Intro to HTML & CSS

Objectives

By the end of this lesson you will...

- Have a development environment for creating a website

- Be able to define and explain:
    - HTML
    - CSS
    - Bootstrap
    - CDN

- Create a multi-page responsive website
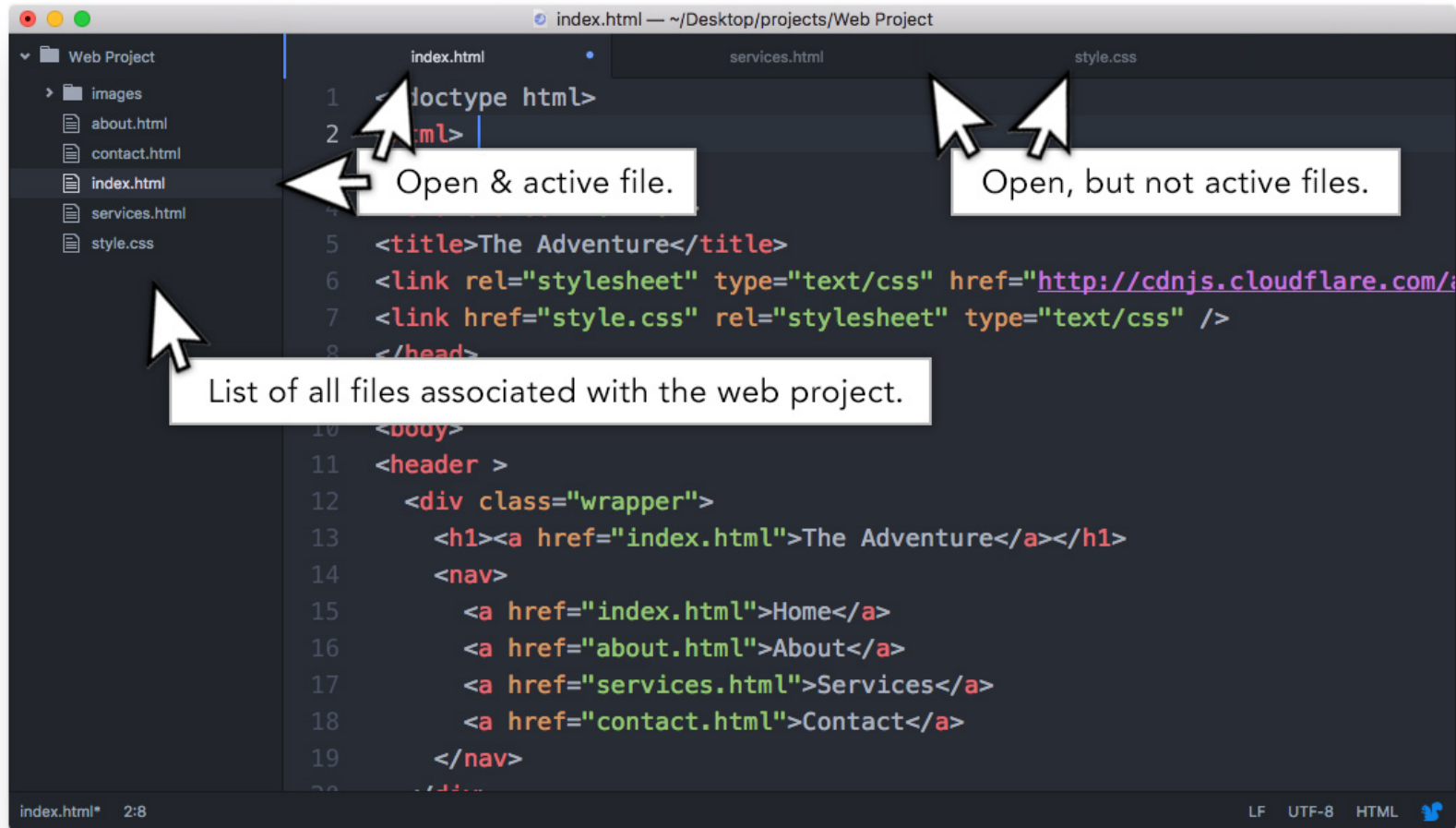
Development Environment

Chrome

Make sure you have Google Chrome installed.

http://google.com/chrome

Atom

- Install Atom: https://atom.io/

- Atom is a text editor that makes writing code much easier!

Check In

- Do you have Atom installed?

- Are you using Chrome as your browser?

Define & Explain

HTML & CSS, The Building Blocks

- HTML – **Hypertext Markup Language** is the structure/bones
- CSS – **Cascading Style Sheets** give your pages their look and feel

Think of a Car

HTML is like the doors, windows, tires. seats.
CSS is like the size, color, and other attributes.

Bootstrap

CSS frameworks, quite simply, are a large collection of CSS rules that play nice with each other. They make it easy for developers to worry less about the visual design of individual html elements, and more about the bigger picture of the page as a whole.

Bootstrap is a front-end framework developed by Twitter.

CDN

Stands for **content delivery network**. It is an online repository of commonly used Javascript and CSS files. It's a good practice to use a CDN in your production code, as many of the files are pre-cached, so your site will actually load faster.

