目录

[一、 基础认知 1](#_Toc23263)

[1. CSS引入方式 1](#_Toc20126)

[二、 基础选择器 1](#_Toc26027)

[1.标签选择器 1](#_Toc13073)

[三、 字体和文本样式 3](#_Toc5864)

[1.字体大小 3](#_Toc15419)

[2.字体粗细 3](#_Toc31469)

[3.字体样式(是否倾斜) 3](#_Toc3908)

[4.字体系列font-family 4](#_Toc2991)

[5.样式的层叠问题 4](#_Toc10768)

[6.字体font相关属性的连写 5](#_Toc5173)

[7.文本缩进 5](#_Toc17753)

[8.文本水平对齐方式 5](#_Toc25824)

[9.水平居中方法总结text-align : center 6](#_Toc10342)

[10.文本修饰 6](#_Toc1718)

[11.行高 7](#_Toc911)

[四、选拌器进阶 8](#_Toc20055)

[1.后代选择器:空格 8](#_Toc20477)

[2.子代选择器:> 8](#_Toc29505)

[3.并集选择器:， 9](#_Toc26176)

[4.交集选择器:紧挨着 9](#_Toc13968)

[5. hover伪类选择器 10](#_Toc19117)

[五、背景相关属性 10](#_Toc15451)

[1.背景颜色 10](#_Toc5470)

[2.背景图片 11](#_Toc20302)

[3.背景平铺 11](#_Toc8851)

[4.背景位置 11](#_Toc20727)

[5.背景相关属性的连写形式 12](#_Toc2784)

[6.(拓展)img标签和背景图片的区别 13](#_Toc5334)

[六、光紫显示模式 13](#_Toc31507)

[1.块级元素 13](#_Toc31349)

[2.行内元素 13](#_Toc27313)

[3.行内块元素 14](#_Toc193)

[4.元素显示模式转换 14](#_Toc13428)

[七、CSS特性 15](#_Toc31014)

[1.继承性的介绍 15](#_Toc1888)

[2.层叠性的介绍 15](#_Toc5302)

[八、CSS三大特性 16](#_Toc7639)

[1.优先级的介绍 16](#_Toc9085)

[2.权重叠加计算 17](#_Toc21024)

[九、Chrome调试工具 17](#_Toc24423)

[十、盒子模型 18](#_Toc9265)

[1.盒子模型的介绍 18](#_Toc10085)

[2.内容的宽度和高度 18](#_Toc8896)

[3.边框(border) -连写形式 18](#_Toc29881)

[4.边框(border) - 单方向设置 19](#_Toc31206)

[5.边框(border)-单个属性 19](#_Toc25478)

[6. CSS3盒模型（自动内减) 19](#_Toc15325)

[7.版心居中 20](#_Toc12581)

[8.外边距折叠现象–①合并现象 20](#_Toc23706)

[9.外边距折叠现象-②塌陷现象 20](#_Toc26052)

[10.如果想要通过margin或padding改变行内标签的垂直位置，无法生效。 21](#_Toc24406)

[十一、结构伪类选择器 21](#_Toc19549)

[十二、伪元素 22](#_Toc27971)

[十三、标准流 23](#_Toc13065)

[十四、浮动 23](#_Toc1792)

[1.浮动的特点 23](#_Toc32563)

[十五、清除浮动 24](#_Toc20099)

[1.清除浮动的介绍 24](#_Toc15429)

[2.清除浮动的方法一①直接设置父元素高度 25](#_Toc5531)

[3.清除浮动的方法一②额外标签法 25](#_Toc16567)

[4.清除浮动的方法一③单伪元素清除法 25](#_Toc32054)

[5.清除浮动的方法—④双伪元素清除法 26](#_Toc9395)

[6.清除浮动的方法—⑤给父元素设置overflow : hidden 26](#_Toc5125)

[十六、定位 27](#_Toc24224)

[1. 网页常见布局方式 27](#_Toc5959)

[2.定位的常见应用场景 27](#_Toc15709)

[3.使用定位的步骤 27](#_Toc20142)

[4.相对定位 28](#_Toc29934)

[5.绝对定位 29](#_Toc1223)

[6.固定定位 29](#_Toc23010)

[7.元素层级问题 30](#_Toc26444)

[十七、定位装饰 30](#_Toc2253)

[1.认识基线 30](#_Toc14144)

[2.文字对齐问题 31](#_Toc25207)

[3.垂直对齐方式 31](#_Toc11984)

