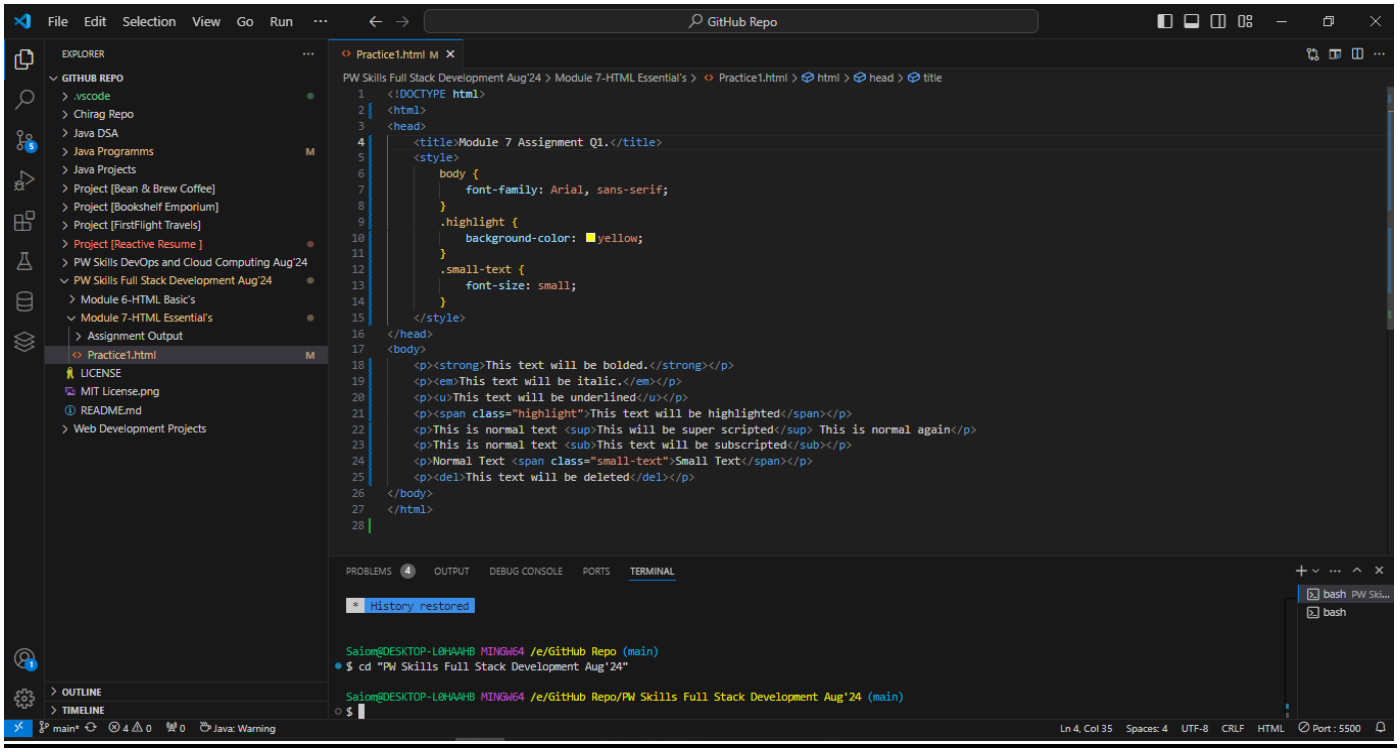
Module -7 Assignment

Q1.) Build a simple webpage that displays text as shown in the below image.

Output-



This text will be bolded.

This text will be italic.

This text will be underlined

This text will be highlighted

This is normal text This will be super scripted This is normal again

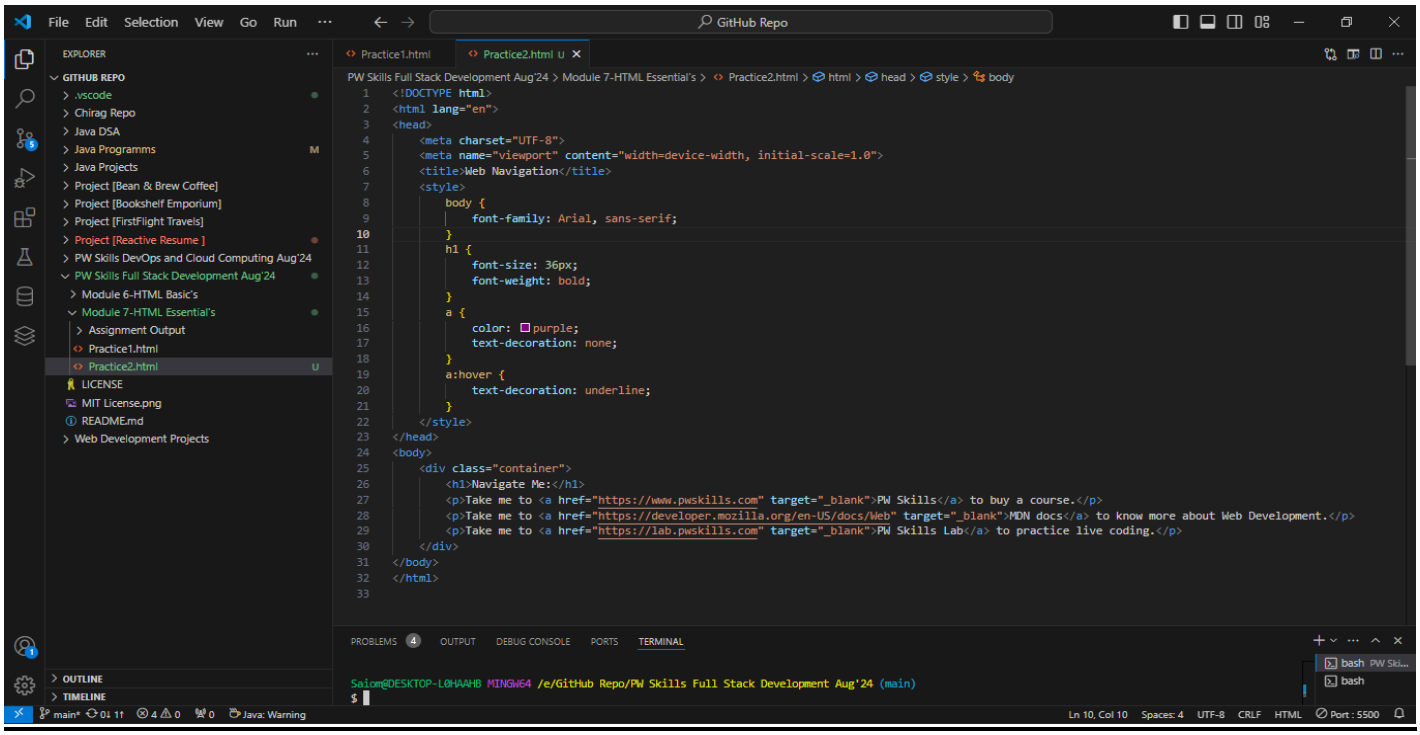
This is normal text This text will be subscripted

Normal Text Small Text

~~This text will be deleted~~

Q2.) Build a simple webpage that helps users navigate different web development-related websites. Note: On clicking the hyperlink, the web pages should open in a new tab. Below is a reference image.

