- ◆ 定位
- ◆ 装饰
- ◆ 选择器拓展



- ◆ 能够说出 定位 的常见应用场景
- ◆ 能够说出 不同定位方式 的特点
- ◆ 能够使用 子绝父相 完成元素水平垂直案例
- ◆ 能够写出三种常见的 光标类型 (cursor)
- ◆ 能够使用 圆角边框 属性完成 正圆 和 胶囊按钮 效果
- ◆ 能够说出 display 和 visibility 让 元素本身隐藏 的区别



- ◆ 定位
- ◆ 装饰
- ◆ 选择器拓展

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

1.1 网页常见布局方式

1. 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

2. 浮动

1. 可以让原本垂直布局的 块级元素变成水平布局

3. 定位

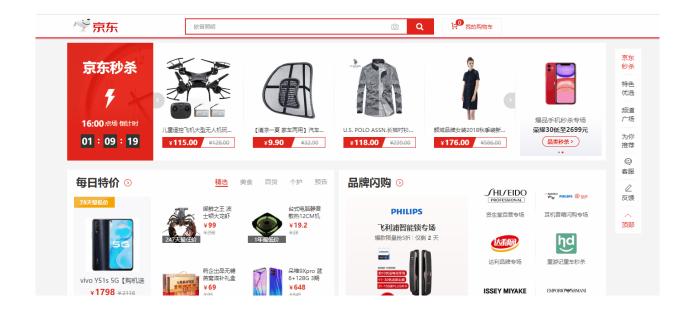
- 1. 可以让元素自由的摆放在网页的任意位置
- 2. 一般用于 盒子之间的层叠情况

1.2 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





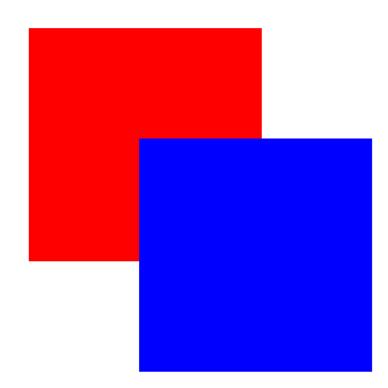
- ▶ 定位的常见应用场景是什么?
 - 1. 解决盒子与盒子之间的层叠问题 → 定位之后元素层级最高,可以层叠在其他盒子上
 - 2. 可以让盒子始终固定在屏幕中的某个位置

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

2.1 定位初体验

- 1. 需求:页面中两个盒子,要求完成图片的效果,可以如何完成?
 - ▶ 针对于盒子与盒子之间的层叠问题,推荐使用定位完成!



2.2 使用定位的步骤

1. 设置定位方式

> 属性名: position

▶ 常见属性值:

定位方式	属性值
静态定位	static
相对定位	relative
绝对定位	absolute
固定定位	fixed

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)

方向	属性名	属性值	含义
水平	left	数字+px	距离左边的距离
水平	right	数字+px	距离右边的距离
垂直	top	数字+px	距离上边的距离
垂直	bottom	数字+px	距离下边的距离

小结

- ▶ 使用定位的步骤是什么?
 - 1. 设置定位的方式: position

定位方式	属性值
静态定位	static
相对定位	relative
绝对定位	absolute
固定定位	fixed

2. 设置偏移值:水平+垂直就近各取一个

方向	属性名	属性值	含义
水平	left	数字+px	距离左边的距离
水平	right	数字+px	距离右边的距离
垂直	top	数字+px	距离上边的距离
垂直	bottom	数字+px	距离下边的距离

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

3.1 静态定位

- > 介绍:静态定位是默认值,就是之前认识的标准流。
- 代码: position:static;
- ▶ 注意点:
 - 1. 静态定位就是之前标准流,不能通过方位属性进行移动
 - 2. 之后说的定位不包括静态定位,一般特指后几种:相对、绝对、固定

- 静态定位能通过方位属性进行移动吗?
 - 不能,静态定位的元素就是默认标准流的情况

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

4.1 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
- ▶ 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

- ▶ 相对定位position的属性值是什么?
 - relative
- ▶ 相对定位是否需要配合方位属性实现移动?
 - 需要配合方位属性实现移动
- ▶ 相对定位的元素相对于谁进行移动?
 - 相对于自己原来位置进行移动
- 相对定位的元素在网页中是否占位置?
 - 在页面中占位置 → 没有脱标

