FULLSTACK COHORT 3

Introduction to HTML

JOAQUIN ANTONIO

Session 1

What is HTML?

1. HTML: HyperText Markup Language.
2. It is the standard markup language used to create web pages.
3. Allows you to add content, structure, and formatting to your web pages, making them easy to read and navigate.
4. HTML is the foundation of every website and is essential for web development.

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1
2017	W3C Recommendation: HTML5.1 2nd Edition
2017	W3C Recommendation: HTML5.2

New features in HTML 5:

1. New tags
2. New properties of existing tags
3. Includes plugins
4. Improved API’S
5. Browser storages
6. Meta tags for responsiveness

The Structure of an HTML Document

1. Every HTML document consists of two parts: the head and the body.
    1. The head contains information about the document, such as the title and meta data.
    2. The body contains the content of the document, such as text, images, and links.
2. Understanding the structure of an HTML document is essential for creating well-organised and functional web pages.

Types of tags:

There are two types of tags in HTML.

1. Paired tags:

```
<h1>.....</h1>
```

2. Single tags:

```
<br>
```

Session 2

Creating a Basic HTML Document

1. HTML is the foundation of every web page.
2. In this session, we will go through how to create a basic HTML document, including:
    1. how to structure your content using headings and paragraphs
    2. how to add images and links
    3. how to use lists to organise your content

Understanding HTML Document Structure

1. When creating an HTML document, it's important to understand its structure:
    1. The declaration defines the document type
    2. The element is the root of the document
        1. The element contains meta information, such as the page title and character encoding and the visible content of the page.
2. Let's explore these elements in more detail

HTML structure

```
<!DOCTYPE html>
<html>
<head>
    <title>My first HTML page</title>
</head>
<body>
     <h1>Hello World!</h1>
     <p>My first paragraph</p>
      </br>
     <p>My second paragraph</p>
</body>
</html>
```

Creating the Basic Skeleton of an HTML Page

1. Before adding content to your HTML page, you need to create its basic structure.

2. This includes defining the document type, adding an HTML element, and including the head and body sections. we'll walk through each step of this process.

Common HTML Elements and Tags

1. HTML provides a wide range of elements and tags to structure and format your content.
2. Some of the most commonly used ones include headings, paragraphs, lists, links, images, and tables.
3. By using these elements and tags effectively, you can create visually appealing and easy-to-read web pages.

HTML Tags

1. Heading Tags
2. Defines headings of different levels, with <h1> being highest level and <h6> the lowest.
    1. `<h1>` to `<h6>`

`Syntax:`

`<h1>.....</h1>`

HTML Tags

1. Paragraph Tag
2. Defines a paragraph of text.

    1.

Syntax:

…….

HTML Tags

1. Division tag
2. Used to group content for styling or scripting purposes.
    1.

Syntax:

```
<div>
        content
</div>
```

HTML Tags

These tags are used to create tabular data:

1. Table
    1. `<table>`
2. Table row
    1. `<tr>`
3. Table data
    1. `<td>`

```
Syntax:

<table>

        <tr>

                <td>

                </td>

         </tr>

</table>
```

Adding Content to the Element

1. Once you've created the basic structure of your HTML page, it's time to add content to the element.

2. This is where you'll include all the visible elements, such as headings, paragraphs, images, and more.

3. Let's dive into how to add and structure content within the body of your web page.

Session 3

Text Formatting with HTML

1. In this session, we'll cover how to add emphasis, create lists, and format text using HTML tags.
2. We'll also show you how to add links to your text and how to use special characters.
3. By the end of this session, you'll be able to add professional-looking text to your web pages with ease.

Text Formatting with HTML

1.……

2. …….

3. …….

4. ……

5. <u>……</u>

6. <s>……</s>

Using Headings in HTML

1. Headings are important for organising your content and making it easier for users to read.
2. In HTML, you can use six different levels of headings
3. For a subheading. we'll show you how to choose the right heading level and how to use them effectively in your web pages.

