Question1: Build a simple webpage that displays text as shown in the below image.

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 Smal Text

~~This text will be deleted~~

Code

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Text_Styling</title>
</head>
<body>
    <p class="first"><b>This text will be bolded</b></p>
    <p class="second"><i>This text will be italic</i></p>
    <p class="third"> <mark>This text will be highlighted</mark></p>
    <p class="forth"><u>This text will be underline</u></p>
    <p class="fifth"><del>This text will be deleted</del></p>
```
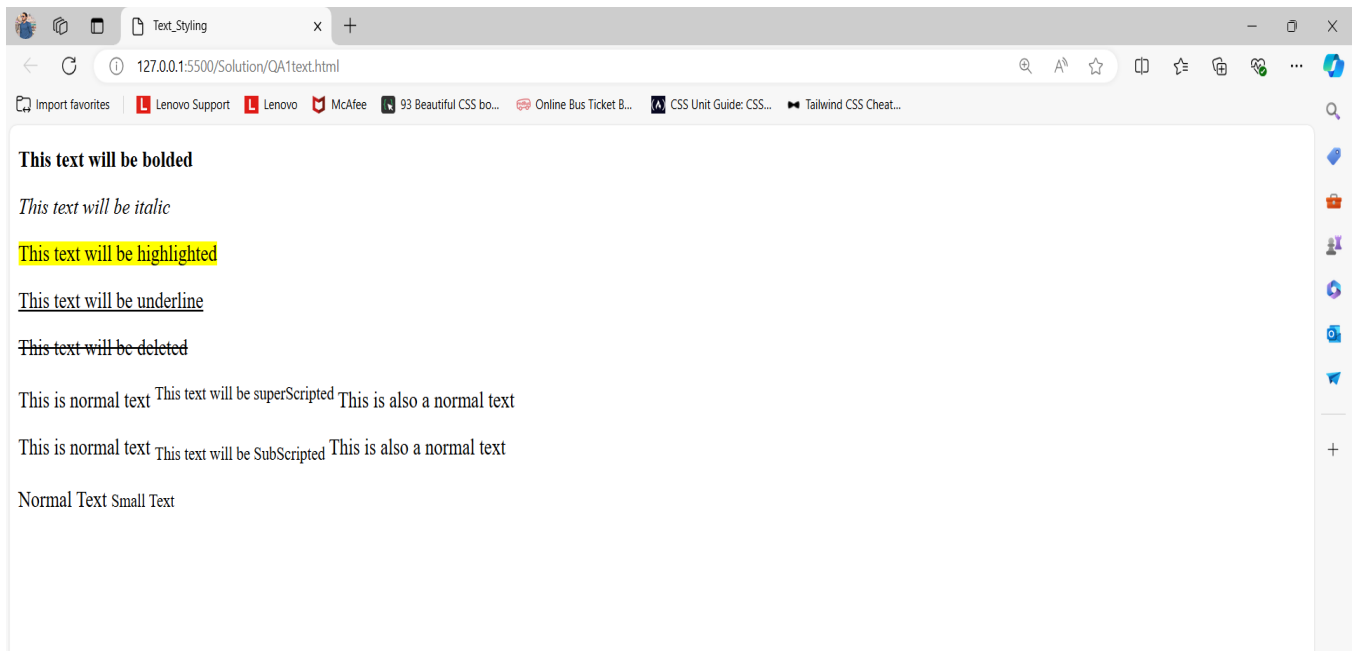
```
    <p class="sixth">This is normal text <sup>This text will be superScripted </sup>This is also a normal
text </p>

    <p class="seventh">This is normal text <sub>This text will be SubScripted </sub>This is also a normal
text</p>

    <p class="eighth">Normal Text <small>Small Text</small></p>
</body>
</html>
```

<p style="text-align:center"><strong style="color:red">Out Put</p>

Question2: Build a simple webpage that help user navigate different web development-related websites. Note: On clicking the hyperlink the hyperlink the web pages should open in a new tab. Below is a reference image.

Code

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>HyperLink</title>
</head>
<body>
    <p>Take me to <a href="https://pwskills.com/" target="_blank">PW Skills</a>to by a course.</p>
    <p>Take me to <a href="https://developer.mozilla.org/en-US/" target="_blank">MDN docs</a>to know more about Web Development</p>
    <p>Take me to <a href="https://lab.pwskills.com/" target="_blank">PW Skills Lab</a> to practice live coding</p>
</body>
</html>
```

Out Put
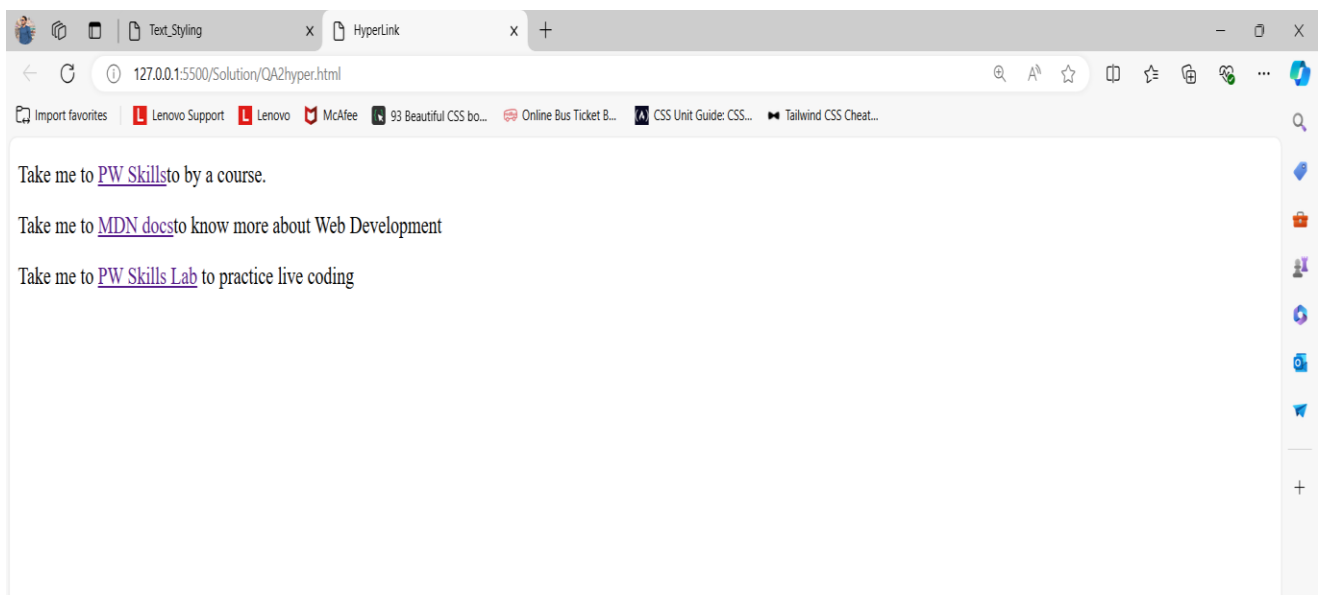
Question3: Build a simple Blog web page with 3 pages home , web development , and web design. Each page must contain hyperlinks to other pages in the top, a heading of the page topic and a paragraph of information. For the home page you can add some information about yourself.