[4.光标类型 31](#_Toc19131)

[5.边框圆角 32](#_Toc29510)

[6.边框圆角的常见应用 32](#_Toc18220)

[7.溢出部分显示效果 33](#_Toc23448)

[(拓展)元素整体透明度 33](#_Toc17389)

[十八、项目样式补充 34](#_Toc22608)

[1.精灵图的介绍 34](#_Toc14449)

[2.精灵图的使用步骤 35](#_Toc28646)

[3.背景图片大小 35](#_Toc29423)

[4.background连写拓展 35](#_Toc6854)

[5.盒子阴影 36](#_Toc18648)

[6.过渡 37](#_Toc25745)

[十九、项目前置认知 38](#_Toc7052)

[1. DOCTYPE文档说明 38](#_Toc9307)

[2.网页语言 38](#_Toc23580)

[3.字符编码（了解) 38](#_Toc27642)

[4. SEO简介 39](#_Toc19949)

[5. SEO三大标签 39](#_Toc3741)

[6. ico图标设置 39](#_Toc1891)

[二十、项目结构搭建 40](#_Toc24898)

[1.文件和目录准备 40](#_Toc26885)

[2.完成后的目录及文件结构 40](#_Toc11210)

# 基础认知

## 1. CSS引入方式

·内嵌式: CSS写在style标签中

提示: style标签虽然可以写在页面任意位置，但是通常约定写在head标签中

·外联式: CSS 写在一个单独的.css文件中

提示:需要通过link标签在网页中引入

·行内式: CSS写在标签的style属性中

提示:基础班不推荐使用，之后会配合js使用

·CSS常见三种引入方式的特点区别有哪些（书写位置、作用范围、使用场景)?



# 基础选择器

## 1.标签选择器

·结构:标签名{ css属性名︰属性值;}

·作用:通过标签名，找到页面中所有这类标签，设置样式

·注意点:

1.标签选择器选择的是一类标签，而不是单独某一个

2.标签选择器无论嵌套关系有多深，都能找到对应的标签

2.类选择器

·结构: **.**类名{ css属性名:属性值;}

·作用:通过类名，找到页面中所有带有这个类名的标签，设置样式

1.所有标签上都有class属性，class属性的属性值称为类名（类似于名字)

2.类名可以由数字、字母、下划线、中划线组成，但不能以数字或者中划线开头

3.一个标签可以同时有多个类名，类名之间以空格隔开

4、类名可以重复，一个类选择器可以同时选中多个标签

3. id选择器

·结构:#id属性值{ css属性名:属性值;}

·作用:通过id属性值，找到页面中带有这个id属性值的标签，设置样式

·注意点:

1.所有标签上都有id属性

2. id属性值类似于身份证号码，在一个页面中是唯一的，不可重复的!

3.一个标签上只能有一个id属性值

4.一个id选择器只能选中一个标签

4.通配符选择器

·结构:\*{css属性名:属性值;}

·作用:找到页面中所有的标签，设置样式

·注意点:

1.开发中使用极少，只会在极特殊情况下才会用到

2．在基础班小页面中可能会用于去除标签默认的margin和padding(后续讲解)

# 字体和文本样式

## 1.字体大小

·属性名: font-size

·取值:数字＋px

·注意点:

谷歌浏览器默认文字大小是16px

单位需要设置，否则无效

## 2.字体粗细

·属性名: font-weight

·取值:



注意点:

·不是所有字体都提供了九种粗细，因此部分取值页面中无变化

·实际开发中以:正常、加粗两种取值使用最多。

## 3.字体样式(是否倾斜)

·属性名:font-style

·取值:

正常（默认值): normal

倾斜: italic

## 4.字体系列font-family

·属性名: font-family

·常见取值:具体字体1,具体字体2,具体字体3,具体字体4...字体系列

具体字体: "Microsoft YaHei"、微软雅黑、黑体、宋体、楷体等.......

字体系列: sans-serif、serif、monospace等....

·渲染规则:

1.从左往右按照顺序查找，如果电脑中未安装该字体，则显示下一个字体

2.如果都不支持，此时会根据操作系统，显示最后字体系列的默认字体

·注意点:

1．如果字体名称中存在多个单词，推荐使用引号包裹

2．最后一项字体系列不需要引号包裹

3. 网页开发时，尽量使用系统常见自带字体，保证不同用户浏览网页都可以正确显示

## 5.样式的层叠问题

·问题:

给同一个标签设置了相同的样式，此时浏览器会如何渲染呢?