Using Paragraphs in HTML

1. Paragraphs are the building blocks of text in HTML.

2. They allow you to break up your content into smaller, more manageable chunks that are easier for users to read.

3. We'll show you how to use the tag to create paragraphs and format your text for maximum readability.

Using Line Breaks in HTML

1. Line breaks are a simple but important part of formatting your text in HTML.
2. They allow you to create white space between lines of text, which can help improve readability and make your content easier to scan.
3. We'll show you how to use the tag to add line breaks to your web pages.

    1. `<br>`

Creating Lists in HTML

1. Lists are an effective way to present information in a structured and organised manner.

2. We'll show you how to use HTML to create both ordered and unordered lists, and provide tips on how to use them effectively to improve the readability of your content.

Creating Lists in HTML

These tags are used to create numbered lists.

1. Ordered list

   1. `<ol>`

2. List item

   1. `<li>`

Syntax:

```
<ol>
        <li>
            </li>
</ol>
```

Creating Lists in HTML

These tags are used to create bulleted lists:

1. Unordered list

    1. `<ul>`

2. List item

    1. `<li>`

```
Syntax:

<ul>

    <li>

    </li>

</ul>
```

Session 4

Adding Images and Links in HTML

1. Images and links are essential elements for creating engaging and informative content on your website.

2. We'll show you how to use HTML to add images and links to your pages, and provide tips on how to optimise them for maximum impact.

Inserting Images in HTML

1. Enhance the visual appeal of your webpages by adding images using the tag in HTML.
2. Learn how to specify image sources, set alt text for accessibility, and adjust image size and alignment for a seamless user experience.
3. Image tag: used to embed images in the document.
    1. `<ul>`
4. Syntax:
5. `<img src = “”/>`

Creating Hyperlinks in HTML

1. Add interactivity to your webpages by creating hyperlinks using the tag in HTML.
2. Learn how to link to other pages within your site, external websites, and downloadable files.
3. Also, discover how to style links with CSS for a consistent look and feel.

Creating Hyperlinks in HTML

1. Anchor tag
   1. <a>
2. Used to create hyperlinks to other web pages or resources.

Syntax:

Linking to External Websites

1. Make your webpages more informative by linking to external websites.

2. Learn how to: open links in a new tab, how to link to specific sections on a page, and how to use anchor tags for email links.

3. Also, discover best practices for linking to external content.

Linking Within the Same Page (Anchors)

1. Make your webpages more user-friendly by using anchors to link to specific sections on the same page.

2. Learn how to create smooth scrolling, how to add anchor links to your navigation menu, and how to use JavaScript to enhance anchor functionality.

Session 5

HTML Forms: Capturing User Input

1. Learn how to create interactive web forms using HTML.
2. Explore different form elements like text inputs, checkboxes, radio buttons, and dropdown menus.
3. Discover how to style and validate forms to enhance user experience.

Introduction to HTML Forms

1. Discover the power of HTML forms and how they allow users to interact with your website.

2. Learn how to create form elements like input fields, buttons, and checkboxes, and explore the different attributes and properties that can be used to customise and validate user input.

Introduction to HTML Forms

1. The Form tag is used to create interactive forms for user input
    1.

Syntax:

```
<form>
            content
</form>
```

Text Inputs, Radio Buttons, Checkboxes, and Dropdown Menus

1. These are some of the most commonly used form elements.

2. Learn how to create and customise these elements using HTML, and explore how to use JavaScript to enhance the user experience with these inputs.

Text Inputs, Radio Buttons, Checkboxes, and Dropdown Menus

1. These tags are used within forms:
2. Input field
    1. <input>
3. Textarea
    1. <textarea>
4. Dropdown selection tag
    1. <select>

Syntax:

```
<form>
    <input type="text">


    <textarea cols="30“rows="10"></textarea>


    <select >
        <option value=""></option>
    </select>
    </form>
```

Submitting and Processing Form Data

1. Learn how to process and validate user input with server-side scripting languages like PHP or Python.

2. Discover how to store and retrieve user data from databases, and explore how to send confirmation emails to users.

Session 6

HTML Comments

1. HTML comments are a way to add notes to your code that are not visible on the webpage.
2. They can be used to provide explanations, make temporary changes, or remove sections of code without deleting them.
3. Learn how to use HTML comments effectively in your web development projects.

