哈喽  我是小K,  首先恭喜你过关斩将来到这里，在这一关我们将会学习到更加深入的知识点，首先来回顾一下之前我们学到的知识点吧~

首先是 width,height 这两个很好理解，所以你只需要记住这两个单词就可以了

A12

当盒子有了宽度和高度之后，又有了边框

A13

Border-width

Border-style

Border-color

Border-top

Border-left

Border-right

Border-bottom

Border-top-color

….

再往后，有背景，包括：

    背景颜色：background-color

    背景图片：background-image

    背景重复：background-repeat

    背景尺寸：background-size

    背景定位：background-position

    背景滚动：background-attachment

回过头来一看，我们已经学会了如此多的的css魔法！你真是太棒了~

不过呢，之前我们学css魔法语句的时候，都是些在html标签里面的，一旦多了的话，页面上就会显得比较混乱，现在你已经是一个成熟的魔法师了，所以我们有必要找个方法来整理一下了。

之前我们知道css代码可以写在标签的 style 属性里面 比如这样：

<div style=“width:100px; height:200px; border:1px solid #333;”></div>

但是这样写的一个缺点刚才也说了，那就是会让页面上的代码看起来比较混乱，而且代码还不能重复使用。

什么意思呢？

比如现在页面当中有两个一模一样的盒子，他们两用相同的背景，相同的边框，相同的尺寸大小，这时候我们会怎么写？

<div style=“width:100px; height:200px; border:1px solid #333;”></div>

<div style=“width:100px; height:200px; border:1px solid #333;”></div>

这样对吧？但是看看这两行代码，其中的css代码是完全一样的，所以这时候我们可以把这段css代码提取出来，放到一个公共的地方，然后给这两个盒子使用

就像孙悟空的72变一样，变一个是变，变两个也是变。那代码少一点，对电脑的压力也就要小一点了。为什么我们经常听说谁谁谁写的程序性能不好，就是因为这个了。所以，我们具体应该怎么做呢？

在html中，有一个标签，叫做style标签。注意哦，它也是一个标签，和div一样的哦~  之前写在div里面的那叫做div的属性，而这个级别更高了。

另外，style标签的位置也是有讲究的，来看看：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        这里可以写css代码

    </style>

</head>

<body>

<div></div>

</body>

</html>

可以看到，style需要写在head标签里面，和title标签平级

不过这时候问题来了，css代码写在这里，要怎么写呢？

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        width: 100px; height:200px; background: #932;

    </style>

</head>

<body>

<div></div>

</body>

</html>

这样，看着总有点怪怪的，而且页面上也没有任何效果

我们回过头来推理一下，之前我们的css代码是直接写在一个标签里面的，这段css代码直接是用来描述它所在的那个标签。而现在css代码写在公共的地方了，那这段css代码到底是用来描述谁的呢？

所以这时候就必须要借助一个工具了，这个工具就叫做选择器，你可以把它理解成一个人的名字，叫这个名字是谁，叫那个名字又是谁：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px; height:200px; background: #932;

        }

    </style>

</head>

<body>

<div></div>

</body>

</html>

现在，可以看到div就能够被看到了，它有了宽，有了高。因为它就叫div  所以这个选择器也直接写 div  后面跟上 { } 就可以了，然后在大括号里面写要给这个div加什么样式 。

就像我们在操场上上体育课，体育老师说： 男生请出列！  于是所有的男生都被选择了出来

体育老师继续说：做20个俯卧撑！   那这就是给男生加上一些状态了

Css选择器也是一样的   在刚才的代码中，我们用div这个选择器选择了页面上所有的div  记住哦，是所有的div哦  因为代码里面只有一个div标签所以才只选中了那一个而已，你再给多加几个div试试？

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px;

            height:200px;

            background: #932;

        }

    </style>

</head>

<body>

<div></div>

</body>

</html>

这时候就可以看到页面上会有好多个盒子了，然后因为每一个盒子都是一样的大小，颜色也一样，所以看起来像是一个长方形  不过如果你给中间的div再加上行间样式，就能看出来区别了：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px;

            height:200px;

            background: #932;

        }

    </style>

</head>

<body>

<div></div>

<div style="width: 300px;background:#200;"></div>

<div></div>

</body>

</html>

可以看到代码中的三个div 他们都被style标签中的css样式所影响，而同时第二个div也会被它自己身上的css样式所影响，这就是css的层叠效果了。你可以无限制的给一个元素加样式，重复加也没关系。

但是要注意的一点是，样式会被覆盖掉，就拿第二个div来说。 页面一开始在读取代码的时候，读到style标签这，电脑知道：哦 所有叫 div 的孩子都得把它养到 100宽，200高，然后得给它们都穿上 #932 这个颜色的衣服

然后继续往下读  读到第一个div  电脑：嗯，这是个div, 来，多吃点，白白胖胖的，宽100  高200 。然后给了一间深红色的衣服。然后继续往下走

哎，又碰到一个div,  于是又继续，来，多吃点，宽100  高两百。穿…  咦，不对，这孩子身上有小纸条，我来看看！ 哦 原来这个孩子是有特殊交代的

这个孩子得宽 300  高200！  嗯  再多吃点！ 哟，衣服的颜色也要不一样的，要求还真多，好吧，给你！ 一件黑色的衣服套在第二个div身上了

再往下走...