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

5.1 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
- ▶ 应用场景:
 - 1. 配合绝对定位组CP (子绝父相)

5.2 绝对定位到底相对于谁进行偏移

- 绝对定位相对于谁移动?
 - 1. 祖先元素中没有定位 → 默认相对于浏览器进行移动
 - 2. 祖先元素中有定位 → 相对于 最近的 有定位 的祖先元素进行移动

- > 绝对定位position的属性值是什么?
 - absolute
- 绝对定位是否需要配合方位属性实现移动?
 - 需要配合方位属性实现移动
- > 绝对定位的元素相对于谁进行移动?
 - 1. 祖先元素中没有定位 → 默认相对于浏览器进行移动
 - 2. 祖先元素中有定位 → 相对于 最近的 有定位 的祖先元素进行移动
- > 绝对定位的元素在网页中是否占位置?
 - 在页面中不占位置 → 已经脱标

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

6.1 子绝父相介绍

> 场景: 让子元素相对于父元素进行自由移动

> 含义:

• 子元素: 绝对定位

• 父元素:相对定位

> 子绝父相好处:

• 父元素是相对定位,则对网页布局影响最小

- > 子绝父相的含义是什么?
 - 子元素: 绝对定位
 - 父元素:相对定位
- > 子绝父相的应用场景是什么?
 - 让子元素相对于父元素进行自由移动
- > 子绝父相的好处是什么?
 - 父元素是相对定位,则对网页布局影响最小

(拓展) 子绝父绝特殊场景

> 场景: 在使用子绝父相的时候, 发现父元素已经有绝对定位了, 此时直接子绝即可!

▶ 原因:

• 父元素已经有定位已经满足要求,如果盲目修改父元素定位方式,可能会影响之前写好的布局

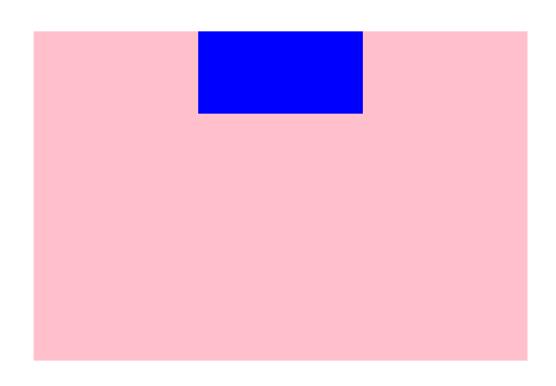
(案例) 卡片模块的hot图标定位案例

▶ 需求:根据设计图,通过PxCook量取数据,在 day05-作业-卡片模块案例 基础上,定位hot图片完成一致的效果



(案例) 子绝父相水平居中案例

》 需求: 使用子绝父相, 让子盒子在父盒子中水平居中(父子元素任意宽度下都能实现)



(案例) 子绝父相水平居中案例-解决方法

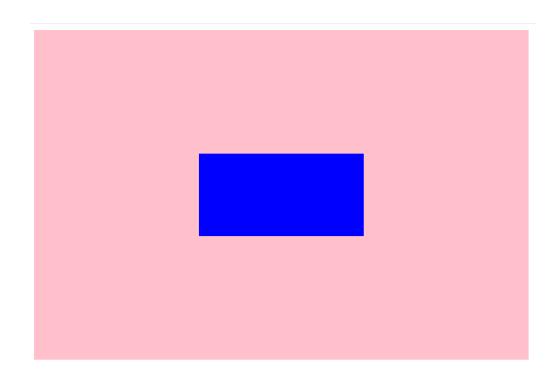
- 1. 子绝父相
- 2. 先让子盒子往右移动父盒子的一半
 - left: 50%
- 3. 再让子盒子往左移动自己的一半
 - 普通做法: margin-left: 负的子盒子宽度的一半
 - 缺点: 子盒子宽度变化后需要重新改代码
 - 优化做法: transform: translateX(-50%)

优点: 表示沿着X轴负方向(往左)始终移动自己宽度的一半,子盒子宽度变化不需要更改代码

- 使用子绝父相水平居中的操作是什么?
 - 1. 子绝父相
 - 2. left: 50%;
 - 3. transform: translateX (-50%);

(案例) 子绝父相水平垂直都居中案例

》 需求: 使用子绝父相, 让子盒子在父盒子中水平垂直都居中(父子元素任意宽高下都能实现)



