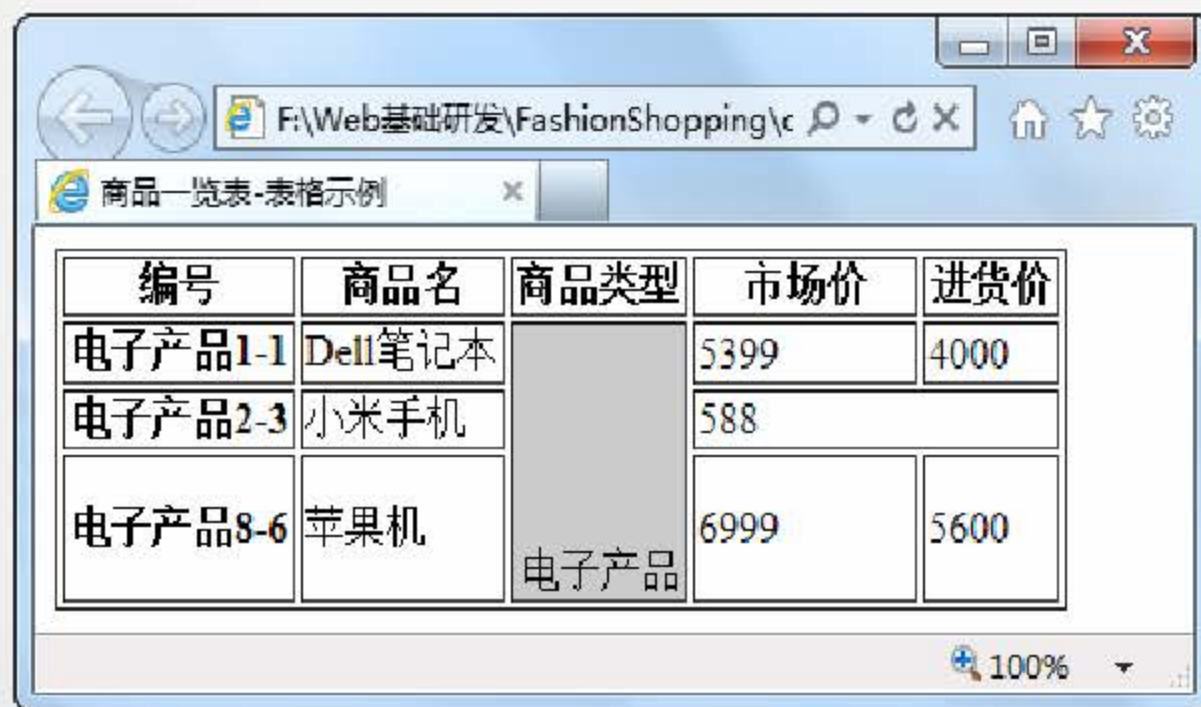
2.1.3 单元格标签

```
<body>
<table width="300" border="1">
  <tr>
    <td colspan="2">单元格占两列</td>
    <!-- <td>&nbsp;</td>--> <!--此空间因前面单元格的colspan而被占-->
    <td width="90">&nbsp;</td>
  </tr>
  <tr>
    <td width="75">&nbsp;</td>
    <td width="90">&nbsp;</td>
    <td rowspan="2">单元格占两行</td>
  </tr>
  <tr>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
    <!-- <td>&nbsp;</td>--> <!--此空间因前面单元格的rowspan而被占-->
  </tr>
</table>
</body>
```



2.1.3 单元格标签

- 标签<th>用来定义表格头部信息
 - 多用于表格的第一行或第一列
 - 其内容通常使用粗体并水平居中显示



| 编号 | 商品名 | 商品类型 | 市场价 | 进货价 |
|---------|---------|------|------|------|
| 电子产品1-1 | Dell笔记本 | 电子产品 | 5399 | 4000 |
| 电子产品2-3 | 小米手机 | | 588 | |
| 电子产品8-6 | 苹果机 | | 6999 | 5600 |

```
<table width="400" border="1">
  <tr>
    <th>编号</th>
    <th>商品名</th>
    <th>商品类型</th>
    <th>市场价</th>
    <th>进货价</th>
  </tr>
  <tr>
    <th>电子产品1-1</th>
    .....
  </tr>
  <tr>
    <th>电子产品2-3</th>
    .....
  </tr>
  <tr>
    <th>电子产品8-6</th>
    .....
  </tr>
</table>
```




