

# How to Center an Image Vertically and Horizontally with CSS

---

 [freecodecamp.org/news/how-to-center-an-image-in-css](https://freecodecamp.org/news/how-to-center-an-image-in-css)

June 14, 2020

There are many ways to align HTML elements with CSS. One of the most common things developers struggle with is trying to center an element in the middle of the page.

So in this post, I will be showing some of the most common ways to center an image both vertically and horizontally using different CSS properties.

I've gone over the CSS Position and Display properties in my previous post. If you you're not familiar with those properties, I recommend checking out those posts before reading this article.

## Centering an Image Horizontally

---

Let's begin with centering an image horizontally by using 3 different CSS properties.

### Text-Align

---

The first way to center an image horizontally is using the `text-align` property. However, this method only works if the image is inside a block-level container such as a `<div>` :

```
<style>
  div {
    text-align: center;
  }
</style>

<div>
  
</div>
```

### Margin: Auto

---

Another way to center an image is by using the `margin: auto` property (for left-margin and right-margin).

However, using `margin: auto` alone will not work for images. If you need to use `margin: auto` , there are 2 additional properties you must use as well.

The margin-auto property does not have any effects on inline-level elements. Since the `<img>` tag is an inline element, we need to convert it to a block-level element first:

```
img {  
  margin: auto;  
  display: block;  
}
```

Secondly, we also need to define a width. So the left and right margins can take the rest of the empty space and auto-align themselves, which does the trick (unless we give it a width of 100%):

```
img {  
  width: 60%;  
  margin: auto;  
  display: block;  
}
```

## Display: Flex

---

The third way to center an image horizontally is by using `display: flex`. Just like we used the `text-align` property for a container, we use `display: flex` for a container as well.

However, using `display: flex` alone will not be enough. The container must also have an additional property called `justify-content`:

```
div {  
  display: flex;  
  justify-content: center;  
}
```

```
img {  
  width: 60%;  
}
```

The `justify-content` property works together with `display: flex`, which we can use to center the image horizontally.

Finally, the width of the image must be smaller than the width of the container, otherwise, it takes 100% of the space and then we can't center it.

**Important:** The `display: flex` property is not supported in older versions of browsers. [See here](#) for more details.

## Centering an Image Vertically

---

### Display: Flex

---

For vertical alignment, using `display: flex` is again really helpful.

Consider a case where our container has a height of 800px, but the height of the image is only 500px:

```
div {  
  display: flex;  
  justify-content: center;  
  height: 800px;  
}  
  
img {  
  width: 60%;  
  height: 500px;  
}
```

Now, in this case, adding a single line of code to the container, `align-items: center`, does the trick:

```
div {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  height: 800px;  
}
```

The `align-items` property can position elements vertically if used together with `display: flex`.

## Position: Absolute & Transform Properties

---

Another method for vertical alignment is by using the `position` and `transform` properties together. This one is a bit complicated, so let's do it step by step.

### Step 1: Define Position Absolute

---

Firstly, we change the positioning behavior of the image from `static` to `absolute`:

```
div {  
  height: 800px;  
  position: relative;  
  background: red;  
}  
  
img {  
  width: 80%;  
  position: absolute;  
}
```

Also, it should be inside a relatively positioned container, so we add `position: relative` to its container div.

## Step 2: Define Top & Left Properties

---

Secondly, we define the top and left properties for the image, and set them to 50%. This will move the starting point(top-left) of the image to the center of the container:

```
img {  
  width: 80%;  
  position: absolute;  
  top: 50%;  
  left: 50%;  
}
```

## Step 3: Define the Transform Property

---

But the second step has moved the image partially outside of its container. So we need to bring it back inside.

Defining a `transform` property and adding -50% to its X and Y axis does the trick:

```
img {  
  width: 80%;  
  position: absolute;  
  top: 50%;  
  left: 50%;  
  transform: translate(-50%, -50%);  
}
```

There are other ways to center things horizontally and vertically, but I've explained the most common ones. I hope this post helped you understand how to align your images in the center of the page.