

# CSS

## THE CENTERING TRICKS CHECKLIST

### CENTER HORIZONTALLY

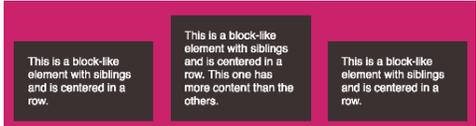
**is it inline?** You can center inline elements horizontally, that are **within** a block-level parent element, by writing so:

CSS CODE SNIPPET

```
.center-children {
  text-align: center;
}
```

**or is it block?** You can apply a margin-left and a margin-right of auto and set a width.

```
.center-this {
  margin: 0 auto;
  width: 400px;
}
```



**is it for several blocks?** For two or more block-level elements that need to be centered horizontally in a row, change them to **inline-block**, as seen below.

HTML CODE SNIPPET

```
<main class="example">
  <div>This is a block-like element with siblings and is centered in a row.</div>
  <div>This is a block-like element with siblings and is centered in a row. Please note this one has more content than the others.</div>
  <div>This is a block-like element with siblings and is centered in a row.</div>
</main>
```

CSS CODE SNIPPET

```
.example {
  text-align: center;
}
div {
  display: inline-block;
  text-align: left;
  padding: 20px;
  max-width: 240px;
  margin: 10px;
}
```

### CENTER VERTICALLY

**is it inline & only a single line?** Inline/text elements can appear vertically centered, with equal padding above and below them.

```
.bio {
  padding-top: 26px;
  padding-bottom: 26px;
}
```

When you aren't using padding the text that you use won't be wrapping, make the **line-height** = to the **height** to center the text.

```
.hack {
  height: 120px;
  line-height: 120px;
}
```

**is it inline with multiple lines?** Equal padding on top and bottom can work for multiple lines of text, too.

If that doesn't work, you can try using **display: table** on the parent and **display: table cell** on the child. Apply a vertical-align property to the child.

**is it block & you know what the height is?** If you do know the height, you can center vertically:

```
.parent {
  position: relative;
}
.child {
  position: absolute;
  top: 50%;
  height: 120px;
  transform: translateY(-50%);
}
```

**is it block & you don't know the height?** You can center it by sliding it back up half of the height

after pushing the element down halfway:

```
.parent {
  position: relative;
}
.child {
  position: absolute;
  top: 50%;
  transform: translateY(-50%);
}
```

### CENTER BOTH

Using a combination of the techniques above can result in both vertical and horizontal centering.

**is it fixed width and height?**

Using negative margins equal to half of that width and height, after you've absolutely positioned it at 50% / 50% will center it with great cross browser support:

```
.parent {
  position: relative;
}
.child {
  width: 300px;
  height: 90px; /*halve this*/
  padding: 20px; /*add this*/
  position: absolute;
  top: 50%;
  left: 50%;
  margin: -50px 0 0 -50px;
}
```

**or unknown width and height?** You can apply the transform property and a negative translate of 50% in both directions to center.

```
.parent {
  position: relative;
}
.child {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}
```