于是就得到我们看到的画面了。

再来把上一关中我们刚见面的小伙伴 P 标签拿过来看看看，不是说选中所有的div标签么，放一个P标签进去，看看会不会被选中呢？

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px;

            height:200px;

            background: #932;

        }

    </style>

</head>

<body>

<div></div>

<div style="width: 300px;background:#200;"></div>

<p>这个是P标签</p>

<div></div>

</body>

</html>

可以看到，页面上只是多了一行文字，也就代表了，div{} 这个选择器并不会对 P标签起作用，那想要给P标签加样式怎么办呢？ 你来试试？

在上面的代码中，给p标签加一个背景颜色，色值为： #30c88d;

然后注意代码中的一个细节，在style标签里面写css样式的时候，每一个属性都单独写一行，所以你以后也要这样写哦~   这是业界通用的格式，所以啦！

现在我们知道两个选择器了，一个是 div{}  它可以选择所有的div标签，  一个是p{}  它可以选择所有的 p 标签。这一类的选择器，我们可以称之为标签选择器

A14

不过如果现在有这样的一个需求：

写出5个盒子，宽度分别是  100px  200px 300px 400px 500px 高度随意，然后背景颜色必须不一样，并且还要给前 3 个盒子加上边框！

哇~  好简单，不就5个div么，写行间样式呗…

不许用行间样式！！！

啥？ 那怎么搞？  5个div  那用 div{} 这个选择器也区分不了谁是谁啊  那怎么办呢？

回想一下刚才我们类比了一下在操场上的情况   现在div是男生，p是女生，男女区分开了，那男生之间怎么区分？

体育老师大手往下一指，那谁谁谁，去办公室帮我把体育器材拿出来！ 在没有体育委员的情况下会有人动么？别开玩笑了，鬼知道你说的那谁谁是谁啊！你自己不会去啊  对吧。所以这时候就需要一个名字：

小K !  去帮我拿一下体育器材，跑步前进！

在这里也是一样的，既然现在有一堆的div区分不开，那么我们就给它们分别起个名字咯！ 有名字就能区分开了嘛，起名字：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px;

            height:200px;

            background: #932;

        }

    </style>

</head>

<body>

<div></div>

<div class="box2"></div>

<div></div>

</body>

</html>

现在，我就给第二个div起了个名字，叫做 box2 。虽然这名字起得确实不咋地，但是名字只是个代号，怎么起名字才有艺术感我们以后再讨论。现在第二个div和其他两个div就能够区分开了  那就是在div的身上写一个class属性，后面跟上的就是它的名字了

名字是有了，怎么利用这个名字把样式加上去呢？

往下看

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        .box2{

            width: 100px;

            height:200px;

            background: #932;

        }

    </style>

</head>

<body>

<div></div>

<div class="box2"></div>

<div></div>

</body>

</html>

这样，页面上就只有一个盒子了，从上往下读一遍代码：

到style这，电脑知道： 哦  得给一个名字叫做 box2的孩子 注意box2的前面有一个点，这个点不是我写错了，这个点代表的意思就是class名字  如果没有这个点就变成标签选择器了，这里也是有区分的。牢记！敲黑板！！！

A15

电脑：哦，得给一个名字叫做 box2 的孩子加样式，宽100  高200  背景颜色…

然后再往下，碰到第一个div  这孩子没有名字，一边去，我要找 box2 !

碰到第二个div ,  拉过来仔细看了一下  class=“box2” 哦 这孩子就叫box2  来，给你吃的，长胖点…

再往下，嗯  好像没有叫 box2 的孩子了，完事！ 收工！

这样我们就能够单独为一个标签添加样式了  你来试试吧~

请以此为下面三个div单独添加一些样式  随你喜欢哦！

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

    </style>

</head>

<body>

<div></div>

<div></div>

<div></div>

</body>

</html>

现在，我们可以给一个元素起名字，并且给它单独加样式了，像上面这种  .box2{}  这样的选择器，我们叫他“类”选择器  因为叫这个名字的元素可以是一个，也可以是多个，所以不管它是多少个，我们都把它区分为一类，所以就叫做类选择器了，通俗一点，也可以叫做class选择器。因为起名字的时候，就是用到了class这个属性。