大部分浏览器都会忽略空白单元格（即<td> </td>之间没有内容）；
当空白单元格格式时，需要在单元格标签内加入一个空白实体引用** **，以确保浏览器能正确显示该单元格。

- 行是表格的水平元素，一行可以包含一个或多个单元格
- 在HTML中，使用<tr>标签进行界定，<td>与<th>标签位于<tr></tr>标签之间
- 表格除了能对单元格进行设置外，还可以对行进行设置，常用的属性有水平对齐、背景颜色、垂直对齐、边框颜色等

| 属性 | 描述 |
|------------------|--|
| align | 设置单元格内容水平对齐方式：left、center、right、justify |
| valign | 设置单元格内容垂直对齐方式：top、middle、bottom、baseline |
| bgcolor | 设置单元格的背景颜色 |
| bordercolor | 设置行内单元格的边框颜色 |
| bordercolordark | 设置行内单元格的左上边框颜色 |
| bordercolorlight | 设置行内单元格的右下边框颜色 |

```
<body>
<table width="400" border="2">
  <tr align="right" bgcolor="#FFFF99" bordercolor="#000099"
      bordercolordark="#003399" bordercolorlight="#FF0000" >
    <td>影片介绍</td>
    <td>影片图片</td>
    <td>上映时间</td>
  </tr>
</table>
</body>
```


- 除了表格主体（行与列）外，表格还提供了标题、表头和表尾部分，使得表格的内容更加丰富，数据的组织更加清晰。
- 使用<thead>、<tfoot>、<tbody>、<caption>标签可以对表格进行横向分组：
 - <thead>标签定义表头，用于创建表格的头部信息
 - <tfoot>标签定义表尾，用于创建表格的脚注部分
 - <tbody>标签定义表格主体，用于表示表格的主体部分
 - <caption>标签定义表格标题，显示在整个表格的上方
- 表格可以包含多个<tbody>标签，用于对表格主体部分的数据进行横向分组；而<thead>和<tfoot>标签在表格中只能出现一次。

2.1.5 表格的行分组

```
<table width="400" border="1" rules="groups">
  <caption>
    企业员工薪水绩效表
  </caption>
  <thead>
    <tr>
      <th>员工编号</th><th>员工岗位</th>
      <th>基本工资</th><th>本月绩效</th>
    </tr>
  </thead>
  <tbody>
    .....
  </tbody>
  <tbody>
    .....
  </tbody>
  <tfoot>
    .....
  </tfoot>
</table>
```

| 员工编号 | 员工岗位 | 基本工资 | 本月绩效 |
|--------|-----------|------|------|
| YF0016 | Java高级工程师 | 6000 | 3000 |
| YF0021 | Java程序员 | 3000 | 2500 |
| YF0062 | Web前端设计师 | 5500 | 2000 |
| YF0081 | 测试成员 | 3000 | 2000 |
| 总计 | | 10W | 10W |

2.1.6 表格的列分组


- 表格的纵向分组又称列分组
- 在HTML中提供了<colgroup>标签，该标签可以将表格按列进行分组

| 属性 | 描述 |
|--------|--|
| align | 设置单元格内容水平对齐方式：left、center、right、justify |
| valign | 设置单元格内容垂直对齐方式：top、middle、bottom、baseline |
| span | 规定该列组应该横跨的列数，默认值是1 |
| width | 设定列组合的宽度 |

