

A Quick Look at CSS Selectors: Type Selectors

We will put together all we've learned so far to build something cool. Well, not particularly cool, but a good step in the direction of building some more amazing things.

Take a look at the markup below:

```
1 <html>
2   <head></head>
3   <body>
4     <h1> I am a header </h1>
5     <p class="info">Some vital
6     <p id="warning">Dont do th
7     <p class="info">Some other
8     <p class="primary">Basic t
9   </body>
10 </html>
```

What's going on up there?

If you've written a lot of `html`, everything above should make some sense.

Some explanation wouldn't hurt.

Here we go:

Line 4 basically adds an `h1` header to the page.

Line 5 adds a paragraph with a `class` of `info`

Line 6 adds a paragraph too. This time, with an `id` of `warning`

Line 7 is pretty straightforward too. Can you figure that out? It says, "Add a paragraph with a class of `info`"

Line 8 also adds a paragraph to the page but with a class of `primary`

Now we are on the same page. At the end of this lesson you'd have styled the markup as seen below:

Output
HTML

I am a header

Some vital info

Dont do this!

Some other vital info - really?

Basic texts



To be honest, I know this isn't pretty. But trust me, there's a lot of foundational stuff to pick up along the way.

You can't fly without learning to walk, right?

Let's walk with CSS. We'll fly soon!

The Element Selector

If you take a look at the `html` responsible for the layout above, you'll notice we have a couple elements there.

The `h1`, `body` and `p` are all elements. Do you remember how elements are selected and styled? Below is an example:

```
body {  
  background-color: #ccc;  
}
```



In the code above, the `body` element is selected and styled. `body` could be substituted for any other html element.

Let's take a look at specific selectors in CSS.

The Class Selector

If you remember clearly from `html`, a single class name can be applied to multiple elements. In the example at the start of this lesson, we have the following:

```
<p class="info">Some vital info</p>
<p class="info">Some other vital info - really?</p>
<p class="primary">Basic texts</p>
```



To select classes in CSS, you do it like this:

```
.info {
  color: white;
}
```



Did you see that?

You add a `.` before the class name. Then style the selected class whichever way you like.

The single declaration `color: white` will now be applied to every element with the class name, `info`.

The ID Selector

Unlike classes which may be applied to as many elements as possible, an ID is specific to a *single* element.

So how do you select an `Id` in CSS?

```
#warning {
  color: red;
}
```



Yes, that's it.

You add the pound sign, `#` before the `ID` name.

If these are completely new to you, don't worry. With some practice, you'll get the hang of it.

Grouping Selectors

We have seen how to individually target elements, classes, and IDs. What if you want to apply the same CSS declaration to 2 different classes?

For instance, you may want the elements with class names, `info` and `primary` to have the same `background-color` of `black`.

Your first instinct may be this:

```
.info {  
  background-color: black;  
}  
  
.primary {  
  background-color: black;  
}
```



Yes that works. But there is a better way to group elements in CSS.

Take a look:

```
.info,  
.primary {  
  background-color: black;  
}
```



Just add a comma `,` and you can group multiple selectors. As seen above.

Exercise

Now you understand the code that make up our design. See below:

Output

HTML

CSS (SCSS)

```
body {  
  background-color: #ccc;  
}  
  
h1 {  
  color: black;  
}
```

```
.info {
    color: white;
}

.primary {
    color: green;
}

#warning {
    color: red;
}

.info,
.primary {
    background-color: black;
}
```



Now go ahead and re-style this with CSS. Give the classes, elements and Id very different styles.

1. Style the `body` element with a background color of `black`
2. Style the `h1` element with a color of `green`
3. Give the classes, `info` and `primary` colors of `green` and `white` respectively.
4. Style the `Id` of warning with a color of `white`
5. Finally, group the classes `info` and `primary`, give them a background color of blue.

Give yourself a pat on the back when you're done with this 😎😊

See you in the next lesson.