# CSS基础入门

第3天课堂笔记（本课程共8天）

# 一、复习

## 1、文本属性

字体属性：

|  |
| --- |
| 1. color: orange; 2. font-size: 16px; 3. font-family: "Arial","Microsoft Yahei","SimSun"; 4. line-height: 200%; 5. font-weight: bold; 6. font-style: italic; |

复合属性写法：

|  |
| --- |
| 1. font: bold italic 16px/30px "Arial","Microsoft Yahei","SimSun"; |

复合属性可以写全，也可只写一部分。

注意：行高。

文字实在行高内垂直居中。

应用：单行文字垂直居中。

|  |
| --- |
| 1. .center{ 2. width: 300px; 3. height: 50px; 4. margin-top: 30px; 5. border: 1px solid red; 6. font-size: 14px; 7. line-height: 50px; 8. } |

缩进：text-indent

属性值：em（汉字字节）、像素、百分比（缩进父盒子的百分比）

|  |
| --- |
| 1. text-indent: 20%; |

文本水平位置：text-align

属性值：left、center、right。

注意：图片、文字、表单都适用。

|  |
| --- |
| 1. text-align: center; |

文本修饰：text-decoration

常用的几个属性值：none、underline、overline、line-through。

|  |
| --- |
| 1. a{ 2. color: #333; 3. text-decoration: none; 4. } |

## 盒模型

盒子：所占的矩形区域由5个属性确定。

属性：width、height、padding、border、margin

书写内容区域：width、height

实际占有区域：width、height、padding、border

背景渲染区域：border内部。

padding、margin：复合属性。

|  |
| --- |
| 1. padding-top 2. padding-right 3. padding-bottom 4. padding-left |

复合属性书写：四值、三值、两值、单值。

border：属性类型、属性方向。

|  |
| --- |
| 1. border-width 线宽 2. border-style 线型 3. border-color 颜色 |

三个属性值缺一不可，必须写。

|  |
| --- |
| 1. border: 1px solid #000; |

属性方向：得补全三个类型。

|  |
| --- |
| 1. border-top: 1px solid #000; |

拆的特别细：

|  |
| --- |
| 1. border-top-width: 1px; 2. border-top-style: solid; 3. border-top-color: blue; |

案例：小三角

|  |
| --- |
| 1. width: 0; 2. height: 0; 3. padding: 0; 4. border: 15px solid #fff; 5. border-bottom-color: red; 6. border-top: 0; |



# 盒模型拓展

## 清除默认样式

我们有一些标签例如body、p、ul、ol、dl等都有默认的margin或padding。

ul还有默认的小圆点样式

a有默认的文本样式

这些默认样式不是我们想要的，制作网页css第一步，清除默认样式。

|  |
| --- |
| 1. \*{ 2. margin: 0; 3. padding: 0; 4. } 5. ul,ol{ 6. list-style: none; 7. } 8. a{ 9. color:#333; 10. text-decoration: none; 11. } |

练习过程用通配符清样式，实际工作中，把所有的标签列出来，用并集选择器来书写。

## height

实际工作中，我们不设置高度。高度用内部的内容撑开。

因为我们为了避免后期的问题。

新闻列表，有加有减。为了让盒子自动撑高减少，不能设置高度。

注意：只是一种应用情况，自动撑高盒子，并不代表所有。

|  |
| --- |
|  |

FW小技巧：

先锁定背景图。



用参考量距离：在两端拉两条参考线，鼠标放在中间，按住shift键。就得到距离的像素值。

## 宽度剩余

网站左侧内边距是固定的，内容在右侧甩出的空白不一样，可以给定一个足够宽度，只设置左边的内边距，右边空白区域自动剩余。

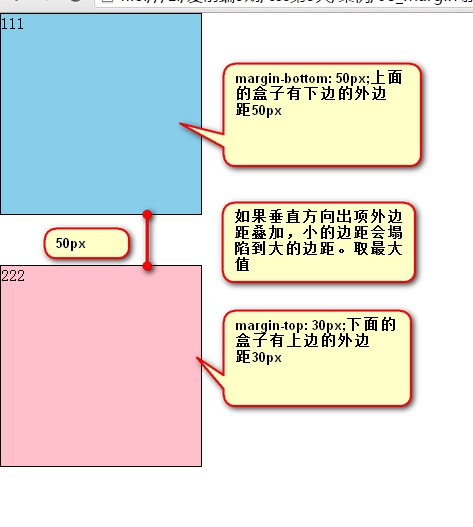
|  |
| --- |
| 1. padding-left: 20px; 2. padding-right: 0px; |