Syntax:

```
<!-- this is comment -->
```

Adding Comments to HTML Code

1. Learn how to add comments to your HTML code to keep track of what you've done, why you've done it, or to temporarily disable sections of code.

2. This simple technique can save you time and improve the readability and maintainability of your code.

The Purpose of Comments in Web Development

1. Comments are an essential part of any programming language, including HTML.
2. They can help you keep track of your code and make it more readable for others.
3. Learn about the different types of comments in web development and how you can use them to improve your code.
    1. Documenting Code
    2. Temporarily Disabling Code
    3. Providing Context for Styling or Scripting
    4. Collaboration
    5. License Information

Session 7

Why is HTML Validation Important?

1. HTML validation is the process of checking if your HTML code follows the rules and standards set by the World Wide Web Consortium (W3C).

2. Validating your HTML code ensures that your website will display correctly across different browsers and devices, improving accessibility and user experience.

The Benefits of HTML Validation

1. By validating your HTML code, you can catch errors and ensure that your website meets industry standards.

2. This helps improve search engine optimization, enhances website performance, and provides a better user experience.

3. Don't underestimate the power of clean and error-free HTML code!

Using Online Validation Tools

1. Take advantage of online HTML validation tools to easily check your code for errors and ensure compliance with web standards.

2. These tools provide instant feedback, helping you identify and fix issues quickly, resulting in a more reliable and well-optimised website.

Session 8

Building a Basic Web Page

1. Learn the fundamentals of creating a web page from scratch.
2. Understand HTML structure, add content, and apply basic styling using CSS.
3. Get started on your web development journey and unleash your creativity!

Combining HTML Elements for a Cohesive Web Page

1. Explore the power of combining HTML elements to create a visually appealing and user-friendly web page.

2. Learn how to structure your content effectively, utilise headings, paragraphs, lists, and more to enhance the readability and organisation of your web page.

Enhancing Your Web Page with Content, Formatting, Images, and Links

1. Take your web page to the next level by adding engaging content, applying text formatting to make it stand out, incorporating eye-catching images, and including relevant links for further exploration.

2. Learn how to captivate your audience and create an immersive web experience.

FULLSTACK COHORT 3

Introduction to CSS

Session 1

What is CSS and Why is it Essential for Web Design?

1. CSS, or Cascading Style Sheets, is a fundamental technology for styling and formatting web pages.
2. It allows designers to control the look and feel of a website:
    1. layout
    2. colours
    3. fonts and more.
3. With CSS, web designers can create visually appealing and consistent user experiences across different devices and platforms.

Inline, Internal, and External CSS

1. When it comes to applying CSS styles to your web pages, you have three options:
    1. Inline CSS
        1. Applied directly to HTML elements
    2. Internal CSS
        1. Embedded within the HTML document
    3. External CSS
        1. Stored in a separate file.
2. Each option has its own advantages and use cases

Linking CSS to HTML Documents

1. To apply external CSS styles to your HTML documents, you need to link the CSS file to your HTML file using the <link> tag.

2. This allows you to maintain a separation between your HTML structure and CSS styles

3. Makes it easier to update and manage your website's design.

Session 2

CSS Syntax and Selectors

1. Understanding CSS syntax and selectors is crucial for creating well-designed and visually appealing web pages.

2. Syntax: refers to the rules and conventions that dictate how CSS code is written.

3. Selectors: used to target specific HTML elements and apply styles to them.

CSS Rule Structure: Selectors, Properties, and Values

1. In CSS, a rule consists of a selector, followed by a set of properties and their corresponding values.
2. Selectors target specific HTML elements, while properties define the styles to be applied.
3. Understanding the structure of CSS rules is essential for effectively styling your web pages.

Understanding CSS Selectors

1. In CSS, selectors are used to target specific HTML elements for styling.
2. There are different types of selectors. Each type has its own purpose and usage:
    1. Element selectors
    2. Class selectors
    3. ID selectors.
3. Let's explore these selectors and how they can enhance your CSS styling.