·结果:

如果给同一个标签设置了相同的属性，此时样式会层叠(覆盖)，写在最下面的会生效.（后面的覆盖前面的）

·TIP:

css (Cascading style sheets)层叠样式表

所调的层叠即叠加的意思，表示样式可以一层一层的层叠覆盖

## 6.字体font相关属性的连写

·属性名: font(复合属性)

·取值:

font : style weight size family;

·省略要求:

只能省略前两个，如果省略了相当于设置了默认值

·注意点:

如果需要同时设置单独和连写形式

要么把单独的样式写在连写的下面

要么把单独的样式写在连写的里面

## 7.文本缩进

·属性名: text-indent>

·取值:

数字+pX

数字+em(推荐:1em =当前标签的font-size的大小)

## 8.文本水平对齐方式

·属性名: text-align

·取值:



·注意点:

如果需要让文本水平居中，text-align属性给文本所在标签(文本的父元素）设置

## 9.水平居中方法总结text-align : center

·text-align : center 能让哪些元素水平居中?!

1．文本

2. span标签、a标签

3. input标签、img标签

·注意点:

1. 如果需要让以上元素水平居中, text-align : center需要给以上元素的父元素设置

## 10.文本修饰

·属性名: text-decoration

·取值:



·注意点:

开发中会使用text-decoration : none ;清除a标签默认的下划线

## 11.行高

·作用:控制一行的上下行间距

·属性名: line-height

·取值:

数字+px

倍数（当前标签font-size的倍数)

·应用

1. 让单行文本垂直居中可以设置line-height:文字父元素高度
2. 网页精准布局时，会设置line-height : 1可以取消上下间距

·行高与font连写的注意点:

如果同时设置了行高和font连写，注意覆盖问题

font : style weight size/line-height family ;

拓展颜色常见取值（了解)

·属性名:

如:文字颜色: color

如:背景颜色: background-color

·属性值:



标签水平居中方法总结margin : 0 auto

·如果需要让div、p、 h (大盒子）水平居中?

可以通过margin : 0 auto ;实现

·注意点:

1.如果需要让div、p. h (大盒子)水平居中，直接给当前元素本身设置即可

2. margin: 0 auto一般针对于固定宽度的盒子，如果大盒子没有设置宽度，此时会默认占满父元素的宽度

# 四、选拌器进阶

## 1.后代选择器:空格

·作用:根据HTML标签的嵌套关系，选择父元素后代中满足条件的元素

·选择器语法:选择器1选择器2{ css }

·结果:

在选择器1所找到标签的后代(儿子、孙子、重孙子...）中，找到满足选择器2的标签，设置样式〉

·注意点:

1.后代包括:儿子、孙子、重孙子......

2．后代选择器中，选择器与选择器之前通过空格隔开

## 2.子代选择器:>

·作用:根据HTML标签的嵌套关系，选择父元素子代中满足条件的元素

·选择器语法:选择器1>选择器2{ css }

·结果:

在选择器1所找到标签的子代(儿子)中，找到满足选择器2的标签，设置样式

·注意点:

1.子代只包括:儿子

2.子代选择器中，选择器与选择器之前通过>隔开

## 3.并集选择器:，

·作用:同时选择多组标签，设置相同的样式

·选择器语法:选择器1，选择器2 { css }

·结果:

找到选择器1和选择器2选中的标签，设置样式

·注意点:

1．并集选择器中的每组选择器之间通过，分隔

2．并集选择器中的每组选择器可以是基础选择器或者复合选择器

3.并集选择器中的每组选择器通常一行写一个，提高代码的可读性

## 4.交集选择器:紧挨着

·作用:选中页面中司时满足多个选择器的标签

·选择器语法:选择器1选择器2{ css }

·结果:

(既又原则)找到页面中既能被选择器1选中，又能被选择器2选中的标签，设置样式

·注意点:

1.交集选择器中的选择器之间是紧挨着的，没有东西分隔

2．交集选择器中如果有标签选择器，标签选择器必须写在最前面

## 5. hover伪类选择器

·作用:选中鼠标悬停在元素上的状态，设置样式

·选择器语法: 选择器:hover { css }

·注意点:

1. 伪类选择器选中的元素的某种状态

5.1 emmet语法

·作用:通过简写语法，快速生成代码

·语法:

类似于刚刚学习的选择器的写法



# 五、背景相关属性

## 1.背景颜色

·属性名: background-color (bgc)

·属性值:

颜色取值:关键字、rgb表示法、 rgba表示法、十六进制......

