

一、布局方式

1. 标准流/静态流
2. 浮动布局
 - 1) 属性
 - 2) 取值
 - 3) 特点
 - 4) 常见问题
 - 5) 解决
3. 定位布局
 - 1) 属性
 - 2) 取值
 - 3) 偏移属性
 - 4) 分类
 - 5) 堆叠次序

二、背景属性

1. 背景颜色
2. 背景图片相关
 - 1) 设置背景图片
 - 2) 设置背景图片的重复方式
 - 3) 设置背景图片的显示位置
 - 4) 设置背景图片的尺寸
3. 背景属性简写

三、文本属性

1. 字体相关
 - 1) 设置字体大小
 - 2) 设置字体粗细程度
 - 3) 设置斜体
 - 4) 设置字体名称
 - 5) 字体属性简写
2. 文本样式
 - 1) 文本颜色
 - 2) 文本装饰线
 - 3) 文本内容的水平对齐方式
 - 4) 行高
 - 5) font属性简写2

一、布局方式

1. 标准流/静态流

默认布局方式,按照代码书写顺序及标签类型从上到下,从左到右依次显示

2. 浮动布局

主要用于设置块元素的水平排列

1) 属性

float

2) 取值

可取left或right, 设置元素向左浮动或向右浮动

```
float:left/right;
```

3) 特点

- 元素设置浮动会从原始位置脱流,向左或向右依次停靠在其他元素边缘,在文档中不再占位
- 元素设置浮动,就具有块元素的特征,可以手动调整宽高
- "文字环绕":浮动元素遮挡正常元素的位置,无法遮挡正常内容的显示,内容围绕在浮动元素周围显示
- 文档流在文档流中找位置, 浮动流在浮动层中找位置。

4) 常见问题

子元素全部设置浮动,导致父元素高度为0,影响父元素背景色和背景图片展示,影响页面布局

5) 解决

- 对于内容固定的元素,如果子元素都浮动,可以给父元素固定高度(例:导航栏)
- 在父元素的末尾添加空的块元素。设置clear:both;清除浮动, 构建了一个兄弟元素。
- 为父元素设置overflow:hidden;解决高度为0

float_html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    #box{
      width:500px;
      height:500px;
      margin:0 auto;
      background:orange;
    }
    #d1,#d2,#d3{
      width:200px;
      height:200px;
    }
    #d1{
      /*height:300px;*/
      background:red;
      float:right;
```

```

    }
    #d2{
        background:green;
        height:300px;
        float:right;
    }
    #d3{
        background:blue;
        float:right;
    }
    span{
        float:right;
        width:200px;
        height:200px;
        background:pink;
    }
</style>
</head>
<body>
    <div id="box">
        <div id="d1"></div>
        <div id="d2"></div>
        <div id="d3"></div>
    </div>
    <!--脱离文档流的元素在文档中不再占位,可以手动调整宽高-->
    <span></span>

</body>
</html>

```

float_02.html

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <style>
        div{
            width:200px;
            height:200px;
            background:red;
            float:left;
            margin-right:30px;
        }
        input{
            float:left;
        }
    </style>

```

```

</head>
<body>
    <div></div>
    <span>行内元素</span>
    <input type="text">
    <h1>浮动元素只能遮挡元素的位置,但是不影响内容显示</h1>
</body>
</html>

```

float_nav.html

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <style>
        #nav{
            width:900px;
            height:30px;
            background:gray;
        }
        .left{
            width:300px;
            float:left;
            background:green;
        }
        .right{
            width:300px;
            float:right;
            background:red;
        }
        ul{
            margin:0;
            padding:0;
            list-style:none;
        }
        li{
            float:left;
            margin-right:30px;
        }
        #main{
            margin-top:50px;
            width:900px;
            /*解决子元素全部浮动造成父元素高度为0的问题*/
            /*height:400px;*/
            /*overflow:auto;*/
            background:pink;
        }
    </style>

```

