

Align - Align in CSS

In CSS there are many properties that allow aligning an element like margin, padding, text-align, position, float ...

Align is the setting of the element's element or content element. There are many properties in CSS that allow aligning an element like margin, padding, text-align, position, float . Align helps you a lot in formatting text and images.

In this article, TipsMake.com will work with you to find some of the most commonly used aligns. Invites you to read the track.

Center for element - Use margin: auto

In CSS, to center a block element

We use the margin attribute with the value of auto to align the element in the middle compared to its parent element.

The element after occupying a certain space, the remaining space will be divided equally for the two sides:

```
.center {  
  margin: auto;  
  width: 50%;  
  border: 3px solid green;  
  padding: 10px;  
}
```

Center the element

To center horizontally for a block element (like div) you can use use margin: auto.

Note: The element after occupying certain space, the space left

will be divided equally for both sides.

Center the image

To center an image, set margin-left and margin-right to auto and make the image become a block- like element as mentioned above.

```
img {
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 40%;
}
```

Center the image

To center an image, set margin-left and margin-right to auto and do for the image to become a block element as mentioned above.

Align text horizontally - Use text-align

In CSS there is a text-align attribute that allows you to align content text to the left, middle, or right of the element by values:

1. text-align: center - center
2. text-align: left - left root
3. text-align: right - right-aligned

Note : The text-align attribute also applies only to block elements because inline only occupies a sufficient width of its content.

Align left / right - Use position

Another method to align elements is to use position: absolute.

```
.right {
  position: absolute;
  right: 0px;
  width: 300px;
  border: 3px solid purple;
  padding: 10px;
}
```

Align left / right - Use float

The float attribute is also used to align elements.

```
.right {
  float: right;
  width: 300px;
  border: 3px solid purple;
  padding: 10px;
}
```

Note : If a floating element is higher than the parent element, it will cause an overflow of content. To fix this, we use `clearfix overflow: auto`:

```
.clearfix {  
  overflow: auto;  
}
```

Centered vertically - Use padding

There are many ways to center a vertical element in CSS. The simplest solution is to use padding.

```
.center {  
  padding: 70px 0;  
  border: 3px solid green;  
}
```

To align both horizontally and vertically, use padding with the attribute `text-align: center`.

```
.center {  
  padding: 70px 0;  
  border: 3px solid green;  
  text-align: center;  
}
```

Centered vertically - Use line-height

You also have another way to center vertically by using the `line-height` attribute with the same value as the `height` attribute .

```
.center {  
  line-height: 200px;  
  height: 200px;  
  border: 3px solid green;  
  text-align: center;  
}  
  
.center p {  
  line-height: 1.5;  
  display: inline-block;  
  vertical-align: middle;  
}
```

Center vertically - Use position and transform

If you don't use the above padding and line-height , you can use the third way to use position and transform:

```
.center {  
  height: 200px;  
  position: relative;  
  border: 3px solid purple;  
}
```

```
}  
  
.center p {  
margin: 0;  
position: absolute;  
top: 50%;  
left: 50%;  
transform: translate(-50%, -50%);  
}
```

You finished reading the article "**Align - Align in CSS**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