·注意点:

1.背景颜色默认值是透明: rgba(0,0,0,0). transparent

2.背景颜色不会影响盒子大小，并且还能看清盒子的大小和位置，一般在布局中会习惯先给盒子设置背景颜色

## 2.背景图片

·属性名: background-image (bgi)

·属性值: background-image: url('图片的路径');

·注意点:

1.背景图片中url中可以省略引号

2.背景图片默认是在水平和垂直方向平铺的

3.背景图片仅仅是指给盒子起到装饰效果，类似于背景颜色，是不能撑开盒子的

## 3.背景平铺

·属性名: background-repeat (bgr)

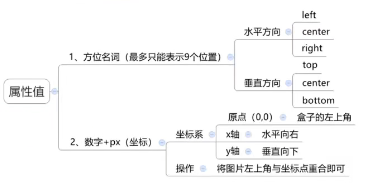
·属性值:



## 4.背景位置

·属性名: background-position (bgp)

·属性值:background-position:水平方向位置垂直方向位置;



·注意点:

方位名词取值和坐标取值可以混使用，第一个取值表示水平，第二个取值表示垂直

## 5.背景相关属性的连写形式

·属性名: background (bg)

·属性值:

单个属性值的合写，取值之间以空格隔开

·书写顺序:

推荐: background: color image repeat position

·省略问题:

可以按照需求省略

特殊情况:在pc端，如果盒子大小和背景图片大小一样，此时可以直接写background: url()

·注意点

如果需要设置单独的样式和连写

①要么把单独的样式写在连写的下面

②要么把单独的样式写在连写的里面

## 6.(拓展)img标签和背景图片的区别

·需求:需要在网页中展示一张图片的效果?

·方法一:直接写上img标签即可

img标签是一个标签，不设置宽高默认会以原尺寸显示

·方法二: div标签＋背景图片

需要设置div的宽高，因为背景图片只是装饰的CSS样式，不能撑开div标签

# 六、光紫显示模式

## 1.块级元素

·显示特点:

1.独占一行（一行只能显示一个)

2．宽度默认是父元素的宽度，高度默认由内容撑开

3.可以设置宽高

·代表标签:

. div、p、 h系列、ul、li、dl、dt、dd、form、header、nav、footer......

## 2.行内元素

·显示特点:

1.一行可以显示多个

2．宽度和高度默认由内容撑开

3.不可以设置宽高

·代表标签:

a、 span. b、u、 i、 s、 strong、ins、em、de.....

## 3.行内块元素

·显示特点:

1. 一行可以显示多个
2. 可以设置宽高

·代表标签:

input、textarea、button、select...

·特殊情况:

img标签有行内块元素特点，但是Chrome调试工具中显示结果是inline

## 4.元素显示模式转换

·目的:改变元素默认的显示特点，让元素符合布局要求

·语法:



拓展1: HTML嵌套规范注意点

1．块级元素一般作为大容器，可以嵌套:文本、块级元素、行内元素、行内块元素等等.....

但是:**p标签中不要嵌套div.p、h等块级元素**

2. a标签内部可以嵌套任意壳素

但是: **a标签不能嵌套a标签**

# 七、CSS特性

## 1.继承性的介绍

·特性:子元素有默认继承父元素样式的特点（子承父业)

·可以继承的常见属性(文字控制属性都可以继承)

1. color

2. font-style、font-weight、font-size、font-family

3. text-indent、text-align

4. line-height

5...…

·注意点:

可以通过调试工具判断样式是否可以继承

只有控制字的能继承

·(拓展）继承失效的特殊情况

如果元素有浏览器默认样式，此时继承性依然存在，但是优先显示浏览器的默认样式

1. a标签的color会继承失效

2. h系列标签的font-size会继承失效

## 2.层叠性的介绍

·特性:

1．给同一个标签设置不同的样式→此时样式会层叠叠加→会共同作用在标签上

2 给同一个标签设置相同的样式→此时样式会层叠覆盖→最终写在最后的样式会生效

·注意点:

1. 当样式冲突时，只有当选择器优先级相同时，才能通过层叠性判断结果

# 八、CSS三大特性

## 1.优先级的介绍

·特性:不同选择器具有不同的优先级，优先级高的选择器样式会覆盖优先级低选择器样式

·优先级公式:

继承<通配符选择器<标签选择器<类选择器<id选择器<行内样式<!Important

·注意点:

1. !important写在属性值的后面，分号的前面!

