



[Course](#) > [Modul...](#) > [5.2 Tab...](#) > Styling ...

Styling your table

Styling your table

We now know how to put a basic table together. However, the tables we have looked at so far could really use some work in terms of how they look.

Here, we will look at some useful CSS elements to style your table. All examples below are using CodePen. Each CodePen has three tabs - Result, HTML and CSS. Make sure to view each tab to get the HTML and CSS code for examples. You are also welcome to modify the code for each example by clicking on 'Edit on CodePen' on the top right corner of the CodePen (click on "RUN" when you completed your changes).

border

Though `border` is a valid attribute of the table element, it is best specified in CSS. It is a shorthand property meaning you can set several CSS properties simultaneously.

The CSS `border` property sets `border-width`, `border-style` and `border-color` in order:

```
table { border: 1px solid black; }
```

Property Value	Possible values
<code>border-width</code>	thin, medium, thick, in pixels
<code>border-style</code>	none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset

border-color	color name or color values, transparent
initial	sets the property to the default value. Defaults for width, style and color are 'medium none current-color-of-element'
inherit	inherits property from parent element

To give a border to `<table>`, `<th>` and `<td>`:

```
1. table, th, td { border: 1px solid black; }
```

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

border-collapse

We gave a border to the table, table-header and table-data above. This creates two borders creating a double line. In order to collapse them all into a single border, we use the `border-collapse` CSS property:

```
1. table { border-collapse: collapse; }
```

Possible values of this property are:

- `separate` - default value where borders are detached like in the example above
- `collapse` - border are collapsed into a single border
- `initial` - sets to default value (`separate`)

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

Table width and height

Browsers automatically set the width and height for the rows and columns for your table based on the content in your cells. Cells with most content usually set the height and width of all cells adjacent to themselves.

With CSS, you can also explicitly set the dimensions of your cells. Width and height can be specified in:

- units of length like pixels, percentage - relative to the table width, etc
- auto: the browser will calculate and select a width for the specified element (default value)

It also supports initial (sets property to default value) and inherit (from parent element).

width/height of `<td>` of one cell will not only affect that cell but the whole column/row. If two cells in one column/row have different widths/heights specified, the larger value is set.

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

text-align

This property is used to align the text of `<th>` and `<td>` cells left, right or center (week 3 recap).

Default:

- `<th>` - center
- `<td>` - left

```
1. td { text-align: right; }
```

vertical-align

This property is used to align the text of `<th>` and `<td>` cells top, bottom or middle.

Default:

- `<th>` - middle
- `<td>` - middle

```
1. th { vertical-align: top; }
```

padding

Right now our table looks quite cramped. We use the `padding` property on `<th>` and `<td>` to provide some space between border and content in cell. It takes its value in units of length like `px`, `cm`, `%` - relative to parent container's width, etc.

```
1. th, td { padding: 15px; }
```

This will add 15px space around the content on all sides.

You can also apply different padding styles for four individual sides by using:

- `padding-top` | `th, td { padding-top: 15px; }`
- `padding-right` | `th, td { padding-right: 25px; }`
- `padding-bottom` | `th, td { padding-bottom: 35px; }`
- `padding-left` | `th, td { padding-left: 45px; }`

Alternatively, `padding` can also be provided as a shorthand property where you can specify all four sides in one go:

```
1. th, td { padding: 20px 30px 40px 50px; }
```

It is specified in the order: top, right, bottom and left padding.

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

border-spacing

`border-spacing` specifies the distance between two cell borders in pixels. This is different from `padding` which is space between content in cell and border. It takes its value in units of length like `px`, `cm`, `%` - relative to parent container's width, etc.

It has the `inherit` property whose default value is 0.

It takes two values for horizontal and vertical spacing. If only one value is provided, it is used for both horizontal and vertical spacing:

```
1. table { border-spacing: 25px 50px; }  
2. table, td, th { border-spacing: 25px; }
```

Some things to keep in mind:

- If you try to provide spacing for only `<th>` and `<td>`, make sure there is space from the table border or you will not see a difference.
- You have to set - `table { border-collapse: separate; }` for it to take effect.

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

Side-borders

The first property `border` will set a border to all four sides. You can also set borders to individual sides - top, right, bottom, left:

```
1. th, td { border-right: 1px solid black; }
```

See the Pen [Styling your table](#) by W3Cx (@w3devcampus) on [CodePen](#).

© All Rights Reserved