

## Css定位

### 1.css定位

改变元素在页面上的位置

### 2.css定位机制

普通流:元素按照其在html中的位置顺序决定排布的过程.

浮动:

绝对布局:

### 3.css的定位属性

position:把元素放在一个静态的.相对的 绝对的 或者固定的位置中

top:元素向上的偏移量

left:元素向左的偏移量

right:元素向右的偏移量

bottom:元素向下的偏移量

overflow:设置元素一处其区域发生的事情

clip:设置元素显示的形状

vertical-align:设置元素垂直对齐方式

z-index:设置元素的堆叠顺序

## Css position属性

static

relative

absolute

fixed

```
1  html
2  <div id="position1"></div>
3  //这就是普通流
4  <script>
5      for(var i =0;i<100;i++){
6          document.write(i+"<br/>");
7      }
8  </script>
9
10  在style.css中
11
12  #position1{
13      width:100px;
14      height:100px;
15      background-color:red;
```

```
16 }
17
18 然后在style.css中加入
19 position:relative; //发现加入之后没有什么变化
20 需要设置偏移量
21 right:100px; //这个是距离右边100px
22 left:10px; //距离左边10px
23 top:10px; //距离上面10px
24
25
26 //换成绝对布局
27 position:absolute; 发现不占位置了
28
29 //固定,滑动页面的时候,始终在固定位置
30 position:fixed;
31
32 //静态,设置偏移量对他是不起任何效果的.对z-index也是不起任何效果的
33 position:static;
34
35
36 //现在有两个div
37 <div id="position1"></div>
38 <div id="position2"></div>
39
40 在style.css中
41 #position1{
42     width:100px;
43     height:100px;
44     background-color:blue;
45     position:relative;
46     left:20px;
47     top:20px;
48 }
49
50 #position2{
51     width:100px;
52     height:100px;
53     background-color:aqua;
54     position:relative;
55     left:10px;
56     top:10px;
57 }
58
59 现在的效果是后面的盖住了前面的
60 怎么更改次序呢??
61 在每个里面加上z-index
```

```
62
63 在position1中加入
64 z-index:2;
65 在position2中加入
66 z-index:1;
67 两个就会反过来了.值越大,呈现出来的越靠前.
68
69
```

## 浮动

float属性可用的值:

left:元素向左浮动

right:元素向右浮动

none:元素不浮动

inherit:从父级继承浮动属性

clear属性:

去掉浮动属性(包括继承来的属性)

clear属性值:

left.right:去掉元素向左向右浮动

both:左右两侧均去掉浮动

inherit:从父级继承来的clear的值

```
1  <div id="fd"></div>
2  <div id="sd"></div>
3  <div id="td"></div>
4
5  在style.css中
6  #fd{
7      width:100px;
8      height:100px;
9      background-color:red;
10 }
11 #sd{
12     width:100px;
13     height:100px;
14     background-color:blue;
15 }
16 #td{
17     width:100px;
18     height:100px;
19     background-color:green;
20 }
```

```
21
22 这样并不能满足我们的需求:
23 需要设置float属性
24 浮动的意思就是不占据位置,相当于向上浮动的感觉.往屏幕外浮动
25 #fd{
26     width:100px;
27     height:100px;
28     background-color:red;
29     float:left;
30 }
31 #sd{
32     width:100px;
33     height:100px;
34     background-color:blue;
35 }
36 #td{
37     width:100px;
38     height:100px;
39     background-color:green;
40 }
41
```