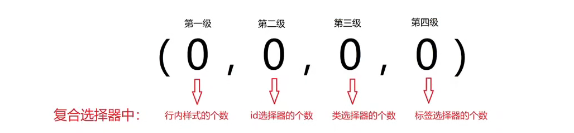
2. !important不能提升继承的优先级，只要是继承优先级最低!3．实际开发中不建议使用!important 。

3.选择器选择范围越广，优先级越低.

## 2.权重叠加计算

·场景:如果是复合选择器，此时需要通过权重叠加计算方法，判断最终哪个选择器优先级最高会生效

·权重叠加计算公式:(每一级之间不存在进位)



·比较规则:

1．先比较第一级数字，如果比较出来了，之后的统统不看

2．如果第一级数字相同，此时再去比较第二级数字，如果比较出来了，之后的统统不看

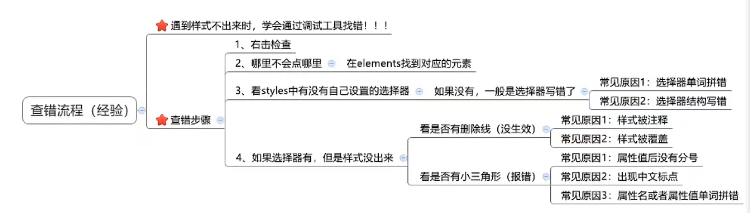
3. .....

4. 如果最终所有数字都相同，表示优先级相同，则比较层叠性（谁写在下面，谁说了算!)

·注意点:!important如果不是继承，则权重最高，天下第一!

# 九、Chrome调试工具

(拓展）查错流程(遇到样式出不来，要学会通过调试工具找错)



# 十、盒子模型

## 1.盒子模型的介绍

·盒子的概念

1.页面中的每一个标签，都可看做是一个“盒子”，通过盒子的视角更方便的进行布局

2．浏览器在渲染(显示〉网页时，会将网页中的元素看做是一个个的矩形区域，我们也形象的称之为盒子

·盒子模型

CSS中规定每个盒子分别由:内容区域(content)、内边距区域(padding)、边框区域(border)、外边距区域（margin)构成，这就是盒子模型

·记忆:可以联想现实中的包装盒

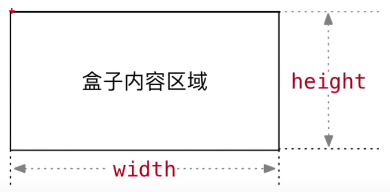
border：盒子边框线 padding：盒子内边距 margin：盒子外边距

## 2.内容的宽度和高度

·作用:利用width 和height属性默认设置是盒子内容区域的大小

·属性: width / height

·常见取值:数字+px



## 3.边框(border) -连写形式

·属性名: border

·属性值:单个取值的连写，取值之间以空格隔开.

如: border : 10px solid red;

·快捷键: bd + tab

## 4.边框(border) - 单方向设置

·场景:只给盒子的某个方向单独设置边框

·属性名: border -方位名词

·属性值:连写的取值

## 5.边框(border)-单个属性

·作用:给设置边框粗细、边框样式、边框颜色效果

·单个属性:



做盒子的时候：从内到外，先宽度高度背景色，放内容，调节内容位置，控制文字细节.

## 6. CSS3盒模型（自动内减)

·需求:盒子尺寸300\*300，背景粉色，边框10px实线黑色，上下左右20px的内边距，如何完成?给盒子设置border或padding时，盒子会被撑大，如果不想盒子被撑大?

·解决方法①:手动内减

操作:自己计算多余大小，手动在内容中减去

缺点:项目中计算量太大，很麻烦

·解决方法②∶自动内减

操作:给盒子设置属性box-sizing : border-box;即可

优点:浏览器会自动计算多余大小，自动在内容中减去

## 7.版心居中

·版心：网页的有效内容

·实现代码：margin：0 auto;

## 8.外边距折叠现象–①合并现象

·场景:垂直布局的块级元素，上下的margin会合并

·结果:最终两者距离为margin的最大值

·解决方法:避免就好

只给其中一个盒子设置margin即可

## 9.外边距折叠现象-②塌陷现象

·场景:互相嵌套的块级元素，子元素的 margin-top 会作用在父元素上

·结果:导致父元素一起往下移动

·解决方法:

1.给父元素设置border-top 或者padding-top (分隔父子元素的margin-top)