Creating an Online Resume

HTML

Start by adding some basic structure:

```html
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

Type this basic webpage structure into your text editor. Save this file as *index.html*.

Tags

Tags form the structure of your page. They surround and apply *meaning* to content. There usually is an opening tag and then a closing tag, like - `<div></div>` , a divider.

Let's look at four tags:

1. title `<title> </title>`
2. heading `<h1> </h1>`
3. paragraph `<p> </p>`
4. img `<img>`

Title

`<title>` displays the title in the browser toolbar. It's also used for the title when its added to your browser's favorites and the title of your page for search engine results.

Headings

- These include the `<h1>` , `<h2>` , `<h3>` , `<h4>` , `<h5>` and `<h6>` tags.
- `<h1>` is the main heading and the remaining headings decrease in size, with `<h6>` being the smallest.

Paragraph

Paragraphs: The `<p>` tag is used for splitting content into separate paragraphs. Each new `<p>` tag will appear on a new line.

Images

- Some tags, like the `<img>` (image) do not require a closing tag.
- The image tag requires a `src` attribute pointing to the location of the image

Putting it all together

Let's add the tags that were discussed.

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Resume: Darth Vader, Lord of the Sith</title>
  </head>
  <body>
    <h1>Darth Vader</h1>
    <p>Lord of the Sith</p>
    <img src="http://upload.wikimedia.org/wikipedia/en/7/
  </body>
</html>
```

Open the page in Chrome; it should look like this:

Darth Vader

Lord of the Sith

You try it

- Create a page that contains a title, some text, and an image
- Note: http://pixabay.com is a great place to get free, open-source images
- Reminder:
    - title `<title> </title>`
    - heading `<h1> </h1>`
    - paragraph `<p> </p>`
    - img `<img src="myimage.jpg">`

Check In

- Are you able to open index.html?

- Do you see your title?

- Do you see text on the page?

- Do you see your image?

CSS

While HTML provides, structure, CSS is used for styling, making webpages look nice. From the size of the text to the background colors to the positioning of HTML elements, CSS gives you control over almost every visual aspect of a page.

Create a file

In your website directory create *main.css* then type this basic structure in the file

```
.heading {
  color: pink;
}
```

Link index.html to main.css

Next, we need to "link" our HTML page and CSS stylesheet. To do so, add the following code to the `<head>` section of the HTML page just below the title:

```
<link rel="stylesheet" href="main.css">
```

Reference the CSS class

use the class attribute to reference the class that you created in the main.css file.

```html
<h1 class="heading">Darth Vader</h1>
```

Your code should now look similar to this:

```html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="main.css">
    <title>Resume: Darth Vader, Lord of the Sith</title>
  </head>
  <body>
    <h1 class="heading">Darth Vader</h1>
    <p>Lord of the Sith</p>
    <img src="http://upload.wikimedia.org/wikipedia/en/7/
  </body>
</html>
```

Look at your site in Chrome

- Save the file.

- Check it out in your browser.

Do you see the difference?

Selectors, Properties, & Values

```css
body{
  background-color: beige;
}

img {
  border: 1px solid black;
}

.card{
  text-align: center;
  margin: 10px;
  padding: 5px;
}

.heading{
  color: pink;
}
```

CSS Values

http://cssvalues.com/ is an excellent resource for finding the acceptable values of a CSS property.

You try it

- Put a border on your image, change the color of your header text, explore additional styling.

- Reminder:
  - `border: 1px solid black;`
  - `color: red;`
  - http://cssvalues.com/

Check In

- Do you see the text color change?

- Do you see a border on your image?

- What styles did you explore and find interesting?

Connect index.html to resume.html

Somewhere in the body, add the following code:

```html
<a href="resume.html">Resume</a>
```

Your code should now look similar to this:

```html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="main.css">
    <title>Resume: Darth Vader, Lord of the Sith</title>
  </head>
  <body>
    <h1 class="heading">Darth Vader</h1>
    <p>Lord of the Sith</p>
    <img src="http://upload.wikimedia.org/wikipedia/en/7/
    <a href="resume.html">Resume</a>
  </body>
</html>
```

You Try it - Create a resumé page

- Create a page called resume.html

- include the html,head,title, and body tags

- In the head, link to the css file

- In the body, add the text "Hello World" in a tag with the class "heading"

Check In

- Can you click the link from index.html to get to resume.html?

- Do you see Hello World on the resume.html page?

- Is the Hello World text the color of your heading CSS class?

Bootstrap

Add the Bootstrap link to the head of your HTML pages.

```
<link rel="stylesheet" href="http://netdna.bootstrapcdn.c
```

Take a look at your pages. That's just from adding one line of code!

Bootstrap Grid System

The grid system allows you to easily organize your content into a responsive design based around a 12-column grid. In order to make use of this grid, we need to wrap the main area of our content in one of Bootstrap's container classes: either .container or .container-fluid:

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>My Resume</title>
  </head>
  <body class="container-fluid">
    <div>Hello, World!</div>
  </body>
</html>
```

Rows and Columns

To make use of the grid layout, we can add column classes to our divs. Every column needs to be inside of a div with the row class in order for Bootstrap's styling magic to work.

```html
<div class="row">
  <div class="col-md-12 bg-primary">Hello, World!</div>
</div>
```

Let's put it all together

```html
<div class="container-fluid">
  <div class="row">
    <div class="col-md-12 bg-primary">Hello, World!</div>
  </div>
  <div class="row">
    <div class="col-md-6 bg-success">Hello,</div>
    <div class="col-md-6 bg-danger">World!</div>
  </div>
  <div class="row">
    <div class="col-md-2 bg-info">Hi!</div>
    <div class="col-md-2 bg-warning">Hi!</div>
    <div class="col-md-2 bg-info">Hi!</div>
    <div class="col-md-2 bg-warning">Hi!</div>
    <div class="col-md-2 bg-info">Hi!</div>
    <div class="col-md-2 bg-warning">Hi!</div>
  </div>
</div>
```

You Try it

- Put the grid into your resume.html page
- Explore what happens if you change the column sizes
- Create the grid layout you need for your resume

Check In

- What surprised you about the grid system?

- What challenges are you experiencing?

Final Site

Thank you!