CSS (Cascading Style Sheets)

The road to learning CSS is a bit less traveled than HTML, only 400 million hits…

I was thinking about how to tie together teaching HTML and CSS with the more business elements of the class. What better way then to assume that we are interested in taking our love of bears, finding others that also love bears, and convincing them to join us at a starting a small conference (that we hope will grow to be huge). CSS can help take boring HTML markup, and add style.

All About Bears

Below is our HTML. We have an image, a link, a list and even a copyright notice.
Obligatory (60 lines of CSS styling later)

Here's that same HTML file with a bit of CSS applied. Definitely seems more professional right?

How does this magic happen?

CSS can change the look of HTML elements. In order to do this, CSS must select HTML elements, then apply styles to them.

For those who are curious, here is the index.html file:
<!DOCTYPE html>
<html>
<head>
<link href='https://fonts.googleapis.com/css?family=
<link href='css/style.css' type='text/css' rel='styl
</head>

<body>
<div class="header">
  <div class="container">
    <h1>All About Bears</h1>
    <p>The Conference That Is All About Bears</p>
    <a class="btn" href="#">Learn More</a>
  </div>
</div>

<div class="nav">
  <div class="container">
    <ul>
      <li>Register</li>
      <li>Schedule</li>
      <li>Sponsors</li>
      <li>About</li>
      <li>Contact</li>
    </ul>
  </div>
</div>

<div class="main">
  <div class="container">
    <img src='https://d1o50x50snmhul.cloudfront.net/
    height="128" width="196">
    <h2 class="hotpink bold">The All About Bears Con
    <p>Connect with the best minds in the field and
    challenging bear issues.</p>
    Hear industry leaders talk about what worked

and here is the  style.css file:
html, body {
  margin: 0;
  padding: 0;
}

#byline {
  color: orange;
}

body {
  font-family: 'Roboto', sans-serif;
  font-weight: 100;
}

p {
  color: darkmagenta;
}

.hotpink {
  color: hotpink;
}

.bold {
  font-weight: bold;
}

.container {
  margin: 0 auto;
  max-width: 940px;
  padding: 0 10px;
}

.header {
  background: url(https://images.unsplash.com/photo)
  background-size: cover;
  height: 800px;
  text-align: center;
}

.header .container {
  position: relative;
  top: 200px;
}

.header h1 {
  color: #fff;
  line-height: 100px;
  font-size: 80px;
  margin-top: 0;
  margin-bottom: 80px;
  text-transform: uppercase;
}

@media (min-width:850px) {
  .header h1 {
    font-size: 120px;
  }
}
Most of the time, you want to avoid mixing HTML and CSS code in the same file. So it’s a good practice to have a separate file.
You can create a CSS file by using the `.css` file name extension, like so: `style.css`

It's a good practice to keep your CSS code linked in your HTML file. This doesn't sacrifice the readability and maintainability of your HTML file.

### Linking your CSS File

You can use the `<link>` element to link HTML and CSS files together. The `<link>` element must be placed between the `<head>` of the HTML file.

- **href** — the value of this attribute must be the address, or path, to the CSS file.
- **type** — this attribute describes the type of document that you are linking to. The value of this attribute should be set to `text/css`.
- **rel** — this attribute describes the relationship between the HTML file and the CSS file. Because you are linking to a stylesheet, the value should be set to `stylesheet`.

For example, between our HTML `<head>` tag, a link would look like:

```html
<link href="css/style.css" type="text/css" rel="stylesheet">
```

### Selecting Elements

HTML Elements can be selected in three ways: by **tag**, by **class** and by **id**

#### Selecting by Tag

CSS can select HTML elements by using an element's tag name. A tag name is the word (or character) between HTML angle brackets.

For example, in HTML, the tag for a paragraph element is `<p>`. The CSS syntax for selecting `<p>` elements is:

```css
p {
}
```
Selecting by Element

CSS is not limited to selecting elements by tag name. HTML elements can have more than just a tag name; they can also have attributes. One very common attribute is the class attribute.

You can also select an element by its class attribute.

css has transitions too!

Selecting by ID

CSS is not limited to selecting elements by tag or class either. You can also select an element by its ID attribute. IDs override the styles of tags and classes, so use them sparingly!
Selecting Multiple Classes

It's possible to add more than one class name to an HTML element's class attribute. We can add multiple classes to an HTML element's class attribute by separating them with a space. This enables us to mix and match CSS classes to create many unique styles without writing a custom class for every style combo.

Classes VS. IDs

CSS can select HTML elements by their tag, class, and ID. CSS classes and IDs have different purposes, which can affect which one you use to...
CSS classes (.): Designed to be used over many elements.

CSS IDs (#): an ID is meant to style only one element. IDs override the styles of tags and classes.

**Descendant Selection**

Nested elements can be selected by separating selectors with a space.

**If you only remember a few things:**

- CSS can change the look of HTML elements. In order to do this, CSS must select HTML elements, then apply styles to them.
- CSS can select HTML elements by **tag**, **class**, or **ID**.
- Multiple CSS classes can be applied to one HTML element.
- Classes can be reusable, while IDs can only be used once.
- IDs are more specific than classes, and classes are more specific than tags.
- Multiple selectors can be chained together to select an element. This raises the specificity, but can be necessary.
- Nested elements can be selected by separating selectors with a space.

**Some Recommended CSS Resources:**

- CSS Cheat Sheet: https://adam-marsden.co.uk/css-cheat-sheet
- CSS Typography Reference: https://cssreference.io/typography/