(案例) 子绝父相水平垂直都居中案例-解决方法

- 1. 子绝父相
- 2. 让子盒子往右走大盒子一半
 - left: 50%
- 3. 让子盒子往下走大盒子一半
 - top: 50%
- 4. 让子盒子往左+往上走自己的一半
 - transform: translate(-50%, -50%);

- 使用子绝父相水平垂直居中的操作是什么?
 - 1. 子绝父相
 - 2. left: 50%;
 - 3. top: 50%;
 - 4. transform: translate (-50%, -50%);

(案例) 导航二维码居中定位案例

» 需求:根据设计图,在 day06-作业-微金所导航案例 基础上,定位二维码图片完成一致的效果



(案例) 底部半透明遮罩效果

》 需求:根据设计图,通过代码在网页中完成一致的效果

> 注意点: 定位的元素会脱标(如:绝对定位),脱标后的元素宽高默认由内容撑开



目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

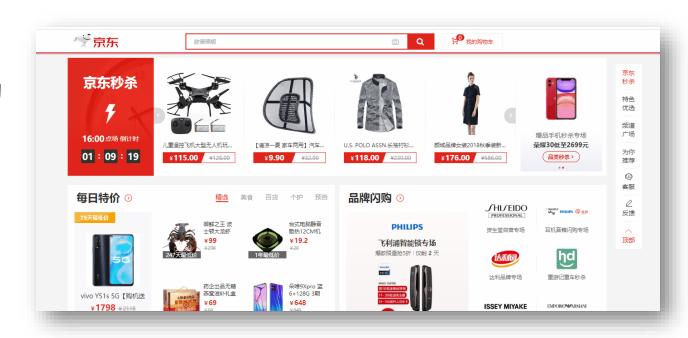
7.1 固定定位

▶ 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
- 3. 在页面中不占位置 → 已经脱标
- ▶ 应用场景:
 - 1. 让盒子固定在屏幕中的某个位置



- > 固定定位position的属性值是什么?
 - fixed
- 固定定位是否需要配合方位属性实现移动?
 - 需要配合方位属性实现移动
- ▶ 固定定位的元素相对于谁进行移动?
 - 相对于浏览器可视区域进行移动
- 固定定位的元素在网页中是否占位置?
 - 在页面中不占位置 → 已经脱标

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

定位方式	属性值	相对于谁移动	是否占位置
静态定位	static	不能通过方位属性移动	占位置
相对定位	relative	相对于自己原来的位置	占位置
绝对地位	absolute	相对于最近的且有定位的祖先元素移动	不占位置 (脱标)
固定定位	fixed	相对于浏览器可视区域	不占位置 (脱标)

目标: 能够说出 定位 的常见应用场景, 并且能够说出 不同定位方式 的特点

- 1. 定位的基本介绍
- 2. 定位的基本使用
- 3. 静态定位
- 4. 相对定位
- 5. 绝对定位
- 6. 子绝父相
- 7. 固定定位
- 8. 元素的层级关系

8.1 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

8.2 更改定位元素的层级

> 场景: 改变定位元素的层级

> 属性名: z-index

》 属性值:数字

• 数字越大,层级越高

- 标准流、浮动、定位元素之间的层级关系是什么样?
 - 标准流 < 浮动 < 定位
- > 不同定位元素之间的层级关系是什么样?
 - 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
- ▶ 通过什么属性可以修改定位元素的层级?
 - z-index: 数字;



- ◆ 定位
- ◆ 装饰
- ◆ 选择器拓展

目标: 能够完成元素的装饰效果

- 1. 垂直对齐方式
- 2. 光标类型
- 3. 边框圆角
- 4. overflow溢出部分显示效果
- 5. 元素本身隐藏

1.1 认识基线 (了解)

▶ 基线:浏览器文字类型元素排版中存在用于对齐的基线 (baseline)

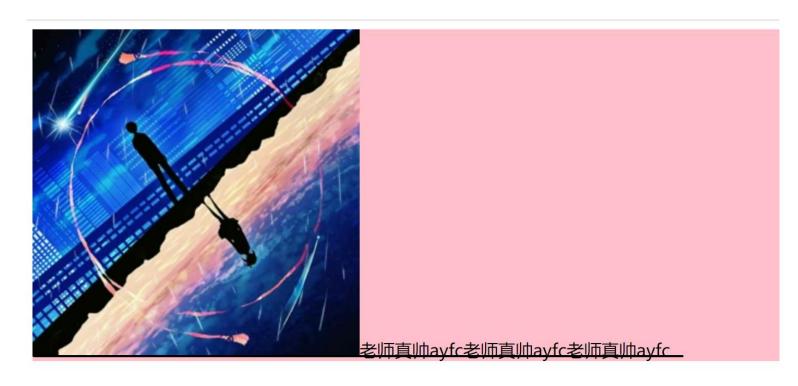


handsome guy

1.2 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



1.3 垂直对齐方式

属性名: vertical-align

▶ 属性值:

属性值	效果
baseline	默认,基线对齐
top	顶部对齐
middle	中部对齐
bottom	底部对齐

- 设置行内/行内块元素垂直对齐方式的属性是什么?
 - vertical-align
- 垂直对齐方式的常见取值有哪些?

属性值	效果
baseline	默认,基线对齐
top	顶部对齐
middle	中部对齐
bottom	底部对齐

(拓展) 项目中 vertical-align 可以解决的问题

- 1. 文本框和表单按钮无法对齐问题
- 2. input和img无法对齐问题
- 3. div中的文本框,文本框无法贴顶问题
- 4. div不设高度由img标签撑开,此时img标签下面会存在额外间隙问题
- 5. 使用line-height让img标签垂直居中问题

▶ 注意点:

- 学习浮动之后,不推荐使用行内块元素让div一行中显示,因为可能会出现垂直对齐问题
- 推荐优先使用浮动完成效果

目标: 能够完成元素的装饰效果

- 1. 垂直对齐方式
- 2. 光标类型
- 3. 边框圆角
- 4. overflow溢出部分显示效果
- 5. 元素本身隐藏

2.1 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:

属性值	效果
default	默认值,通常是箭头
pointer	小手效果,提示用户可以点击
text	工字型,提示用户可以选择文字
move	十字光标,提示用户可以移动

小结

光标类型属性的属性名是什么?

• 属性名: cursor

> 光标类型属性的常见属性值有哪些?

属性值	效果
default	默认值,通常是箭头
pointer	小手效果,提示用户可以点击
text	工字型,提示用户可以选择文字
move	十字光标,提示用户可以移动

目标: 能够完成元素的装饰效果

- 1. 垂直对齐方式
- 2. 光标类型
- 3. 边框圆角
- 4. overflow溢出部分显示效果
- 5. 元素本身隐藏

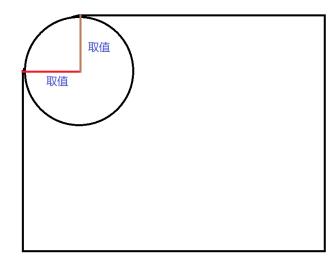
3.1 边框圆角

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

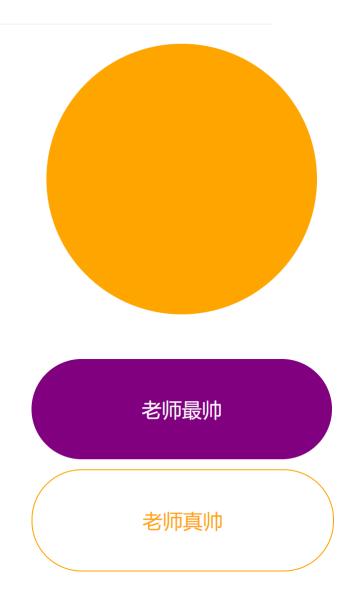
▶ 原理:



▶ 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!

3.2 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



- 如果需要在网页中用盒子展示出一个正圆,应该如何完成?
 - 1. 盒子必须是正方形
 - 2. 设置 → border-radius:50%
- 如果需要在网页中用盒子展示出一个胶囊按钮,应该如何完成?
 - 1. 盒子要求是长方形
 - 2. 设置 → border-radius: 盒子高度的一半

目标: 能够完成元素的装饰效果

- 1. 垂直对齐方式
- 2. 光标类型
- 3. 边框圆角
- 4. overflow溢出部分显示效果
- 5. 元素本身隐藏

4.1 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:

属性值	效果
visible	默认值,溢出部分可见
hidden	溢出部分隐藏
scroll	无论是否溢出,都显示滚动条
auto	根据是否溢出,自动显示或隐藏滚动条

以是内容找是内容找是内容找是内容找是 内容我是内容我是内容我是内容 我是内容我是内容我是内容我是 内容我是内容我是内容我是内容 我是内容我是内容我是内容 我是内容我是内容我是内容 大容我是内容我是内容 我是内容我是内容我是内容 我是内容

小结

> 设置溢出部分显示效果的属性名是什么?