不过这也会带来另外一个问题，一个元素，一个div，给它起个名字，叫做 box 。 那这时候 div{} 这个选择器对它还能不能起作用呢？

A 能

B 不能

很明显是可以的嘛，难道因为你有了一个名字，你就不再属于男生或者女生这个分类了？  一个人是可以拥有多种身份的嘛  现在这个div就拥有了两重身份，首先它是 div  所以 div{} 这个选择器肯定是能对它起作用的，  同时它又 是 box 。 所以 .box{} 这个选择器对它也是起作用的。那这时候问题来了，两个选择器都能选中这个元素，听谁的呢？看下面的案例：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        div{

            width: 100px;

            height: 100px;

            background: red;

        }

        .box{

            width: 100px;

            height: 200px;

            background: blue;

        }

    </style>

</head>

<body>

<div class="box"></div>

</body>

</html>

现在很明显，style标签中的两个选择器都能够选中 下面的div  那 div 为什么是蓝色呢？我们来解读一下代码

到style标签这

电脑： 哦  一会要给所有的 div 加上这些样式，

然后，嗯？还要给box这孩子加上这些样式？ 好的好的

然后往下走，碰到div了，然后给它加样式，不过在加的过程中发现，哎，这孩子不就叫box么？ 给它也要加样式来着？ 这时候电脑犯难了，这两套衣服都是给这孩子的，但是孩子只能穿一套啊！电脑是没有判断力的，如果没有人为干预，那它在这就死机了，因为它不知道该怎么办，我们平常碰到电脑死机就是这么个情况

所以怎么办呢？ Css语言的创作者支招了：class能选择的范围明显比div要小一些，所以，当碰到这种情况的时候  优先听从 class选择器的！！ 这时候电脑就开心了，电脑是个选择器恐惧症最晚期患者，所以以后我们在写代码的时候千万不要让电脑去做选择，你敢让它选，它就敢死机给你看！！

所以在这里有一个规则，class > div  什么意思呢，就是当一个元素，同时被class选择器选中，同时又被标签选择器所选中的时候，优先听从 class 的

这就是所谓的 选择器的权重了！ 关于选择器的权重还有一些规定： style > ID > class > div

这里的style指的就是我们之前写在标签的style属性里面的样式，也叫做行间样式

然后是 ID

我们都知道，我们自己不仅有名字，而且有身份证号，名字可能会重复，但是身份证号是不会重复的，身份证号也就是我们的ID了，那么标签也可以有一个ID。

既然叫做ID  那么也就代表了当你给这个标签用了这个ID ， 那么别的标签就不能再使用这个ID了

ID的使用方法和class类似，在标签的ID属性里定义一个名字，那么这个名字就是这个标签的ID了  而ID选择器的前面是 “#”

来看这个案例：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        #mark{

            width: 100px;

            height: 100px;

            background: black;

        }

        .box{

            background: red;

        }

        div{

            background: blue;

        }

    </style>

</head>

<body>

<div class="box" id="mark" style="background:yellow;"></div>

</body>

</html>

在这个案例中，div的颜色是？

    A  yellow

    B  red

    C  black

    D  blue

A: 很棒哦！一下就记住了上面的公式呢

B: 还差一点，记住上面的公式哦

C: 还差一点，记住上面的公式哦

D: 还 差一点，记住上面的公式哦

根据上面的公式呢，style是最大的，所以很明显，div的颜色应该是 yellow

如果把style属性去掉，则是ID选择器里的 black .  style标签最上面那个就是ID选择器啦

再后面是 class选择器，再然后是 标签选择器

其实这里是有一定规律的  掌握了这个规律之后就比较好判断到底会听谁的了 ： 选择器能选中的元素越少，则它的优先级越高

Style肯定只能对当前这个标签起作用，所以它的优先级是最高的

ID选择器只能选择一个，要么这个，要么那个，它还算是有一定的自由，但是也只能选择一个，所以它的优先级在style之后

class选择器可以选择多个标签，不过明显没有标签选择器那么多，所以它排在第3位

最后则是标签选择器了

不过关于class选择器还有一个特点，那就是： 一个元素可以同时拥有多个class

还是拿我们人类作为类比，比如李白，他的大名是李白，字太白，号青莲居士。这么多个名字都是他

而标签也是一样，可以叫 box 同时也可以叫 smallK …

看下面的案例：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <style>

        .box{

            width: 100px;

            height: 100px;

            background: red;

        }

        .small{

            background: blue;