Code (Home)

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Home</title>
    <style>
      nav {
        margin-bottom: 20px;
      }
      nav a {
        margin-right: 10px;
      }
      img{
        height: 400px;
        width: 50%;
        background-size: cover;
      }
    </style>
  </head>
  <body>
    <nav>
      <a href="blog.html">Home</a>
      <a href="QA3wd.html">Web Development</a>
      <a href="design.html">Web Design</a>
    </nav>
```

```
  <header>
    <h1>Home</h1>
  </header>


  <main>
    <p>
      Welcome to my blog! My name is Rajkamal Yadav, and I am passionate about
      web development and design. On this blog, I will share my knowledge,
      tips, and tutorials on various topics related to web technologies.
    </p>
  </main>
  <img src="assets/DSC_0012.JPG" alt="my_pic">
 </body>
</html>
```

Code (Web Development)

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Web Development</title>
    <style>
      nav {
        margin-bottom: 20px;
      }
      nav a {
        margin-right: 10px;
      }
    </style>
  </head>
```

```html
  <body>
    <nav>
      <a href="blog.html">Home</a>
      <a href="QA3wd.html">Web Development</a>
      <a href="design.html">Web Design</a>
    </nav>
    <header>
      <h1>Web Development</h1>
    </header>
    <main>
      <p>
        Web development is the work involved in developing a website for the
        Internet or an intranet. This can range from creating a simple single
        static page of plain text to complex web applications, electronic
        businesses, and social network services.
      </p>
    </main>
    <img src="assets/img.avif" alt="img">
  </body>
</html>
```

Code (Web Design)

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Web Design</title>
    <style>
      nav {
        margin-bottom: 20px;
      }
      nav a {
        margin-right: 10px;
      }
    </style>
  </head>
```