2.1.6 表格的列分组

```
<table width="400" border="1">
  <colgroup span="2" align="left" valign="bottom" />
  <colgroup style="background-color:#CCC" />
<tr height="60">
  <th>员工编号</th>
  <th>员工岗位</th>
  <th>基本工资</th>
  <th>本月绩效</th>
</tr>
<tr>
  <td>YF0016</td>
  <td>Java高级工程师</td>
  <td>6000</td>
  <td>3000</td>
</tr>
<tr>
  <td>YF0016</td>
  <td>Java高级工程师</td>
  <td>6000</td>
  <td>3000</td>
</tr>
</table>
```

第一组占据两列，使用默认样式；
第二组由于没有span属性，默认值占一列，
通过style属性将背景样式设为灰色



| 员工编号 | 员工岗位 | 基本工资 | 本月绩效 |
|--------|-----------|------|------|
| YF0016 | Java高级工程师 | 6000 | 3000 |
| YF0016 | Java高级工程师 | 6000 | 3000 |



浏览器对align与valign属性支持不够好，建议通过CSS样式来实现。
<colgroup> 标签由于没有内容部分，写成单标签或双标签的形式均可。

- 使用表格嵌套布局时，页面排版更加灵活，可以轻松设计出更加复杂而精美的效果。

```
<table width="100" border="1">
  <tr>
    <td>&nbsp;</td>
    <td>
      <table width="100" border="1">
        <tr>
          <td>&nbsp;</td>
        </tr>
      </table>
    </td>
    <td>&nbsp;</td>
    .....
  </tr>
</table>
```




在嵌套表格时，内部表格<table>应该位于外层表格的<td> </td>标签之间。

表格虽然允许多重嵌套，但在页面设计时，当嵌套层次太多时不利于搜索引擎对页面内容的检索；因此，表格嵌套的层次不能过深，**一般不要超过3~4层。**

- 框架（frame）能够将浏览器窗口划分为多个独立的窗格，每个窗格包含一个独立的HTML页面
- 用户可以通过框架加载或重新加载单个窗格内容，而不需要重新加载浏览器窗口的所有内容。相对框架而言，整个浏览器窗口对应的框架集合称为框架集（frameset）
- 在HTML中，使用框架集标签<frameset>来划分页面的框架，使用属性rows（或cols）说明框架的行数（或列数）以及所占窗口的比例

```
<frameset rows="行高度所占窗口的像素或比例,..."  
           cols="列宽度所占窗口的像素或比例,..." >  
    <frame src="..." />  
    .....  
</frameset>
```


2.2.2 框架集的属性

- 框架集<frameset>主要负责整个页面的布局，属性包括行、列、边框、边框颜色、空白距离等属性

| 属性 | 描述 |
|--------------|---------------------------------------|
| rows | 设置框架集中包含框架的行数，以及对应的高度 |
| cols | 设置框架集中包含框架的列数，以及对应的宽度 |
| frameborder | 设置框架集的边框是否显示，取值为1、0或yes、no，边框本身不能调整宽度 |
| bordercolor | 设置框架集的边框的颜色 |
| framespacing | 框架与框架间的空白距离 |

2.2.1 框架集的基本结构

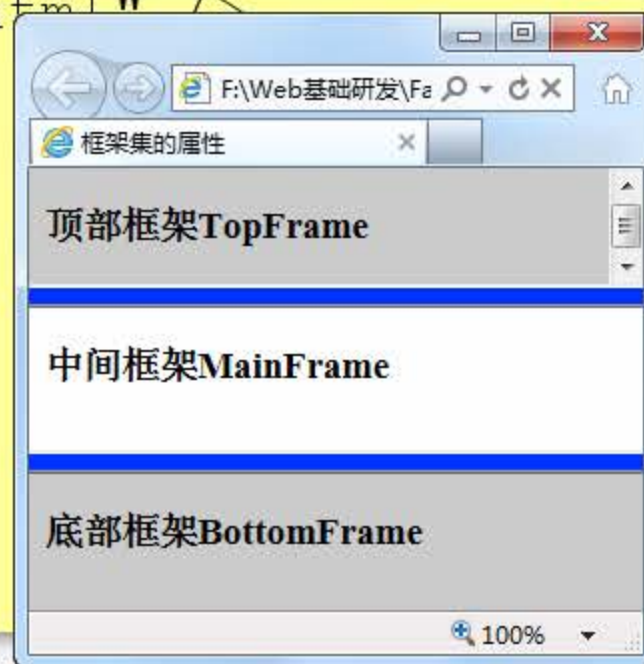
```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>框架集的基本结构</title>
</head>
  <frameset rows="60,*,100">
    <frame src="引入页面的URL" />
    <frame src="引入页面的URL" />
    <frame src="引入页面的URL" />
  <noframes>
    <body>
      该浏览器不支持框架集!
    </body>
  </noframes>
</frameset>
</html>
```