Output-



Navigate Me:

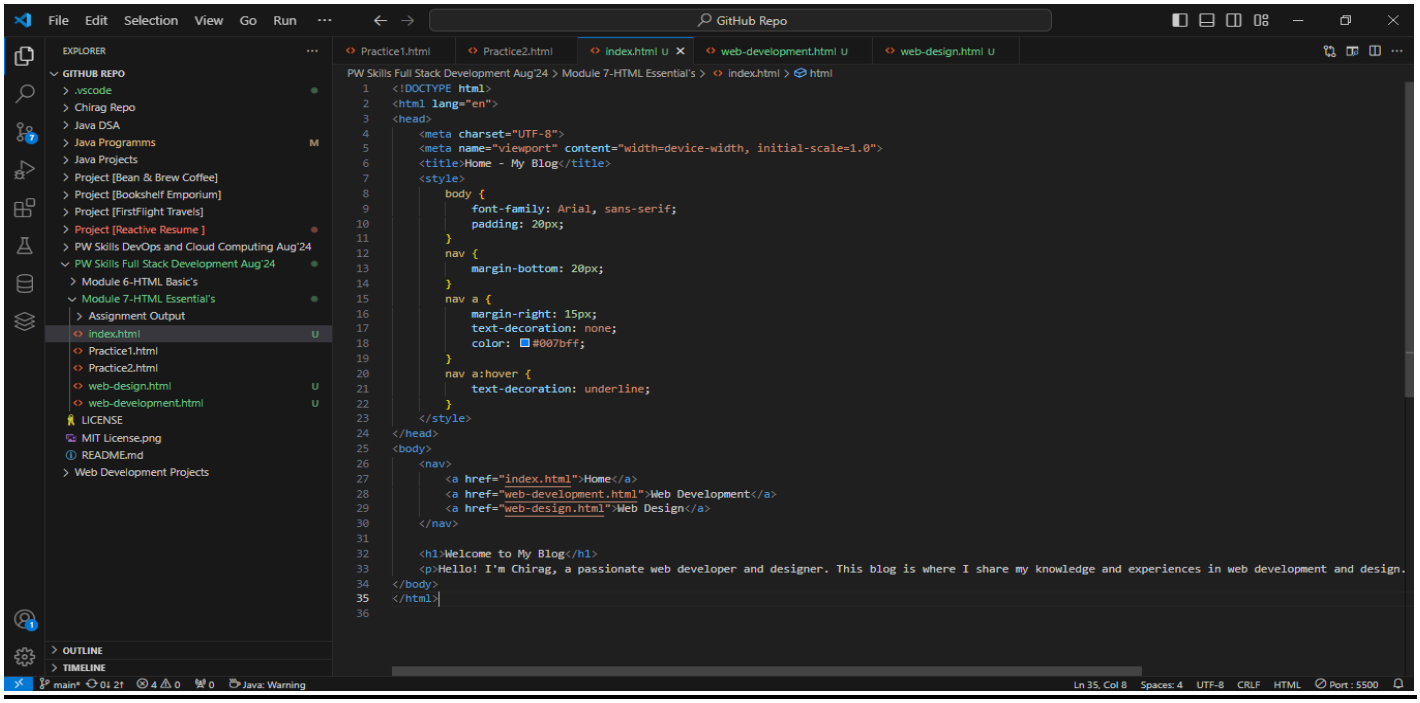
Take me to PW Skills to buy a course.

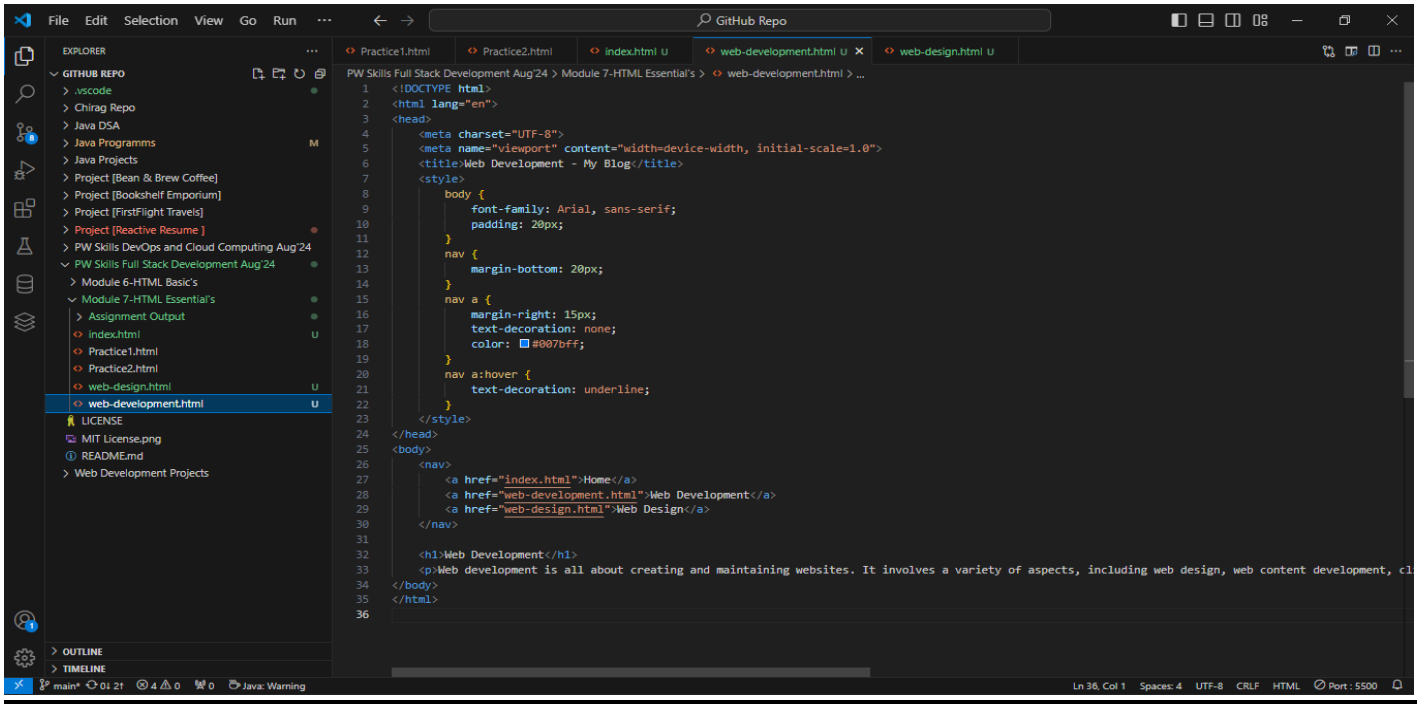
Take me to MDN docs to know more about Web Development.

