# CSS样式设置小技巧

## 水平居中设置-行内元素

我们在实际工作中常会遇到需要设置水平居中的场景，比如为了美观，文章的标题一般都是水平居中显示的。

这里我们又得分两种情况：行内元素 还是 块状元素 ，块状元素里面又分为定宽块状元素，以及不定宽块状元素。我们先来了解一下行内元素怎么进行水平居中？

如果被设置元素为文本、图片等行内元素时，水平居中是通过给父元素设置 text-align:center 来实现的。(父元素和子元素：如下面的html代码中，div是“我想要在父容器中水平居中显示”这个文本的父元素。反之这个文本是div的子元素 )如下代码：

html代码：

<body>

<div class="txtCenter">我想要在父容器中水平居中显示。</div>

</body>

css代码：

<style>

.txtCenter{

text-align:center;

}

</style>

## 水平居中设置-定宽块状元素

当被设置元素为 块状元素 时用 text-align：center 就不起作用了，这时也分两种情况：定宽块状元素和不定宽块状元素。

定宽块状元素：块状元素的宽度width为固定值。

满足定宽和块状两个条件的元素是可以通过设置“左右margin”值为“auto”来实现居中的。我们来看个例子就是设置 div 这个块状元素水平居中：

html代码：

<body>

<div>我是定宽块状元素，哈哈，我要水平居中显示。</div>

</body>

css代码：

<style>

div{

border:1px solid red;/\*为了显示居中效果明显为 div 设置了边框\*/

width:200px;/\*定宽\*/

margin:20px auto;/\* margin-left 与 margin-right 设置为 auto \*/

}

</style>

也可以写成：

margin-left:auto;

margin-right:auto;

注意：元素的“上下 margin” 是可以随意设置的。

## 水平居中总结-不定宽块状元素方法

在实际工作中我们会遇到需要为“不定宽度的块状元素”设置居中，比如网页上的分页导航，因为分页的数量是不确定的，所以我们不能通过设置宽度来限制它的弹性。(不定宽块状元素：块状元素的宽度width不固定。)

不定宽度的块状元素有三种方法居中（这三种方法目前使用的都很多）：

1、加入 table 标签

2、设置 display: inline 方法：与第一种类似，显示类型设为 行内元素，进行不定宽元素的属性设置

3、设置 position:relative 和 left:50%：利用 相对定位 的方式，将元素向左偏移 50% ，即达到居中的目的

### 加入table

为什么选择方法一加入table标签? 是利用table标签的长度自适应性---即不定义其长度也不默认父元素body的长度（table其长度根据其内文本长度决定），因此可以看做一个定宽度块元素，然后再利用定宽度块状居中的margin的方法，使其水平居中。

第一步：为需要设置的居中的元素外面加入一个 table 标签 ( 包括 <tbody>、<tr>、<td> )。

第二步：为这个 table 设置“左右 margin 居中”（这个和定宽块状元素的方法一样）。

举例如下：

html代码：

<div>

<table>

<tbody>

<tr><td>

<ul>

<li>我是第一行文本</li>

<li>我是第二行文本</li>

<li>我是第三行文本</li>

</ul>

</td></tr>

</tbody>

</table>

</div>

css代码：

<style>

table{

border:1px solid;

margin:0 auto;

}

</style>

### 设置 display: inline 方法

改变块级元素的 display 为 inline 类型（设置为 行内元素 显示），然后使用 text-align:center 来实现居中效果。如下例子：

html代码：

<body>

<div class="container">

<ul>

<li><a href="#">1</a></li>

<li><a href="#">2</a></li>

<li><a href="#">3</a></li>

</ul>

</div>

</body>

css代码：

<style>

.container{

text-align:center;

}

/\* margin:0;padding:0（消除文本与div边框之间的间隙）\*/

.container ul{

list-style:none;

margin:0;

padding:0;

display:inline;

}

/\* margin-right:8px（设置li文本之间的间隔）\*/

.container li{

margin-right:8px;

display:inline;

