1. 什么是网页的布局方式

--网页的布局方式其实就是指浏览器是如何对网页中的元素进行排版的

--标准流（文档流/普通流）排版方式

--浏览器默认的排版方式就是标准流的排版方式

--在CSS中，元素分为三类，分别是块级元素、行内元素、行内块级元素

--在标准流中有两种排版方式

--垂直排版

--如果元素是块级元素，那么就会垂直排版

--水平排版

--如果元素是行内元素、行内块级元素，那么就会水平排版

--浮动流排版方式

--浮动流是一种“半脱离标准流”的排版方式

--浮动流只有一种排版方式，就是水平排版。它只能设置某个元素左对齐或者右对齐

--注意点：

--浮动流中没有居中对齐，也就是没有center这个取值

--在浮动流中是不可以使用margin: 0 auto; 的

--特点：

--无论是块级元素、行内元素还是行内块级元素都可以通过浮动流来水平排版

--在浮动流中，无论是行内元素、块级元素还是行内块级元素，都可以设置宽高

--综上所述，浮动流中的元素和标准流中的行内块级元素很像

--浮动元素脱标

--脱标：

--当某一个元素浮动之后，这个元素看上去就像是在标准流中被删除了一样，这就是浮动元素的脱标

--浮动元素脱标之后会有什么影响

--如果前面一个元素浮动了，而后面一个元素没有浮动，那么这个时候前面的一个元素就会盖住后面的元素

--浮动元素排序规则

--相同方向上的浮动元素，先浮动的元素会显示在前面，后浮动的元素会显示在后面

--不同方向上的浮动元素，左浮动的会找左浮动，右浮动的会找右浮动

--浮动元素浮动之后的位置，由浮动元素浮动之前在标准流中的位置来确定

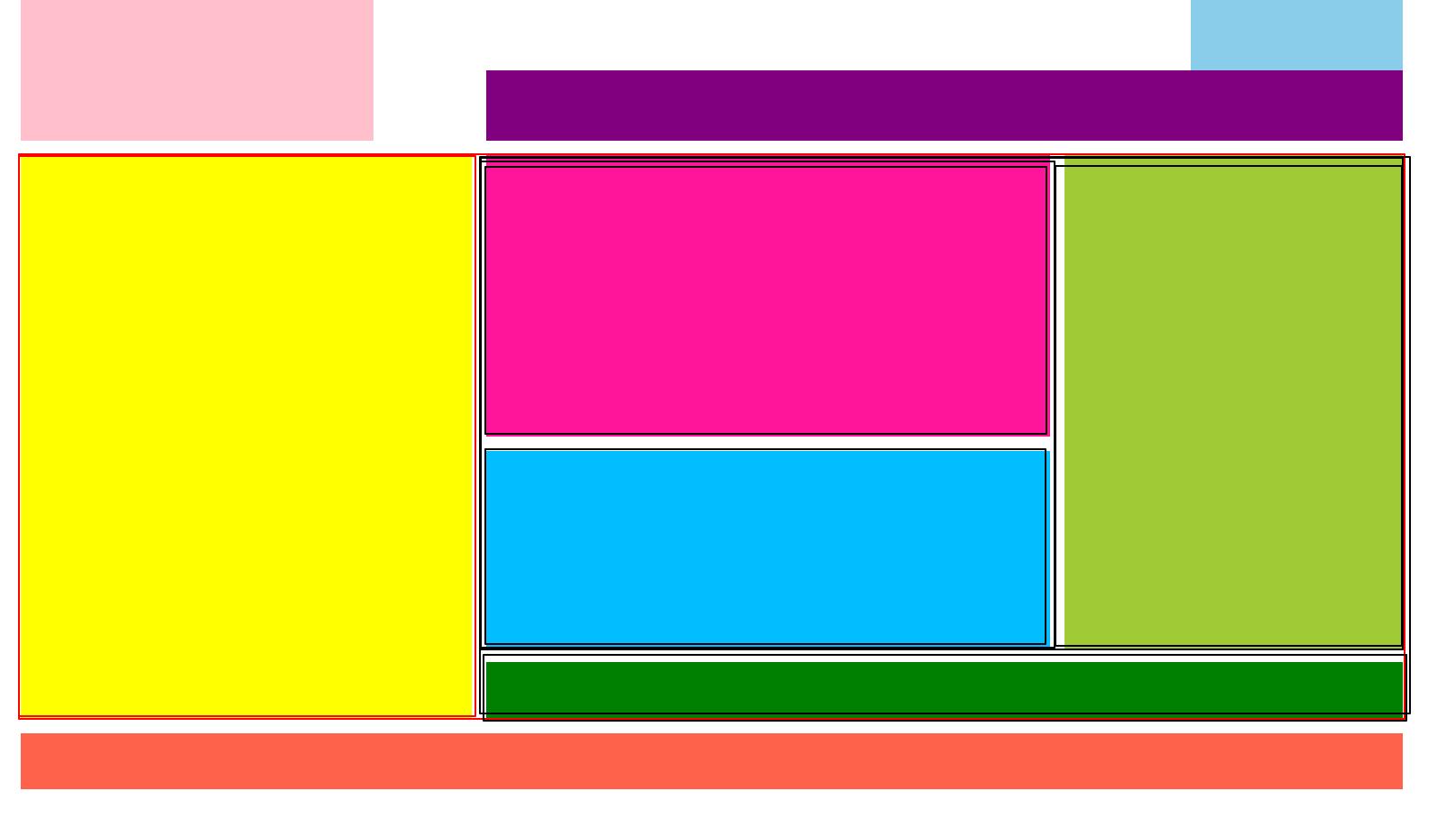
如：

box1左浮，box2不浮动，box3左浮，box4左浮

结果见下图



--浮动练习模型



--在企业开发中什么时候使用标准流，什么时候使用浮动流？

--垂直方向使用标准流，水平方向使用浮动流

--拿到一个很复杂的界面如何入手？

--从上至下布局

--从外向内布局

--水平方向可以先分为一左一右，再对左边或者右边进一步布局

--浮动元素的高度问题