```

        .item1{
            width:580px;
            height:400px;
            background:gray;
            float:left;
        }
        .item2,.item3{
            float:right;
            width:300px;
            height:190px;
            background:gray;
        }
        .item2{
            margin-bottom:20px;
        }
        #d1,#d2{
            width:200px;
            height:200px;
            background:red;
        }
        #d1{
            background:green;
            float:left;
        }
        #d2{
            /*使文档中正常元素不受前面浮动元素的影响
            clear:left/right/both;
            */
            clear:left;
        }
        .clear{
            clear:both;
        }
    </style>
</head>
<body>
    <div id="nav">
        <div class="left">
            <ul>
                <li>首页</li>
                <li>物流</li>
                <li>客服</li>
            </ul>
        </div>
        <div class="right">右侧导航</div>
    </div>
    <div id="main">
        <div class="item1">1</div>
        <div class="item2">2</div>
        <div class="item3">3</div>
        <div class="clear"></div>

    </div>

```

```
<div>
    联系我们
    <div id="d1"></div>
    <div id="d2"></div>
</div>
</body>
</html>
```

banner_test.html

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <style>
        #banner{
            width:300px;
            height:250px;
            position:relative;
        }
        a{
            color:#fff;
            background:orange;
            position:absolute;
            width:100px;
            height:30px;
            right:10px;
            /*根据参照物对应方向的值计算偏移量
            (250*50%)* /
            top:50%;
            /*通过外边距微调元素的位置*/
            margin-top: -15px;
        }
    </style>
</head>
<body>
    <div id="banner">
        
        <a href="">下一张</a>

    </div>
</body>
</html>
```

3. 定位布局

结合偏移属性调整元素的显示位置

1) 属性

position

2) 取值

可取relative（相对定位）/absolute（绝对定位）/fixed（固定定位）

```
position: relative/absolute/fixed
```

3) 偏移属性

设置定位的元素可以使用偏移属性调整距离参照物的位置

top	距参照物的顶部
right	距参照物的右侧
bottom	距参照物的底部
left	距参照物的左侧

4) 分类

- relative 相对定位

元素设置相对定位, 可参照元素在文档中的原始位置进行偏移, 不会脱离文档流

- absolute 绝对定位

1. 绝对定位的元素参照离他最近的已经定位的祖先元素进行偏移, 如果没有, 则参照窗口进行偏移
2. 绝对定位的元素会脱流, 在文档中不占位, 可以手动设置宽高

使用绝对定位: "父相子绝": 父元素设置相对定位, 子元素绝对定位, 参照已定位的父元素偏移.

- fixed 固定定位

1. 参照窗口进行定位, 不跟随网页滚动而滚动
2. 脱离文档流

5) 堆叠次序

元素发生堆叠时可以使用 z-index 属性调整已定位元素的显示位置, 值越大元素越靠上:

- 属性: z-index
- 取值: 无单位的数值, 数值越大, 越靠上
- 堆叠:

1. 定位元素与文档中正常元素发生堆叠, 永远是已定位元素在上
2. 同为已定位元素发生堆叠, 按照 HTML 代码的书写顺序, 后来者居上

relative.html

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    #d1,#d2{
      width:200px;
      height:200px;
      background:red;
      margin:0 auto;
    }
    #d1{
      background:green;
      /*相对定位,已定位的元素可以设置偏移属性,调整
      元素的的显示位置*/
      position:relative;
      /*
      top:100px;
      left:100px;
      */
      bottom:-100px;
      right:100px;
    }
  </style>
</head>
<body>
  <div id="d1"></div>
  <div id="d2"></div>
</body>
</html>

```

absolute.html

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    #box{
      width:500px;
      margin:0 auto;
    }
  </style>
</head>
<body>
  <div id="box">
    <div id="d1"></div>
    <div id="d2"></div>
  </div>
</body>
</html>

```



```

        background:orange;
        /*设置为定位元素*/
        position:relative;
    }
    #d1,#d2{
        width:200px;
        height:200px;
        background:red;
    }
    #d1{
        background:green;
        /*绝对定位*/
        position:absolute;
        top:0;
        left:0;
    }
    body{
        /*设置为定位元素*/
        position:relative;
    }
</style>
</head>
<body>
    <div id="box">
        <div id="d1"></div>
        <div id="d2"></div>
    </div>
</body>
</html>

```

fixed.html

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <style>
        #main{
            height:1500px;
            background:pink;
        }
        #chat{
            width:300px;
            height:300px;
            background:orange;
            /*固定定位:脱离文档流,参照窗口进行偏移;不跟随
            页面滚动而滚动*/
            position:fixed;

```