## 浮动的应用

```
1  index.html
2  <!DOCTYPE html>
3  <html>
4      <head>
5          <meta charset="UTF-8"/>
6          <title>浮动的应用</title>
7          <link rel="stylesheet" type="text/css" href="style.css"/>
8      </head>
9      <body>
10         <div id="div1">
11             <ul>
12                 <li></li>
13                 <li></li>
14                 <li></li>
15             </ul>
16             <ul>
17                 <li></li>
18                 <li></li>
19                 <li></li>
20             </ul>
21
```

```

22         </ul>
23         <ul>
24             <li></li>
25             <li></li>
26             <li></li>
27         </ul>
28     </div>
29 </body>
30 </html>
31
32
33 style.css
34 这里使用一个通配符*:代表所有的属性
35 *{
36     margin:0px;
37     padding:0px;
38
39 }
40 li{
41     list-style:none;
42 }
43
44 #div1{
45     width:950px;
46     height:auto;//代表自适应的
47     margin:20px auto;//上下 左右自适应
48
49 }
50
51 ul{
52     width:250px;
53     float:left;
54 }

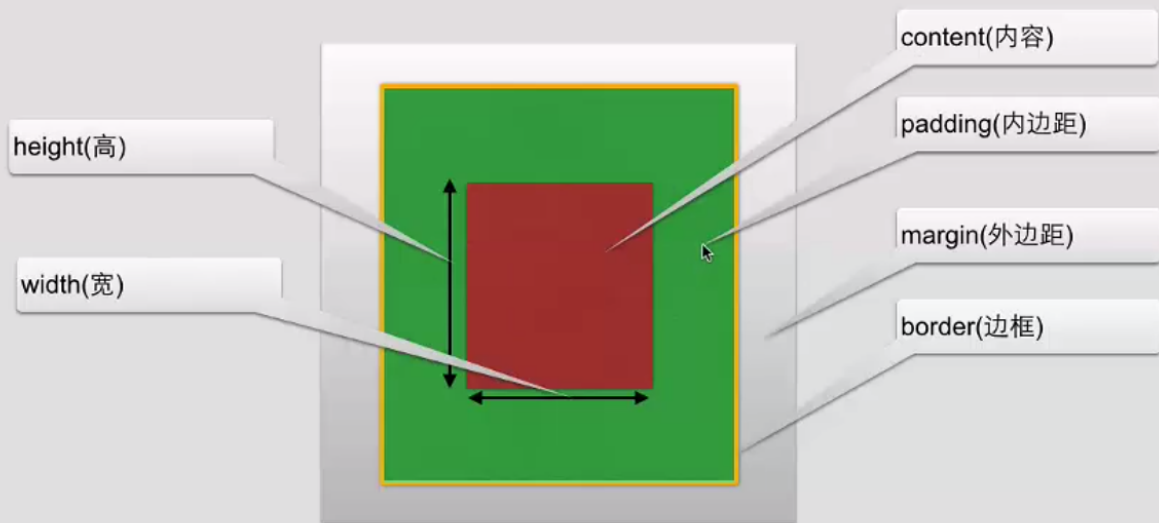
```

## css3的盒子模型

盒子模型的内容范围包括:

margin(外边距) border(边框) padding(内边距) content(内容)部分组成

# CSS盒子模型概述



极客学院

## 1.内边距

内边距在content外,边框内

## 2.内边距的属性

padding:设置所有的边距

padding-bottom:设置底边距

padding-left:设置左边距

padding-right:设置右边距

padding-top:设置上边距

```
1 <table border="1">
2   <tr>
3     <td>内边距</td>
4
5   </tr>
6 </table>
7
8
9 style.css
10 td{
11   padding:100px;
12   //或者可以各自规定
13   padding-left:100px;
14   padding-right:100px;
15   padding-top:100px;
16 }
```

## 边框

### 1.css边框:

我们可以创建出效果出色的外边框,并且可以应用于任何元素.

### 2.边框的样式

border-style:定义了10个不同的非继承样式,包括none

### 3.边框的单边样式

border-top-style

border-left-style:

border-right-style:

border-bottom-style:

### 4.边框的宽度

border-width

### 5.单边框的宽度

border-top-width

bordr-left-width

border-rigth-width

boeder-bottom-width

### 6.边框颜色

border-color

### 7.单边框颜色

border-top-color

border-left-color

border-right-color

border-bottom-color

```
1  <p>Css边框</p>
2
3  styele.css
4
5  p{
6      border-style:double/dotted//双线.虚线
7      border-top-style:double//顶部就变成了双线的效果
8      //bordr-width:10px//边框整体的宽度
9      border-top-width:10px//单边框的宽度
10 }
11
12
```

## Css3边框

border-radius:圆角边框

box-shadow:边框阴影

border-image:边框图片

```
1  p{
2      border-radius:"10px";
3      width:100px;和border-width有什么区别??这里的宽度是指p元素内容的宽度,而border-
width是边框的宽度
4      height:50px;
5      background-color:aquamarine1;
6
7      border:2px solid blue;//
8  }
9
10  边框阴影
11  .cssid{
12      width:100px;
13      height:100px;
14      text-align:center;
15      box-shadow:10px 10px 5px #FFFFFF;//向右移动 向下移动 透明度 颜色
16
17  }
18
19
```