Combining CSS Selectors for More Specific Styling

1. Combining CSS selectors allows you to target HTML elements with even greater precision.
2. By using multiple selectors in a single rule, you can apply styles to elements that match all of the selectors.
3. This technique is particularly useful when styling complex web pages.
4. Let's dive into how to combine selectors in CSS.

Session 3

Text Styling with CSS

1. Understanding how to use CSS to style text is essential for creating visually appealing web pages.
2. CSS can be used to style text in a variety of ways, including changing the:
    1. font
    2. colour
    3. size
    4. spacing.
3. Let's explore some common techniques for text styling in CSS.

Changing Text Colour, Font Size, and Font Family with CSS

1. In CSS, you can easily change the colour, font size, and font family of text to enhance the visual appeal of your web pages.

2. By using CSS properties like colour, font-size, and font-family, you can customise the appearance of text to match your design requirements.

3. Let's dive into how to modify these text properties using CSS.

Adding Emphasis to Text with CSS

1. In CSS, you can add emphasis to your text by using properties like font-weight, font-style, and text-decoration.

2. These properties allow you to make your text bold, italicised, or underlined, helping you to highlight important information or create visual interest.

3. Let's explore how to emphasise text using CSS.

Controlling Text Alignment and Spacing with CSS

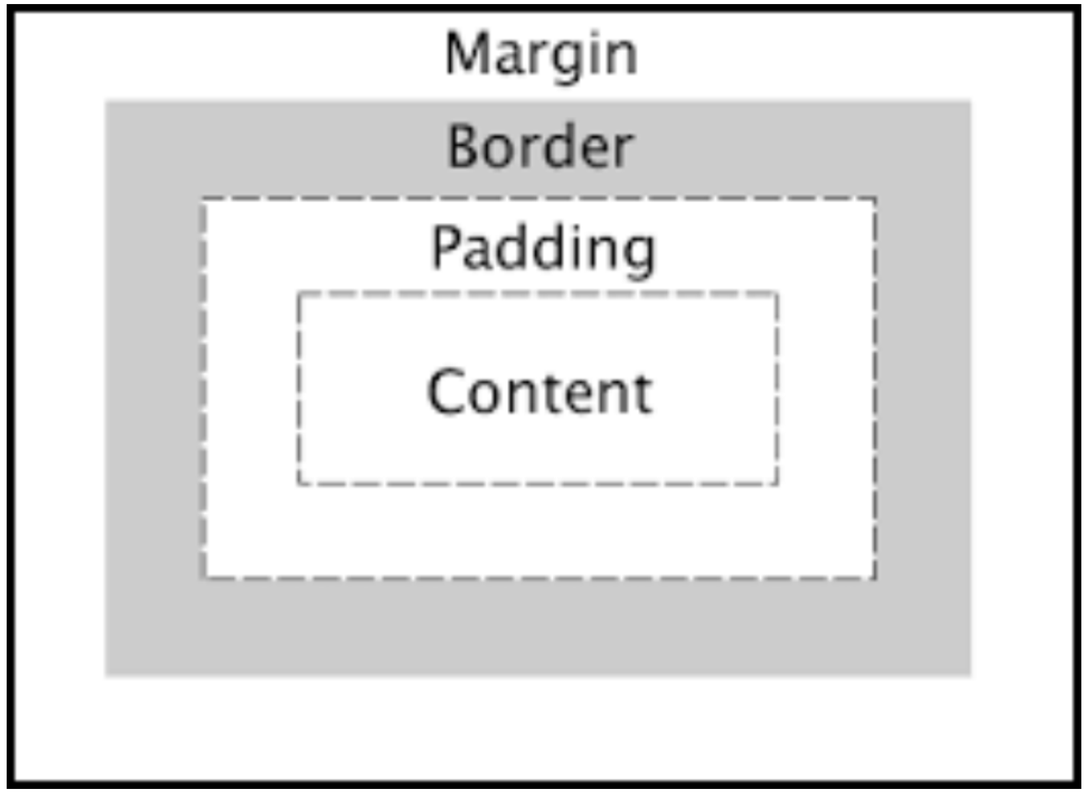
1. In CSS, you have control over how your text is aligned and spaced on the web page.
2. With properties like text-align, line-height, and letter-spacing, you can adjust the alignment, spacing, and readability of your text.
3. Let's learn how to fine-tune text alignment and spacing using CSS.