```
<body>
 <nav>
   <a href="blog.html">Home</a>
   <a href="QA3wd.html">Web Development</a>
   <a href="design.html">Web Design</a>
 </nav>
 <header>
  <h1>Web Design</h1>
 </header>
 <main>
  <p>
   Web design encompasses many different skills and disciplines in the
   production and maintenance of websites. The different areas of web
   design include web graphic design, user interface design, authoring,
   including standardized code and proprietary software, user experience
   design, and search engine optimization.
  </p>
 </main>
 <img src="assets/img2.jpeg" alt="img">
 </body>
</html>
```
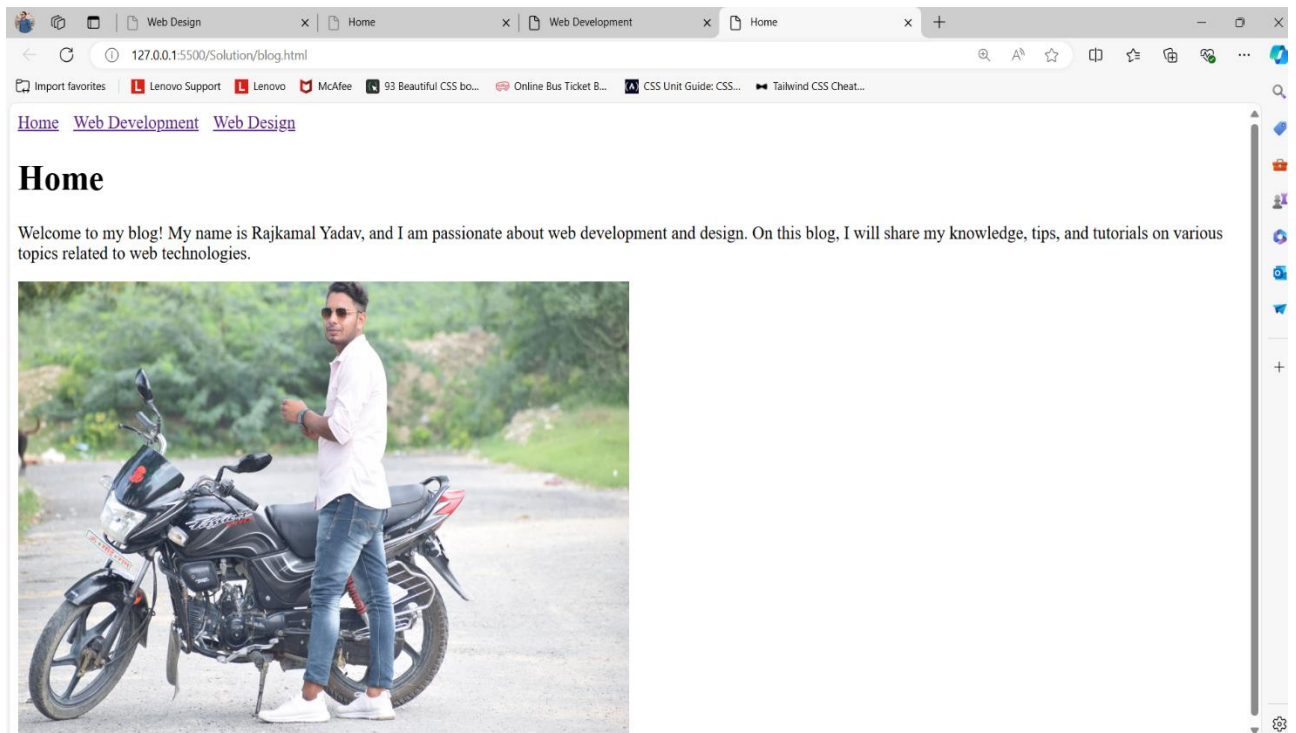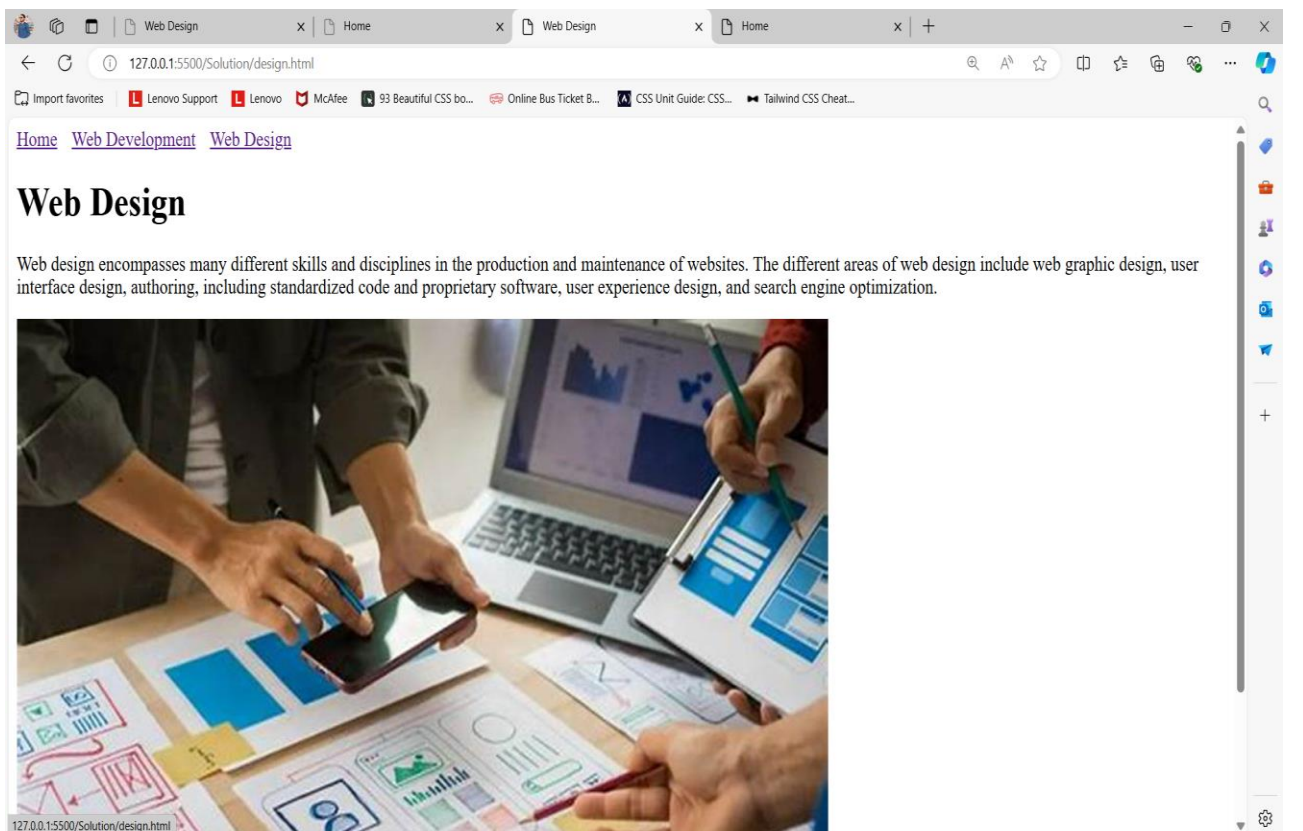
Out Put

Web Development

Web development is the work involved in developing a website for the Internet or an intranet. This can range from creating a simple single static page of plain text to complex web applications, electronic businesses, and social network services.
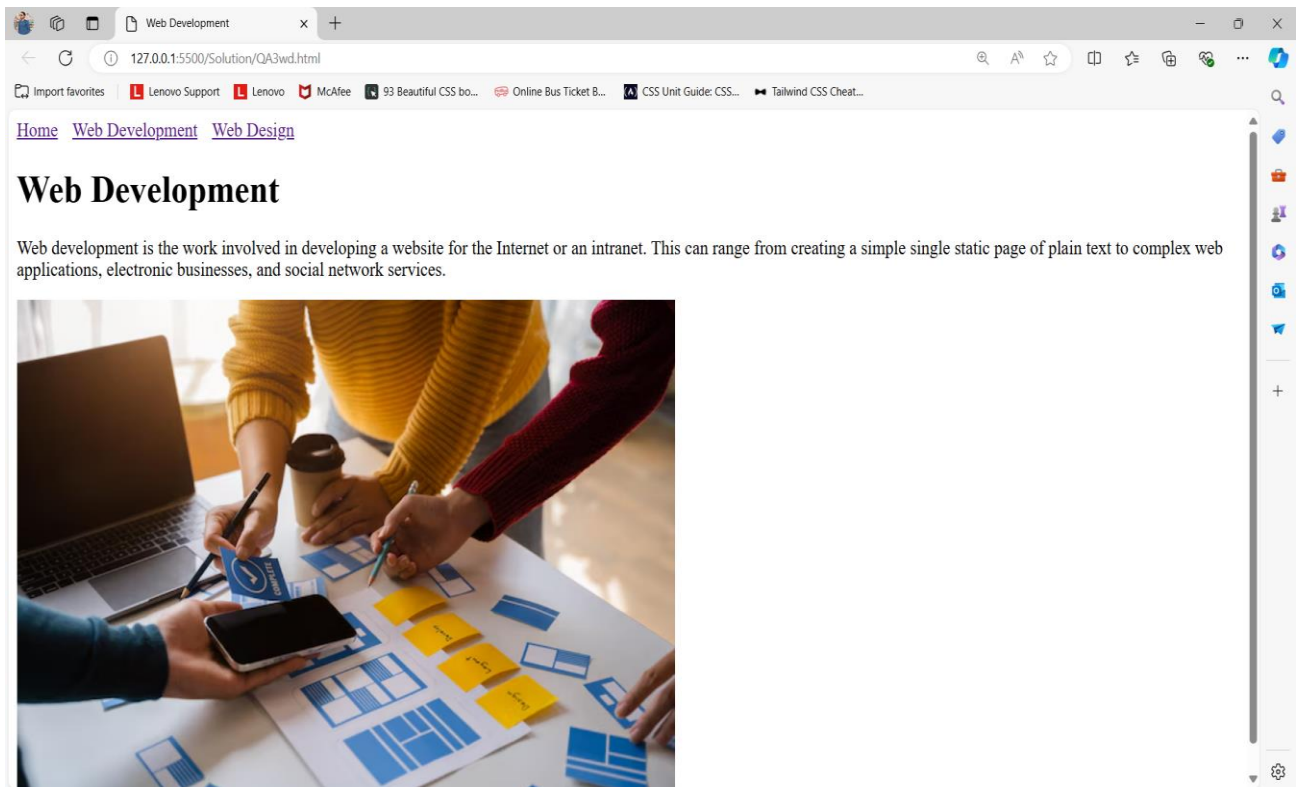
Web Design

Web design encompasses many different skills and disciplines in the production and maintenance of websites. The different areas of web design include web graphic design, user interface design, authoring, including standardized code and proprietary software, user experience design, and search engine optimization.

Question4: Create a ordered list of HTML tags. Each list item include the tag name and some information about the tag.

Code

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>HTML Tags</title>
    <style>
      body {
        font-family: Arial, sans-serif;
      }
      h1 {
        text-align: center;
      }
      ol {
        max-width: 600px;
        margin: 0 auto;
        padding: 0;
      }
      li {
        margin-bottom: 10px;
      }
      code {
        background-color: #f4f4f4;
        padding: 2px 4px;
        border-radius: 3px;
      }
    </style>
  </head>
  <body>
    <h1>HTML Tags</h1>
    <ol>
      <li>
        <strong>&lt;html&gt;</strong>: The &lt;html&gt; tag
```