## Css外边距

外边距:围绕在内容边框的区域就是外边距,外边距默认为透明区域

外边距接受任何长度单位,百分数值

外边距的通用属性:

margin:设置所有边距

margin-bottom:设置底边距

margin-left:设置左边距

margin-right;设置右边距

margin-top:设置上边距





```
2
3 <div class="mg">外边距</div>
4
5 style.css
6
7 body{
8     margin:0;
9 }
10 .mg{
11     background-color:blue;
12     width:100px;
13     height:100px;
14     margin:100px;
15 }
16
```

```
1 <div class="container">
2     <div class="bd">
3         <div class="pd">
4             <div class="content">hello</div>
5
6         </div>
7     </div>
8
9 </div>
10
11 style.css
12 body{
13     margin:0;
14 }
15 .container{
16     margin:100px;
17 }
18
19 .bd{
20     border-style:dotted;
21 }
22
23 .pd{
24     padding:100px;
25 }
26
27 .content{
28     background-color:blue;

```

```
29 }  
30
```

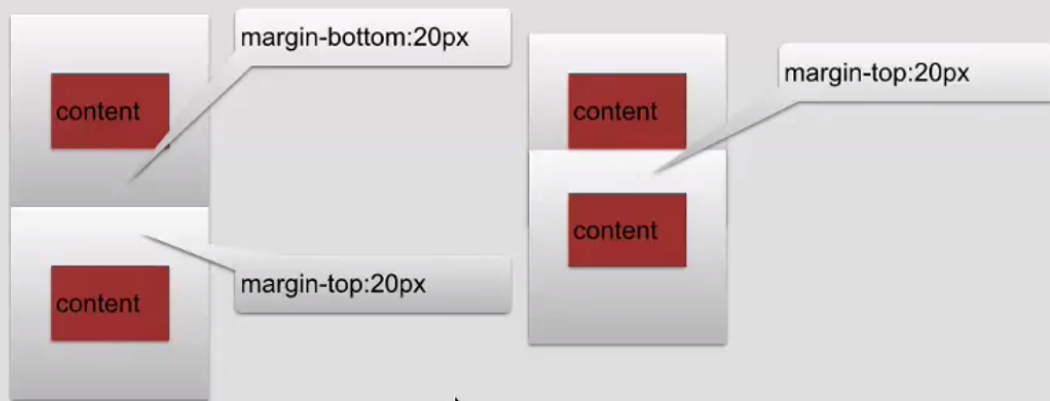
## Css外边距合并

外边距合并就是一个叠加的概念(遵循多的那个部分)

### CSS外边距合并

#### 1、外边距合并:

外边距合并就是一个叠加的概念



极客学院

## 盒子模型的应用



```
1 <!DOCTYPE html>  
2 <html>  
3  
4   <head>  
5     <meta charset="UTF-8"/>  
6     <title>盒子模型的应用</title>  
7     <link rel="stylesheet" type="text/css" href="style.css">  
8   </head>  
9   <body>  
10    <div class="top">  
11      <div class="top_content"></div>  
12    </div>  
13  
14    <div class="body">
```

```

15     <div class="body_img"></div>
16     <div class="body_content">
17         <div class="body_no"></div>
18     </div>
19
20
21 </div>
22 <div class="footing">
23     <div class="footing_content"></div>
24     <div class="footing_menu"></div>
25 </div>
26
27 </body>
28 </html>

```

```

1  *{
2      <!--将所有的内边距和外边距都调成0-->
3      margin:0;
4      padding:0;
5  }
6
7  .top{
8      width:100%;
9      height:50px;
10     background-color:black;
11
12 }
13
14 .top_content{
15     width:75%;
16     height:50px;
17     margin:0px auto;
18     background-color: blue;
19 }
20
21 .body{
22     width:75%;
23     height:1500px;/*有内容撑开,就不需要制订了*/
24     margin:20px auto;
25     background-color: aqua;
26 }
27 .body_img{
28     width:100%;
29     height:400px;

```

```
30     background-color:rgb(241, 255, 240);
31 }
32 .body_content{
33     width:100%;
34     height:1100px;
35     background-color: red;
36 }
37 .body_no{
38     width:100%;
39     height:50px;
40     background-color: saddlebrown;
41 }
42 .footing{
43     width:75%;
44     height:400px;
45     background-color:khaki;
46     margin:0px auto;
47 }
48 .footing_content{
49     width:100%;
50     height:350px;
51     background-color: lightgray;
52
53 }
54
55 .footing_menu{
56     width:100%;
57     height:70px;
58     background-color:black;
59 }
```