

1. css的三大特性

层叠性、继承性、优先级

1.1层叠性

相同选择器给设置相同的样式，此时一个样式会覆盖（层叠）另一个冲突的样式。
层叠性主要解决样式冲突的问题

层叠性原则：

- 样式冲突。遵循的原则是就近原则，哪个样式离结构近，就执行哪个样式
- 样式不冲突，不会层叠

1.2 继承性

子标签会继承父标签的某些样式，比如文本颜色和字号
子元素可以继承父元素的样式（text-,font-,line-这些元素开头的可以继承，以及color属性）

行高的继承性

```
body{
  font:12px/1.5 Microsoft YaHei;
}
```

- 行高可以跟单位，也可以不跟单位
- 1.5指的是当前元素文字大小（font-size)的1.5倍

1.3 优先级

当同一个元素指定多个选择器，就会有优先级的产生

- 选择器相同，则执行层叠性
- 选择器不同，则根据选择器权重执行

选择器	选择器权重
继承或*	0,0,0,0
元素选择器	0,0,0,1

选择器	选择器权重
类选择器，伪类选择器	0,0,1,0
id选择器	0,1,0,0
行内样式style=""	1,0,0,0
!important 重要的	无穷大

- 权重是由4组数字组成，不会有进位
- 等级判断从左到右，如果某一数值相同，则判断下一数值
- 继承的元素是0，如果该元素没有直接选中，不管父元素权重多高，子元素得到的权重都是0

权重叠加

如果是复合选择器,则会有权重叠加,需要计算权重

2.盒子模型

页面布局要学习三大核心，盒子模型，浮动，定位

2.1看透网页布局的本质

网页布局过程：

- 先准备好相关的网页元素，网页元素基本都是盒子box
- 利用css设置好盒子样式，然后摆放到相应位置（核心）
- 往盒子里装内容

2.2盒子模型（box model）的组成

所谓盒子模型，就是把HTML页面中的布局元素看作是一个矩形的盒子，也就是一个盛装内容的容器

css盒子本质上是一个盒子，封装周围的HTML元素，包括：边框，外边距，内边距和实际内容

组成：border**边框**、content**内容**、padding**内边距**、margin**外边距**

2.3边框（border）

```
border-weight:5px;  
border-style:solid|dashed|dotted;  
border-color:pink;
```

边框简写：

```
border:1px solid red;
```

- **边框简写没有顺序**

边框分开写法：

```
border-top|bottom|right|left:1px solid red;
```

- 只是设定上边框，其余同理

2.4表格的细线边框

border-collapse属性控制浏览器绘制表格边框的方式。它控制相邻单元格的边框。
语法：

```
border-collapse:collapse;
```

- 表示相邻边框合并在一起

2.5边框会影响盒子实际大小

边框会额外增加盒子的实际大小，因此我们有两种解决方案

- 测量盒子大小的时候不量边框
- 如果测量点时候包含了边框，则需要width/height减去边框宽度

2.6内边距(padding)

内边距即边框与内容之间的距离

padding-top|padding-bottom|padding-right|padding-left

padding简写：

`padding:5px;` 上下左右都有5像素内边距
`padding:5px 10px;` 上下内边距5像素，左右内边距10像素
`padding:5px 10px 20px;` 上边距5像素 左右内边距10像素 下内边距20像素
`padding: 5px 10px 20px 30px;` 上5 右10 下20 左30

- padding会影响盒子的实际大小
- 如果盒子已经有了宽度和高度，此时再指定内边框，会撑大盒子
- 如果保证盒子跟效果图大小保持一致，则让width/height减去多出来的内边距大小即可
- 如果盒子本身没有指定width/height属性，则此时padding不会撑开- 盒子大小

问题:在div标签直接写英文内容时内容会冲出盒子范围，但写中文就不会

2.7外边距(margin)

margin属性用于控制外边距，即控制盒子与盒子之间的距离

`margin-left|right|top|bottom` 左|右|上|下外边距

简写：margin的简写方式与padding完全一致

外边距典型应用

外边距可以让块级盒子水平居中，但必须满足两个条件：

- 盒子必须指定了宽度
- 盒子左右的外边距设置为auto

```
.header{  
    width:900px;  
    margin:0 auto;  
}
```

常见的写法有三种：

```
margin-left:auto;margin-right:auto;
```

```
margin:auto;
```

```
margin:0 auto;(最常见)
```

注意：以上方法是让块级元素水平居中，行内元素或者行内块元素水平居中给其父元素添加text-align:center即可

2.8外边距合并

使用margin定义块元素的垂直外边距时，可能出现外边距的合并

嵌套块元素垂直外边距的塌陷

对于两个嵌套关系（父子关系）的块元素，父元素有上边距同时子元素也有上边距，此时父元素会塌陷较大的外边距值，子元素和父元素谁的上边距值大，就会塌陷谁的值。

解决方案：

- 可以为父元素定义上边框
- 可以为父元素定义上内边距
- 可以为父元素添加overflow:hidden(这种方案不会让盒子变大)
- 还有其它方法，比如浮动、固定绝对定位的盒子不会有塌陷问题

2.9清除内外边距

网页元素很多带有默认的内外边距，而且不同的浏览器默认的也不一致，因此我们在布局前，首先要清除网页元素的内外边距

```
*{  
  padding:0;  
  margin:0;  
}
```

这通常是css的第一行代码

注意：行内元素为了照顾兼容性，尽量只设置左右内外边距，不要设置上下内外边距。但是转换为块级和行内块元素就可以了

3.ps基本操作

打开文件

Ctrl+r打开标尺

右击标尺，改单位为像素

Ctrl+加号/减号放大缩小（或者Ctrl+alt+鼠标滚轮）

移动图片为空格键

用选区工具（矩形小框）拖动，可以测量大小

Ctrl+d可以取消选区