--在标准流中，内容的高度可以支撑起父元素的高度

--在浮动流中，浮动的元素是不可以支撑起父元素的高度的

--清除浮动方式

--给前面一个父元素设置高度（一）

--注意点

--在企业开发中，我们能不写高度就不写高度，所以这种方式用的很少

--给后面的盒子添加clear属性（二）

--取值：

--none：默认取值，按照浮动元素的排序顺序来排序（左浮动找左浮动，右浮动找右浮动）

--left：不要找前面的左浮动元素

--right：不要找前面的右浮动元素

--both：不要找前面的左浮动和右浮动元素

--注意点：

--当我们给某个元素添加clear属性之后，这个属性的margin属性会失效

--隔墙法

--外墙法

--在两个盒子中间添加一个额外的块级元素（div标签）

--给这个额外添加的块级元素设置clear: both;属性

--注意点：

--外墙法可以让第二个盒子使用margin-top属性，却不能让第一个盒子使用margin-bottom属性

--可以通过设置添加的盒子的hight属性来实现上面的效果

--内墙法

--在第一个盒子中所有的子元素最后边添加一个额外的块级元素

--给这个额外添加的块级元素设置clear: both; 属性

--注意点：

--可以让第二个盒子使用margin-top属性

--可以让第一个盒子使用margin-bottom属性

--可以通过设置添加的盒子的hight属性来实现上面的效果

--外墙法和内墙法的区别

--外墙法不能撑起第一个盒子的高度，而内墙法可以撑起第一个盒子的高度

--在企业开发中，我们不常用隔墙法来清除浮动。因为添加的盒子是用来修改样式的，而这违背了HTML标签的初衷，而且额外添加的盒子使得代码的可阅读性变得更差。所以，隔墙法用得少

--伪元素选择器（掌握）

--作用：

--伪元素选择器的作用就是给指定标签的内容前面或者后面添加一个子元素

--格式：

--标签名称::before{

属性名称: 值;

}

--给指定标签的内容前面添加一个子元素

如图所示：



--标签名称::after{

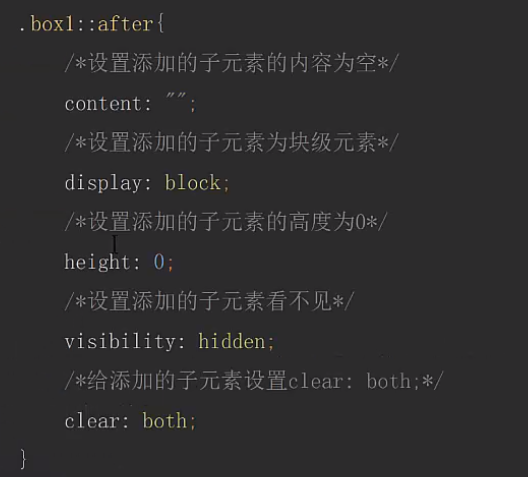
属性名称: 值;

