

The display Property

The `display` property specifies if/how an element is displayed.

Every HTML element has a default display value depending on what type of element it is. The default display value for most elements is `block` or `inline`.

Click to show panel

This panel contains a `<div>` element, which is hidden by default (`display: none`).

It is styled with CSS, and we use JavaScript to show it (change it to `display: block`).

Block-level Elements

A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).

The `<div>` element is a block-level element.

Examples of block-level elements:

- `<div>`
- `<h1>` - `<h6>`
- `<p>`
- `<form>`
- `<header>`
- `<footer>`
- `<section>`

Inline Elements

An inline element does not start on a new line and only takes up as much width as necessary.

This is an inline element inside a paragraph.

Examples of inline elements:

- ``
- `<a>`
- ``

Display: none;

`display: none;` is commonly used with JavaScript to hide and show elements without deleting and recreating them. Take a look at our last example on this page if you want to know how this can be achieved.

The `<script>` element uses `display: none;` as default.

Override The Default Display Value

As mentioned, every element has a default display value. However, you can override this.

Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way, and still follow the web standards.

A common example is making inline `` elements for horizontal menus:

Example

```
li {  
    display: inline;  
}
```

[Try it Yourself »](#)

Note: Setting the display property of an element only changes **how the element is displayed**, NOT what kind of element it is. So, an inline element with `display: block;` is not allowed to have other block elements inside it.

The following example displays `` elements as block elements:

Example

```
span {  
    display: block;  
}
```

[Try it Yourself »](#)

The following example displays <a> elements as block elements:

Example

```
a {  
    display: block;  
}
```

[Try it Yourself »](#)

Hide an Element - display:none or visibility:hidden?

`display:none`



Remove

`visibility:hidden`

Hiding an element can be done by setting the `display` property to `none`. The element will be hidden, and the page will be displayed as if the element is not there:

Example

```
h1.hidden {  
    display: none;  
}
```

[Try it Yourself »](#)

`visibility:hidden;` also hides an element.

However, the element will still take up the same space as before. The element will be hidden, but still affect the layout:

Example

```
h1.hidden {  
    visibility: hidden;  
}
```