}

</style>

这种方法相比第一种方法的优势是不用增加无语义标签，但也存在着一些问题：它将块状元素的 display 类型改为 inline，变成了行内元素，所以少了一些功能，比如设定长度值。

## 设置 position:relative 和 left:50%

通过给父元素设置 float，然后给父元素设置 position:relative 和 left:50%，子元素设置 position:relative 和 left: -50% 来实现水平居中。

1、父元素设置float，然后设置position:relative 和 left:50%，则父元素的左边框正好是窗口的中间。

2、子元素设置position:relative 和 left: -50%，使子元素相对父元素向右偏移50%，即子元素相对于原来位置向左移动了父元素宽度的50%，则子元素处于窗口的中间位置。

代码如下：

<body>

<div class="container">

<ul>

<li><a href="#">1</a></li>

<li><a href="#">2</a></li>

<li><a href="#">3</a></li>

</ul>

</div>

</body>

css代码：

<style>

.container{

float:left;

position:relative;

left:50%

}

.container ul{

list-style:none;

margin:0;

padding:0;

position:relative;

left:-50%;

}

.container li{float:left;display:inline;margin-right:8px;}

</style>

这三种方法使用得都非常广泛，各有优缺点，具体选用哪种方法，可以视具体情况而定。

## 垂直居中

我们在实际工作中也会遇到需要设置垂直居中的场景，比如好多报纸的文章标题在左右一侧时，常常会设置为垂直居中，为了用户体验性好。

这里我们又得分两种情况：父元素高度确定的单行文本，以及父元素高度确定的多行文本。

### 父元素高度确定的单行文本

父元素高度确定的单行文本的竖直居中的方法是通过设置父元素的 height 和 line-height 高度一致来实现的。(height: 该元素的高度，line-height: 顾名思义，行高（行间距），指在文本中，行与行之间的基线间的距离 )。

这种文字行高与块高一致带来了一个弊端：当文字内容的长度大于块的宽时，就有内容脱离了块。

如下代码：

<div class="container">

hi,imooc!

</div>

css代码：

<style>

.container{

height:100px;

line-height:100px;

background:#999;

}

</style>

### 父元素高度确定的多行文本

父元素高度确定的多行文本、图片等的竖直居中的方法有两种:

**方法一：使用插入 table (包括tbody、tr、td)标签，同时设置 vertical-align：middle。**

css 中有一个用于竖直居中的属性 vertical-align，在父元素设置此样式时，会对inline-block类型的子元素都有用。下面看一下例子：

html代码：

<body>

<table><tbody><tr><td class="wrap">

<div>

<p>看我是否可以居中。</p>

</div>

</td></tr></tbody></table>

</body>

css代码：

table td{height:500px;background:#ccc}

因为 td 标签默认情况下就默认设置了 vertical-align 为 middle，所以我们不需要显式地设置了。

方法二：在 chrome、firefox 及 IE8 以上的浏览器下可以设置块级元素的 display 为 table-cell（设置为表格单元显示），激活 vertical-align 属性，但注意 IE6、7 并不支持这个样式, 兼容性比较差。

html代码：

<div class="container">

<div>

<p>看我是否可以居中。</p>

<p>看我是否可以居中。</p>

<p>看我是否可以居中。</p>

</div>

</div>

css代码：

<style>

.container{

height:300px;

background:#ccc;

display:table-cell;/\*IE8以上及Chrome、Firefox\*/

vertical-align:middle;/\*IE8以上及Chrome、Firefox\*/

}

</style>

这种方法的好处是不用添加多余的无意义的标签，但缺点也很明显，它的兼容性不是很好，不兼容 IE6、7而且这样修改display的block变成了table-cell，破坏了原有的块状元素的性质。

## 隐性改变display类型

有一个有趣的现象就是当为元素（不论之前是什么类型元素，display:none 除外）设置以下 2 个语句之一：

1. position : absolute

2. float : left 或 float:right

元素的display显示类型就会自动变为以 display:inline-block（块状元素）的方式显示，当然就可以设置元素的 width 和 height 了，且默认宽度不占满父元素。