.wrapper{

display: flex;

align-items: flex-start;

}

.left{

flex: 0 0 auto;

}

.right{

flex: 1 1 auto;

}

Flex布局，重要的是不要忘了align-items ，因为其默认的是stretch，会纵向拉伸，导致两则伸缩项目同高。

.wrapper{

overflow: hidden;

}

.left{

float: left;

margin-right: 20px;

}

.right{

overflow: hidden;

}

利用浮动+BFC。外层的盒子要变成BFC，这样可以消除浮动。右边的盒子设置为BFC，由于BFC不会和浮动元素重叠，因此可以实现自适应。

.wrapper .left {

position: absolute;

}

.wrapper .right {

margin-left: 150px;

}

Absolute+margin-left

缺点：左侧脱离文档流了，如果左侧高于右侧盒子，左侧会超出父容器的高度。因此智能通过设置父容器的min-height来控制。

. .wrapper {

overflow: hidden;

}

.left {

float: left;

}

.wrapper .right {

margin-left: 150px;

}

Float+margin-left：父元素需要清除浮动，还要计算margin-left。

剩下一种是calc，这种比较烦，一般还要设置box-sizing，双inline-block还要注意消除空白，不喜欢。

.wrapper{

overflow: auto;

}

.left{

float: left;

}

.right{

float: left;

width: calc(100%-140px)

}