框架集**<frameset>**替代了
<body>部分

- rows和cols属性的四种取值形式:
 - 以像素为单位的绝对值
 - 浏览器窗口的百分比
 - 行（或列）之间的相对宽度
 - 混合度量尺寸

2.2.2 框架集的属性

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>框架集的属性</title>
</head>
  <frameset rows="60,* ,70" frameborder="yes" framespacing="8"
            bordercolor="#0033FF">
    <frame src="topFrame.html" />
    <frame src="mainFrame.html" />
    <frame src="bottomFrame.html" />
  </frameset>
  <noframes>
    <body>
      该浏览器不支持框架集!
    </body>
  </noframes>
</html>
```



在IE、Firefox等浏览器显示的样式略有不同

2.2.3 框架的属性

- `<frame>` 标签用于指示框架集中每个框架的内容。

```
<frame src="url" name=" " ...> </frame>
```

- 其中：

- `src` 和 `name` 是 `<frame />` 标签的两个常用属性
- 属性 `src` 用于指向一个页面资源的 URL 路径，可以是绝对路径，也可以是相对路径
- 属性 `name` 为框架指定一个名称
- `<frame />` 标签可以使用单标签形式，也可以使用双标签形式

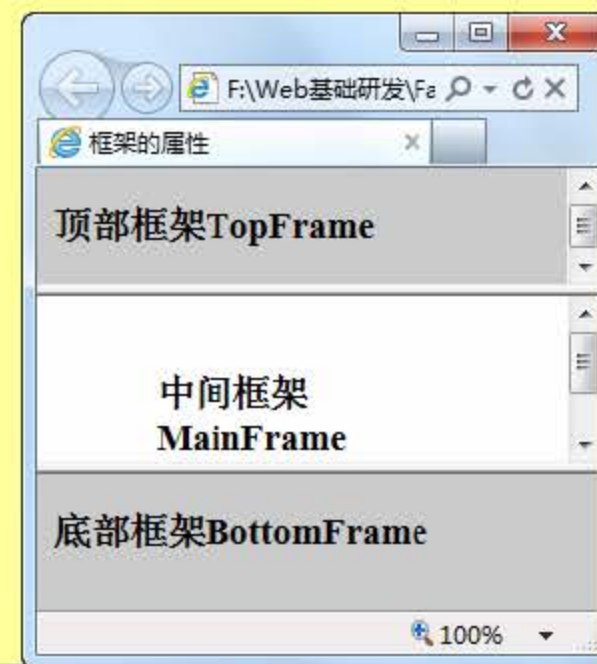
```
<frame src="topFrame.html" name="topFrame" />  
<frame src="mainFrame.html" name="mainFrame" ></frame>