}

--给指定标签的内容后面添加一个子元素

--利用伪元素选择器（四）

--格式：



--注意点：

--这种方式在IE6中是不兼容的，为了能兼容IE6浏览器，需要在下面添加一段代码：

类名{

\*zoom: 1;

}

-- overflow: hidden;

--作用：

--可以将超出标签范围的内容剪裁掉

--清除浮动

--可以通过overflow: hidden; 让里面的盒子设置margin-top之后，外面的盒子不被顶下来

--定位流排版方式

--相对定位

--相对定位就是相对于自己以前在标准流中的位置来移动的

--格式：

标签名{

其他属性: 值;

position: relative;

top:

right:

bottom:

left:

}

--注意点：

--相对定位是不脱离标准流的，会继续占用在标准流中的空间

--在相对定位中同一个方向上的定位属性只能使用一个

--由于相对定位是不脱离标准流的，所以在相对定位中是区分 块级元素、行内元素、行内块级元素的

--由于相对定位是不脱离标准流的，并且相对定位的元素会占用标准流中的位置，所以当给相对定位的元素设置margin/padding等属性时会影响到标准流的布局

--应用场景

--用于对元素进行微调

--配合后面学习的绝对定位来使用

--绝对定位

--绝对定位就是相对于body来定位的

--格式：

标签名称{

其他属性: 值;

position: absolute;

top:

right:

bottom:

left:

}

--注意点：

--绝对定位的元素是脱离标准流的

--绝对定位的元素是不区分 块级元素、行内元素、行内块级元素的

--绝对定位的参考点

--默认情况下，所有的绝对定位元素，无论有没有祖先元素，都会以body作为参考点

--如果一个绝对定位的祖先元素，并且祖先元素也是定位流，那么这个绝对定位的元素都会以定位流的那个祖先元素作为参考点

--只要是这个绝对定位元素的祖先元素都可以

--这个定位流指的是 绝对定位、相对定位、固定定位

--定位流中只有静态定位不行

--如果一个绝对定位的元素有祖先元素，并且祖先元素也是定位流，祖先元素中有多个元素都是定位流，那么这个绝对定位的元素会以离它最近的那个定位流的祖先元素作为参考点

--绝对定位参考点的注意点

--如果一个绝对定位的元素是以body作为参考点，那么该元素其实是以网页的首屏的宽高来作为参考点的，而不是以整个网页的宽高作为参考点的

--一个绝对定位的元素会忽略祖先元素的padding属性，直接贴近祖先元素的边框

在企业开发中，子绝父相（子元素用绝对定位，父元素用相对定位）

--如何让绝对定位的元素水平居中

--只需要设置绝对定位元素的left: 50%;

然后再设置绝对定位元素的margin-left: -元素宽度的一半px;

--固定定位

--固定定位和前面学习的背景关联方式很像，都可以让图片（盒子）不随着滚动条的滚动而滚动

--格式：

标签名称{

其他标签: 值;

position: fixed;

}

--注意点：

--固定定位的元素是脱离标准流的，不会占用标准流中的空间

--固定定位和绝对定位一样不区分 行内元素、块级元素、行内块级元素

--z-index

--什么是z-index属性

--默认情况下，所有的元素都有一个默认的z-index属性，取值是0，z-index属性的作用是专门用于控制定位流元素的覆盖关系

--默认情况下，定位流的元素会盖住标准流的元素

--默认情况下，定位流的元素后面编写的会盖住前面编写的

--如果定位流的元素设置了z-index属性，那么谁的z-index属性比较大，谁就会显示在上面

--注意点：

--从父现象：

--如果两个元素的父元素都没有设置z-index属性，那么哪个元素的z-index属性比较大就显示在上面

--如果两个元素的父元素设置了z-index属性，那么子元素的z-index属性就会失效，也就是说谁的父元素的z-index属性大就会显示在上面

--静态定位