• 属性名: overflow

▶ 设置溢出部分显示效果的常见属性值有哪些?

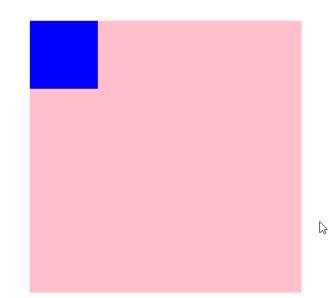
属性值	效果
visible	默认值,溢出部分可见
hidden	溢出部分隐藏
scroll	无论是否溢出,都显示滚动条
auto	根据是否溢出,自动显示或隐藏滚动条

目标: 能够完成元素的装饰效果

- 1. 垂直对齐方式
- 2. 光标类型
- 3. 边框圆角
- 4. overflow溢出部分显示效果
- 5. 元素本身隐藏

5.1 元素本身隐藏

- > 场景: 让某元素本身在屏幕中不可见。如: 鼠标:hover之后元素隐藏
- ▶ 常见属性:
 - 1. visibility: hidden
 - 2. display: none
- ▶ 区别:
 - 1. visibility: hidden 隐藏元素本身,并且在网页中 占位置
 - 2. display: none 隐藏元素本身,并且在网页中 不占位置
- ▶ 注意点:
 - 开发中经常会通过 display属性完成元素的显示隐藏切换
 - display: none; (隐藏) 、 display: block; (显示)



- > 设置元素本身显示与隐藏的常用代码是什么?
 - 元素隐藏 → display: none
 - 元素显示 → display: block

5.2 元素显示隐藏切换案例

▶ 需求:通过代码完成:默认son元素隐藏,当鼠标移入father后让son显示的效果

V

(案例) 导航二维码显示隐藏切换案例

> 需求:通过代码,在刚刚完成的导航二维码居中定位案例基础上,完成如效果图中元素显示隐藏切换效果

我要投资	平台介绍	新手专区	手机微金所	个人中心
				₿

(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

• 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……

(拓展) 边框合并

> 场景: 让相邻表格边框进行合并,得到细线边框效果

> 代码: border-collapse: collapse;

前端与移动开发学员学习情况

编号	姓名	性别	成绩
1	小姐姐	女	100
2	小哥哥	男	100
3	大姐姐	女	100
总成绩		300	



前端与移动开发学员学习情况

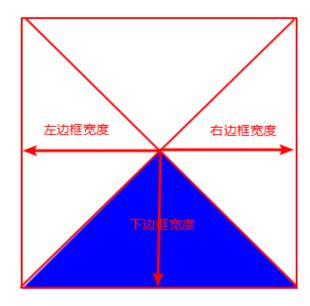
编号	姓名	性别	成绩
1	小姐姐	女	100
2	小哥哥	男	100
3	大姐姐	女	100
总成绩		300	

- 如果让表格单元格之间边框进行合并,可以使用什么属性完成?
 - 边框合并 → border-collapse: collapse;

(拓展) 用CSS画三角形技巧 (面试题)

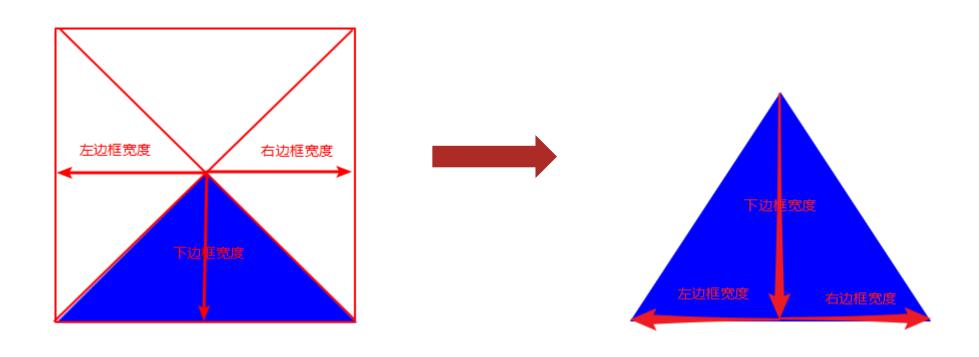
- 》 场景: 在网页中展示出小三角形时,除了可以使用图片外,还可以使用代码完成。
- > 实现原理:
 - 利用盒子边框完成
- > 实现步骤:
 - 1. 设置一个盒子
 - 2. 设置四周不同颜色的边框
 - 3. 将盒子宽高设置为0,仅保留边框
 - 4. 得到四个三角形,选择其中一个后,其他三角形(边框)设置颜色为透明





