

CSS的继承性和层叠性(四)

一. 继承性

继承：有一些属性，当给自己设置的时候，自己的后代都继承上了，这便是继承性。继承性从自己开始，直到最小的元素。

继承的属性有:color、text-开头的、line-开头的、font-开头的、文字样式的，都能继承,而所有关于盒子的、定位的、布局的属性都不能继承。

二. 层叠性

层叠性:就是css处理冲突的能力。所有的权重计算，没有任何兼容问题

2.1 当选择器，选择上了某个元素的时候，那么要这样统计权重

统计权重的顺序:id的数量,类的数量 ,标签的数量

```
<title>title</title>
<style type="text/css">
div{
  font-size: 20px;
}
/*id的数量,类的数量 ,标签的数量*/
/*1,1,1*/
#box1.whatcolor1 p{
  color: red;
}
/*1,0,3*/
div div #box3 p{
  color: yellow;
}
/*0,3,4*/
div.whatcolor1 div.whatcolor2 div.whatcolor3 p{
  color: blue;
}
</style>

</head>

<body>
  <div class="whatcolor1" id="box1">
    <div class="whatcolor2" id="box2">
      <div class="whatcolor3" id="box3">
        <p>我是啥颜色</p>
      </div>
    </div>
  </div>
</body>
```

最终显示为红色

Snip20161108_11.png

2.2 如果权重一样，那么写在后面的为准

```

22      /*}*/
23      #box2 div.whatcolor3{
24          color: blue;
25      }
26      #box2 .whatcolor3 p{
27          color: red;
28      }
29  </style>
30
31  </head>
32
33  <body>
34
35
36      <div class="whatcolor1" id="box1">
37          <div class="whatcolor2" id="box2">
38              <div class="whatcolor3" id="box3">
39                  <p>我是啥颜色</p>
40              </div>
41          </div>
42
43      </div>

```

Snip20161108_12.png

2.3 如果不能直接选中某个元素，通过继承性影响的话，那么权重为0

```

#box1 #box2 #box3{
    color: red;
}

div.whatcolor div.whatcolor div.whatcolor{
    color: green;
}
p{
    color: orange;
}
</style>
</head>
<body>

<div class="whatcolor" id="box1">
    <div class="whatcolor" id="box2">
        <div class="whatcolor" id="box3">
            <p>我是啥颜色</p>
        </div>
    </div>
</div>

```

开始数权重之前，
一定要看看有没有
选中文字所在的最
内层标签

→
0

→
0

→
0,0,1

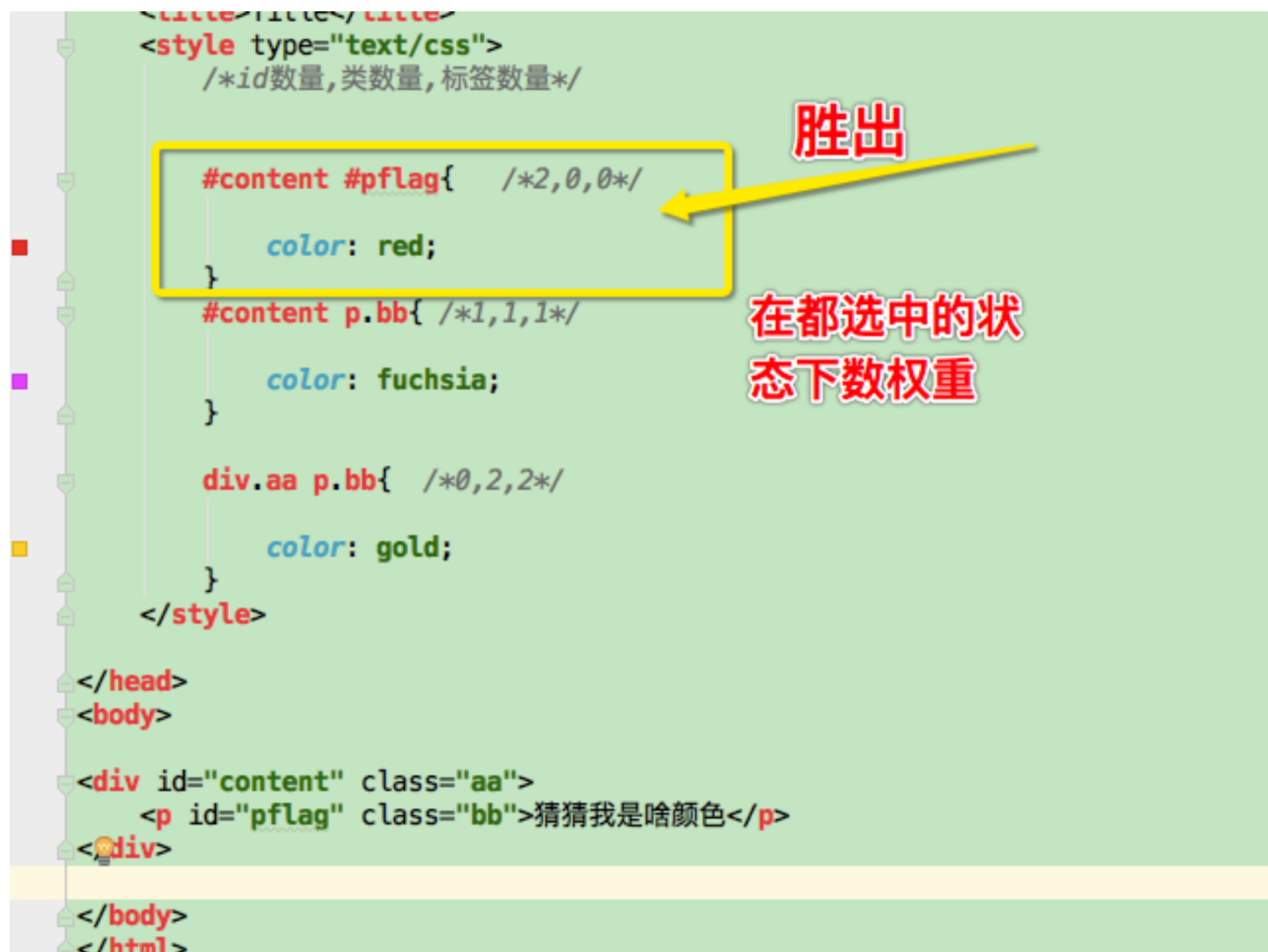
Snip20161108_13.png

2.4 如果大家都是0,则就近原则(谁描述近，就听谁的)



Snip20161108_14.png

大杂烩



Snip20161109_1.png

```
6      <style type="text/css">
7          /*id数量,类数量,标签数量*/
8
9          div div{ /*0,0.,2*/
10             color: gold;
11
12         }
13         div{ /*0,0,1*/
14             color: fuchsia;
15
16         }
17
18     </style>
19
20 </head>
21 <body>
22
23 <div>
24     <div>
25         <div>
26             猜猜我是啥颜色
27         </div>
28     </div>
29 </div>
30
31 </body>
```

看谁离的更加近

Snip20161109_2.png

```
<!-- lang= en -->
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <style type="text/css">
    /*id数量,类数量,标签数量*/

    /*选择上的是#content1中所有后代div,所以选择上了最内存的div,并不是通过继承影响*/
    #content1 div{ /*1,0,1*/
      color: fuchsia;
    }

    /*选择上了最内层div,也是有权重的*/
    #content3{ /*1,0,0*/
      color: forestgreen;
    }

  </style>
</head>
<body>

<div id="content1" content="c1">
  <div id="content2" content="c2">
    <div id="content3" content="c3">
      猜猜我是啥颜色
    </div>
  </div>
</div>
</body>
```

Snip20161109_3.png

额外补充关于权重

1.同一标签，携带了多个类名，有冲突

```
<p class="cls1 cls2">我是啥颜色???
```

```
</p>  
<p class="cls2 cls1">我是啥颜色???
```

与结构中的顺序无关，而与css的顺序有关

```
.cls1{  
    color:red;  
}  
.cls2{  
color:yellow;  
}
```

结果是yellow.因为yellow写在后面

2.important(重要的)标记

!important来给一个属性提交权重，这个属性的权重就是无穷大。语

法:k:v!important



Snip20161109_4.png

!important提升的是一个属性，而不是一个选择器

!important无法提升继承的权重，该是0还是0

HTML结构

```
<div>
  <p> 猜猜我是啥颜色</p>
</div>
```

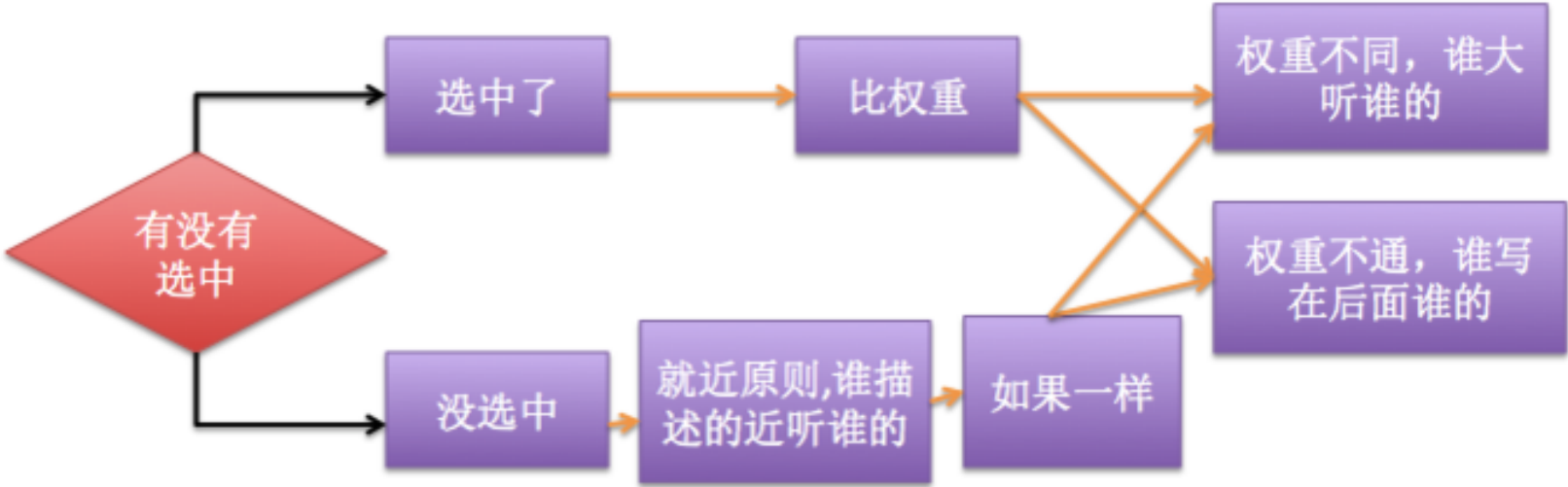
css样式

```
div{
  color: red!important;
}
p{
  color: gold;
}
```


最终显示的颜色为gold,因为!important无法提升它的权重， 因为p标签已经选中了,故显示颜色为gold

!important不影响就近原则

3.权重计算图例总结



权重计算图例

权重总结

- 1>先看有没有选中，如果选中，则以(id数量，类数量，标签数量)来计算权重，谁大听谁的，如果都一样，听后写的为准。
- 2>如果都没有选中，那么权重为0,如果大家都是0，就近原则。