```

        right:0;
        bottom:0;
    }

</style>
</head>
<body>
    <div id="main">

        </div>
        <div id="chat"></div>
</body>
</html>

```

z-index.html

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
    <style>
        div{
            width:200px;
            height:200px;
            background:red;
        }
        #d1{
            background:green;
            position:relative;
            z-index:1;
            /*只有定位元素才能使用偏移属性调位置*/
            /*

            top:100px;
            left:100px;
            */

        }
        #d2{
            position:relative;
            left:100px;
            bottom:-100px;
        }
        #d2:hover{
            /*调整堆叠次序.属性:z-index
            取值为无单位的整数,值越大越靠上;只有定位元素能够
            使用z-index属性调整次序*/
            z-index:10;
        }
    }

```

```

</style>
</head>
<body>
  <!-- 正常元素与定位元素发生堆叠, 永远是定位元素在上方显示 -->
  <!-- 同为定位元素发生堆叠, 根据标签书写顺序, 后来者居上 -->
  <div id="d2"></div>
  <div id="d1"></div>
  <h1>
    实现网页:
    上方为导航栏(包含若干导航项, 包含下拉菜单)
    下方为轮播图区域(包含图片, 图片索引, 左右按钮)
  </h1>
</body>
</html>

```

homework

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    /*清除浏览器默认样式*/
    body, ul, ol{
      margin:0;
      padding:0;
      /*取消列表默认样式(项目符号)*/
      list-style:none;
    }
    body{
      color:#ddd;
    }
    a{
      text-decoration:none;
      color:#ddd;
    }
    /*外围结构的样式*/
    #nav, #banner{
      width:990px;
      margin:0 auto;
    }
    /*单独设置每个模块的样式*/
    /*1. 导航栏*/
    #nav{
      background: green;
      height:30px;
    }
    /*匹配ul中的直接子元素li*/

```

```
#nav ul>li{
    float:left;
    margin-left:50px;
    line-height:30px;
}
#nav ul>li:first-child{
    position:relative;
}
#nav ol{
    display:none;
    position:absolute;
    z-index:10;
}
/*子元素过滤选择器,
li:first-child匹配作为第一个子元素存在的li */
#nav ul>li:first-child:hover ol{
    background:red;
    display:block;
}
#nav ol li:hover{
    background:orange;
}
/*2.轮播图*/
#banner{
    /*margin-top:30px;*/
    margin:30px auto;
    height:300px;
    position:relative;
}
#banner>img{
    width:990px;
    height:300px;
}
#banner ul{
    width:140px;
    position:absolute;
    bottom:20px;
    /*参照父元素的宽度计算偏移量*/
    left:50%;
    margin-left:-70px;
    background:green;
}
#banner li{
    float:left;
    width:20px;
    height:20px;
    background:red;
    border-radius:50%;
    margin-left:10px;
}
#banner ul>li:first-child{
    margin:0;
}
```



```
</html>
```

二、背景属性

1. 背景颜色

```
background-color: red;
```

2. 背景图片相关

1) 设置背景图片

```
background-image : url("路径");
```

设置背景图片，指定图片路径，如果路径中出现中文或空格，需要加引号

2) 设置背景图片的重复方式

默认背景图片从元素的左上角显示，如果图片尺寸与元素尺寸不匹配时，会出现以下情况：

1. 如果元素尺寸大于图片尺寸，会自动重复平铺，直至铺满整个元素
2. 如果元素尺寸小于图片尺寸，图片默认从元素左上角开始显示，超出部分不可见

```
background-repeat: repeat/repeat-x/repeat-y/no-repeat
```

取值：

repeat 默认值，沿水平和垂直方向重复平铺
repeat-x 沿X轴重复平铺
repeat-y 沿Y轴重复平铺
no-repeat 不重复平铺

3) 设置背景图片的显示位置

默认显示在元素左上角

```
background-position: x y;
```

取值方式：

1. 像素值
设置背景图片的在元素坐标系中的起点坐标
2. 方位值
水平：left/center/right
垂直：top/center/bottom
注：如果只设置某一个方向的方位值，另外一个方向默认为center

3. 百分比

类似于方位值，根据百分比计算背景图片的显示坐标。

计算方式：

横坐标 = (元素宽度 - 背景图片宽度) * x%

纵坐标 = (元素高度 - 背景图片高度) * y %

特殊值：

0% 0% 左上角

100% 100% 右下

50% 50% 居中显示

精灵图技术：为了减少网络请求，可以将所有的小图标拼接在一张图片上，一次网络请求全部得到；借助于 background-position 进行背景图片位置的调整，实现显示不同的图标

4) 设置背景图片的尺寸