```

2.2.3 框架的属性

| 属性 | 描述 |
|--------------|-----------------------------|
| name | 设置框架的名称，在设置超链接时用作框架的标记 |
| src | 设置在框架中显示页面的URL |
| frameborder | 设置框架的边框是否显示，取值为0或1 |
| marginheight | 定义内容与框架的上下边缘高度，默认为1 |
| marginwidth | 定义内容与框架的左右边缘宽度，默认为1 |
| scrolling | 设置框架中是否显示滚动条，取值为yes、no和auto |
| noresize | 设置框架不能调整大小 |

2.2.3 框架的属性

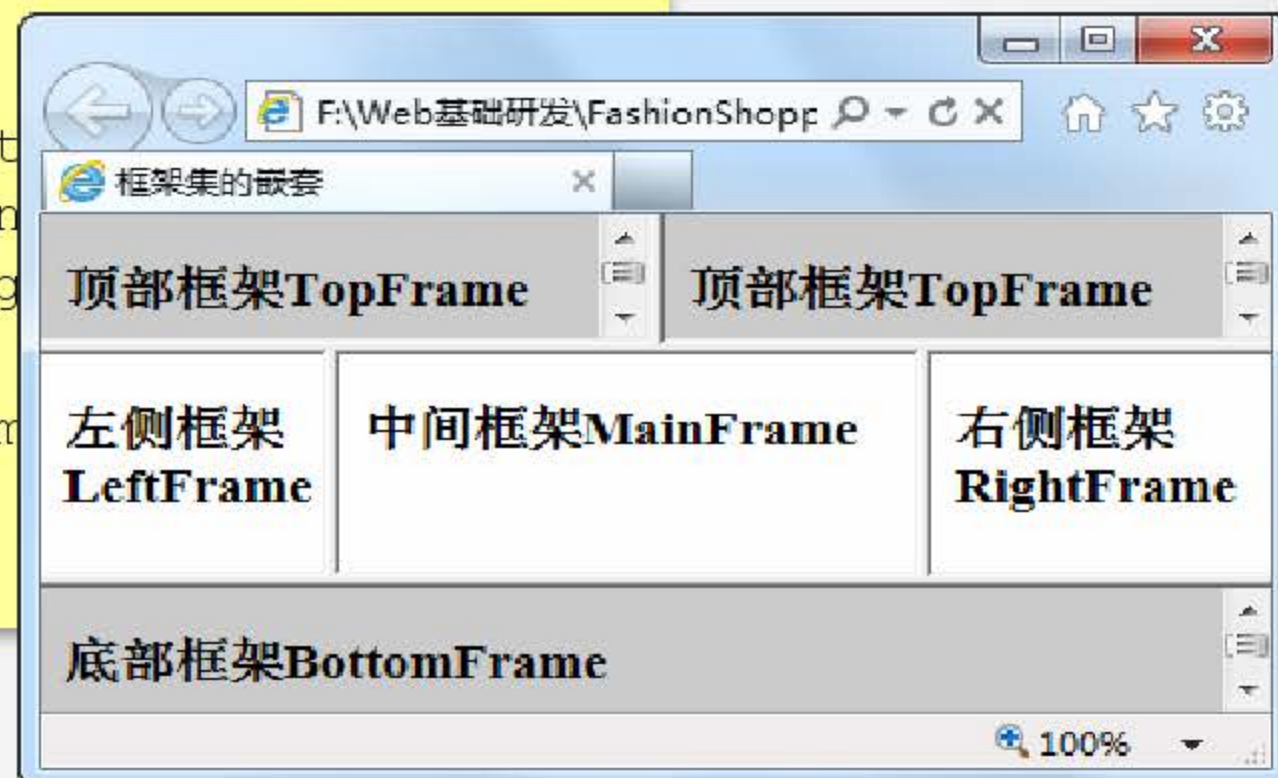
```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>框架的属性</title>
</head>
<frameset rows="60,*,70">
  <frame src="topFrame.html" scrolling="yes" frameborder="yes"
    noresize="noresize" marginheight="40" marginwidth="60" />
  <frame src="mainFrame.html" />
  <frame src="bottomFrame.html" />
</frameset>
<noframes>
  <div>
    顶部框架不能通过鼠标拖拽来调整大小，且右侧一直显示滚动条。
    而中间框架会随窗口大小的改变，根据框架内容的多少决定是否
    出现滚动条。
  </div>
</noframes>
</html>
```



2.2.4 框架集的嵌套

- 单个框架集只能实现具有行和列（类似网格）的固定结构的布局，当希望更加复杂的结构时，就需要通过嵌套框架集来实现

