TASK 1:Build a simple webpage that displays text as shown in the below image.

SOLUTION:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>mohitwebsite</title>

</head>

<body>

<p>

<b>This text will be bolded.</b>

</p>

<p>

<i>this text will be italic</i>

</p>

<p>

<ul>this text will be underlined</ul>

</p>

<p>

<mark>this text will be highlighted</mark>

</p>

<p>

This is normal text<sup>This will be super scripted</sup>This is normal text

</p>

<p>

this is normal text <sub>This text will be subscripted</sub>

</p>

<p>

normal text <small>small text</small>

</p>

<p>

<del>this text will be deleted</del>

</p>

<body>

</html>

TASK 2: 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.

SOLUTION:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>mohitwebsite</title>

</head>

<body>

<h1> navigate me </h1>

<p>

take me to

<a href="https://pwskills.com/"target="-blank">pw skills</a> to buy a course

</p>

<p>

take me to

<a href="https://developer.mozilla.org/en-US/" target="\_blank">MDN docs</a> to know 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>

TASK 3 : 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.

SOLUTION:

<title>web design</title>

<head>

<body>

<div>

<a href="home.html">home</a> |

<a href="webdevelopment.html">web development</a>|

<a href="webdesign.html">web design</a>

</div>

<div>

<p>Web design refers to the design of websites. It usually refers to the user

experience aspects of website development rather than software

development. Web design used to be focused on designing websites for

desktop browsers; however, since the mid-2010s, design for mobile and

tablet browsers has become ever-increasingly important.

</p>

</div>

</body>

</head>

<title>web development</title>

<head>

<body>

<div>

<a href="home.html">home</a> |

<a href="webdevelopment.html">web development</a>|

<a href="webdesign.html">web design</a>

</div>

<div>

<p>Web development, also known as website development, refers to the tasks associated with creating,

building, and maintaining websites and web applications that run online on a browser.It may,

however, also include web design, web programming, and database management.

</p>

</div>

</body>

</head>

TASK 4 : Create an ordered list of HTML tags. Each list item must include the tag name and some information about the tag.

SOLUTION :

<title>web development</title>

<head>

<body>

<ol>

<li>h1 : define a heading tag level 1</li>

<li>p : define paragraph tag that contain elements</li>

<li> div : a container of element</li>

<li> table : define a table with rows and cloumn</li>

</ol>

</body>

</head>

TASK 5 : 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

SOLUTION :

<title>web development</title>

<head>

<body>

<h1> FULL STACK WEB DEVELOPMENT</h1>

<dt>HTML</dt>

<dd>HTML elements are the building blocks of HTML pages. With HTML constructs, images and other

objects such as interactive forms may be embedded into the rendered page. HTML provides a

means to create structured documents by denoting structuralsemantics for text such as headings

, paragraphs, lists, links, quotes, and other items.

</dd>

<dt>CSS</dt>

<dd>CSS is designed to enable the separation of content and presentation, including layout,

colors, and fonts.[3] This separation can improve content accessibility; provide more

flexibility and control in the specification of presentation characteristics;enable

multiple web pages to share formatting by specifying the relevant CSS in a separate

</dd>

<dt>javascript</dt>

<dd>JavaScript is a high-level, often just-in-time compiled language that conforms to the

ECMAScript standard.It has dynamic typing, prototype-based object-orientation,and first-class

functions. It is multi-paradigm, supporting event-driven, functional, and imperative

programming styles. It has application programming interfaces (APIs)for working With

text, dates, regular expressions, standard data structures, and theDocument Object Model (DOM).

</dd>

<dt>node.js</dt>

<dd>

Node.js lets developers use JavaScript to write command line tools and for server-side scripting.

The ability to run JavaScript code on the server is often used to generate dynamic web page content

before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript

everywhere" paradigm,[6] unifying web-application development around a single programming language,

as opposed to using different languages for the server- versus client-side programming.

</dd>

<dt>express.js</dt>

<dd>

Express.js, or simply Express, is a back end web application framework for building RESTful APIs

with Node.js, released as free and open-source software under the MIT License. It is designed

for building web applications and APIs.[3] It has been called the de facto standard server

framework for Node.js

</dd>

<dt>MongoDB</dt>

<dd>

MongoDB is a source-available cross-platform document-oriented database program. Classified

as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB

is developed by MongoDB Inc. and current versions are licensed under the Server Side Public

License (SSPL) which is considered non-free by some organizations and distributions

</dd>

<dt>react.js</dt>

<dd>

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript

library[3][4] for building user interfaces based on components. It is maintained by Meta

(formerly Facebook) and a community of individual developers and companies

</dd>

</body>

</head>

TASK 6 : 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 techstack name, its primary use cases, and some key features or benefits. Below is a reference image.

SOLUTION :

<title>web development</title>

<head>

<body>

<ol>

<li>

<h2>HTML</h2>

<table border="1">

<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>

<h2>CSS</h2>

<table border="1">

<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>

</body>

</head>

TASK 7 : Build a complex nested list structure representing a multi level table of contents use unordered list (<ul>) and list items (<li>) with inline-block styling to create a structured layout. Apply formatting tags to enhance the presentation of the list items.

SOLUTION :

<title>web development</title>

<head>

<body>

<h1> TABLE OF CONTENT</h1>

<ul>

<li><a href="#">Part 1: Introduction</a></li>

<li><a href="#">Part 2: Getting Started</a>

<ul>

<li><a href="#">2.1 Installing the Software</a></li>

<li><a href="#">2.2 Creating a New Project</a>

<ul>

<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>

<ul>

<li><a href="#">2.3.1 toolbar features</a></li>

<li><a href="#">2.3.2 pannel layout</a></li>

</ul>

<ul>

<li><a href="#">2.3.2.1 dock pannel</a></li>

<li><a href="#">2.3.2.2 table interface</a></li>

</ul>

<li><a href="#">2.3 Exploring the Interface</a>

<ul>

<li><a href="#">2.3.1 Toolbar Features</a></li>

<li><a href="#">2.3.2 Panel Layout</a>

<ul>

<li><a href="#">2.3.2.1 Docking Panels</a></li>

<li><a href="#">2.3.2.2 Tabbed Interface</a></li>

</ul>

</li>

</ul>

</li>

</ul>

</li>

<li><a href="#">Part 3: Advanced Topics</a>

<ul>

<li><a href="#">3.1 Working with Plugins</a>

<ul>

<li><a href="#">3.1.1 Installing Plugins</a></li>

<li><a href="#">3.1.2 Plugin Configuration</a></li>

</ul>

</li>

<li><a href="#">3.2 Customizing the UI</a>

<ul>

<li><a href="#">3.2.1 Changing Themes</a></li>

<li><a href="#">3.2.2 Configuring Shortcuts</a></li>

</ul>

</li>

<li><a href="#">3.3 Optimizing Performance</a>

<ul>

<li><a href="#">3.3.1 Caching Strategies</a></li>

<li><a href="#">3.3.2 Resource Minification</a></li>

</ul>

</li>

</ul>

</li>

<li><a href="#">Part 7: Conclusion</a></li>

</ul>

</body>

</head>

TASK 8 : 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 ).

SOLUTION :

<title>web development</title>

<head>

<body>

<table border="1" cellpadding="10">

<thead>

<tr>

<th>Time</th>

<th>Room 1</th>

<th>Room 2</th>

<th>Room 3</th>

<th>Room 4</th>

</tr>

</thead>

<tbody>

<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>

<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>

</tbody>

</table>

</body>

</head>