represents the root of an HTML document. All other HTML elements must be descendants of this element.

  <head>: The <head> tag contains meta-information about the HTML document, such as its title and links to scripts and stylesheets.

  <title>: The <title> tag defines the title of the document, which is shown in the browser's title bar or tab.

  <body>: The <body> tag represents the content of an HTML document. It contains all the content that is displayed in the browser.

  <h1> to <h6>: The <h1> to <h6> tags define HTML headings. <h1> defines the most important heading, and <h6> defines the least important heading.

  <p>: The <p> tag defines a paragraph of text.

  <a>: The <a> tag defines a hyperlink, which is used to link from one page to another. The href attribute specifies the URL of the page the link goes to.

  <img>: The <img> tag is used

to embed an image in an HTML page. The src attribute
specifies the path to the image.

  : The <ul> tag defines an
  unordered list.

  : The <ol> tag defines an
  ordered list.

  : The <li> tag defines a
  list item and is used inside <ul> or
   tags.

  <div>: The <div> tag defines a
  division or a section in an HTML document. It is used as a container for
  other HTML elements.

  <span>: The <span> tag is used
  to group inline-elements in a document. It provides a way to style parts
  of the text or content.

  <form>: The <form> tag is used
  to create an HTML form for user input.

  <input>: The <input> tag
  specifies an input field where the user can enter data.

  <button>: The <button> tag

defines a clickable button.

      <table>: The <table> tag
      defines a table.

      <tr>: The <tr> tag defines a
      row in a table.

      <td>: The <td> tag defines a
      cell in a table.

      <th>: The <th> tag defines a
      header cell in a table.

      <style>: The <style> tag is
      used to define CSS styles within an HTML document.

      <link>: The <link> tag defines
      a relationship between the current document and an external resource. It
      is most commonly used to link to stylesheets.

      <script>: The <script> tag is
      used to embed or reference executable code, typically JavaScript.
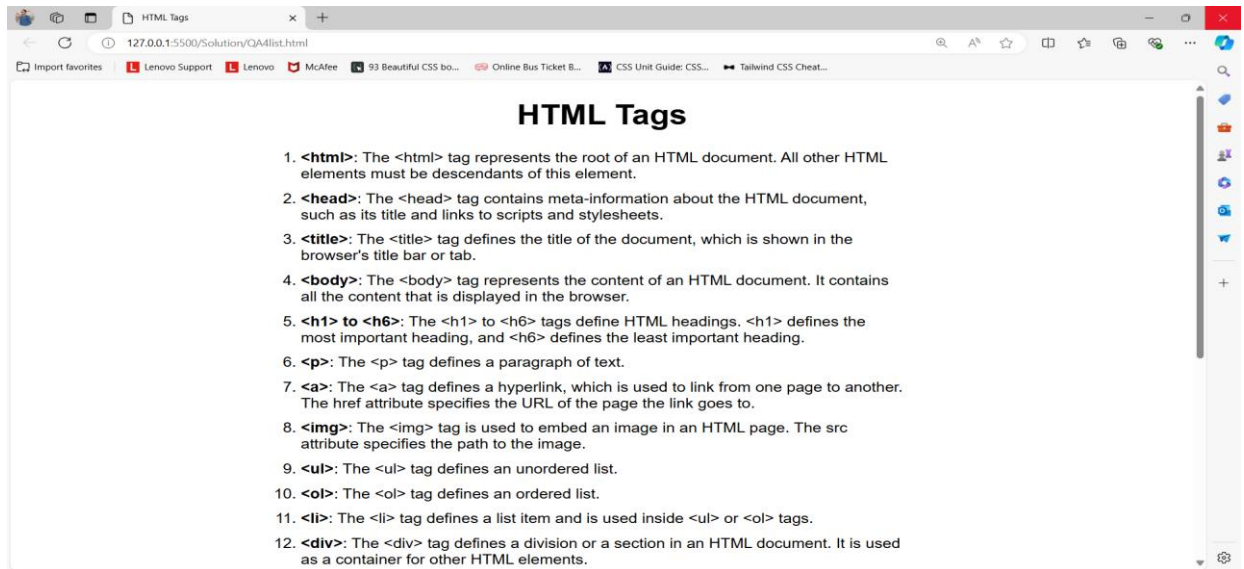