```
<frameset rows="50,*,50" >
  <frameset cols="1*,1*">
    <frame name="topFrame1" src="topFrame.html" />
    <frame name="topFrame2" src="topFrame.html" />
  </frameset>
  <frameset cols="100,*,120">
    <frame name="leftFrame" src="leftFrame.html" />
    <frame name="mainFrame" src="mainFrame.html" />
    <frame name="rightFrame" src="rightFrame.html" />
  </frameset>
  <frame name="bottomFrame" src="bottomFrame.html" />
</frameset>
```



- 内联框架（又称行内框架），可以出现在页面的任何位置，比框架集更加灵活。
- 内联框架是嵌入到页面中的一个区域，通过<iframe>标签引入另外一个页面资源，无需<frameset>标签协助。

```
<iframe src="url" name=" " width=" " height=" " ...> </iframe>
```

2.2.5 内联框架

| 属性 | 描述 |
|--------------|---|
| align | 设置iframe与周围文本对齐方式，取值可以是left、right、top、middle、bottom |
| frameborder | 设置iframe的边框是否显示，取值0或1 |
| marginheight | 定义iframe的顶部和底部的边距 |
| marginwidth | 定义iframe的左侧和右侧的边距 |
| height | 设置iframe的高度 |
| width | 设置iframe的宽度 |
| scrolling | 设置iframe中是否显示滚动条，取值yes、no、auto |
| src | 设置iframe中显示页面的URL |
| name | 设置iframe的名称 |

2.2.5 内联框架

<body>

iframe一般用来包含别的页面，例如我们可以在我们自己的网站页面加载别人网站的内容，为了更好的效果，可能需要使iframe透明效果，那么就需要了解更多的iframe属性。不带边框的iframe因为能和网页无缝的结合从而不刷新页面的情况下更新页面的部分数据成为可能，可是iframe的大小却不像层那样可以“伸缩自如”，所以带来了使用上的麻烦，给iframe设置高度的时候多了也不好，少了更是不行，现在，让我来告诉大家一种iframe动态调整高度的方法……

```
<iframe name=
frameborder
```

```
</iframe>
```

不带边框的iframe是iframe的大小却不佳也不好，少了更是不行</body>



```
ag_iframe.asp"
align="right">
```

分数据成为可能，可设置高度的时候多了

2.2.6 框架之间的链接

- 使用框架布局页面时，常常在框架中放置一个导航栏；当点击导航栏中的链接时，在指定的框架中加载目标页面。
- 每个<frame>标签都有一个name属性为框架分别指定一个名称。在导航栏中点击链接时，会根据target属性所指定的框架中加载目标页面内容。

```
<a href="goods.htm">商品列表</a><br/>
<a href="contacts.htm">通讯录</a><br/>
<a href="http://www.qsrq.com">青软实训</a><br/>
<a href="http://www.qsrq.com">锐聘</a><br/>
<a href="http://www.qsrq.com">码客学院</a>
```

| 编号 | 商品名 | 商品类型 | 市场价 | 进货价 |
|---------|---------|------|------|------|
| 电子产品1-1 | Dell笔记本 | 电子产品 | 5399 | 4000 |
| 电子产品2-3 | 小米手机 | | 588 | |
| 电子产品8-6 | 苹果机 | 电子产品 | 6999 | 5600 |

- 表格元素主要由行、列、单元格、标题、表头行、正文行、表尾行等构成
- 表格的基本元素是单元格，使用标签<td>表示
- 行是表格的水平元素，使用标签<tr>表示
- 表格通过<thead>、<tfoot>、<tbody>、<caption>标签对表格进行横向分组
- 在HTML中提供了<colgroup>标签，该标签可以对表格按列分组
- 整个浏览器窗口对应的框架集合称为框架集（frameset）
- <frame>标签用于指示框架集中每个框架的内容
- 内联框架是嵌入到页面中的一个区域，通过<iframe>标签引入另外一个页面资源
- 在创建超链接时，通过target属性，指明新的页面应该在哪个框架中进行加载

有前途更有“钱”途 全新课程体系

Web·大前端



— Web前端设计与开发 —