Monday - 6/5/2024

Front-end and Back-end Development

- The frontend refers to the graphical user interface (GUI) that your users can directly interact with, such as navigation menus, design elements, buttons, images, and graphs.
- In technical terms, a page or screen that your user sees with several UI components is called a document object model (DOM).
- Three primary computer languages affect how your users interact with your frontend:
    - HTML defines the frontend structure and the different DOM elements.
    - Cascading Style Sheets (CSS) defines the style of a web application, including layout, fonts, colors, and visual style.
    - JavaScript adds a layer of dynamic functionality by manipulating the DOM.
- JavaScript can trigger changes on a page and display new information.
- This means the frontend can handle fundamental user interactions (or requests), like displaying a calendar or checking if your user has entered a valid email address.
- The frontend passes on more complex requests to the backend.
- Essentially, the difference between frontend and backend web development is that frontend serves the client side (what we see on the front i.e. a screen) and backend supports the server side (what's under the hood of a website).
- The backend (or "server side") is the portion of the website you don't see.
- It's responsible for storing and organizing data, and ensuring everything on the client-side actually works.
- The backend communicates with the frontend, sending and receiving information to be displayed as a web page.
- Whenever you fill out a contact form, type in a web address, or make a purchase (any user interaction on the client-side), your browser sends a request to the server side, which returns information in the form of frontend code that the browser can interpret and display.

Front-end Programming Languages

- HTML
- CSS

- JavaScript
- Node.js
- AJAX
- Design frameworks like Foundation and Bootstrap
- Libraries like AngularJS, jQuery, React
- APIs

Back-end Programming Languages

- PHP
- Ruby
- Python, Django
- Java
- SQL

Curriculum Breakdown

Month 1
Week 1 - HTML + CSS + Git + GitHub + GitHub Pages
Week 2 - 4 - JavaScript

Month 2
Week 1 - 3 - React JS
Week 4 - Capstone projects

Month 3
Week 1 - 3 - Python
Week 4 - Django

Month 4
Week 1-2 - Django
Week 3-4 - Final Capstone Project

Month 5
Week 1-2 - Final Capstone Project

HTML

Introduction

- HTML is the language in which most websites are written.
- HTML is used to create pages and make them functional.
- The code used to make them visually appealing is known as CSS
- HTML was first created by Tim Berners-Lee, Robert Cailliau, and others starting in 1989. It stands for Hyper Text Markup Language.
- Hypertext means that the document contains links that allow the reader to jump to other places in the document or to another document altogether.
- A Markup Language is a way that computers speak to each other to control how text is processed and presented. To do this HTML uses two things: tags and attributes.

Tags and Attributes

- They work together but perform different functions.
- Tags mark up the start of an HTML element and are usually enclosed in angle brackets. An example of a tag is: <h1>.
- Most tags must be opened <h1> and closed </h1> to function.
- Attributes contain additional pieces of information.
- Attributes are an opening tag and additional info is placed inside.
- An example of an attribute is:

- In this instance, the image source (src) and the alt text (alt) are attributes of the <img> tag.

HTML Inline and Block Elements

- HTML elements can be broadly categorized into one of two categories:
    - Inline Elements: , <a>, , <img> etc.
    - Block Elements: <p>, , <h1>, <figure> etc.

Inline Elements

- Inline elements are displayed on the same line.
- They do not start on a new line and take up only as much width as their contents require.
- An example of an inline element is the tag.

This is how span works.

- If we have multiple span tags, they get placed on the same line. For example,
   The following spans will be inline; Span 1 and Span 2.
- Some examples of inline elements are:
  - HTML <a> tag
  - HTML <input> tag
  - HTML tag
- Most HTML formatting tags are also inline. For example:
  - HTML <b> tag
  - HTML tag
  - HTML tag, etc

Block Elements

- Block elements take up the whole horizontal space available in its container.
- They start on a new line and take up as much height as their contents require.
- An example of a block element is the HTML Paragraph Tag.
  This is how block elements works.
- If we have multiple block elements, they will each take a separate line. For example,
  Paragraphs are Block Elements.
- Some frequently used Block elements are:
  - HTML tag
  - HTML Headings <h1> - <h6>
  - HTML tag, etc

Create Your First HTML Webpage

- These tags should be placed underneath each other at the top of every HTML page that you create.

  <!DOCTYPE html> — This tag specifies the language you will write on the page. In this case, the language is HTML 5.