**2．给父元素设置overflow: hidden**

3.转换成行内块元素

4．设置浮动

## 10.如果想要通过margin或padding改变行内标签的垂直位置，无法生效。

解决办法：通过line-height（行高）改变行内标签的垂直位置，用padding或者margin改变行内标签的水平位置。

# 十一、结构伪类选择器

**目标:能够使用结构伪类选择器在HTML中定位元素**

·作用与优势:

·作用:根据元素在HTML中的结构关系查找元素

·优势:减少对于HTML中类的依赖，有利于保持代码整洁

·场景:常用于查找某父级选择器中的子元素

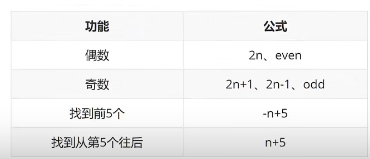
·选择器



·n的注意点:

1.n为:0、1、2、3、4、5、6、......

2.通过n可以组成常见公式



# 十二、伪元素

目标:能够使用伪元素在网页中创建内容

·伪元素:一般页面中的非主体内容可以使用伪元素

·区别:

1.元素:HTML设置的标签

2.伪元素:由CSS模拟出的标签效果

·种类:



·注意点:

1.必须设置content属性才能生效

2.伪元素默认是行内元素

3.默认是行内元素，加宽高不生效，且content必须添加。

# 十三、标准流

**目标:能够认识标准流的默认排布方式及其特点**

·标准流:又称文档流，是浏览器在渲染显示网页内容时默认采用的一套排版规则，规定了应该以何种方式排列元素

·常见标准流排版规则:

·浏览器解析行内块或行内标签的时候，如果标签换行书写会产生一个空格的距离。

# 十四、浮动

·早期的作用：图文环绕

·现在的作用：网页布局：块标签在一行排列

## 1.浮动的特点

1.浮动元素会脱离标准流（简称:脱标)，在标准流中不占位置

·相当于从地面飘到了空中

2.浮动元素比标准流高半个级别，可以覆盖标准流中的元素，但是不能覆盖文字。

3.浮动的标签是顶对齐的。

3.浮动找浮动，下一个浮动会在上一个浮动元素后面左右浮动。

4.浮动元素有特殊的显示效果

·一行可以显示多个

·可以设置宽高

1. 浮动后的标签具备行内块特点。
2. 如果父级宽度不够，子级自动换行。

·注意点:

浮动的元素不能通过text-align:center或者margin:0 auto居中。

3.2 CSS的最佳书写顺序：1.放浮动或者display（行内和块的转换）

2.盒子模型：margin（外边距）、border（边框线）、padding（内边距）、宽度高度背景色

3.文字样式

# 十五、清除浮动

## 1.清除浮动的介绍

·含义:清除浮动带来的影响

影响:如果子元素浮动了，此时子元素不能撑开标准流的块级父元素(父子级标签，子级浮动，父级没有高度，后面的标准流盒子会受影响，显示到上面的位置)

·原因:

子元素浮动后脱标→不占位置

·目的:

需要父元素有高度，从而不影响其他网页元素的布局

## 2.清除浮动的方法一①直接设置父元素高度

·特点:

优点:简单粗暴，方便

缺点:有些布局中不能固定父元素高度。如:新闻列表、京东推荐模块

## 3.清除浮动的方法一②额外标签法

·操作:

1.在父元素内容的最后添加一个块级元素

2．给添加的块级元素设置clear:both（清除左右两侧浮动的影响）

·特点:

缺点:会在页面中添加额外的标签，会让页面的HTML结构变得复杂

## 4.清除浮动的方法一③单伪元素清除法

·操作:用伪元素替代了额外标签



·特点:

优点:项目中使用,直接给标签加类即可清除浮动

## 5.清除浮动的方法—④双伪元素清除法

·操作：



.clearfix::before的作用：解决外边距塌陷问题

塌陷问题：父子级标签，都是块级，子级加margin会影响父级的位置。

·特点:

优点:项目中使用，直接给标签加类即可清除浮动

## 6.清除浮动的方法—⑤给父元素设置overflow : hidden

·操作:

1.直接给父元素设置overflow : hidden

·特点:

优点:方便

# 十六、定位

## 1. 网页常见布局方式

·标准流

1.块级元素独占一行→垂直布局

2.行内元素/行内块元素一行显示多个→水平布局

·浮动

1.可以让原本垂直布局的块级元素变成水平布局

·定位

1. 可以让元素自由的摆放在网页的任意位置

2.一般用于盒子之间的层叠情况

## 2.定位的常见应用场景

1．可以解决盒子与盒子之间的层叠问题

·定位之后的元素层级最高，可以层叠在其他盒子上面

2.可以让盒子始终固定在屏幕中的某个位置

## 3.使用定位的步骤

1.设置定位方式

·属性名: position

·常见属性值:



2．设置偏移值

·偏移值设置分为两个方向，水平和垂直方向各选一个使用即可

·选取的原则一般是就近原则（离哪边近用哪个)



## 4.相对定位

·介绍:自恋型定位，相对于自己之前的位置进行移动

·代码:position:relative;

·特点:

1.需要配合方位属性实现移动

2.相对于自己原来位置进行移动

3.在页面中占位置→没有脱标

4.占有原来的位置。

5.仍然具有标签原来的显示模式.

6.如果left和right都有，以left为准；top和bottom都有，以top为准。

·应用场景:

1.配合绝对定位组CP(子绝父相)

2.用于小范围的移动

## 5.绝对定位

·介绍:拼爹型定位，相对于**非静态定位**的父元素进行定位移动

·代码:position : absolute;

·特点:

1.需要配合方位属性实现移动

2.默认相对于浏览器可视区域进行移动

3.在页面中不占位置→已经脱标

4.先找已经定位的父级，如果有这样的父级就以这个父级为参照物进行定位。

5.有父级，但父级没有定位，以浏览器窗口为参照物进行定位。

6.不占用原来的位置.

7.改变标签的显示模式特点：具有行内块特点，加宽度高度生效，如果没有宽度也没有内容，盒子的宽度就是0（一行共存，宽高生效）

8.绝对定位的盒子不能使用margin：0 auto 居中。

·应用场景:

1.配合绝对定位组CP(子绝父相：父级用相对定位，子级用绝对定位。)

2.绝对定位的盒子居中：position：absolute；

Left：50%；

Top：50%；

Transform：translate（-50%，-50%）；

## 6.固定定位

·介绍:死心眼型定位，相对于浏览器进行定位移动

·代码:position : fixed;

·特点:

1.需要配合方位属性实现移动

2.相对于浏览器可视区域进行移动（改变位置参考浏览器窗口）

3.在页面中不占位置→已经脱标

5.具有行内块的特点，设置尺寸时要设置宽高。

·应用场景:

1.让盒子固定在屏幕中的某个位置

## 7.元素层级问题

·不同布局方式元素的层级关系:

标准流<浮动<定位

·不同定位之间的层级关系:

相对、绝对、固定默认层级相同。

此时HTML中写在下面的元素层级更高，会覆盖上面的元素。

默认情况下，定位的盒子，后来者居上。

1. index：整数；取值越大，显示顺序越靠上，z-index的默认值是0，且必须配合定位才生效。

# 十七、定位装饰

## 1.认识基线

基线:浏览器文字类型元素排版中存在用于对齐的基线(baseline)

## 2.文字对齐问题

·场景:解决行内/行内块元素垂直对齐问题

·问题:当图片和文字在一行中显示时，其实底部不是对齐的

## 3.垂直对齐方式

属性名: vertical-align

属性值：



浏览器遇到行内和行内块标签当做文字处理，默认文字是按基线对齐。在较大的标签里面加vertical-align。

## 4.光标类型

·场景:设置鼠标光标在元素上时显示的样式

·属性名:cursor

·常见属性值:



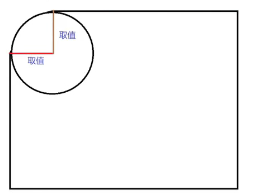
## 5.边框圆角

·场景:让盒子四个角变得圆润，增加页面细节，提升用户体验

·属性名: border-radius

·常见取值:数字+px、百分比

·原理：



·赋值规则:从左上角开始赋值，然后顺时针赋值，没有赋值的看对角!

## 6.边框圆角的常见应用

·画一个正圆:

1.\*盒子必须是正方形

2.设置边框圆角为盒子宽高的一半→border-radius:50%

·胶囊按钮:

1.\*盒子要求是长方形

2设置→border-radius:盒子高度的一半

## 7.溢出部分显示效果

·溢出部分:指的是盒子内容部分所超出盒子范围的区域

·场景:控制内容溢出部分的显示效果，如:显示、隐藏、滚动条......

