DAY08 定位、偏移和元素水平垂直居中

2023年3月30日 17:38

定位和偏移

定位：元素定位 把元素在页面上自由的安排位置

定位的三要素：

1.确定目标元素

2.坐标系

3.定位流

定位的属性:

position:

static 默认 元素不定位 在正常的文档流

relative 相对定位 相对于元素本身在浏览器的位置去定位

absolute 绝对定位 相对于包含块定位

fixed 固定定位 相对于窗口定位

sticky 粘性定位

偏移属性:

top right left bottom

只有设置了定位属性的元素(属性值不是默认值)才可以添加偏移属性

定位属性必须配合偏移属性一起使用

网页布局

浮动

盒子模型

元素类型

定位

相对定位

position:relative

配合偏移属性:top、right、bottom、left

相对定位的特征:

1.参考位置:相对于元素本身在浏览器的位置去定位

2.相对定位不脱离文档流

3.支持负值

绝对定位

position:absolute

配合偏移属性:top、right、bottom、left

绝对定位的特征:

1.参考位置:相对于包含块定位

有定位属性并且属性值不是默认值的父元素

2.脱离文档流

3.支持负值

4.子绝父相: 父元素相对定位 子元素绝对定位

绝对定位的参考位置

1.相对于包含块定位

有定位属性并且属性值不是默认值的父元素

2.如果父元素没有定位属性,那么该属性会继续往上找，直到找到最近的 有定位属性的并且属性值不是默认值的 祖先级元素

3.如果祖先级元素都没有定位属性，那么该元素会参考窗口

固定定位

position:fixed

配合偏移属性:top、right、bottom、left

固定定位的特征:

1.参考位置:相对于窗口定位

2.脱离文档流

偏移属性

只有设置了定位属性，并且属性值不是默认值的元素才能设置偏移属性

定位属性和偏移属性要一起使用

top

right

bottom
left
auto 默认值 元素会处于原来在文档流中的位置不变
数值
百分比
相对定位 参考父元素的宽高
绝对定位 参考包含块的宽高
固定定位 参考窗口的宽高

一个发生绝对定位，固定定位的元素：
设置了宽度，同时设置left,right，以left为准
设置了高度，同时设置top,bottom，以top为准
一个发生绝对定位，固定定位的元素：
没有设置宽度，同时设置left, right, 会拉伸自身以适应left right
没有设置高度，同时设置top, bottom, 会拉伸自身以适应top bottom

粘性定位 css3
position:sticky
配合偏移属性:top、right、bottom、left

粘性定位的特征:
1.参考临界点
2.不脱离文档流
浮动、绝对定位、固定定位的行内元素可以直接设置宽高

层级
层级:定位元素之间的层叠关系
z-index:
auto 默认值 效果等于0
直接设置没有单位的整数，也可以设置负数
数值越大越靠上
数值相同，看书写顺序
负数会跑到所有元素的最后面，一般不推荐
跨度尽量10以上

最大值是21亿多

元素垂直水平居中

方法一:自我计算

必须知道父子元素的宽高

```
.father {
  width: 500px;
  height: 400px;
  background-color: pink;
  /* 外边距溢出 */
  overflow: hidden;
}
.son {
  width: 300px;
  height: 200px;
  background-color: blue;
  /* (400-200)/2 */
  margin-top: 100px;
  /* (500-300)/2 */
  margin-left: 100px;
}
```

```
<div class="father">
  <div class="son"></div>
</div>
```

方法二:必须知道子元素的宽高

1.子绝父相

2.子{

top: 50%;

left: 50%;

margin-top: -自身高度的一半;

margin-left: -自身宽度的一半;

}

```
<style>
  .father {
    width: 500px;
    height: 400px;
    background-color: pink;
    position: relative;
```

```
  }
  .son {
    width: 300px;
    height: 200px;
    background-color: blue;
    position: absolute;
    /* 移到中间去 */
    top: 50%;
    left: 50%;
    /* 回调，子元素宽高的一半 */
    /* 因为用户的视角并不是以元素左上角为原点 */
    margin-top: -100px;
    margin-left: -150px;
  }
</style>

<div class="father">
  <div class="son"></div>
</div>
```

方法三:不需要知道父子元素的宽高，但是子元素必须有宽高

1.子绝父相

2.子{

top: 0;

left: 0;

right: 0;

bottom: 0;

margin: auto;

}

```
<style>
  .father {
    width: 500px;
    height: 400px;
    background-color: pink;
    position: relative;
  }
  .son {
    width: 300px;
    height: 200px;
    background-color: blue;
    position: absolute;
```

```
    top: 0;
    left: 0;
    right: 0;
    bottom: 0;
    margin: auto;
  }
</style>

<div class="father">
  <div class="son"></div>
</div>
```

元素相对于窗口水平垂直居中

1.固定定位

2.水平垂直居中

元素{
top: 0;
left: 0;
right: 0;
bottom: 0;
margin: auto;
}

```
<style>
  div {
    width: 200px;
    height: 200px;
    background-color: cyan;
    position: fixed;
    top: 0;
    left: 0;
    right: 0;
    bottom: 0;
    margin: auto;
  }
</style>

<div></div>
```