Lesson   Wednesday

Introduction to Programming (/introduction-to-programming) / Git, HTML and CSS (/introduction-to-programming/git-html-and-css) / Styling with Classes and Ids

Text | Cheat sheet

In this lesson, we'll learn about `class` and `id` attributes that we can apply to HTML elements and target in our external CSS stylesheets. A `class` attribute is used to group one or more HTML elements, while an `id` is used to distinguish an HTML element from the rest.

The `class` Attribute

The `class` attribute is used to group one or more HTML elements together under a single name. Let's learn how they work by working through some examples!

Let's make a new page with two types of paragraphs: regular paragraphs, and a summary at the top of the page:

paragraphs.html

```
<!DOCTYPE html>
<html lang="en-US">
<head>
  <link href="css/styles.css" rel="stylesheet" type="text/c
ss">
  <title>Paragraphs!</title>
</head>
<body>
  <h1>Paragraphs of text</h1>
  <h2>Plus an intro at the top!</h2>

  <p>Here is an intro. Lorem ipsum dolor sit amet, consecte
tur adipisicing elit, sed do eiusmod tempor incididunt ut l
abore et dolore magna aliqua.</p>

  <p>And here is the full version. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat. Duis aute irure dolor in reprehende
rit in voluptate velit esse cillum dolore eu fugiat nulla p
ariatur. Excepteur sint occaecat cupidatat non proident, su
nt in culpa qui officia deserunt mollit anim id est laboru
m.Sed ut perspiciatis unde omnis iste natus error sit volup
tatem accusantium doloremque laudantium, totam rem aperiam,
eaque ipsa quae ab illo inventore veritatis et quasi archit
ecto beatae vitae dicta sunt explicabo. Nemo enim ipsam vol
uptatem quia voluptas sit aspernatur aut odit aut fugit, se
d quia consequuntur magni dolores eos qui ratione voluptate
m sequi nesciunt.</p>

  <p>Here is a bit more stuff. At vero eos et accusamus et
iusto odio dignissimos ducimus qui blanditiis praesentium v
oluptatum deleniti atque corrupti quos dolores et quas mole
stias excepturi sint occaecati cupiditate non provident, si
milique sunt in culpa qui officia deserunt mollitia animi,
id est laborum et dolorum fuga. Et harum quidem rerum facil
is est et expedita distinctio. Nam libero tempore, cum solu
ta nobis est eligendi optio cumque nihil impedit quo minus
id quod maxime placeat facere possimus, omnis voluptas assu
menda est, omnis dolor repellendus. Temporibus autem quibus
dam et aut officiis debitis aut rerum necessitatibus saepe
eveniet ut et voluptates repudiandae sint et molestiae non
```

```
recusandae. Itaque earum rerum hic tenetur a sapiente delec
tus, ut aut reiciendis voluptatibus maiores alias consequat
ur aut perferendis doloribus asperiores repellat.</p>

  <p>And, this is the end. Lorem ipsum dolor sit amet, cons
ectetur adipisicing elit, sed do eiusmod tempor incididunt
ut labore et dolore magna aliqua. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat. Duis aute irure dolor in reprehende
rit in voluptate velit esse cillum dolore eu fugiat nulla p
ariatur. Excepteur sint occaecat cupidatat non proident, su
nt in culpa qui officia deserunt mollit anim id est laboru
m.</p>

  <p>Here is a bit about the author at the end. Nam libero
tempore, cum soluta nobis est eligendi optio cumque nihil i
mpedit quo minus id quod maxime placeat facere possimus, om
nis voluptas assumenda est, omnis dolor repellendus.</p>
</body>
</html>
```

The dummy text I put in here is called **lorem Ipsum** text. It looks like Latin but is actually just a bunch of fake words. Lorem ipsum text is often used as a placeholder when doing design; you can search online for *lorem ipsum* and find many websites where you can copy the text.

This page is nice, but I'd like for the top paragraph to look more like an intro. I want it to be italicized, and to be skinnier than the other paragraphs. Here's how we do that without changing the other paragraphs. First, we add a class to the `<p>` tag just for that paragraph:

paragraphs.html

```
<p class="intro">Here is an intro. Lorem ipsum dolor sit am
et, consectetur adipisicing elit, sed do eiusmod tempor inc
ididunt ut labore et dolore magna aliqua.</p>
```

Then, we make a CSS rule for `<p>` tags with the `intro` class:

---

styles.css

```css
.intro {
  width: 50%;
  font-style: italic;
}
```

And now our intro paragraph has a width of only 50% and is italicized, while the rest of the paragraphs stay the same.