·属性名: overflow

·常见属性值:



## (拓展)元素整体透明度

·场景:让某元素整体（包括内容)一起变透明

·属性名: opacity

·属性值: 0~1之间的数字，半透明

1:表示完全不透明

0:表示完全透明

·注意点:

opacity会让元素整体透明，包括里面的内容，如:文字、子元素等.......

# 十八、项目样式补充

## 1.精灵图的介绍

·场景:项目中将多张小图片，合并成一张大图片，这张大图片称之为精灵图

·优点:减少服务器发送次数，减轻服务器的压力，提高页面加载速度

·例如:需要在网页中展示8张小图片

8张小图片分别发送→发送8次

合成一张精灵图发送→发送1次

## 2.精灵图的使用步骤

1. 创建一个盒子,设置盒子的尺寸和小图尺寸相同（精灵图的标签都用行内标签 span、吧b、i）
2. 将精灵图设置为盒子的背景图片

3.修改背景图位置

通过PxCook测量小图片左上角坐标，分别取**负值**设置给盒子的background-position: x y;(x:水平方向，y：垂直方向)

## 3.背景图片大小

·作用:设置背景图片的大小,

·语法: background-size:宽度高度;

·取值:



## 4.background连写拓展

·完整连写:background : color image repeat position/size;

·注意点:

background-size和background连写同时设置时，需要注意覆盖问题

·解决:

1.要么单独的样式写连写的下面

2.要么单独的样式写在连写的里面

## 5.盒子阴影

·作用:给盒子添加阴影效果，吸引用户注意，体现页面的制作细节

·属性名:box-shadow

·取值:



## 6.过渡

·作用:让元素的样式慢慢的变化，常配合hover使用，增强网页交互体验

·属性名: transition

·常见取值:



·注意点:

1.过渡需要:默认状态和hover状态样式不同，才能有过渡效果

2. transition属性给需要过渡的元素本身加（哪个标签需要过渡，就在哪个标签的css里面加）

3. transition属性设置在不同状态中，效果不同的

①给默认状态设置，鼠标移入移出都有过渡效果

②给hover状态设置，鼠标移入有过渡效果，移出没有过渡效果

4.如果变化的属性多，直接写all，表示所有需要过渡的属性。

# 十九、项目前置认知

## 1. DOCTYPE文档说明

·<!DOCTYPE html>文档类型声明，告诉浏览器该网页的HTML版本



## 2.网页语言

·<html lang="en">标识网页使用的语言

·作用:搜索引擎归类＋浏览器翻译

·常见语言: zh-CN简体中文/ en英文

## 3.字符编码（了解)

·<meta charset="UTF-8"> 标识网页使用的字符编码

·作用:保存和打开的字符编码需要统一设置，否则可能会出现乱码

·常见字符编码:

1.UTF-8:万国码，国际化的字符编码，收录了全球语言的文字

2. GB2312: 6000+汉字

3. GBK: 20000+汉字

·注意点:开发中统一使用UTF-8字符编码即可

## 4. SEO简介

·SEO (Search Engine Optimization) :搜索引擎优化

·作用:让网站在搜索引擎上的排名靠前

·提升SEO的常见方法:

1．竞价排名

2．将网页制作成html后缀

3.标签语义化(在合适的地方使用合适的标签)

4.……

## 5. SEO三大标签

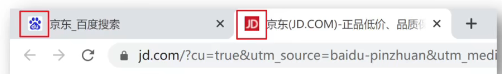
1. title: 网页标题标签

2. description: 网页描述标签

3. keywords: 网页关键词标签

## 6. ico图标设置

·场景:显示在标签页标题左侧的小图标，习惯使用.ico格式的图标



·常见代码:

<link rel="shortcut icon" href="ico图标路径" type="image/x-icon">

# 二十、项目结构搭建

## 1.文件和目录准备

1.新建项目文件夹xtx-pc-client，在VScode中打开

·在实际开发中，项目文件夹不建议使用中文

·所有项目相关文件都保存在xtx-pc-client目录中

2复制favicon.ico 到xtx-pc-client目录

·一般习惯将ico图标放在项目根目录

3.复制images和uploads目录到xtx-pc-client目录中

·images:存放网站固定使用的图片素材，如: logo、样式修饰图片...等.

·uploads:存放网站非固定使用的图片素材，如:商品图片、宣传图片..等

4.新建index.html在根目录

5.新建css文件夹保存网站的样式，并新建以下CSS文件:

· base.css:基础公共样式

·common.css:该网站中多个网页相同模块的重复样式，如:头部、底部

·index.css:首页样式

## 2.完成后的目录及文件结构

