



# 6 CSS+DIV布局3

---

浙江师范大学

主讲老师：徐晓丹（663601）



# 主要内容

- 字体单位
- 块级元素和行内元素
- CSS实例应用--电子相册



# 字体单位

- 字体的大小单位
- px, px是pixel缩写, 是基于像素的单位（屏幕上显示数据的最基本的点）。用像素定义的文字、图片等会随屏幕的分辨率变化而变化。如果在定义字体大小时, 使用px作为单位, 分辨率调高, 实际看到的文字就变“小”。



# 字体单位

- em, 相对单位, 是一个相对长度单位。现指的是字符宽度的倍数, 如: 0.8em, 1.2em, 2em等。通常1em=16px
- 注: 当用于指定字体大小时, em单位是指父元素的字体大小



## em和px

- 浏览器的默认字体高都是16px，所以未经调整的浏览器在显示1em=16px，也就是说1px=0.0625em。
- 为了简化font-size的换算，可以在css中的body中先全局声明font-size=62.5%，也就是定义了默认字体大小为16px\*0.625=10px,子元素会继承父级元素的字体大小，于是1em=10px，所以12px=1.2em。
- 。



- rem, CSS3新增的一个相对长度单位, 只相对根目录即HTML元素。所以可以在html标签上设置字体大小为标准, 文档中的字体大小都会以此为参照。

如:

- `html {font-size:62.5%; /* font-size:10px */}`
- `body {font-size:1.2rem ; }`
- `p {font-size:1.4rem;}`





## rem兼容性

- 兼容性：IE9+，Firefox、Chrome、Safari、Opera 的主流版本都支持，为了兼容不支持 rem 的浏览器，我们需要在 rem 前面写上对应的 px 值，这样不支持的浏览器可以优雅降级。

```
html { font-size: 10px; /* 不建议设置 font-size: 62.5%; 在 IE 9-11 上有  
偏差，具体表现为 1rem = 9.93px。 */ }
```

```
.square { width: 5rem; /* 50px */  
          height: 5rem; /* 50px */ }
```



## em和rem

```
<style>
```

```
html{ font-size: 20px; }
```

```
body{ font-size: 1.4rem; /* 1.4rem = 28px */  
      padding: 0.7rem; /* 0.7rem = 14px */ }
```

```
div{ padding: 1em; /* 1em = 28px */ }
```

```
span{ font-size: 1rem; /* 1rem = 20px */  
      padding: 0.9rem; /* 0.9rem = 18px */ }
```

```
</style>
```

**em**相对于父元素，**rem**相对于根元素





# 块级元素和行内元素

- 块级元素：块级元素排斥其他元素与其位于同一行，块级元素一般做为容器使用。
- 行内元素：可以与其他元素同行，一般不做容器使用。
- 常见块级元素：div p form ul ol li h1--h6等；
- 常见的行内元素：span strong a；



# 各级标题样式

H1,H2 , H3 , H4 , H5 , H6默认样式有margin设置

- h1 { font-size: 2em; margin: .67em 0 }
- h2 { font-size: 1.5em; margin: .75em 0 }



# 块级元素和行内元素

- 块级元素独自占一行且宽度会占满父元素宽度，行内元素不会独占一行，相邻行内元素可以排在同一行。

<p>块级元素一</p>

<p>块级元素二</p>

<span>行内元素一</span>

<span>行内元素二</span>



# 块级元素和行内元素

- 块级元素可以设置width和height，行内元素设置width和height无效。



# 块级元素和行内元素

```
p {background-color: red; height: 50px; width: 50%;}
```

```
div {background-color: green; height: 50px; width: 40%;}
```

```
span {background-color: gray; height: 70px;}
```

```
strong {background-color: blue; height: 70px;}
```

<p>块级元素一</p>

<div>块级元素二</div>

<span>行内元素一</span>

<strong>行内元素二</strong>



- 块级元素可以设置margin和padding属性，行内元素水平方向的margin和padding如margin-left、padding-right可以产生边距效果，但是竖直方向的如padding-top和margin-bottom不会产生边距效果





- 块级元素对应display: block, 行内元素对应display: inline。可以通过修改元素的display属性来切换行内元素和块级元素display: inline-block; 可以让元素具有块级元素和行内元素的特性: 既可以设置长宽, 可以让padding和margin生效, 又可以和其他行内元素并排。是一个很实用的属性



# 行高

- line-height属性
- <div>内文字垂直居中
- 单行文字：行高设为所在div的宽度



# 选择器分组

```
h2 { color: red; }
```

```
.abc { color: red; }
```

```
.abccom { color: red; }
```

可以这样写：

```
h2, .abc, .abccom { color: red; }
```

使用组合，可以一次定义多个CSS，节省很多字节和时间。



# CSS 派生选择器 ( contextual selectors )

- 根据文档的上下文关系来确定某个标签的样式
- 例子：

```
li strong { font-style: italic; font-weight: normal; }
```

```
<p> <strong>浙江</strong> </p>
```

```
<ol>
```

