.father{

width: 400px;

height: 200px;

position: relative;

border: 1px solid red;

}

.son{

width: 300px;

height: 100px;

position: absolute;

border: 1px solid blue;

left: 50%;

top: 50%;

margin: -50px 0 0 -150px;

}

缺点是：需要知道子元素的宽度和高度等等。兼容性很好

.father{

width: 400px;

height: 200px;

position: relative;

border: 1px solid red;

}

.son{

width: 300px;

height: 100px;

position: absolute;

border: 1px solid blue;

left: 0;

top: 0;

bottom: 0;

right: 0;

margin: auto;

}

这是利用了left、right、top、bottom都被设置为0的时候，它会自动利用父元素所有的可用空间，但是width和height被顶死了，margin为auto时会自动平分的。

.father{

width: 400px;

height: 200px;

position: relative;

border: 1px solid red;

}

.son{

width: 300px;

height: 100px;

position: absolute;

left: 50%;

top: 50%;

border: 1px solid blue;

transform: translate(-50%,-50%);

}

这个主要是利用了translate为百分比的时候，移动的距离是自身高度和宽度的百分比。在移动端比较流行，但是需要高版本浏览器支持。（IE9以上）

.father{

width: 80%;

height: 200px;

display: flex;

align-items: center;

justify-content: center;

border: 1px solid red;

}

.son{

width: 600px;

height: 100px;

border: 1px solid blue;

}

利用弹性盒子flex布局。主要是利用了f伸缩项目如果不设置的话 flex：0 1 auto；

也就是默认不伸展，不会利用空余的空间，然后再设置伸缩容器的align-items以及jusitify-content。