```
background-size:width height;
```

取值方式：

1. 像素值

1. 500px 500px; 同时指定宽高
2. 500px; 指定宽度，高度自适应

2. 百分比

百分比参照元素的尺寸进行计算

1. 50% 50%; 根据元素宽高, 分别计算图片的宽高
2. 50%; 根据元素宽度计算图片宽高, 图片高度等比例缩放

3. 背景属性简写

```
background:color url("") repeat position;
```

注意：

1. 如果需要同时设置以上属性值，遵照相应顺序书写
2. background-size 单独设置

三、文本属性

1. 字体相关

1) 设置字体大小

```
font-size:20px;
```

2) 设置字体粗细程度

```
font-weight:normal;
```

取值：

1. normal (默认值) 等价于400
2. bold (加粗) 等价于700

3) 设置斜体

```
font-style:italic;
```

4) 设置字体名称

```
font-family:Arial,"黑体";
```

取值: 1. 可以指定多个字体名称作为备选字体,使用逗号隔开 2. 如果字体名称为中文,或者名称中出现了空格,必须使用引号 例:

```
font-family:Arial;  
font-family:"黑体","Microsoft YaHei",Arial;
```

5) 字体属性简写

```
font : style weight size family;
```

注意: 1. 如果四个属性值都必须设置,严格按照顺序书写 2. size family 是必填项

```
<!doctype html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta name="viewport"  
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-  
scale=1.0, minimum-scale=1.0">  
    <meta http-equiv="X-UA-Compatible" content="ie=edge">  
    <title>Document</title>  
    <style>  
        div,h1{  
            width:100px;  
            height:100px;  
            margin:200px auto;  
            background-color:pink;  
            background-image:url(wxy.jpeg);  
            background-repeat:no-repeat;  
            /*background-position:-100px -100px;*/  
            /*background-position:right bottom;*/  
            /*background-position:10% 10%;*/  
        }  
        div:hover{  
            background-position:-130px -10px;  
        }  
    </style>  
</head>  
<body>  
    <div>  
        <h1>Hello World</h1>  
    </div>  
</body>  
</html>
```



```
h1{
    width:500px;
    height:500px;
    margin:0 auto;
    /*
    background-size特殊取值:
    cover将图片等比拉伸至足够大,完全覆盖元素,超出部分不可见
    contain将图片拉伸至刚好被元素容纳
    */
    background-size:cover;

}
h2{
    width:500px;
    height:500px;
    background:cyan url(wxy.jpeg) no-repeat center;
    font:italic 400 32px youyuan;
}

</style>
</head>
<body>
    <h2>字体样式展示</h2>
    <h1></h1>
    <div></div>

</body>
</html>
```

2. 文本样式

1) 文本颜色

```
color:red;
```

2) 文本装饰线

```
text-decoration:none;
```

取值: underline 下划线 overline 上划线 line-through 删除线 none 取消装饰线

3) 文本内容的水平对齐方式

```
text-align:center;
```

取值：

left(默认值)	左对齐
center	居中对齐
right	右对齐
justify	两端对齐

4) 行高

```
line-height:30px;
```

使用：文本在当前行中永远垂直居中,可以借助行高调整文本在元素中的垂直显示位置 $\text{line-height} = \text{height}$ 设置一行文本在元素中垂直居中 $\text{line-height} > \text{height}$ 文本下移显示 $\text{line-height} < \text{height}$ 文本靠上显示 特殊：line-height可以采用无单位的数值,代表当前字体大小的倍数,以此计算行高

5) font属性简写2

```
font : size/line-height family;
```

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    p{
      font-size:32px;
      font-weight:bold;
      /*font-style:italic;*/
      font-family:"黑体","宋体";
      color:blue;
      /*text-decoration:overline;*/
      width:200px;
      height:200px;
      background:orange;
      text-align:justify;
      /*根据字体大小计算行高*/
      line-height:2;
    }

    span{
      font:italic 700 32px/2 "黑体","宋体";
      background:red;
    }
  </style>
</head>
<body>
```

```
<h1>小泽老师</h1>
<p>隔壁老王hello隔壁老text-decoration王隔壁老王</p>
<span>王老师最帅</span>
```

```
</body>
```

```
</html>
```