<html> — This tag signals that from here on we are going to write in HTML code.

<head> — This is where all the metadata for the page goes — stuff mostly meant for search engines and other computer programs.

<body> — This is where the content of the page goes.

- Inside the <head> tag, there is one tag that is always included: <title>, but there are others that are just as important:

  <title>

- This is where we insert the page name as it will appear at the top of the browser window or tab.

  <meta>

- This is where information about the document is stored: character encoding, name (page context), description.
- Character encoding is a method of converting bytes into characters. To validate or display an HTML document properly, a program must choose a proper character encoding.
- The <meta> tag defines metadata about an HTML document.
- Metadata is data (information) about data.
- <meta> tags always go inside the <head> element and are typically used to specify the character set, page description, keywords, author of the document, and viewport settings.
- Metadata will not be displayed on the page.
- Metadata is used by browsers (how to display content or reload page), search engines (keywords), and other web services.
- There is a method to let web designers take control over the viewport (the user's visible area of a web page), through the <meta> tag.

Add Content

- The HTML <body> is where we add the content which is designed for viewing by human eyes.

HTML tag

- The tag is an empty container that is used to define a division or a section.
- It does not affect the content or layout and is used to group HTML elements to be styled with CSS or manipulated with scripts.

```
<body>

    <div>

        <h1>Sample Web Page</h1>

        <p>This is my first Web Page</p>

    </div>
```

</body>

HTML Headings

- <h1> Coding is fun! </h1>
- <h2> Coding is fun! </h2>
- <h3> Coding is fun! </h3>
- <h4> Coding is fun! </h4>
- <h5> Coding is fun! </h5>
- <h6> Coding is fun! </h6>

- As you might have guessed <h1> and <h2> should be used for the most important titles, while the remaining tags should be used for sub-headings and less important text.
- Search engine bots use this order when deciphering which information is most important on a page.

Add Text to HTML

- Adding text to our HTML page is simple using an element opened with the tag <p> which creates a new paragraph.
- We place all of our regular text inside the element <p>.
- When we write text in HTML, we also have several other elements to control the text or make it appear in a certain way.

Element	**Meaning**	**Purpose**
<b>	Bold	Highlight important information
	Strong	Similarly to bold, to highlight key text
<i>	Italic	To denote text
	Emphasized Text	Usually used as image

		captions
<mark>	Marked Text	Highlight the background of the text
<small>	Small Text	To shrink the text
<strike>	Striked Out Text	To place a horizontal line across the text
<u>	Underlined Text	Used for links or text highlights
<ins>	Inserted Text	Displayed with an underline to show an inserted text
<sub>	Subscript Text	Text appears half a character below the normal line
<sup>	Superscript Text	Text appears half a character above the normal line

Add Links to HTML

- Almost everything you click on while surfing the web is a link that takes you to another page within the website you are visiting or to an external site.
- Links are included in an attribute opened by the <a> tag (Anchor Tag)
- This element is the first that we've met which uses an attribute that looks different to the previously mentioned tags.

  Zindua School

- The first part of the attribute points to the page that will open once the link is clicked.
- Meanwhile, the second part of the attribute contains the text that will be displayed to a visitor to entice them to click on that link.
- Internal links

  About Page

Add Images

- The <img> tag has everything you need to display images on your site. Much like the <a> anchor element, <img> also contains an attribute.
- The attribute features information for your computer regarding the source, height, width, and alt text of the image.
- Alt text is important to ensure that your site is ranked correctly on search sites and also for visually impaired visitors to your site.

```
<img src="zindua.png" alt="Zindua School Logo">
```

Semantic Elements

- HTML5 introduced semantic elements, which provide more meaning to the content and structure of a web page.
- Some common semantic elements include:
    - <header>: Represents the header of a document or section.
    - <nav>: Represents a section of a page that contains navigation links.
    - <main>: Represents the main content of a document.
    - <article>: Represents a self-contained composition in a document, such as a blog post or a news story.
    - <section>: Represents a standalone section of a document, such as a chapter or a tabbed content area.
    - <aside>: Represents a section of a page that contains content that is tangentially related to the main content.
    - <footer>: Represents the footer of a document or section.

Exercise

1. Create a personal portfolio with different pages
    a. About Page
    b. Education Page
    c. Interests/Hobbies Page
2. Link the various pages to each page.
3. Make use of images and various formatting elements.