      <meta>: The <meta> tag
      provides metadata about the HTML document. It is used within the
      <head> section.


```
        </ol>
    </body>
</html>
```

Question5: Create a description list of full stack web development

Tech stack , using the <dl> tag. Each term should be a tech stack name and description should be a brief explanation of what the tech stack is used for.

```html
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Description List</title>

    <style>

        dl{

            font-size: 15px;

        }

        .main{

            font-size: 20px;
```

```
            font-weight: bold;

        }

    </style>

</head>

<body>

    <h1>Full Stack Web Development Tech Stack</h1>

    <hr>

    <dl>

        <dt class="main">Frontend (Client-Side):</dt>

        <dd>

         <dl>

           <dt>HTML (Hypertext Markup Language):</dt>

           <dd>The standard markup language used to create the structure of web pages.</dd>


           <dt>CSS (Cascading Style Sheets):</dt>

           <dd>Styles and layouts the web pages, making them visually appealing.</dd>


           <dt>JavaScript:</dt>

           <dd>A scripting language that adds interactivity to web pages. Can manipulate HTML and CSS
in real-time.</dd>

          </dl>

        </dd>


        <dt class="main">Frontend Frameworks/Libraries:</dt>

        <dd>

         <dl>

           <dt>React.js:</dt>

           <dd>A JavaScript library for building user interfaces, especially single-page applications.</dd>


           <dt>Vue.js:</dt>

           <dd>A progressive JavaScript framework for building user interfaces and single-page
applications.</dd>
```

```html
        <dt>Angular:</dt>

        <dd>A TypeScript-based open-source web application framework led by the Angular Team at
Google.</dd>

     </dl>

     </dd>


     <dt class="main">Backend (Server-Side):</dt>

     <dd>

      <dl>

       <dt>Node.js:</dt>

       <dd>A JavaScript runtime built on Chrome's V8 engine that allows executing JavaScript server-
side.</dd>


        <dt>Express.js:</dt>

        <dd>A minimalist web framework for Node.js, used to build web applications and APIs.</dd>


        <dt>Django:</dt>

        <dd>A high-level Python web framework that encourages rapid development and clean,
pragmatic design.</dd>


        <dt>Ruby on Rails:</dt>

        <dd>A server-side web application framework written in Ruby, designed to simplify and speed
up web application development.</dd>

     </dl>

     </dd>


     <dt class="main">Databases:</dt>

     <dd>

      <dl>

       <dt>MySQL:</dt>

        <dd>An open-source relational database management system that uses SQL (Structured Query
Language).</dd>
```

```html
        <dt>MongoDB:</dt>

        <dd>A NoSQL database that stores data in JSON-like documents, making it flexible and
scalable.</dd>


        <dt>PostgreSQL:</dt>

        <dd>An open-source, powerful, and feature-rich relational database management
system.</dd>

      </dl>

     </dd>


     <dt class="main">Version Control:</dt>

     <dd>

      <dl>

       <dt>Git:</dt>

       <dd>A distributed version control system for tracking changes in source code during software
development.</dd>


       <dt>GitHub/GitLab/Bitbucket:</dt>

       <dd>Web-based platforms for hosting Git repositories and collaborating on code.</dd>

      </dl>

     </dd>


     <dt class="main">Web Servers:</dt>

     <dd>

      <dl>

       <dt>Apache:</dt>

       <dd>A popular open-source HTTP server that allows hosting of websites.</dd>


       <dt>Nginx:</dt>

       <dd>A high-performance HTTP server and reverse proxy, as well as an IMAP/POP3 proxy
server.</dd>

      </dl>

     </dd>
```

```html
<dt class="main">DevOps/Deployment:</dt>

<dd>

 <dl>

  <dt>Docker:</dt>

   <dd>A platform that uses OS-level virtualization to deliver software in packages called
containers.</dd>


   <dt>Kubernetes:</dt>

   <dd>An open-source platform designed to automate deploying, scaling, and operating
application containers.</dd>


   <dt>Jenkins:</dt>

   <dd>An open-source automation server used to automate parts of software development
related to building, testing, and deploying.</dd>

  </dl>

 </dd>


<dt class="main">APIs (Application Programming Interfaces):</dt>

<dd>

 <dl>

  <dt>REST (Representational State Transfer):</dt>

   <dd>A set of principles for designing networked applications, using stateless protocols and
standard operations like GET, POST, PUT, DELETE.</dd>


   <dt>GraphQL:</dt>

   <dd>A query language for APIs that allows clients to request only the data they need.</dd>

  </dl>

 </dd>


<dt class="main">Authentication & Authorization:</dt>

<dd>

 <dl>
```

```
        <dt>OAuth:</dt>

        <dd>An open standard for token-based authentication and authorization on the internet.</dd>


        <dt>JWT (JSON Web Token):</dt>

        <dd>A compact, URL-safe means of representing claims to be transferred between two
parties.</dd>

      </dl>

    </dd>


    <dt class="main">Testing:</dt>

    <dd>

     <dl>

      <dt>Jest:</dt>

      <dd>A JavaScript testing framework maintained by Facebook, often used with React.</dd>


      <dt>Mocha:</dt>

      <dd>A JavaScript test framework running on Node.js, featuring browser support, asynchronous
testing, and more.</dd>


      <dt>Selenium:</dt>

      <dd>An open-source tool for automating web browsers, often used for testing web
applications.</dd>

     </dl>

    </dd>


    <dt class="main">Package Managers:</dt>

    <dd>

     <dl>

      <dt>npm (Node Package Manager):</dt>

      <dd>A package manager for JavaScript, included with Node.js, used for managing
dependencies.</dd>


      <dt>Yarn:</dt>
```

```html
        <dd>An alternative package manager for JavaScript that focuses on speed, security, and
consistency.</dd>

      </dl>

      </dd>


      <dt class="main">Build Tools:</dt>

      <dd>

       <dl>

       <dt>Webpack:</dt>

       <dd>A module bundler for JavaScript applications, used to bundle and serve web assets.</dd>


       <dt>Babel:</dt>

       <dd>A JavaScript compiler that allows using next-generation JavaScript syntax.</dd>

       </dl>

       </dd>

     </dl>
</body>

</html>
```
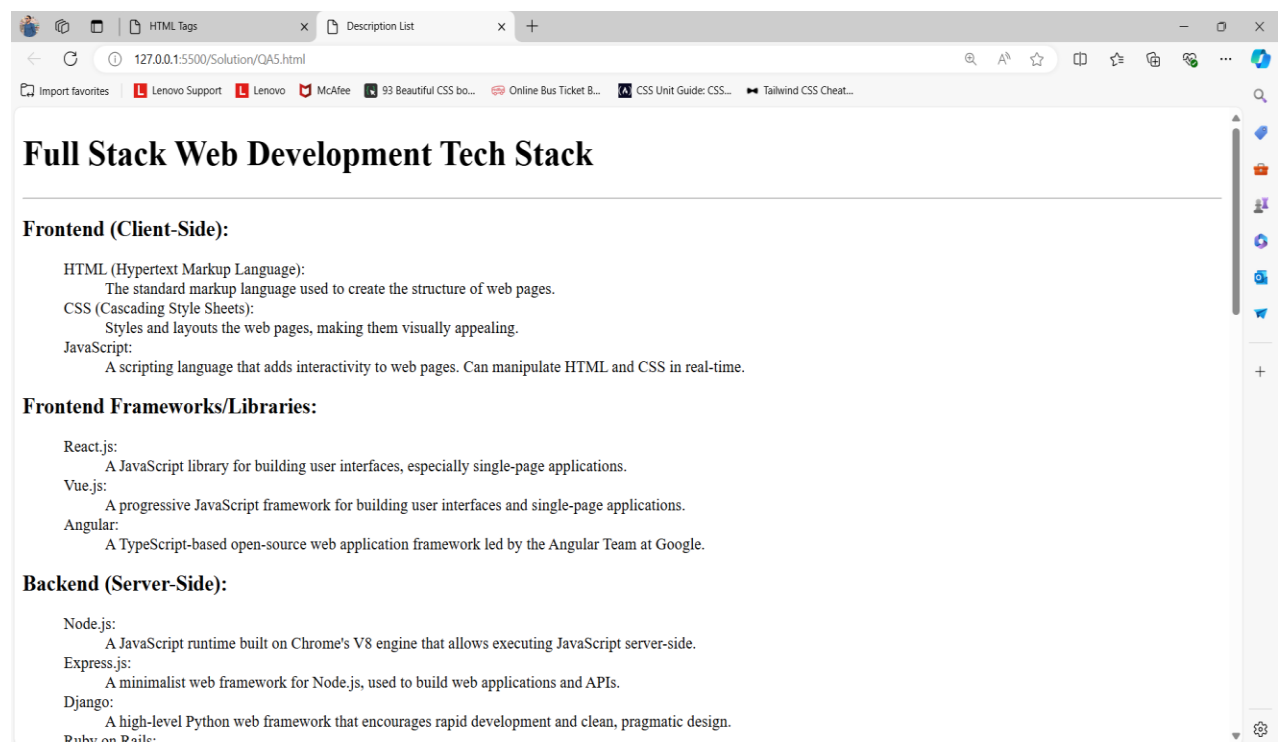
Full Stack Web Development Tech Stack

