1.text-align:center方式水平居中：行内元素解决方案单行行内元素

这种方式可以水平居中块级元素中的行内元素，如inline，inline-block；

但是如果用来居中块级元素中的块级元素时，如div中的div，一旦内层的div有自己的宽度，这种方法就会失效。只能让里面div的文字等内容居中，而div仍然是左对齐的。

2.margin:0 auto方式 水平居中：块状元素解决方案

这种对齐方式要求内部元素(.content\_text)是块级元素，并且不能脱离文档流（如设置position:absolute）,否则无效。

3.脱离文档流的居中方式。

这种通常应用在自定义弹框当中，把背景层设置成透明灰色，内容居中显示在最前面。

<div class="mask">

　　<div class="content"><br>　　　　我是要居中的板块

　　</div>

</div>

.mask{

　　display: block;

　　position: fixed;

　　top: 0;

　　left: 0;

　　width: 100%;

　　height: 100%;

　　background: #000;

　　filter: alpha(opacity=30);

　　-ms-filter:"alpha(opacity=30)";

　　opacity: .3;

　　z-index: 10000;

}

.center{

    display: block;

    position: fixed;

    \_position: absolute;

    top: 50%;

    left: 50%;

    width: 666px;

    height:400px;

    margin-left: -333px;

    margin-top: -200px;

    z-index: 10001;

    box-shadow: 2px 2px 4px#A0A0A0, -2px -2px 4px #A0A0A0;

 4水平居中：多个块状元素解决方案

将元素的display属性设置为inline-block，并且把父元素的text-align属性设置为center即可

使用flexbox布局，只需要把待处理的块状元素的父元素添加属性display:flex及justify-content:center即可:

 5.垂直居中display:table-cell 多行的行内元素解决方案组合使用display:table-cell和vertical-align:middle属性来定义需要居中的元素的父容器元素生成效果

parent { background: #222;

width: 300px;

height: 300px; /\* 以下属性垂直居中 \*/

display: table-cell;

vertical-align:middle;

}

大小不固定块级元素垂直居中

.center {

　　display: table;

　　width: 100%;

}

.center\_text {

　　display: table-cell;

　　text-align: center;

　　vertical-align: middle;

6.垂直居中：单行的行内元素解决方案

height和line-height的值设置一样

9水平垂直居中：已知高度和宽度的元素解决方案

.item{ position: absolute;

top: 50%;

left: 50%;

margin-top: -75px; /\* 设置margin-left /

margin-top 为自身高度的一半 \*/

margin-left: -75px; }

10水平垂直居中：未知高度和宽度元素解决方案

.item{ position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); /\* 使用css3的transform来实现 \*/ }