(拓展) 用CSS画三角形技巧 (面试题)

> 拓展:通过调整不同边框的宽度,可以调整三角形的形态



- ▶ 用CSS画三角形的步骤是什么呢?
 - 1. 设置一个盒子
 - 2. 设置四周不同颜色的边框
 - 3. 将盒子宽高设置为0,仅保留边框
 - 4. 得到四个三角形,选择其中一个后,其他三角形(边框)设置颜色为透明



- ◆ 定位
- ◆ 装饰
- ◆ 选择器拓展

目标: 能够使用 伪类选择器 选择元素的不同状态

- 1. 链接伪类选择器
- 2. 焦点伪类选择器
- 3. 属性选择器

1.1 链接伪类选择器

> 场景: 常用于选中超链接的不同状态

▶ 选择器语法:

选择器语法	功能
a:link { }	选中a链接 未访问过 的状态
a:visited { }	选中a链接 访问之后 的状态
a:hover { }	选中 鼠标悬停 的状态
a:active { }	选中 鼠标按下 的状态

▶ 注意点:

• 如果需要同时实现以上四种伪类状态效果,需要按照 LVHA 顺序书写记忆口诀: 男盆友送了你一个 LV 包包, 你开心的 HA 哈笑

• 其中:hover伪类选择器 使用更为频繁,常用于选择各类元素的悬停状态

小结

▶ 选中超链接的不同状态,常用哪些伪类选择器?

选择器语法	功能
a:link { }	选中a链接 未访问过 的状态
a:visited { }	选中a链接 访问之后 的状态
a:hover { }	选中 鼠标悬停 的状态
a:active { }	选中 鼠标按下 的状态

- 如果同时实现以上四种伪类状态效果,需要按照什么顺序书写?
 - 按照 LVHA 顺序书写

目标: 能够完成 伪类选择器 选择元素的不同状态

- 1. 链接伪类选择器
- 2. 焦点伪类选择器
- 3. 属性选择器

2.1 焦点伪类选择器

- 》 场景: 用于选中元素获取焦点时状态, 常用于表单控件
- ▶ 选择器语法:

```
input:focus {
   background-color: ■skyblue;
}
```

- > 效果:
 - 表单控件获取焦点时默认会显示外部轮廓线

- ▶ 选中表单控件获取焦点的状态,可以使用什么伪类选择器?
 - :focus { }

目标: 能够完成 伪类选择器 选择元素的不同状态

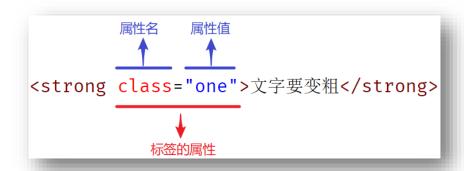
- 1. 链接伪类选择器
- 2. 焦点伪类选择器
- 3. 属性选择器

3.1 属性选择器

> 场景:通过元素上的HTML属性来选择元素,常用于选择 input 标签

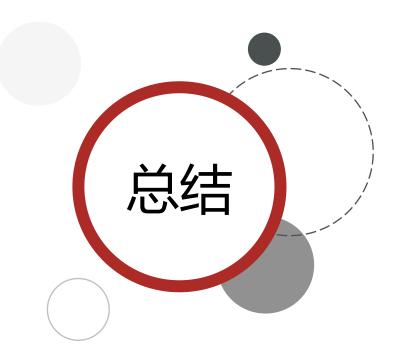
> 选择器语法:

选择器	功能			
E[attr]	选择具有 attr 属性的 E 元素			
E[attr="val"]	选择具有 attr 属性并且属性值等于 val 的 E 元素			



- 选中页面中所有的文本框,使用属性选择器如何实现呢?
 - input[type="text"] { }

目标确认



- ◆ 能够说出 定位 的常见应用场景
- ◆ 能够说出 不同定位方式 的特点
- ◆ 能够使用 子绝父相 完成元素水平垂直案例
- ◆ 能够写出三种常见的 光标类型 (cursor)
- ◆ 能够使用 圆角边框 属性完成 正圆 和 胶囊按钮 效果
- ◆ 能够说出 display 和 visibility 让 元素本身隐藏 的区别