

Week 4_2(en)

Display

We've learned about block and inline elements few weeks ago.

For example, `<div>` and `<h1>` are *block elements*, which come above and below each other, while `` and `` are *inline elements* that only take up as much horizontal space as their content requires and can be directly next to other elements.

This is because of the **default value of `display` property** that HTML elements have.

The CSS `display` property provides the ability to change this default display into what we want.

We will cover 3 values for the `display` property:

- `inline`
- `block`
- `inline-block`

`display: inline;`

Inline elements

- have a box that *wraps tightly* around their content, only taking up the amount of space necessary to display their content.
- cannot be altered in size with the `height` or `width` properties.

Currently, our navigation bar on the top of the page has each `` element on a separate line.

This is because `` elements are block-level elements by default.

We want to make the navigation bar look like the one on this GitHub page, all `` elements on one line.

To do this, let's add `class="blog-nav"` to the `` element in our navigation bar and add the following CSS code.

```
.blog-nav li {  
  display: inline;  
}
```

As we learned last time, the above code applies the style specified inside the declaration block to all `` elements that are nested within the `blog-nav` class.

`display: block;`

Block-level elements

- fill the entire width of the page by default, but their width property can also be set.
- unless otherwise specified, they are the height necessary to accommodate their content.

For example, let's try adding this CSS code.

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

By doing this, we can change all `` elements, which are inline by default, to block-level elements.

display: inline-block

The `inline-block` display combines features of both inline and block elements.

Inline-block elements

- can appear next to each other
- can have specific values `width` and `height` properties

Let's create another html file named `movies.html` and copy & paste the following code.

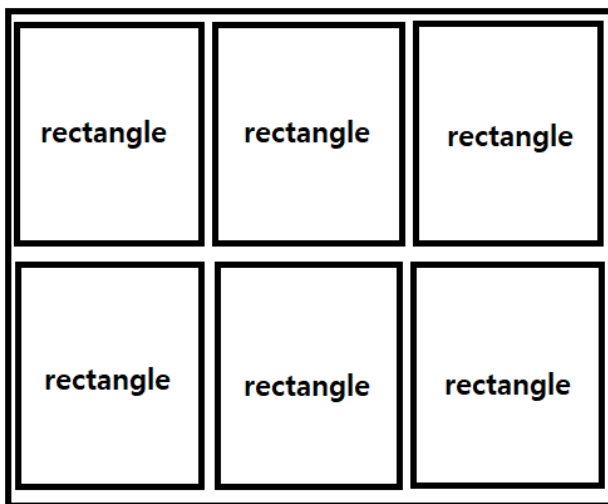
[movies.html](#)

If you look into the code, the overall structure of the page is same as `index.html`, except for the content inside the `main-area`.

The `main-area` contains 6 `<div>` s with `class="rectangle"`.

Each rectangle has a `rectangle-header` and `rectangle-body`.

We want to position each rectangle like below.



div class="main-area"

To do this, we will make the `rectangle` to an `inline-block`, and specify the `width` and `height`.

Let's write the following CSS code.

```
.rectangle {  
  display: inline-block;  
  width: 30%;  
  height: 200px;  
  overflow: hidden;  
}
```

The `width` is set to 30% so that each rectangle takes up about 1/3 of the whole width of its parent element(main-area).

Since the content of the `rectangle-body` is very long and will overflow the rectangle, we will set the `overflow` property to `hidden`.