        }

    </style>

</head>

<body>

<div class="box small"></div>

</body>

</html>

在这个案例中，div拥有两个名字，一个叫box 一个是small  中间用空格分开就可以了。甚至它可以拥有很多个名字。

再仔细看上面的css代码，那大家说现在这个div会是什么颜色呢？

两个都是 class选择器，并没有优先级的差别，所以这时候有一个原则，叫后者覆盖前者

什么是后者呢？ 就是css写在后面的会覆盖前面的代码 我们来读一遍代码就知道了：

代码都是从上到下来读取的，所以电脑在读到 .box{} 这个选择器的时候，知道要给 box这个孩子穿一件红色的衣服。

再往后读到small 的时候，哦  要给这孩子穿一件蓝色的衣服。你里面穿了一件红色的衣服，外面穿了一件蓝色的衣服，最终你的衣服颜色是什么颜色？

肯定是蓝色嘛，所以在这里也是一样，最终div的颜色就是蓝色啦

如果在css代码里，把box 和 small 换一个位置，最终div是什么颜色？

A: blue

B: red

A：罚你返回去再看一遍

B:  真棒！奖励你一朵小红花！  A20

现在，我们的css代码有单独的地方可以写了，然后也有选择器可以单独对某一个或多个标签添加样式了，我们来试着做个项目吧…

A16

可以看到，这是一款社交软件的几个设置界面，而且长得还很像。那如果说用我们现在的知识点来写的话…  困难还是有的，不过你可以尝试去写哦~

切作为附赠给你的小福利吧！哈哈

我也写了，只用我们当前学到的知识点去写，当然这只是个尝试，不过在写的过程中我发现了一件事情，那就是这几个页面几乎都长一样，这导致很多样式需要从这个页面复制到另一个页面里去。

感觉这个情况和当初我们一个页面中有多个相同的盒子差不多~

你说会不会有一种方法，可以让我们把样式再次找个更加公共的地方放起来，然后多个页面都能用呢？

别说，还真有！ 容我想想哈

其实啊，css是可以用单独的一个文件来编写的

我们可以在编辑器里面点击新建，然后选择css文件，以我用为webstorm为例：大概是长这样子的:

A17

之前我们说过了css 的全称是 cascading style sheets  所以右边选择的文件类型是 styleSheet 点开之后：

A18

点开之后选择css File就可以了 不过在上面的输入框里必须要先输入文件的名称才行哦！ 这个文件的名字可以叫style  也可以叫其他的名字，只要不是中文名字就可以了~

哎，其实我也多希望可以用中文命名啊  但是代码是人家老外发明的，用中文命名可能会导致文件加载错误，所以加油咯！ 写代码的过程中记点单词就好了，我相信你是最棒（pang）的！

文件新建好了之后现在我们的整个文件夹应该是这个样子的：

A19

也就是找个地方新建一个文件夹，然后在里面新建一个html文件以及一个css文件

重点：文件的管理一定要做好，最好是每一关都新建一个文件夹，或者每一天都新建一个文件夹。然后把当前的所有东西都放在里面，在里面每一个练习再建立一个文件夹！

把文件新建好之后我们就可以来愉快的敲代码啦！

现在我们有一个html文件和一个css文件了，css文件里面默认是空的，我们可以在里面写选择器，然后把css文件和html文件关联上就可以了 比如css文件里写：

.box{

    width: 200px;

    height: 200px;

    background: #000;

}

除此以外没有任何东西了

然后 html代码：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

</head>

<body>

<div class="box"></div>

</body>

</html>

哎，恰好，html文件里面就有一个叫做box的div   那么现在要怎么样把css和html文件关联上呢？

方法就是在html文件里面放一个标签：link

Link也就是链接的意思，link标签是专门用于关联文件用的  它完整的写法是这样的：

<link rel="stylesheet" href="style.css”>

引入之后：

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Title</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

<div class="box"></div>

</body>

</html>

注意：我们之前接触的标签都是双标签，有一个开头，有一个结尾，而link标签是一个单标签，也就是说它没有单独的结尾部分，它的结尾部分被省略了 千万不要在后面加上 </link> 哦

然后里面有两个属性  一个 rel  一个 href  ：

Ref的作用是告诉html 关联进来的文件是个什么文件：很明显，是一个stylesheet

Href 的作用和我们之前在background-image里面的 url 比较类似，都是用于指定文件的路径  当文件引入好了之后，效果就和之前在style标签里写css样式是一样的了

然后其他的html文件需要用到这个css文件的之后直接用link标签引入就可以了  是不是很方便呢？

现在，总结一下这一关中我们学到的知识点吧：

选择器：

    标签选择器

类选择器

ID选择器

Css引入方式：

    写在style属性里    这种方式也叫做行间样式

    写在style标签里    这种方式也叫做内联样式

    写在css文件里      这种方式也叫做外联样式

现在，试着用这一关学到的知识点来完成下面的练习吧~

下一关我们将会学习更多关于文字的知识点哦！ 下一关见…