## margin

margin垂直方向塌陷。

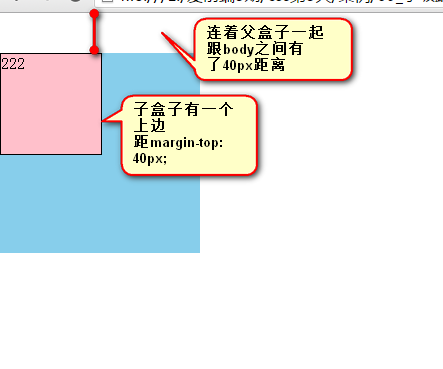
如果两个盒子上下排列，上盒子有下边距，下面的盒子有上边距，它两个之间的距离并不是两个边距的和。小的边距会陷落到大的边距里面。

|  |
| --- |
| 1. .box1{ 2. width: 200px; 3. height: 200px; 4. border: 1px solid #000; 5. margin-bottom: 50px; 6. background: skyblue; 7. } 8. .box2{ 9. width: 200px; 10. height: 200px; 11. border: 1px solid #000; 12. margin-top: 80px; 13. background: pink; 14. } |



不能用儿子去踹父亲。

|  |
| --- |
| 1. .box1{ 2. width: 200px; 3. height: 200px; 4. background: skyblue; 5. } 6. .box2{ 7. width: 100px; 8. height: 100px; 9. border: 1px solid #000; 10. margin-top: 40px; 11. background: pink; 12. } |

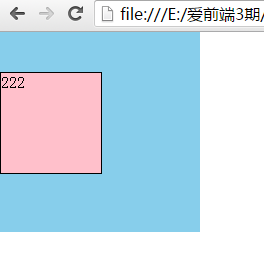


解决方法：

①给父盒子强制限定区域。

②想要之间距离，用父级的padding挤，而不用子级的margin。

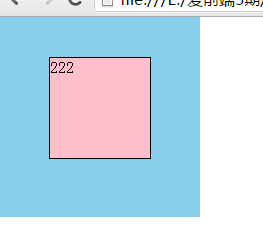
|  |
| --- |
| 1. .box1{ 2. width: 200px; 3. height: 160px; 4. padding-top: 40px; 5. background: skyblue; 6. } |



制作盒子居中。子盒子在父盒子中水平居中。

给子盒子设置。

|  |
| --- |
| margin: 0 auto; |



制作水平居中的盒子，必须给宽度。

|  |
| --- |
| 1. .inner{ 2. width: 1200px; 3. height: 40px; 4. margin: 0 auto; 5. } |

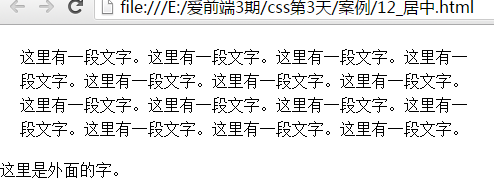
## 5、居中

单行文本的在盒子里垂直居中。

|  |
| --- |
| 1. height: 40px; 2. line-height: 40px; |

多行文本在盒子垂直居中。盒子不给高度，用上下padding挤出来。

|  |
| --- |
| 1. .header{ 2. width: 460px; 3. padding: 20px 20px; 4. line-height: 24px; 5. } |



文本水平方向居中：

|  |
| --- |
| 1. text-align：center； |

盒子在父盒子内水平居中：

给子盒子宽度，设置margin

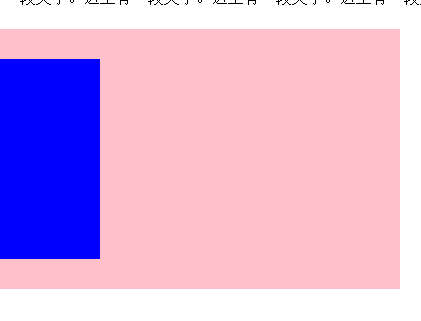
|  |
| --- |
| 1. width: 960px; 2. margin: 0 auto; |

auto自动他会尽可能大的撑开边距。左右都撑的一样的值。

盒子在父盒子内垂直居中：

不给父盒子设置高度，用子盒子的高度撑开。给父盒子设置同样的上下边距。

|  |
| --- |
| 1. .box{ 2. width: 400px; 3. padding:30px 0; 4. background: pink; 5. } 6. .inner{ 7. width: 100px; 8. height: 200px; 9. background: blue; 10. } 11. <div class="box"> 12. <div class="inner"></div> 13. </div> |



