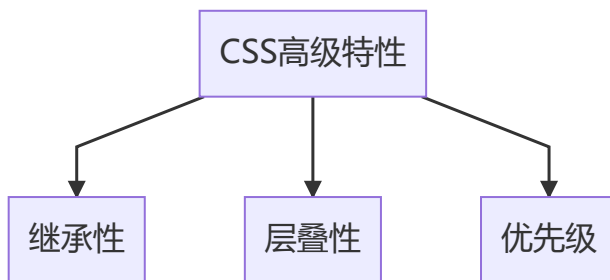


CSS高级特性

之前的时候我们已经学过了CSS的3个基本点分别是选择器的便捷性，CSS属性的统一性，以及CSS与HTML代码的分离性，这三个特点已经解决了我们平常开发过程中的很多样式问题，但是仍然有一些问题无法实现，这个时候就需要借助于我们CSS的高级特性

CSS的中文名称叫“层叠样式表”



继承性

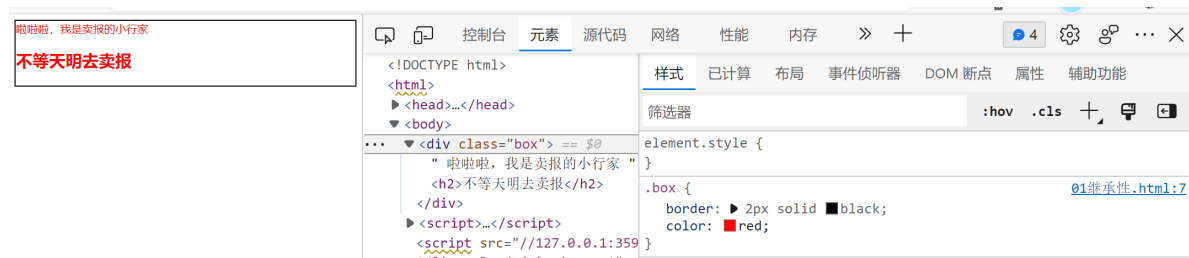
继承性指的是内部元素默认继承外部元素的样式

怎么理解继承？王思聪和王健林是父子关系，王思聪就可以继承王健林的财产

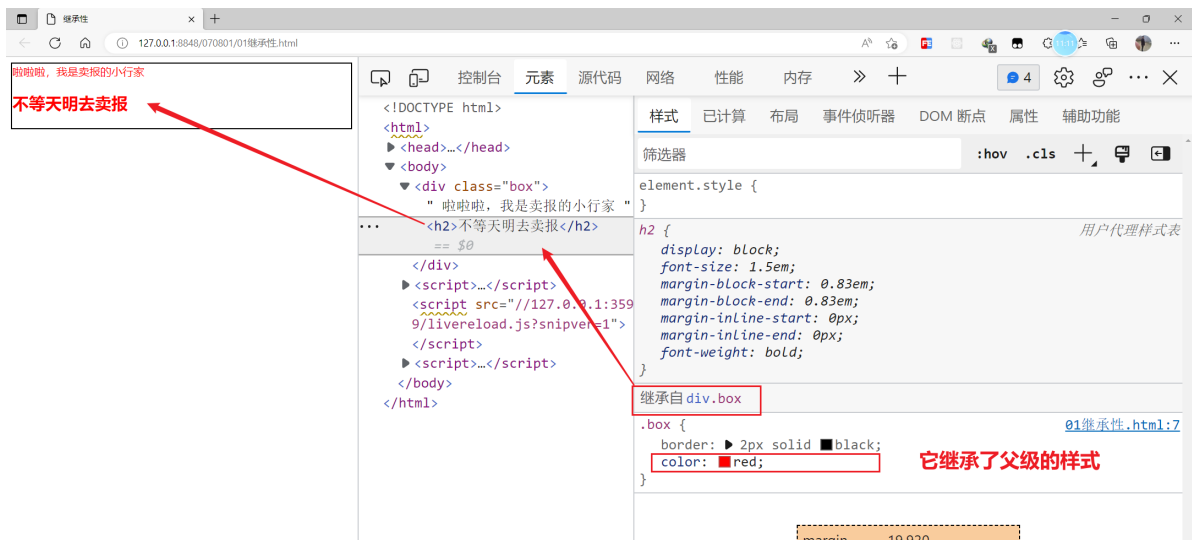
通俗意义上来说，父级里面有的东西，子级会默认继承下来

```
1 <style>
2   .box{
3     border: 2px solid black;
4     color: red;
5   }
6 </style>
7 <div class="box">
8   啦啦啦，我是卖报的小行家
9   <h2>不等天明去卖报</h2>
10 </div>
```

在上面的代码里面，我们明明只是把 `box` 设置为了红色，为什么 `h2` 标签也会变成红色呢？



上面是 `box` 的样式，我们在浏览器里可以看到，它设置成了红色，现在我们再去看内部的 `h2` 标签



通过上面的图片我们可以得到结果，因为 `h2` 标签继承了父级元素 `box` 的，所以它会有这个红色的文字样式，这个特点就是CSS里面的继承性

在上面图片里面，我们也可以看到一个特点，`color` 这个属性是被继承下来了，而 `border` 没有被继承，这是因为能够默认被继承是样式属性，而布局属性是不能被继承的

其它我们还可以通过一个特殊的CSS属性值来**主动的继承**

```
1 h2{
2     /* 我想让我的边框与父级元素的边框一模一样 */
3     border: inherit;
4 }
```

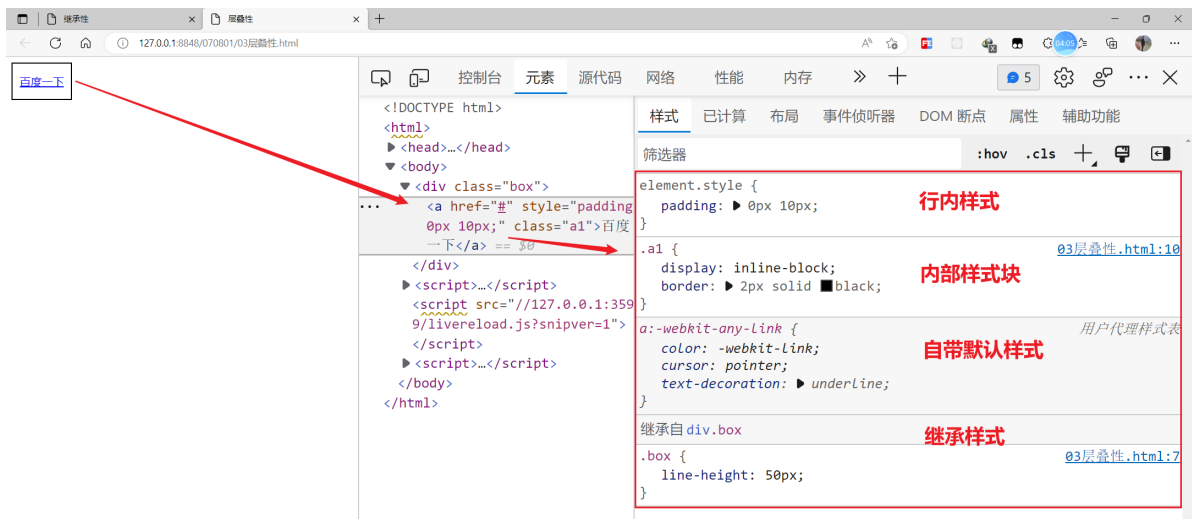
层叠性

层叠性的本意是指CSS的样式应该是由多个部分共同组成的

```
1 <style>
2     .box{
3         line-height: 50px;
4     }
5     .a1{
6         display: inline-block;
7         border: 2px solid black;
8     }
9 </style>
10 <div class="box">
11     <a href="#" style="padding: 0px 10px;" class="a1">百度一下</a>
12 </div>
```

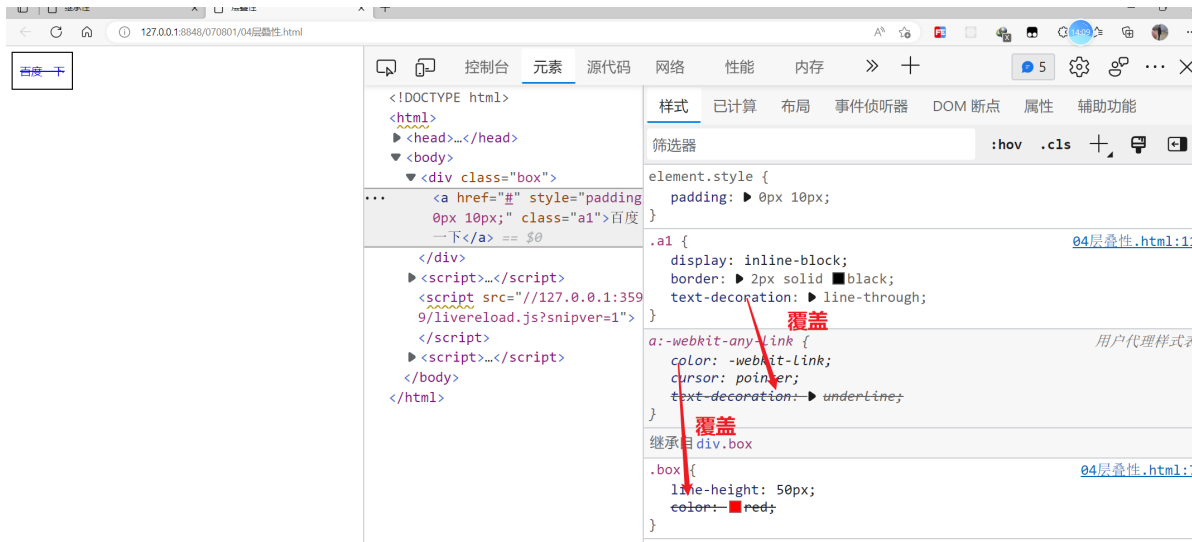
百度一下

我们现在想知道为什么 `a` 标签展示出来的效果是这样的？



上面图片我们可以看到，层叠性指的是元素最终的表现形式是由 **继承样式** + **自带的默认样式** + **内部样式块** + **行内样式** 来共同决定的

层叠性除了决定样式的最终表现形式外，它还可以让我们看到如果样式冲突了，它是怎么解决了，请看下面的效果图



当样式出现冲突的时候，我们就可以看到，谁层叠在了上面，就听谁的

行内样式 > 内部样式块 > 默认样式 > 继承样式

优先级

优先级是为了解决在同一层里面出现样式冲突以后的问题，优先级一般是指选择器的优先级

选择器相同的时候

```
1 .box{
2     color:red;
3     border:1px solid black;
4     color:yellow;
5 }
```

在上面的代码里面，如果选择器相同的时候样式冲突了，我们可以找最后一次出现的样式就可以了，所以 **yellow** 会覆盖 **red**

选择器不相同的时候

在我们的开发当中，这种情况是最常见的，我们有很多个选择器，我们可以使用不同的方式来选择同一个元素，这个时候 如果出现了样式冲突，我们应该计算一权重值再来做比较

选择器的权重值

选择器	权重值
星号选择器	0, 0, 0, 0
标签选择器, 伪元素	0, 0, 0, 1
类选择器, 属性选择器, 伪类	0, 0, 1, 0
ID选择器	0, 1, 0, 0
style属性	1, 0, 0, 0
!important	无穷大

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8">
5          <title>选择器的权重值计算</title>
6          <style>
7              /* 1 */
8              p{
9                  color: red !important;
10             }
11             /* 1+1=2 */
12             fieldset>p{
13                 color: blue;
14             }
15             /* 10 */
16             .p1{
17                 color: yellow;
18             }
19             /* 1+10=11 */
20             p.p1{
21                 color: pink;
22             }
23             /* 1+1+10=12 */
24             fieldset>p.p1{
25                 color: green;
26             }
27             /* 10+10=20 */
28             .f1>.p1{
29                 color: black;
30             }
31             /* 10+1+10=21 */
32             .f1>p.p1{
33                 color: red;
34             }
35             /* 1+10+1+10=22 */
36             fieldset.f1>p.p1{
37                 color: yellow;
```

```

38         }
39         /* 100 */
40         #aaa{
41             color: blue;
42         }
43         /* 1+100=101 */
44         p#aaa{
45             color: red;
46         }
47         /* 10+100=110 */
48         .p1#aaa{
49             color: green;
50         }
51     </style>
52 </head>
53 <body>
54     <fieldset class="f1">
55         <p class="p1" id="aaa" style="color: blue;">震惊，日本前首相安倍晋三突遭不
幸福</p>
56     </fieldset>
57 </body>
58 </html>

```

上面的代码就是用来展示选择器的权重值的计算过程，但有一个根本原则不能违背

style>ID>class>标签 这是一个总体原则，官方的说法就是选择器权重不进位

```

1     <style>
2         /* 10 */
3         .div1{
4             color: blue;
5         }
6         /* 11 选择器权重不进位 */
7         div>div>div>div>div>div>div>div>div>div{
8             color: red;
9         }
10
11 </style>
12 <div>
13     <div>
14         <div>
15             <div>
16                 <div>
17                     <div>
18                         <div>
19                             <div>
20                                 <div>
21                                     <div>
22                                         <div class="div1">我是第11个</div>
23                                     </div>
24                                 </div>
25                             </div>
26                         </div>
27                     </div>
28                 </div>
29             </div>
30         </div>
31     </div>

```

在上面的代码里面，我们可以看到最终显示的效果是 `blue`，因为标签选择器无论怎么叠加，都不能大于 `class` 选择器，也就是值不能进位