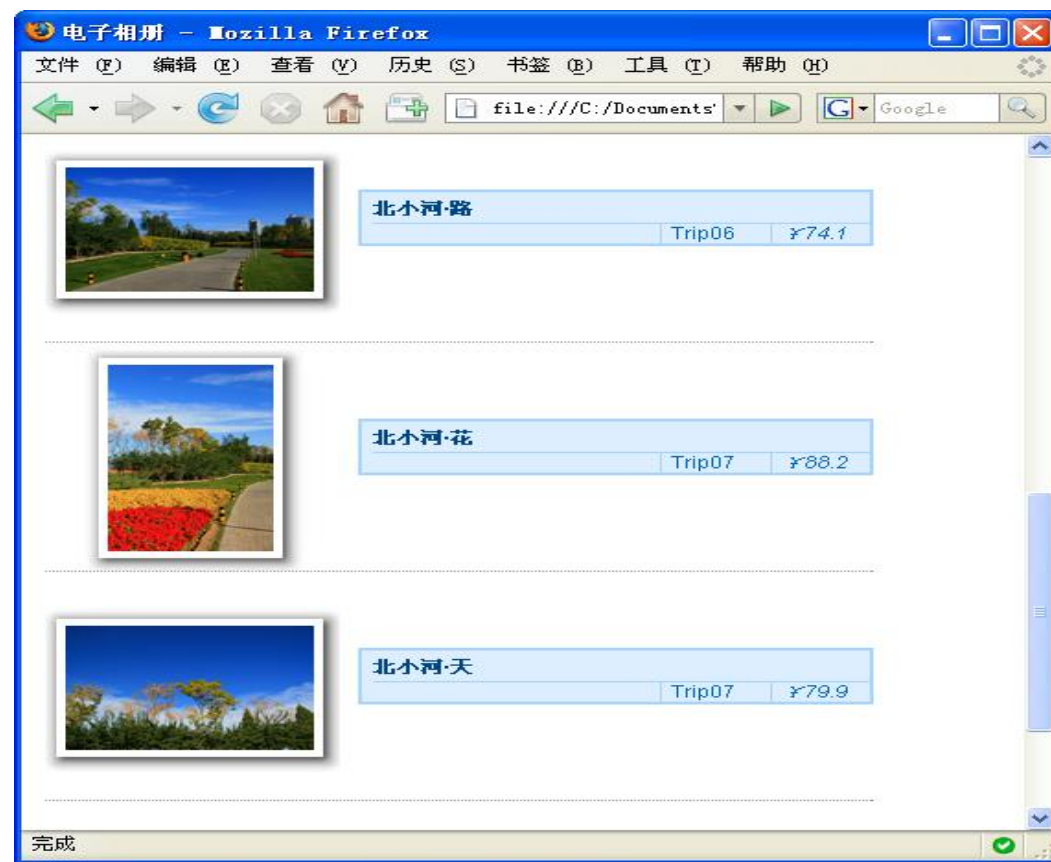
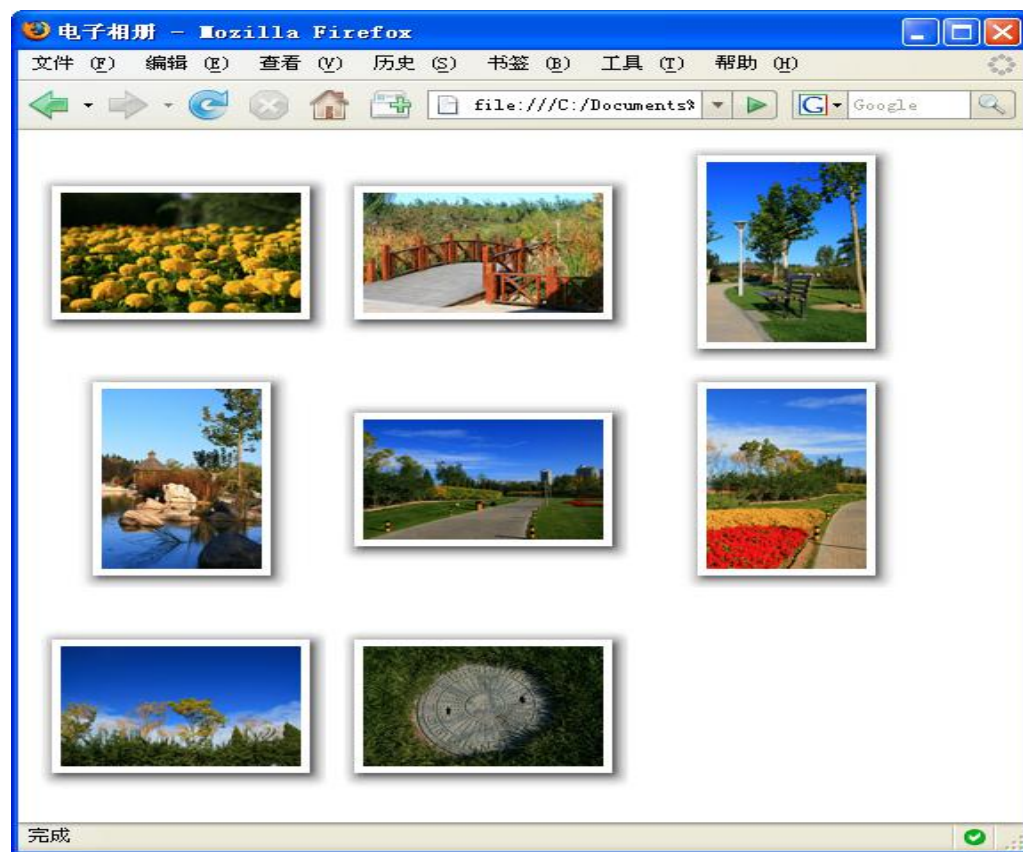
```
<li> <strong>金华</strong> </li>
```

```
<li>师大</li>
```

```
</ol>
```



# 布局例子—电子相册



这两个看起来完全不同的页面实际上HTML代码是完全相同的，只是使用了不同的CSS设计。



谢谢！

web开发

主讲老师：徐晓丹

浙江师范大学