Frontend (Client-Side):

HTML (Hypertext Markup Language):
  The standard markup language used to create the structure of web pages.
CSS (Cascading Style Sheets):
  Styles and layouts the web pages, making them visually appealing.
JavaScript:
  A scripting language that adds interactivity to web pages. Can manipulate HTML and CSS in real-time.

Frontend Frameworks/Libraries:

React.js:
  A JavaScript library for building user interfaces, especially single-page applications.
Vue.js:
  A progressive JavaScript framework for building user interfaces and single-page applications.
Angular:
  A TypeScript-based open-source web application framework led by the Angular Team at Google.

Backend (Server-Side):

Node.js:
  A JavaScript runtime built on Chrome's V8 engine that allows executing JavaScript server-side.
Express.js:
  A minimalist web framework for Node.js, used to build web applications and APIs.
Django:
  A high-level Python web framework that encourages rapid development and clean, pragmatic design.
Ruby on Rails:

Question6: Create an ordered list of the full stack web development tech stack HTML, CSS and JS. For each tech stack , created a table that lists the tech stack name, its primary use cases, and some key features or benefits. Below is a reference image.

Eg.

Code

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Describe HTML CSS JS</title>
  <style>
    th,td{
      border: 1px solid black;
    }
    table{
      text-align: center;
      border: 1px solid black;
      width: 800px;
    }
  </style>
```

```
</head>
<body>
 <h2>
   Here's an ordered list of the full stack web development tech stack,
   including HTML, CSS, and JavaScript, along with tables describing each
   tech stack:
 </h2>
 <table>
 <h5>1. HTML(Hyper Text Markup Language):</h5>
 <ol>
 <tr>
   <th>Teck stack</th>
   <th>Primery Use Cases</th>
   <th>Key Features/Benefits</th>
 </tr>
 <tr>
   <td rowspan="6">HTML</td>
 </tr>
 <tr>
   <td>Structure and organize content on webpages</td>
   <td>Defines the structure of web documents</td>
 </tr>
 <tr>
   <td>Create headings, paragraphs, lists, and tables</td>
   <td>Supports multimedia elements with tags</td>
 </tr>
 <tr>
   <td>Embed images, videos, audio, and other media</td>
   <td>Enables semantic markup for accessibility</td>
 </tr>
 <tr>
   <td>Build forms for user input and data submission</td>
```

```html
    <td>Easy to learn and widely supported</td>
</tr>
<tr>
    <td>Provide semantic meaning to web content</td>
    <td>Integrates with other web technologies</td>
</tr>
</ol>
<!----------------------------CSS Table------------------------------------------->
<table>
    <h5>2. CSS (Cascading Style Sheets):</h5>
    <ol>
    <tr>
        <th>Teck stack</th>
        <th>Primery Use Cases</th>
        <th>Key Features/Benefits</th>
    </tr>
    <tr>
        <td rowspan="6">CSS</td>
    </tr>
    <tr>
        <td>Styling and visual presentation of webpages</td>
        <td>Controls the layout and design of elements</td>
    </tr>
    <tr>
        <td>Define colors, fonts, spacing, and backgrounds</td>
        <td>Supports responsive design for different devices</td>
    </tr>
    <tr>
        <td>Apply animations and transitions</td>
        <td>Enhances user experience with visual effects</td>
    </tr>
    <tr>
```

```html
      <td>Implement media queries for responsive design</td>

      <td>Separates presentation from HTML structure</td>

    </tr>

    <tr>

      <td>Enables reusability with classes and selectors</td>

      <td>Supports modern CSS frameworks like Flexbox and Grid</td>

    </tr>

    </ol>

</table>

<!----------------------------------Javascript Table--------------------------------------->