写字

## 父子盒模型

子盒子的最大实际占有位置宽度，不能超过父盒子的width。

制作的时候，要注意有padding和border，得给子盒子的width进行内减。

|  |
| --- |
| 1. .box{ 2. width: 400px; 3. padding:30px 0; 4. background: pink; 5. } 6. .box div{ 7. width: 360px; 8. height: 50px; 9. padding: 0 20px; 10. margin-bottom: 20px; 11. background: blue; 12. } |



# 标准文档流

类似我们的Word文档，内容书写的时候必须在光标位置书写。光标位置受前面元素的影响。前面变大字号、变大图片，光标往下移动。

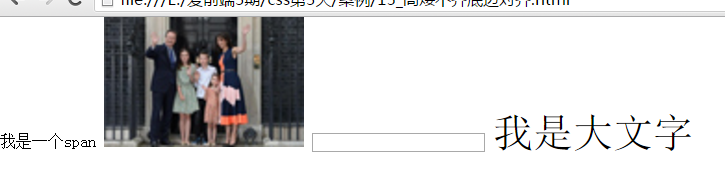
Word是标准文档流：书写时必须从上到下、从左到右书写。

网页制作的时候：web制作也是标准文档流，必须从上往下制作。

## 1、在微观上有几个特性：

①文本类的内容有空白折叠现象。

②文本类的内容：高矮不齐，地边对齐。



③文本类的内容如果超过了盒子宽度会自动换行。

## 块级元素和行内元素

标准文档流将所有的标签区分成两种：块级和行内级。

块级：所有的容器级标签都是块级元素。p标签也是块级元素。

行内级：除了p之外的所有文本级的标签都是行内元素。

|  |
| --- |
| 1. 块级：div h系列 p ul ol li dl dt dd 等。 2. 行内：span a img input b u i等 |

①块级元素的性质：

块级元素独占一行。

可以设置宽度和高度。

如果没设置宽度，会自动撑满父亲。父级的100%

②行内元素的性质：

行内元素不能独占一行，有其他的行内元素，排在一行书写。

不能设置宽高的。

宽度由内容撑开。

③行内块性质：

既可以是设置宽高，又能在一行排列。

块级元素和行内元素可以互相转换。

css有一个属性叫做显示模式（display）.

两个属性值：block（块级）、inline（行内）、inline-block（行内块）。

行内转块：（常用）

|  |
| --- |
| 1. display: block; |

块转行内：（1年都用不到1次）

|  |
| --- |
| 1. display: inline; |

标准文档流做网页，做不出效果。必须让元素脱离标准文档流，才能进行网页灵活制作。

脱标：浮动、绝对定位、固定定位。

# 浮动

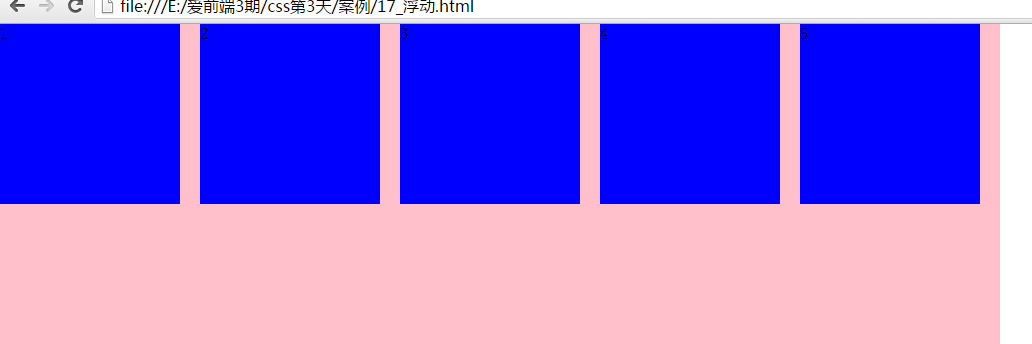
浮动：可以让我们的元素并排在一行排列。

float：浮动、漂浮。

属性值：left、right。

浮动的元素会贴父级盒子的边。

|  |
| --- |
| 1. ul li{ 2. float: left; 3. width: 180px; 4. margin-right: 20px; 5. margin-bottom: 20px; 6. height: 180px; 7. background: blue; 8. } |



浮动作用：让元素脱离了标准文档流，既可以设置宽高，还可以并排在一行。

右浮动，会贴父盒子的右边。后面的元素，会依次贴前面元素的左边。

|  |
| --- |
|  |



如果想要子级元素都在一行，得保证父盒子的宽度足够。

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |