

0.概述



1.设置css样式/读取css值

1.1 css()

css()操作的是style属性

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>11_css</title>
6 </head>
7 <body>
8   <p style="color: blue;">尚硅谷的后裔</p>
9   <p style="color: green;">太阳的后裔</p>
10
11 <!--
12 设置css样式/读取css值
13   css()
14 -->
15 <script src="../../课件/jQuery库/jquery-1.10.1.js" type="text/javascript">
16 <script type="text/javascript">
17   /*
```

```

18     1. 得到第一个p标签的颜色
19     2. 设置所有p标签的文本颜色为red
20     3. 设置第2个p的字体颜色(#ff0011),背景(blue),宽(300px), 高(30px)
21     */
22     // 1. 得到第一个p标签的颜色
23     console.log($('p:first').css('color'))      //'color' 属性名
24     // 2. 设置所有p标签的文本颜色为red
25     $('p').css('color','red')
26     // 3. 设置第2个p的字体颜色(#ff0011),背景(blue),宽(300px), 高(30px)
27     //:eq(1)   : 第2个
28
29     //法1
30     $('p:eq(1)').css({
31         color:'#ff0011',
32         background:'blue',
33         width : 300,
34         height :30
35     })
36
37     //法2
38     //p.style.color='#ff0011'
39
40
41 </script>
42 </body>
43 </html>
44

```

2.获取/设置标签的位置数据

2.1offset()

相对页面左上角的坐标

offset([coordinates])

返回值:Object{top,left}

概述

获取匹配元素在当前视口的相对偏移。

返回的对象包含两个整型属性: top 和 left。此方法只对可见元素有效。

2.2position()

相对于父元素左上角的坐标

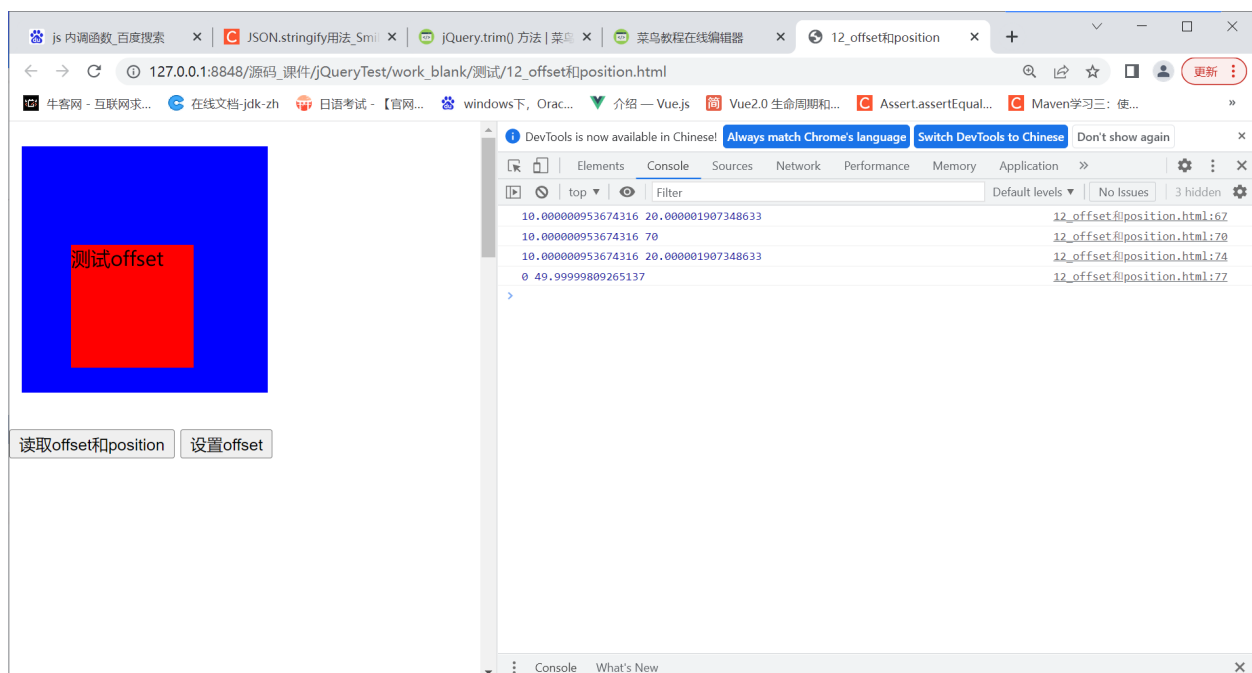
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>12_offset和position</title>
6 </head>
7 <style type="text/css">
8   * {
9     margin: 0px;
10  }
11
12  .div1 {
13    position: absolute;
14    width: 200px;
15    height: 200px;
16    top: 20px;
17    left: 10px;
18    background: blue;
19  }
20
21  .div2 {
22    position: absolute;
23    width: 100px;
24    height: 100px;
25    top: 50px;
26    background: red;
27  }
28
29  .div3 {
30    position: absolute;
31    top: 250px;
32  }
33 </style>
34 <body style="height: 2000px;">
35
36 <div class="div1">
37   <div class="div2">测试offset</div>
38 </div>
39
40 <div class='div3'>
```

```
41 <button id="btn1">读取offset和position</button>
42 <button id="btn2">设置offset</button>
43 </div>
44
45 <!--
46 获取/设置标签的位置数据
47  * offset(): 相对页面左上角的坐标
48  * position(): 相对于父元素左上角的坐标
49 -->
50 <script src="../../课件/jQuery库/jquery-1.10.1.js" type="text/javascript">
51 <script type="text/javascript">
52  /*
53  需求:
54  1. 点击 btn1
55      打印 div1 相对于页面左上角的位置
56      打印 div2 相对于页面左上角的位置
57      打印 div1 相对于父元素左上角的位置
58      打印 div2 相对于父元素左上角的位置
59  2. 点击 btn2
60      设置 div2 相对于页面的左上角的位置
61  */
62
63  //1. 点击 btn1
64  $('#btn1').click(function () {
65      // 打印 div1 相对于页面左上角的位置
66      var offset = $('.div1').offset()
67      console.log(offset.left,offset.top)    //10 20
68      //打印 div2 相对于页面左上角的位置
69      offset = $('.div2').offset()
70      console.log(offset.left,offset.top)    //10 70
71
72      //打印 div1 相对于父元素左上角的位置
73      var position = $('.div1').position()
74      console.log(position.left,position.top)    //10 20
75      // 打印 div2 相对于父元素左上角的位置
76      position = $('.div2').position()
77      console.log(position.left,position.top)    //0 50
78
79
80  })
81
82  // 2. 点击 btn2
83  // 设置 div2 相对于页面的左上角的位置
```

```

84 $('#btn2').click(function () {
85     $('.div2').offset({
86         top: 100,
87         left: 50
88     })
89 })
90
91 </script>
92 </body>
93 </html>
94

```



2.3scrollTop()

- scrollTop():

读取/设置滚动条的Y坐标

- \$(**document.body**).scrollTop()+\$(**document.documentElement**).scrollTop()

读取页面滚动条的Y坐标(兼容chrome和IE)

- \$('body,html').scrollTop(60);

滚动到指定位置(兼容chrome和IE)

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="UTF-8">
6   <title>13_元素滚动</title>
7 </head>
8 <body style="height: 2000px;">
9 <div style="border:1px solid black;width:100px;height:150px;overflow:auto">
10   This is some text. This is some text. This is some text. This is some text.
11   This is some text. This is some text. This is some text. This is some text.
12   This is some text. This is some text. This is some text. This is some text.
13   This is some text. This is some text. This is some text. This is some text.
14   This is some text. This is some text. This is some text. This is some text.
15   This is some text. This is some text. This is some text. This is some text.
16   his is some text.
17 </div>
18 <br>
19 <br>
20 <br>
21 <button id="btn1">得到scrollTop</button>
22 <button id="btn2">设置scrollTop</button>
23
24 <!--
25 1. scrollTop():
26   读取/设置滚动条的Y坐标
27 2. $(document.body).scrollTop()+$(document.documentElement).scrollTop()
28   读取页面滚动条的Y坐标(兼容chrome和IE)
29 3. $('body,html').scrollTop(60);
30   滚动到指定位置(兼容chrome和IE)
31 -->
32 <script src="../../../../课件/jQuery库/jquery-1.10.1.js"></script>
33 <script>
34   /*
35    需求:
36    1. 得到div或页面滚动条的坐标
37    2. 让div或页面的滚动条滚动到指定位置
38   */
39
40   // 1. 得到div或页面滚动条的坐标
41   $('#btn1').click(function(){
42     //得到div滚动条的坐标
43     console.log($('div').scrollTop())
```

```

44 //得到页面滚动条的坐标 法1
45 //总是一个有值一个为0，所以相加
46 console.log($('html').scrollTop()+$('body').scrollTop())
47 /* 对于任何非空 HTML 文档，调用 document.documentElement
48 总是会返回一个 <html> 元素，
49 且它一定是该文档的根元素。借助这个只读属性，
50 能方便地获取到任意文档的根元素。
51 返回当前文档的body 元素 语法格式：document.body */
52
53 //得到页面滚动条的坐标 法2（兼容多个浏览器）
54 console.log($(document.documentElement).scrollTop()+$(document.body).sc
55
56
57 })
58
59 // 2. 让div或页面的滚动条滚动到指定位置
60 $('#btn2').click(function(){
61
62     $('div').scrollTop(200)
63     $('html,body').scrollTop(150) //并集 $('html,body')
64
65 })
66
67 </script>
68 </body>
69
70 </html>

```

2.4 scrollLeft()

2.5补充

- 滚动类特效常用属性：

scrollLeft、scrollTop：设置或获取位于对象最顶/左端和窗口中可见内容的最顶/左端之间的距离。即当前上滚或左滚的距离。

scrollHeight、scrollWidth：获取对象可滚动的总高度/宽度

offsetLeft、offsetTop：获取当前对象与父元素之间的距离（不包含父元素的边框）

offsetWidth、offsetHeight：获取元素自身的宽度/高度

clientLeft、clientTop：效果和边框宽度相同，很少使用

clientWidth、clientHeight：不含边框的元素自身的宽度/高度

- 关于滚动条：

在12种属性当中，存在如下两对：offsetHeight&clientHeight；

offsetWidth&clientWidth。它们的区别在于，offsetWidth和offsetHeight是包含元素边框以及边框内部内容的，而clientWidth和clientHeight是不含元素边框的，是从元素的padding区域开始计算的。

特意这么说，其实是希望各位注意，从边框开始计算的，也会把滚动条的宽度/高度计算到结果之内，谷歌浏览器中，滚动条的宽/高度是17px，如果希望能够获取一个元素的padding和内容，需使用**clientWidth和clientHeight**，才能保证没有误差。

3.获取/设置标签的尺寸数据

3.1 内容尺寸

- height(): height
- width(): width

3.2 内部尺寸

- innerHeight(): **height+padding**
- innerWidth(): **width+padding**

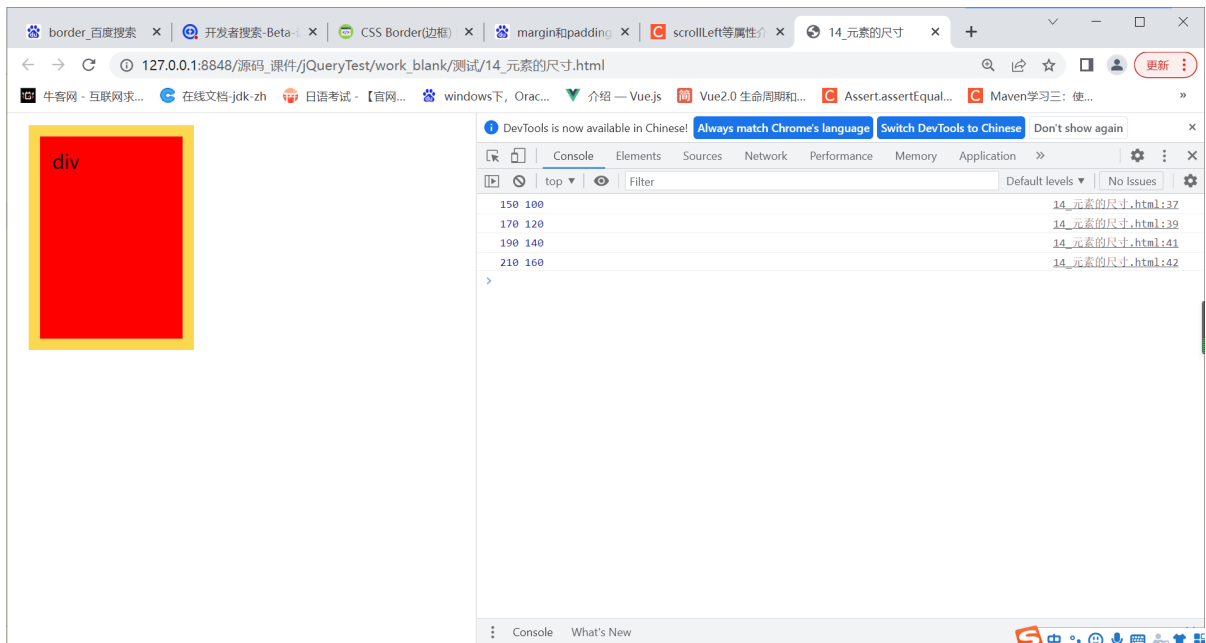
3.3 外部尺寸

- outerHeight(false/true): **height+padding+border 如果是true, 加上margin**
- outerWidth(false/true): **width+padding+border 如果是true, 加上margin**

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>14_元素的尺寸</title>
6 </head>
7 <style>
```



```
8   div {
9     width: 100px;
10    height: 150px;
11    background: red;
12    padding: 10px;
13    border: 10px #fbd850 solid;
14    margin: 10px;
15  }
16 </style>
17 </head>
18
19 <body>
20 <div>div</div>
21
22 <!--
23 1. 内容尺寸
24   height(): height
25   width(): width
26 2. 内部尺寸
27   innerHeight(): height+padding
28   innerWidth(): width+padding
29 3. 外部尺寸
30   outerHeight(false/true): height+padding+border 如果是true, 加上margin
31   outerWidth(false/true): width+padding+border 如果是true, 加上margin
32 -->
33 <script src="../../../../课件/jQuery库/jquery-1.10.1.js" type="text/javascript">
34 <script>
35   var $div = $('div')
36   // 1. 内容尺寸 红色部分
37   console.log($div.height(),$div.width()) //150 100
38   // 2. 内部尺寸 橙色包着的部分
39   console.log($div.innerHeight(),$div.innerWidth()) //120 170
40   // 3. 外部尺寸
41   console.log($div.outerHeight(),$div.outerWidth()) //140 190
42   console.log($div.outerHeight(true),$div.outerWidth(true)) //160 210
43
44 </script>
45 </body>
46
47 </html>
```



3.4 基础补充

<style>

div {

width: 100px;

height: 150px;

background: red;

padding: 10px;

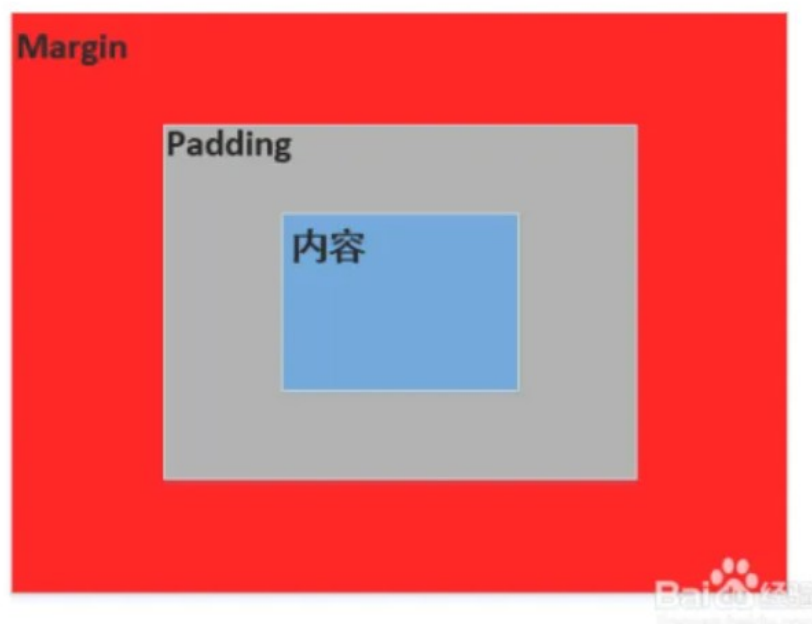
border: 10px #fbd850 solid; **边界**

margin: 10px;

}

</style>

红色方框的边界为margin; 灰色方框的边界为padding; 蓝色方框为内容。



3.4.1 margin

margin是组件与组件的间距。



<style>

div {

width: 300px;

height: 150px;

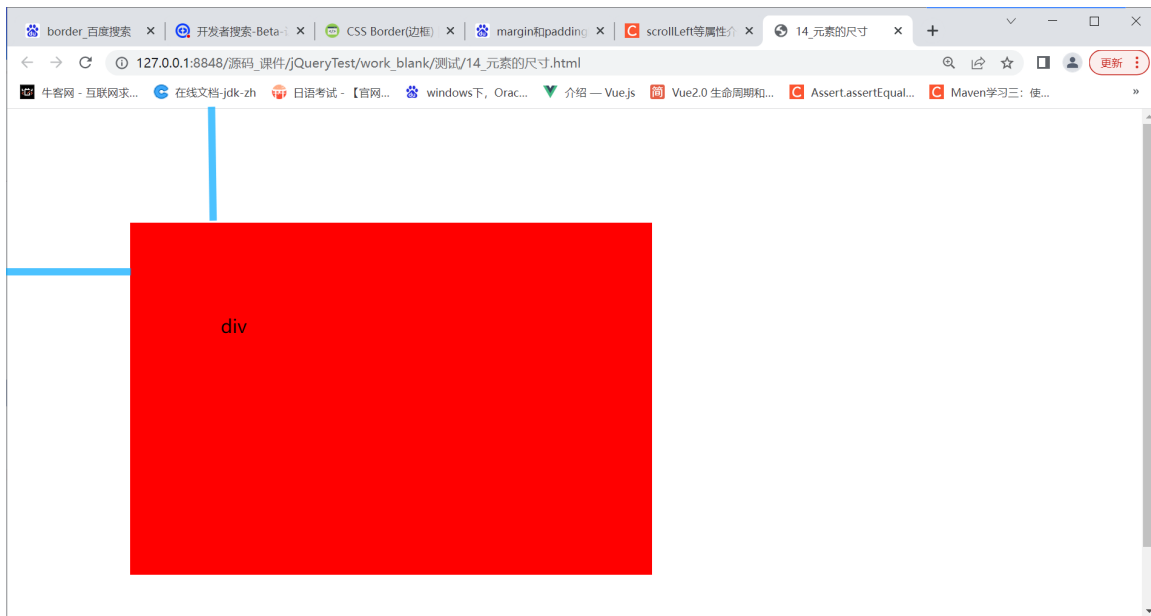
background: red;

padding: 80px;

margin: 100px;