<table>

  <h5>3. JS (JavaScript):</h5>

  <ol>

  <tr>

    <th>Teck stack</th>

    <th>Primery Use Cases</th>

    <th>Key Features/Benefits</th>

  </tr>

  <tr>

    <td rowspan="6">JavaScript</td>

  </tr>

  <tr>

    <td>Client-side interactivity and dynamic web content</td>

    <td>Enables interactivity and user engagement</td>

  </tr>

  <tr>

    <td>Validate user input and perform form handling</td>

    <td>Manipulates HTML and CSS dynamically</td>

  </tr>

  <tr>

    <td>Implement complex behavior and logic on webpages</td>

    <td>Supports asynchronous operations (AJAX)</td>
```

```html
    </tr>
    <tr>
      <td>Fetch data from servers and update web content</td>
      <td>Integrates with various libraries and frameworks</td>
    </tr>
    <tr>
      <td>Build interactive web applications and games</td>
      <td>Executes code directly in the web browser</td>
    </tr>
    </ol>
</table>
 </body>
</html>
```

Out Put

Here's an ordered list of the full stack web development tech stack, including HTML, CSS, and JavaScript, along with tables describing each tech stack:

1. HTML(Hyper Text Markup Language):

Teck stack	Primery Use Cases	Key Features/Benefits
HTML	Structure and organize content on webpages	Defines the structure of web documents
	Create headings, paragraphs, lists, and tables	Supports multimedia elements with tags
	Embed images, videos, audio, and other media	Enables semantic markup for accessibility
	Build forms for user input and data submission	Easy to learn and widely supported
	Provide semantic meaning to web content	Integrates with other web technologies

2. CSS (Cascading Style Sheets):

Teck stack	Primery Use Cases	Key Features/Benefits
CSS	Styling and visual presentation of webpages	Controls the layout and design of elements
	Define colors, fonts, spacing, and backgrounds	Supports responsive design for different devices
	Apply animations and transitions	Enhances user experience with visual effects
	Implement media queries for responsive design	Separates presentation from HTML structure
	Enables reusability with classes and selectors	Supports modern CSS frameworks like Flexbox and Grid

3. JS (JavaScript):

Teck stack	Primery Use Cases	Key Features/Benefits

Question7: Build a complex nested list structure representing a multi-level table of contents. Use unordered list <ul> and lists item <li> with inline-block styling create a structured layout . Apply formatting tags to enhance the presentation of list items.

Output should look like this:

Table of Contents

- Part 1: Introduction
- Part 2: Getting Started
  - 2.1 Installing the Software
  - 2.2 Creating a New Project
    - 2.2.1 Project Templates
    - 2.2.2 Customizing Settings
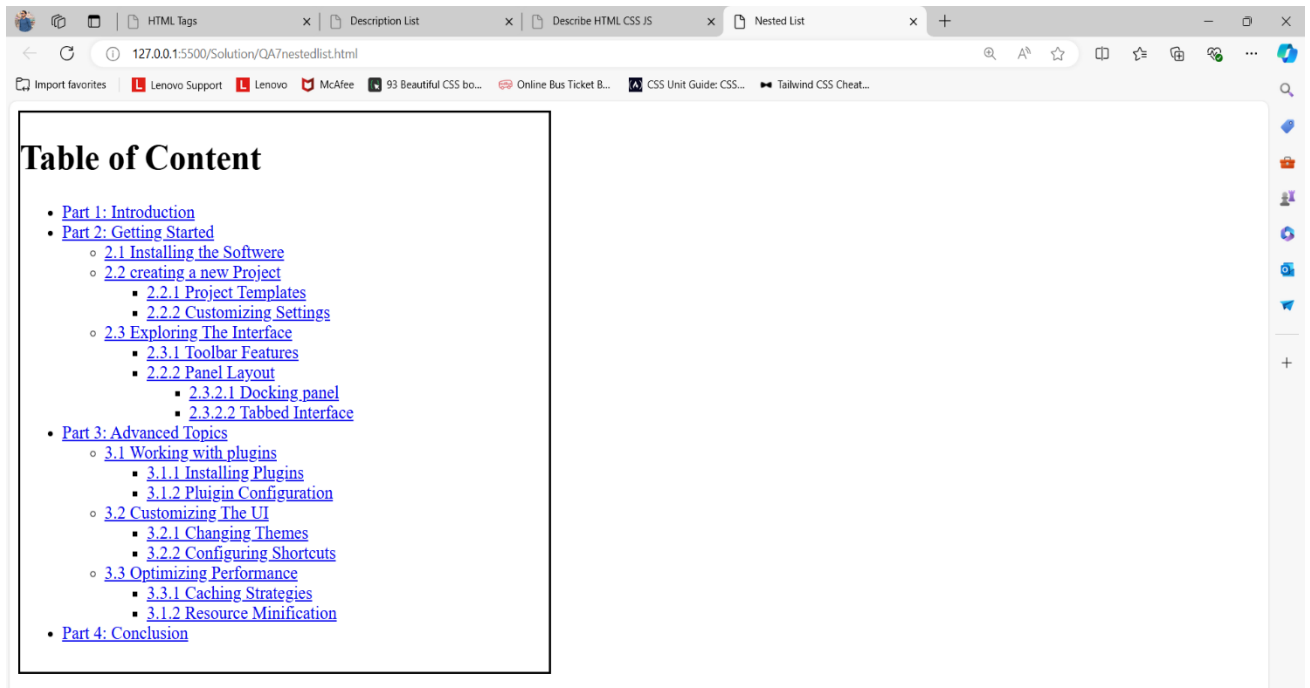 - 2.3 Exploring the Interface
    - 2.3.1 Toolbar Features
    - 2.3.2 Panel Layout
      - 2.3.2.1 Docking Panels
      - 2.3.2.2 Tabbed Interface
- Part 3: Advanced Topics
  - 3.1 Working with Plugins
    - 3.1.1 Installing Plugins
    - 3.1.2 Plugin Configuration
  - 3.2 Customizing the UI
    - 3.2.1 Changing Themes
    - 3.2.2 Configuring Shortcuts
  - 3.3 Optimizing Performance
    - 3.3.1 Caching Strategies
    - 3.3.2 Resource Minification
- Part 4: Conclusion

Code

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```html
<title>Nested List</title>
<style>
    .table{
        border: 2px solid black;
        height: 500px;
        width: 500px;
    }
</style>
</head>
<body>

    <div class="table">
    <h1>Table of Content</h1>
    <ul>
        <li><a href="#">Part 1: Introduction</a></li>
        <li><a href="#">Part 2: Getting Started</a></li>
        <ul>
            <li><a href="#">2.1 Installing the Softwere</a></li>
            <li><a href="#">2.2 creating a new Project</a></li>
            <ul type="squre">
             <li><a href="#">2.2.1 Project Templates</a></li>
             <li><a href="#">2.2.2 Customizing Settings</a></li>
            </ul>
            <li><a href="#">2.3 Exploring The Interface</a></li>
            <ul type="squre">
                <li><a href="#">2.3.1 Toolbar Features</a></li>
                <li><a href="#">2.2.2 Panel Layout</a></li>
                <ul type="squre">
                    <li><a href="#">2.3.2.1 Docking panel</a></li>
```

```html
            <li><a href="#">2.3.2.2 Tabbed Interface</a></li>
          </ul>
        </ul>
      </ul>
      <li><a href="#">Part 3: Advanced Topics</a></li>
      <ul>
        <li><a href="#">3.1 Working with plugins</a></li>
        <ul type="squre">
          <li><a href="#">3.1.1 Installing Plugins</a></li>
          <li><a href="#">3.1.2 Pluigin Configuration</a></li>
        </ul>
        <li><a href="#">3.2 Customizing The UI</a></li>
        <ul type="squre">
          <li><a href="#">3.2.1 Changing Themes</a></li>
          <li><a href="#">3.2.2 Configuring Shortcuts</a></li>
        </ul>
        <li><a href="#">3.3 Optimizing Performance</a></li>
        <ul type="squre">
          <li><a href="#">3.3.1 Caching Strategies</a></li>
          <li><a href="#">3.1.2 Resource Minification</a></li>
        </ul>
      </ul>
      <li><a href="#">Part 4: Conclusion</a></li>
    </ul>
  </div>
</body>
</html>
```

Out Put

Question7: Create a table to display a conference schedule. Each row corresponds to a time slots, and each column corresponds to a room. Some time slots might have multiple sessions running simultaneously in different rooms. Utilize rowspan and colspan attribute as necessary to accommodate this complex schedule.

Output should look like this:

Code

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        div{
            border: 2px solid black;
            height: 420px;
            width: 750px;
            margin: auto;
        }
        table{
            border: 1px solid black;
            height: 300px;
            width: 700px;
            text-align: center;
            margin: auto;
        }
        tr,td,th{
            border: 1px solid black;
        }
        h1{
            text-align: center;
        }
    </style>
</head>
```

```html
</head>
<body>
  <div>
  <table>
  <h1>Confrance Schedule</h1>
  <tr>
    <th>Time</th>
    <th>Room1</th>
    <th>Room2</th>
    <th>Room3</th>
    <th>Room4</th>
  </tr>
  <tr>
    <td rowspan="3">9:00 AM - 10:00 Am</td>
    <td rowspan="2">Keynote</td>
    <td>Session A</td>
    <td>Session B</td>
    <td rowspan="3">Session c</td>
  </tr>
  <tr>
    <td>Session D</td>
    <td>Session E</td>
  </tr>
  <tr>
    <td>10:30 AM - 11:30 AM</td>
    <td colspan="2">Session F</td>
  </tr>
  <tr>
    <td>12:00 PM - 1:00 PM</td>
```

```html
      <td colspan="4">Lunch Break</td>
    </tr>
    <tr>
      <td rowspan="2">1:00 PM - 2:00 PM</td>
      <td>Session G</td>
      <td rowspan="2">Session H</td>
      <td>Session I</td>
      <td>Session J</td>
    </tr>
    <tr>
     <td>Session K</td>
     <td>Session L</td>
     <td>Session M</td>
    </tr>
   </table>
</div>
</body>
</html>
```

Out Put