We can target any class within our CSS stylesheet by referencing the class name with a period in front of it. In the above example our class is called `intro` and we've targeted it in our stylesheet with `.intro`.

Next, let's add a class to the final paragraph about the author and give it styling:

---

paragraphs.html

```html
<p class="author">Here is a bit about the author at the end. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus.</p>
```

Let's styles it like this to:

---

styles.css

```
p.author {
  width: 75%;
  font-style: italic;
  font-weight: bold;
}
```

Notice that we've targeted our class a little differently: `p.author`. With this CSS selector we're targeting all paragraphs that have `class="author"`. If we had a header with `class=author`, it would not be targeted by the `p.author` CSS selector.

Classes Can Be Applied to Multiple Tags

When we create a `class`, we can use it on multiple tags, not just one. Let's work through another example of a web page that show you how; we'll add `class="important"` to multiple HTML elements:

fido.html

```html
<!DOCTYPE html>
<html lang="en-US">
<head>
  <link href="css/styles.css" rel="stylesheet" type="text/css">
  <title>How to take care of Fido</title>
</head>
<body>
  <h1>Instructions for Fido's babysitter</h1>

  <p>Thanks so much for watching Fido this weekend! Here's everything you need to know while I'm gone.</p>

  <p>Here's a bit of history about Fido. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.</p>

  <p class="important">Fido is alergic to a lot of foods. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>

  <p>Fido likes to play. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.</p>

  <p>Here are a list of things that you should do with Fido every day:</p>

  <ul class="important">
    <li>Walk him</li>
    <li>Feed him</li>
    <li>Pet him</li>
  </ul>
```

```
    <p>If you have any questions while I'm out of town, I <st
rong class="important">won't have cell service</strong>, bu
t you can <a href="mailto:michael@epicodus.com" class="impo
rtant">email me</a> if you need anything.</p>

    <p>Thanks again!</p>
</body>
</html>
```

Here we've added the `important` class to a paragraph, an unordered list, and a `<strong>` tag. Now, we can style everything with the `important` class like this:

fido-styles.css

```css
.important {
  color: red;
}
```

Applying Multiple Classes to One Element

We can apply multiple classes to one element by separating each by a space. For example, if we had an `emphasize` and `highlight` class:

fido-styles.css

```css
.emphasize {
  font-weight: bold;
}

.highlight {
  background-color: yellow;
}
```

We could add then both to an element like so:

```
<p>I <span class="emphasize highlight">love</span> the movi
e "All Dogs Go to Heaven"!</p>
```

The `id` Attribute

The `id` attribute is meant to distinguish one HTML element from all the rest. Because of this, `id` attributes must be unique and only used once in an HTML document. This is in contrast to classes, which are not unique and can be applied to multiple elements.

Let's look at a simple example that compares how to target an `id` and a `class` attribute. In the following example, we'll style two paragraph tags with red text, though one we'll target through a class and the other through an id:

index.html

```
<html lang="en-US">
<head>
  <link href="styles.css" rel="stylesheet" type="text/css">
  <title>Some Exciting Red Text</title>
</head>
<body>
  <p class="intro">This text will appear red.</p>
  <p id="line2">This text will also be red. </p>
</body>
</html>
```

styles.css

```
/* This is an example CSS selector for a class attribute */
.intro {
  color: red;
}

/* This is an example CSS selector for an id attribute */
#line2 {
  color: red;
}
```

As we can see, to target an `id` in our CSS, we'll prefix the `id` name with a hash `#`. For classes, we'll prefix the name of the `class` with a period `.`.

When to Use an `id` or a `class`

So when should we use an `id`, and when should we use a `class`? Use a `class` when you want to group more than one element together. Use an `id` when you want to make an HTML element unique from all other elements.

To explore this distinction more, we can revisit our initial example from this lesson. At the beginning of the lesson we created an `intro` class to style the intro paragraph, and an `author` class to style the paragraph about the author. If we were planning to only use those set of styles once in the HTML document, then we should update those classes to use `id` attributes instead.

However, if our plans were to create a blog site in which we would many post book reviews, and we wanted our `author` and `intro` styles to be applied to every blog post we create, then we should stick to using `class` attributes.

Previous (/introduction-to-programming/git-html-and-css/journal-1-discussion)

Next (/introduction-to-programming/git-html-and-css/practice-styling-with-classes-and-ids)

Lesson 39 of 64
Last updated March 24, 2023

disable dark mode

(http://www.epicodus.com)

© 2023 Epicodus (http://www.epicodus.com/), Inc.