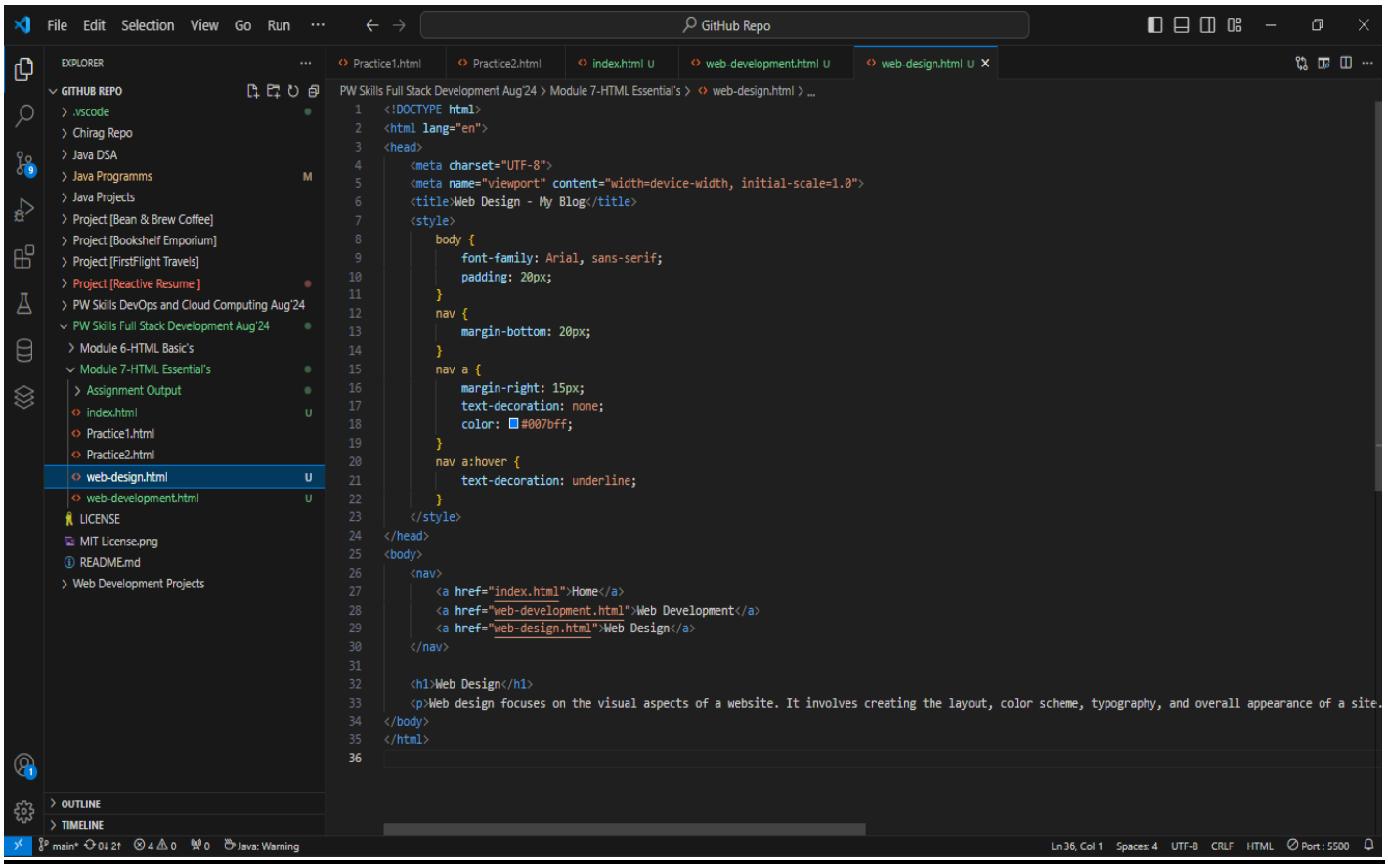
Take me to PW Skills Lab to practice live coding.

Q3.) Build a simple blog web page with 3 pages home, web development, and web design. Each page must contain hyperlinks to other pages in the top, a heading of the page topic and a paragraph of information. For the home page you can add some information about yourself.

Output-







Home Web Development Web Design

Welcome to My Blog

Hello! I'm Chirag, a passionate web developer and designer. This blog is where I share my knowledge and experiences in web development and design. Explore the different sections to learn more!

Web Development

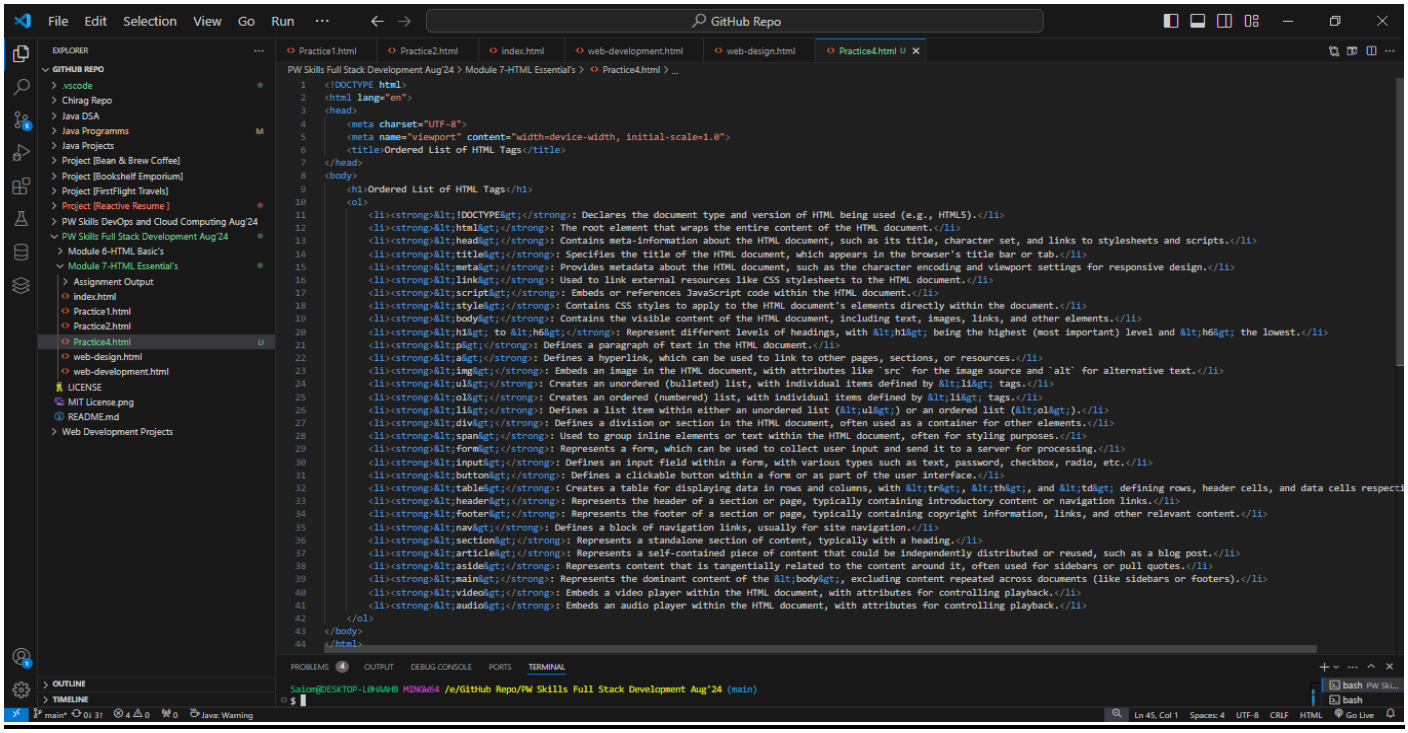
Web development is all about creating and maintaining websites. It involves a variety of aspects, including web design, web content development, client-side/server-side scripting, and network security configuration. It is a constantly evolving field, with new tools and technologies being introduced regularly.

Web Design

Web design focuses on the visual aspects of a website. It involves creating the layout, color scheme, typography, and overall appearance of a site. Good web design is essential for creating a user-friendly and aesthetically pleasing experience, which can significantly impact the success of a website.

Q4.) Create an ordered list of HTML tags. Each list item must include the tag name and some information about the tag.

Output-

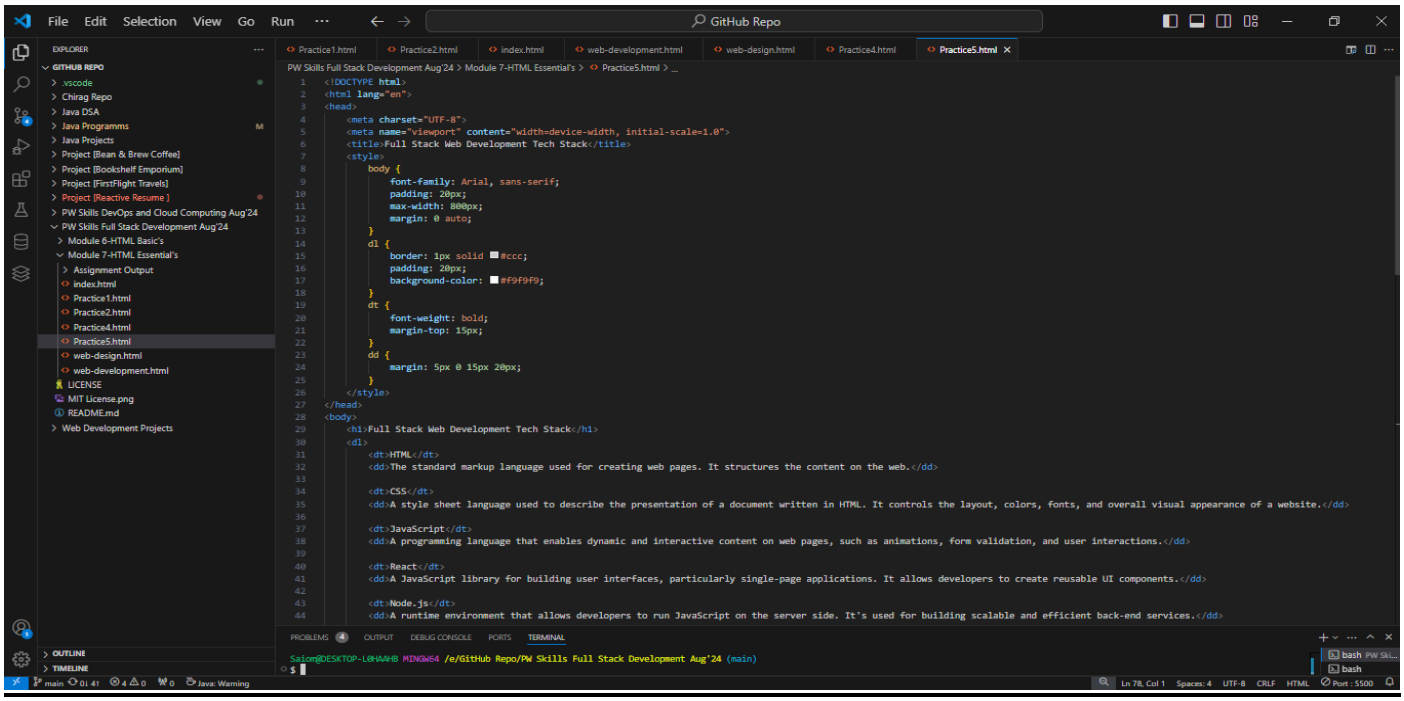


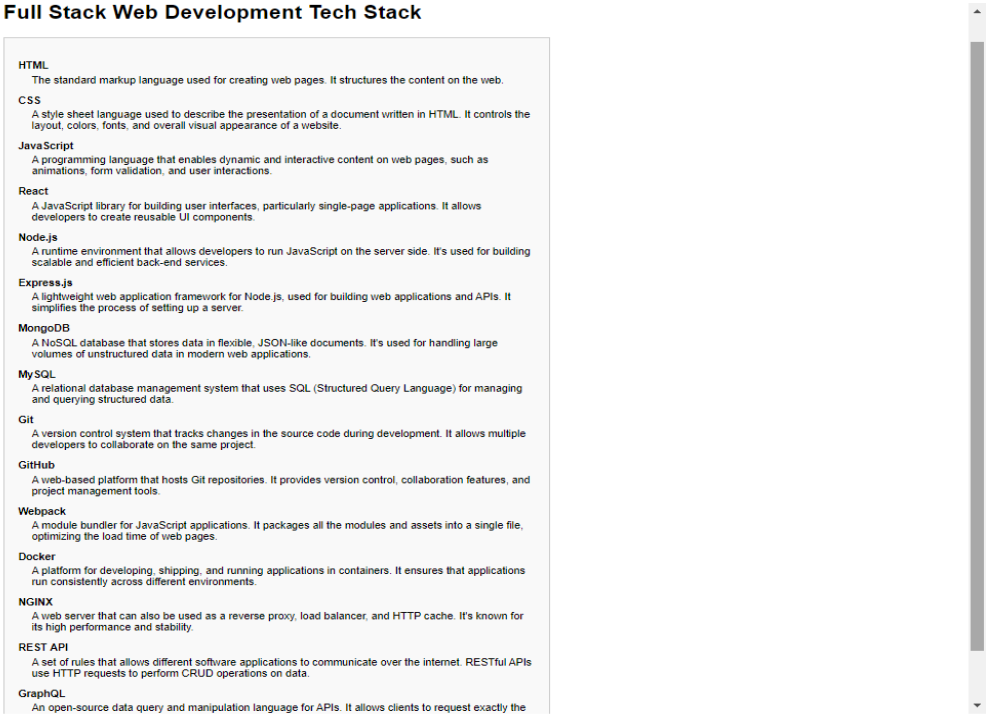
Ordered List of HTML Tags

1. <**!DOCTYPE**>: Declares the document type and version of HTML being used (e.g., HTML5).
2. <**html**>: The root element that wraps the entire content of the HTML document.
3. <**head**>: Contains meta-information about the HTML document, such as its title, character set, and links to stylesheets and scripts.
4. <**title**>: Specifies the title of the HTML document, which appears in the browser's title bar or tab.
5. <**meta**>: Provides metadata about the HTML document, such as the character encoding and viewport settings for responsive design.
6. <**link**>: Used to link external resources like CSS stylesheets to the HTML document.
7. <**script**>: Embeds or references JavaScript code within the HTML document.
8. <**style**>: Contains CSS styles to apply to the HTML document's elements directly within the document.
9. <**body**>: Contains the visible content of the HTML document, including text, images, links, and other elements.
10. <**h1**> **to** <**h6**>: Represent different levels of headings, with <h1> being the highest (most important) level and <h6> the lowest.
11. <**p**>: Defines a paragraph of text in the HTML document.
12. <**a**>: Defines a hyperlink, which can be used to link to other pages, sections, or resources.
13. <**img**>: Embeds an image in the HTML document, with attributes like `src` for the image source and `alt` for alternative text.
14. <**ul**>: Creates an unordered (bulleted) list, with individual items defined by <li> tags.
15. <**ol**>: Creates an ordered (numbered) list, with individual items defined by <li> tags.
16. <**li**>: Defines a list item within either an unordered list (<ul>) or an ordered list (<ol>).
17. <**div**>: Defines a division or section in the HTML document, often used as a container for other elements.
18. <**span**>: Used to group inline elements or text within the HTML document, often for styling purposes.
19. <**form**>: Represents a form, which can be used to collect user input and send it to a server for processing.
20. <**input**>: Defines an input field within a form, with various types such as text, password, checkbox, radio, etc.
21. <**button**>: Defines a clickable button within a form or as part of the user interface.
22. <**table**>: Creates a table for displaying data in rows and columns, with <tr>, <th>, and <td> defining rows, header cells, and data cells respectively.
23. <**header**>: Represents the header of a section or page, typically containing introductory content or navigation links.
24. <**footer**>: Represents the footer of a section or page, typically containing copyright information, links, and other relevant content.
25. <**nav**>: Defines a block of navigation links, usually for site navigation.
26. <**section**>: Represents a standalone section of content, typically with a heading.
27. <**article**>: Represents a self-contained piece of content that could be independently distributed or reused, such as a blog post.
28. <**aside**>: Represents content that is tangentially related to the content around it, often used for sidebars or pull quotes.
29. <**main**>: Represents the dominant content of the <body>, excluding content repeated across documents (like sidebars or footers).
30. <**video**>: Embeds a video player within the HTML document, with attributes for controlling playback.
31. <**audio**>: Embeds an audio player within the HTML document, with attributes for controlling playback.

Q5.) Create a description list of full stack web development tech stack, using the <dl> tag. Each term should be a tech stack name and each description should be a brief explanation of what the tech stack is used for.

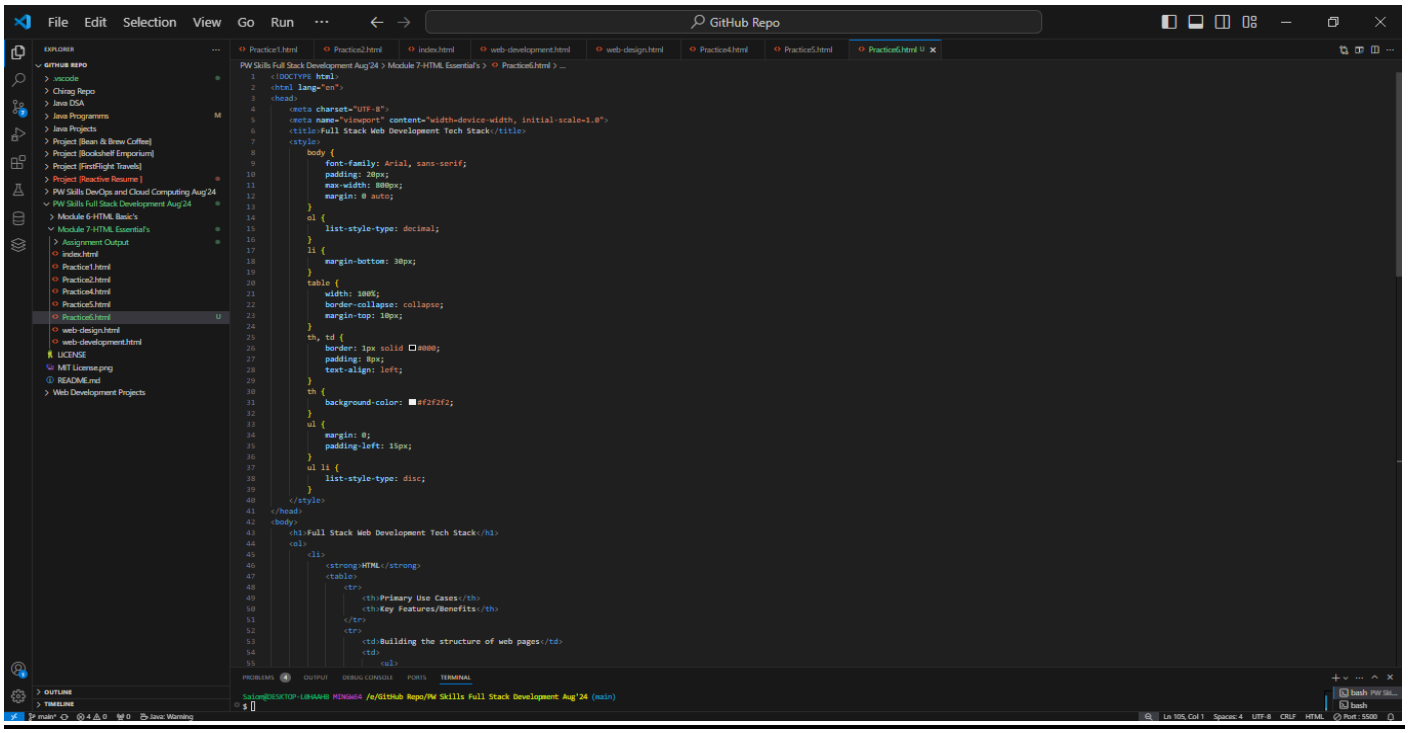
Output-



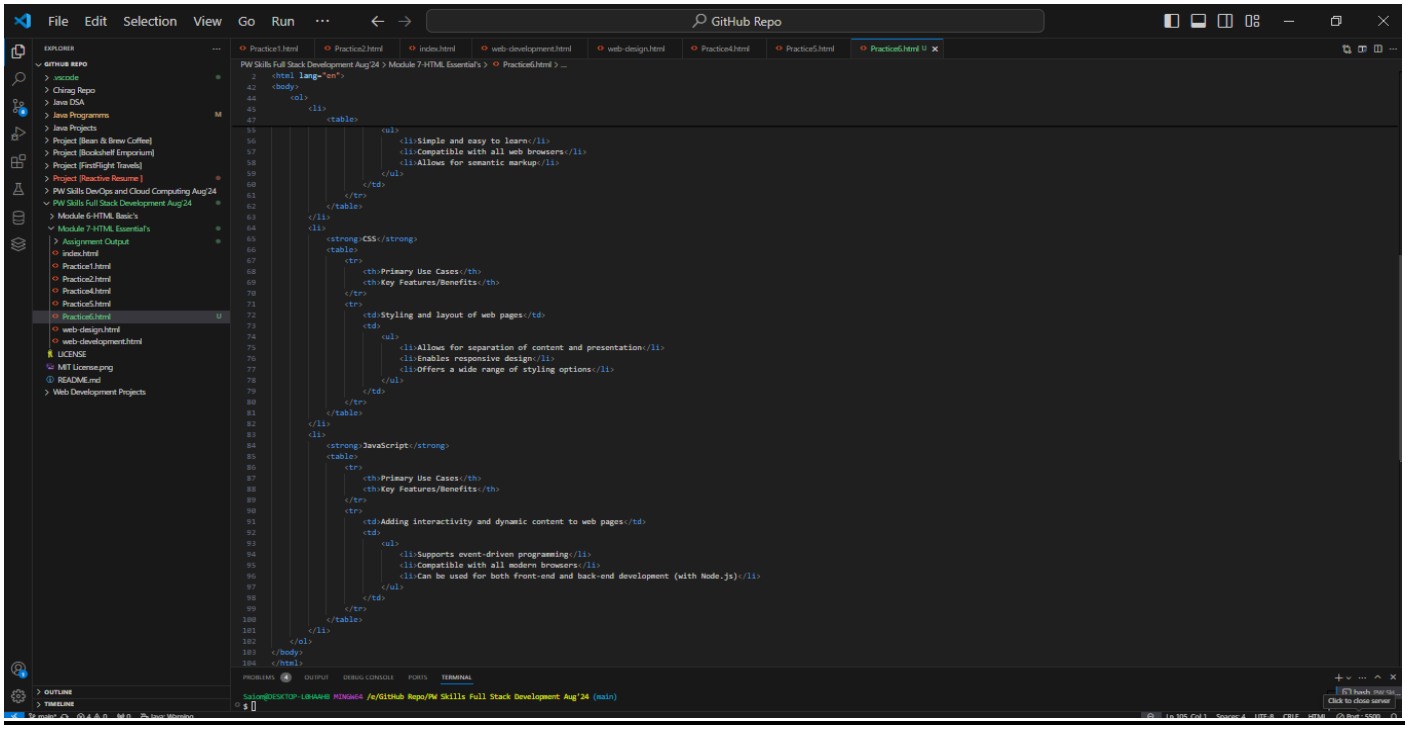


Q6.) Create an ordered list of the full stack web development tech stack HTML, CSS, and JS. For each tech stack, create a table that lists the tech stack name, its primary use cases, and some key features or benefits. Below is a reference image.

Output-



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Full Stack Web Development Tech Stack</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            padding: 20px;
            max-width: 800px;
            margin: 0 auto;
        }
        ol {
            list-style-type: decimal;
        }
        li {
            margin-bottom: 30px;
        }
        table {
            width: 100%;
            border-collapse: collapse;
            margin-top: 10px;
        }
        th, td {
            border: 1px solid #000;
            padding: 8px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
        ul {
            margin: 0;
            padding-left: 15px;
        }
        ul li {
            list-style-type: disc;
        }
    </style>
</head>
<body>
    <h1>Full Stack Web Development Tech Stack</h1>
    <ol>
        <li>
            <strong>HTML</strong>
            <table>
                <tr>
                    <th>Primary Use Cases</th>
                    <th>Key Features/Benefits</th>
                </tr>
                <tr>
                    <td>Building the structure of web pages</td>
                    <td>
                        <ul>
```



```
                            <li>Simple and easy to learn</li>
                            <li>Compatible with all web browsers</li>
                            <li>Allows for semantic markup</li>
                        </ul>
                    </td>
                </tr>
            </table>
        </li>
        <li>
            <strong>CSS</strong>
            <table>
                <tr>
                    <th>Primary Use Cases</th>
                    <th>Key Features/Benefits</th>
                </tr>
                <tr>
                    <td>Styling and layout of web pages</td>
                    <td>
                        <ul>
                            <li>Allows for separation of content and presentation</li>
                            <li>Enables responsive design</li>
                            <li>Offers a wide range of styling options</li>
                        </ul>
                    </td>
                </tr>
            </table>
        </li>
        <li>
            <strong>JavaScript</strong>
            <table>
                <tr>
                    <th>Primary Use Cases</th>
                    <th>Key Features/Benefits</th>
                </tr>
                <tr>
                    <td>Adding interactivity and dynamic content to web pages</td>
                    <td>
                        <ul>
                            <li>Supports event-driven programming</li>
                            <li>Compatible with all modern browsers</li>
                            <li>Can be used for both front-end and back-end development (with Node.js)</li>
                        </ul>
                    </td>
                </tr>
            </table>
        </li>
    </ol>
</body>
</html>
```

Full Stack Web Development Tech Stack

1. **HTML**

Primary Use Cases	Key Features/Benefits
Building the structure of web pages	• Simple and easy to learn • Compatible with all web browsers • Allows for semantic markup

2. **CSS**

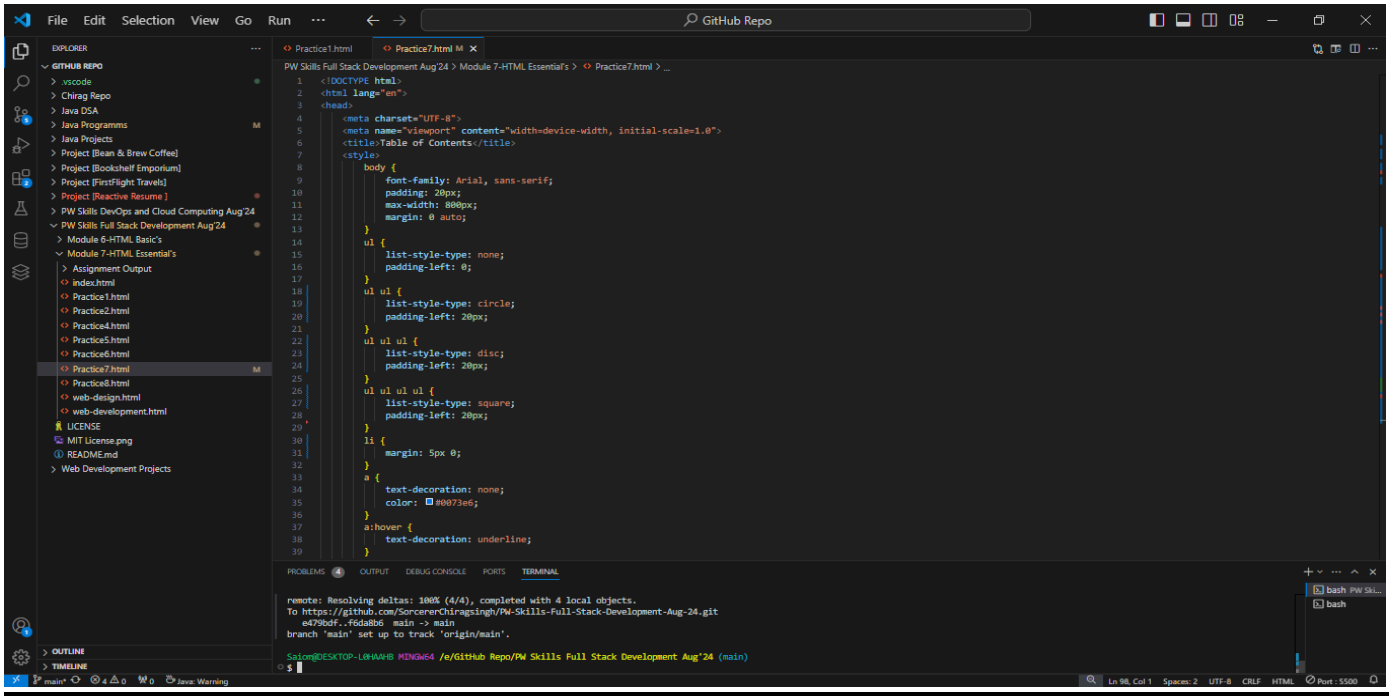
Primary Use Cases	Key Features/Benefits
Styling and layout of web pages	• Allows for separation of content and presentation • Enables responsive design • Offers a wide range of styling options

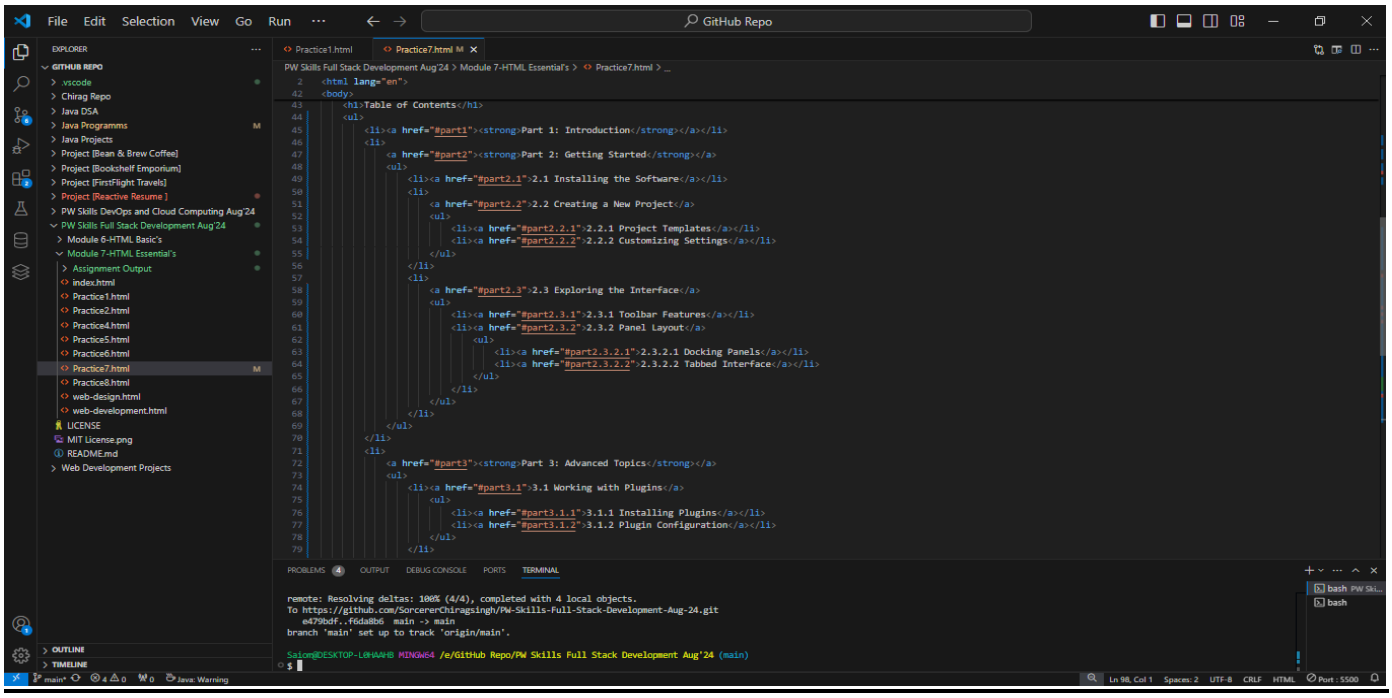
3. **JavaScript**

Primary Use Cases	Key Features/Benefits
Adding interactivity and dynamic content to web pages	• Supports event-driven programming • Compatible with all modern browsers • Can be used for both front-end and back-end development (with Node.js)

Q7.) Build a complex nested list structure representing a multi-level table of contents. Use unordered lists (<ul>) and list items (<li>) with inline-block styling to create a structured layout. Apply formatting tags to enhance the presentation of list items.

Output-





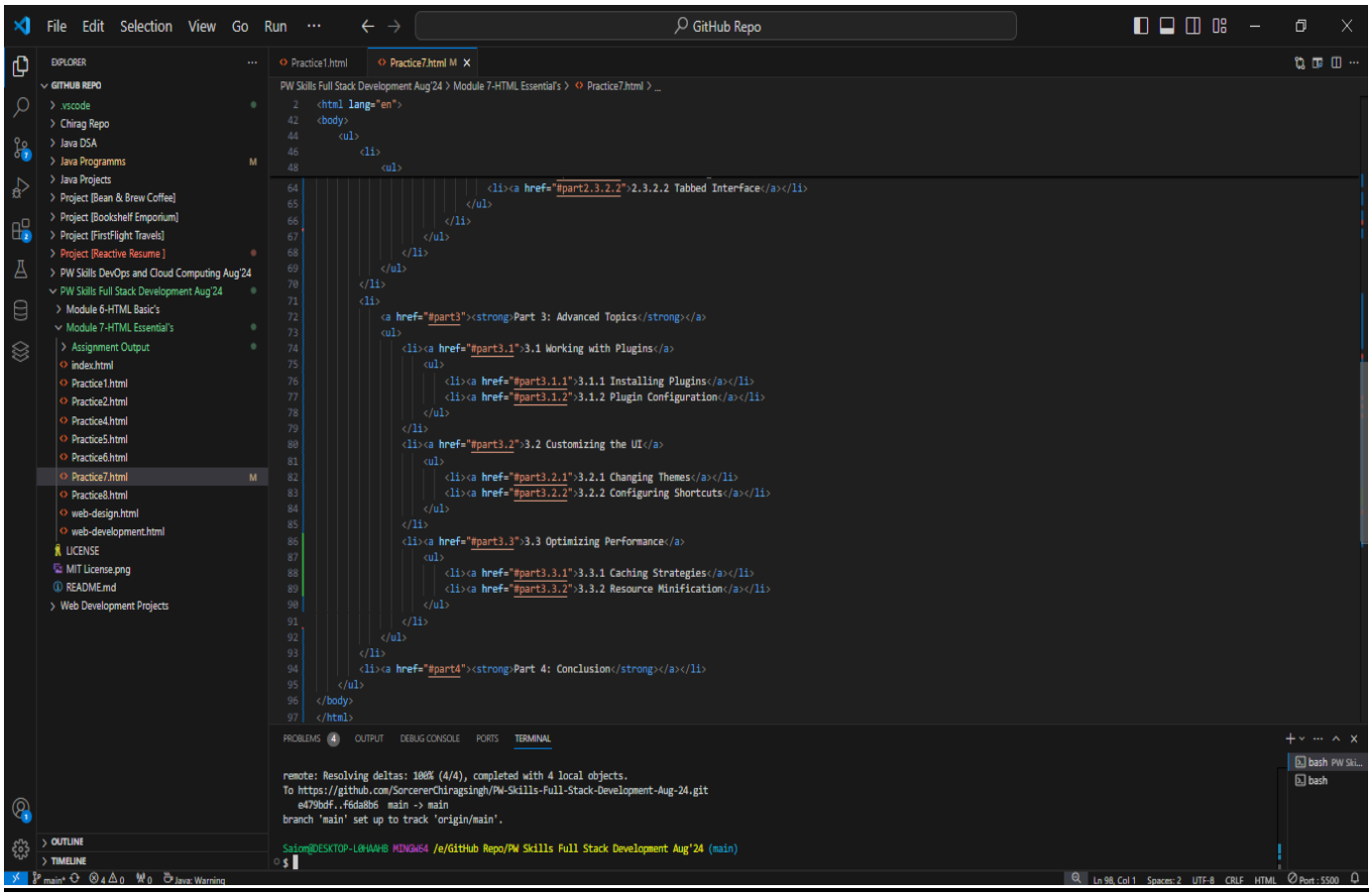
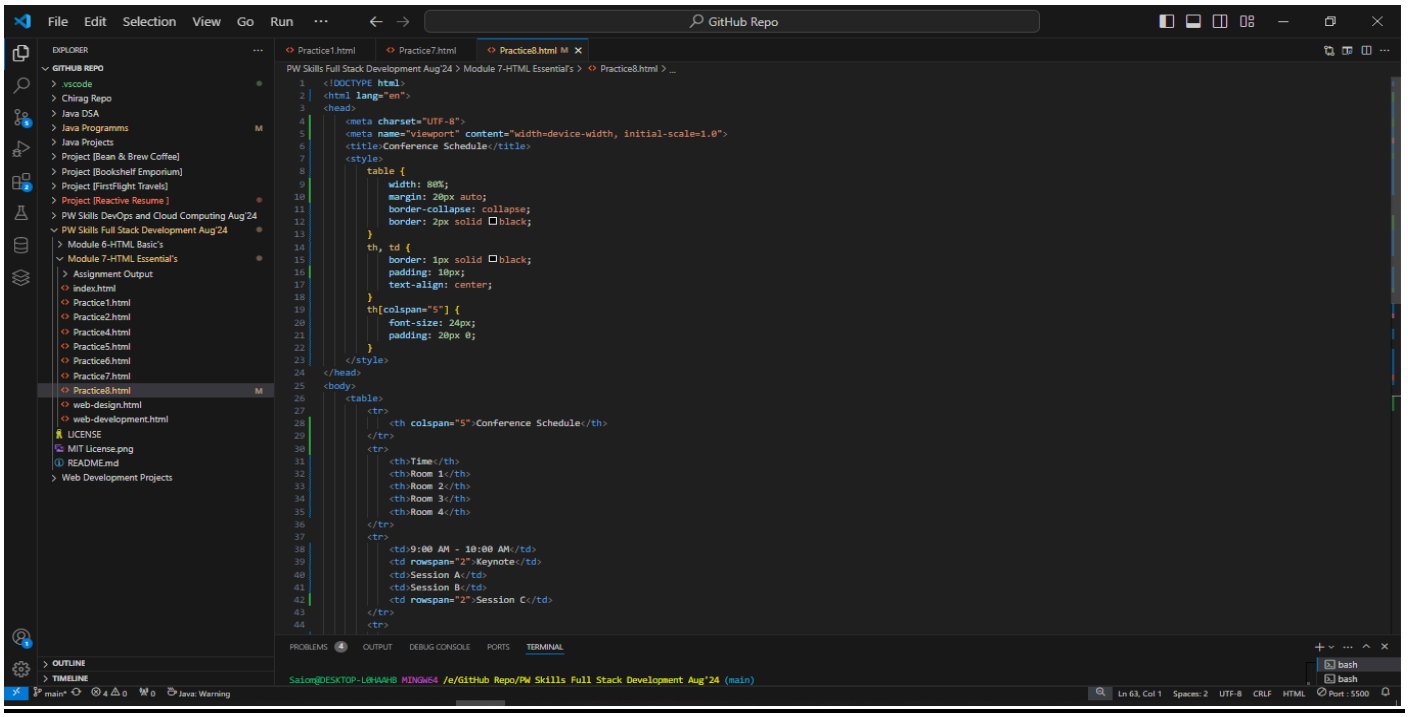
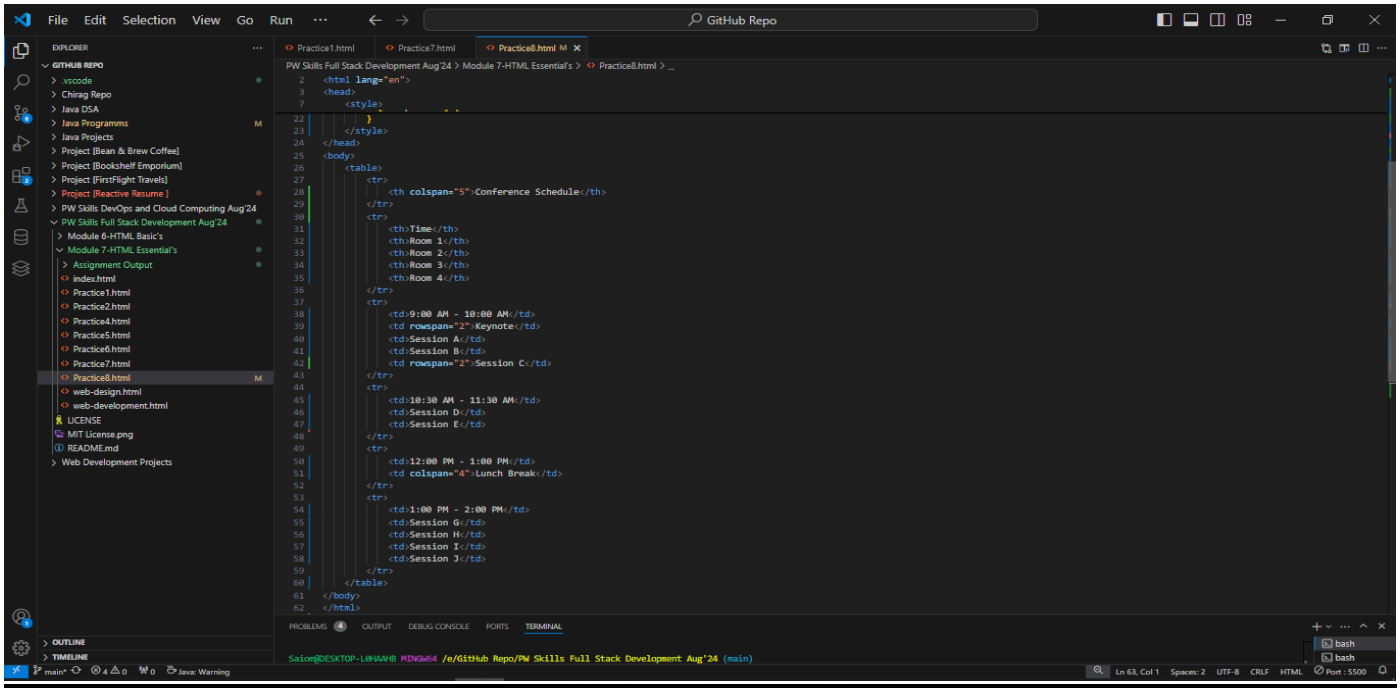


Table of Contents

Q8.) Create a table to display a conference schedule. Each row corresponds to a time slot, and each column corresponds to a room. Some time slots might have multiple sessions running simultaneously in different rooms. Utilize rowspan and colspan attributes as necessary to accommodate this complex schedule.

(use table attribute “cellpadding” to give extra padding in each table cell).





Conference Schedule				
Time	**Room 1**	**Room 2**	**Room 3**	**Room 4**
9:00 AM - 10:00 AM	Keynote	Session A	Session B	Session C
10:30 AM - 11:30 AM		Session D	Session E	
12:00 PM - 1:00 PM	Lunch Break			
1:00 PM - 2:00 PM	Session G	Session H	Session I	Session J