Session 4

Understanding the Box Model and Layout in CSS

1. In CSS, every element on a webpage is treated as a box.
2. The box model is a fundamental concept that determines how these boxes are sized, spaced, and positioned.
3. By understanding the box model and layout techniques, you can create well-structured and visually appealing web designs.
4. Let's explore the box model and layout in CSS.

Understanding the Box Model and Layout in CSS



Understanding the Box Model and Layout in CSS

1. Content:
    1. Actual content of your element (eg. text, images, any other HTML elements)
2. Padding:
    1. The space between the content and the element’s border. Purpose: control space inside the element
3. Border:
    1. Surrounds padding and content. It can have a specific style, colour, width.
4. Margin:
    1. The space between the border of an element and the surrounding elements. Purpose is to help control the spacing inside the element.

Controlling Element Dimensions: Width and Height

1. When designing webpages, it's crucial to have control over the dimensions of your elements.
2. With CSS, you can specify the width and height of elements, ensuring they fit perfectly into your layout.
3. Let's explore how to control element dimensions with CSS.

Positioning Elements: Relative, Absolute, and Fixed

1. Positioning elements in CSS is essential for creating complex layouts.
2. Learn about static, relative, absolute, and fixed positioning, and how they affect the position of your elements on the webpage.
3. Take control of your design with CSS positioning!

Session 5

Working with Colours and Backgrounds

1. Colours and backgrounds play a crucial role in web design.

2. Learn how to choose the perfect colour palette, apply colours to text and backgrounds, create gradients, and use background images to enhance your website's visual appeal.

3. Let your creativity shine with CSS colour and background properties.

Applying Background Colors and Images

1. Elevate your web design with stunning backgrounds.

2. Discover how to set background colours, use background images, and apply different background properties to create visually appealing webpages.

3. Unleash your creativity and make your website stand out with CSS background techniques.

Specifying Foreground (Text) Colors

1. Take your web design to the next level by choosing the perfect text colours.

2. Learn how to specify text colors using CSS, explore different colour formats, and understand the importance of contrast for readability.

3. Create visually appealing and accessible websites with well-chosen text colors.

Adding Gradients and Transparency

1. Elevate your web design with the power of gradients and transparency.

2. Discover how to create stunning gradient backgrounds, apply transparency to elements, and add visual depth to your website.

3. Learn the techniques to make your designs stand out with CSS gradients and transparency.

Session 6

Understanding CSS Flexbox

1. Discover the power of CSS Flexbox and revolutionise your website layouts.
2. Learn how to create flexible and responsive designs, align and distribute elements, and easily create complex layouts.
3. Take your web design skills to the next level with the flexibility and versatility of CSS Flexbox.

Introduction to CSS Flexbox: Creating Flexible Layouts

1. CSS Flexbox is a powerful tool for creating flexible and responsive layouts that adapt to different screen sizes and devices.

2. Learn how to use Flexbox to create complex layouts with ease, control the positioning and alignment of elements, and build websites that look great on any device.

Exploring Flex Containers and Flex Items

1. Take your CSS layouts to the next level with Flex containers and flex items.

2. Understand how to create flexible and responsive designs by manipulating the properties of flex containers and flex items.

3. Learn how to control the layout, alignment, and distribution of elements using CSS Flexbox.

Controlling Alignment and Order in CSS Flexbox

1. Discover how to fine-tune the alignment of flex items within flex containers using CSS Flexbox.
2. Learn techniques to align items horizontally and vertically, and explore the power of the order property to rearrange elements dynamically.
3. Take full control of your flexible layouts with CSS Flexbox!

Session 7

Exploring CSS Grid Layout

1. Take your web design skills to the next level with CSS Grid Layout.
2. Learn how to create complex and responsive grid-based layouts with ease.
3. Discover the power of grid containers, grid items, and grid properties to achieve highly customisable designs.
4. Master the art of CSS Grid Layout!