

display: none vs visibility: hidden

We had a look at `display: none;` - this value removes the element to which you apply it from the document flow. This means that the element is not visible and it also doesn't "block its position". Other elements can (and will) take its place instead.

There is an alternative to that though.

If you only want to hide an element but you want to keep its place (i.e. other elements don't fill the empty spot), you can use `visibility: hidden;`

Here's a visual example:

```
1. .box-1 {
2.     display: none;
3. }
4.
5. .box-2 {
6.     display: inline-block;
7. }
```

Will render:

x

where **x** has the class `box-2`. The first element just isn't displayed. **It's still part of the DOM though**, you can still access it via JavaScript for example.

Here's an example for `visibility: hidden`:

```
1. .box-1 {
2.     visibility: hidden;
3. }
4.
5. .box-2 {
6.     display: inline-block;
7. }
```

Will render:

x

where simply is an empty spot and **x** has the class `box-2`.

The element is only invisible, **it's not removed from the document flow and of course also not from the DOM.**