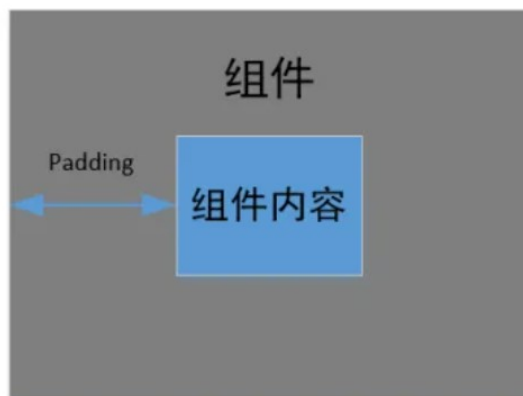
}

</style>

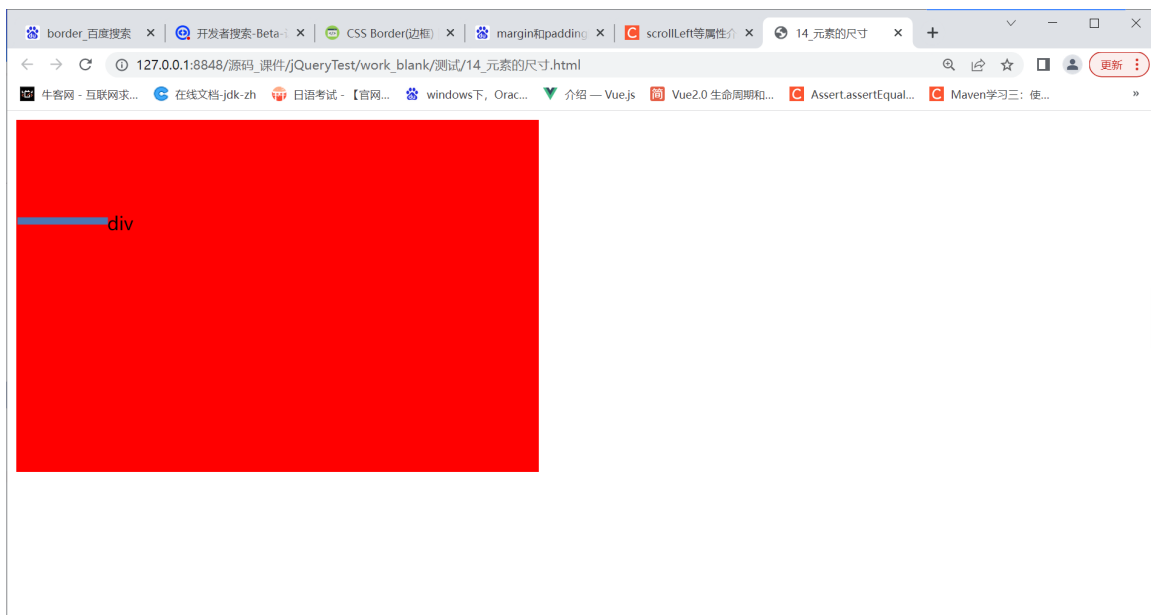


3.4.2 padding

padding是组件与组件内容的间距。



```
<style>
div {
  width: 300px;
  height: 150px;
  background: red;
  padding: 80px;
}
</style>
```



3.4.3 width

3.4.4 height

<style>

div {

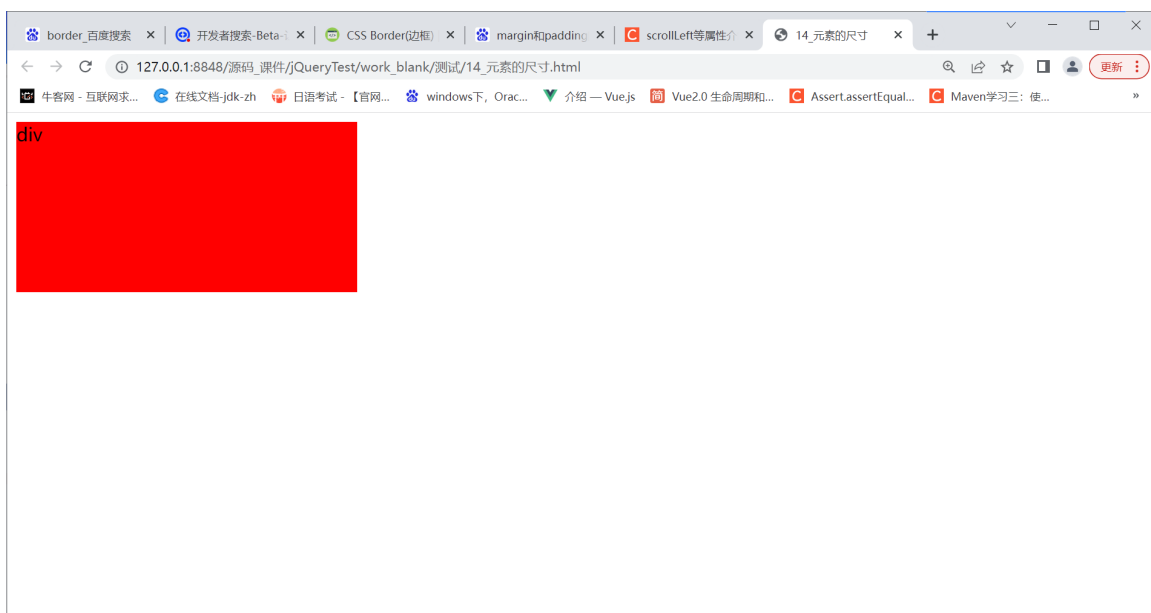
width: 300px;

height: 150px;

background: red;

}

</style>



3.4.5 border

```
<style>
```

```
div {
```

```
width: 300px;
```

```
height: 150px;
```

```
background: red;
```

```
padding: 10px;
```

```
border: 80px deeppink solid;
```

```
margin: 10px;
```

```
}
```

```
</style>
```

