

一、记录下几种盒子居中的方法：

- 1.0、margin固定宽高居中；
- 2.0、负margin居中；
- 3.0、绝对定位居中；
- 4.0、table-cell居中；
- 5.0、flex居中；
- 6.0、transform居中；
- 7.0、不确定宽高居中(绝对定位百分数)；
- 8.0、button居中。

二、代码演示(html使用同一个Demo)：

html Demo：

```
<body>    <div id="container">        <div id="box"></div>    </div></body>
```

1.0、margin固定宽高居中(演示)

这种定位方法纯粹是靠宽高和margin拼出来的，不是很灵活。

CSS：

```
#container { width: 600px; height: 500px; border: 1px solid #000;
margin: auto;}#box { width: 200px; height: 200px; margin: 150px 200px;
background-color: #0ff;}
```

2.0、负margin居中(演示)

利用负的margin来进行居中，需要知道固定宽高，限制比较大。

CSS:

```
#container { position: relative; width: 600px; height: 500px;
border: 1px solid #000; margin: auto;}#box { position: absolute; width:
200px; height: 200px; left: 50%; top: 50%; margin: -100px -100px;
background-color: #0ff;}
```

3.0、绝对定位居中(演示)

利用绝对定位居中，非常常用的一种方法。

CSS:

```
#container { position: relative; width: 600px; height: 500px;
border: 1px solid #000; margin: auto;}#box { position: absolute; width:
200px; height: 200px; left: 0; top: 0; right: 0; bottom: 0;
margin: auto; background-color: #0ff;}
```

4.0、table-cell居中(演示)

利用table-cell来控制垂直居中。

CSS:

```
#container {    display: table-cell;    width: 600px;    height: 500px;
vertical-align: middle;    border: 1px solid #000;}#box {    width: 200px;
height: 200px;    margin: 0 auto;    background-color: #0ff;}
```

5.0、flex居中(演示)

CSS3中引入的新布局方式，比较好用。缺点：IE9以及IE9一下不兼容。

CSS:

```
#container {    display: -webkit-flex;    display: flex;    -webkit-align-items:
center;            align-items: center;    -webkit-justify-content: center;
justify-content: center;    width: 600px;    height: 500px;    border: 1px solid
#000;    margin: auto;}#box {    width: 200px;    height: 200px;    background-
color: #0ff;}
```

6.0、transform居中(演示)

这种方法灵活运用CSS中transform属性，较为新奇。缺点是IE9下不兼容。

CSS:

```
#container {    position: relative;    width: 600px;    height: 600px;
border: 1px solid #000;    margin: auto;}#box {    position: relative;    top:
50%;    left: 50%;    width: 200px;    height: 200px;    transform:
translate(-50%, -50%);    -webkit-transform: translate(-50%, -50%);    -ms-
transform: translate(-50%, -50%);    -moz-transform: translate(-50%, -50%);
background-color: #0ff;}
```

7.0、不确定宽高居中(绝对定位百分数)(演示)

这种不确定宽高的居中，较为灵活。只需要保证left和right的百分数一样就可以实现水平居中，保证top和bottom的百分数一样就可以实现垂直居中。

CSS:

```
#container {    position: relative;    width: 600px;    height: 500px;
border: 1px solid #000;    margin: auto;}#box {    position: absolute;    left:
30%;    right: 30%;    top: 25%;    bottom: 25%;    background-color: #0ff;}
```

8.0、button居中(演示)

利用button做外容器，里边的块元素会自动垂直居中，只需要控制一下水平居中就可以达到效果。

HTML:

```
<button>    <div></div></button>
```

CSS:

```
button {    width: 600px;    height: 500px;    border: 1px solid #000;}div {
width: 200px;    height: 200px;    margin: 0 auto;    background-color: #0ff;}
```