

Core Html

Assignment Solution



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

Solution

```
<body>
  <p>
    <b> This text will be bolded. </b>
  </p>
  <p>
    <i> This text will be italic. </i>
  </p>
  <p>
    <ins> This text will be underlined</ins>
  </p>
  <mark> This text will be highlighted </mark>
  <p>
    This is normal text<sup> This will be super scripted </sup> This is normal
    again
  </p>
  <p>This is normal text <sub> This text will be subscripted </sub></p>
  <p>Normal Text <small> Smal Text </small></p>
  <p><del> This text will be deleted</del></p>
</body>
```

Browser 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 Smal Text

~~This text will be deleted~~

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

```
<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 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>
```

Browser 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.

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

home.index

```
<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>
    <h1>Web Design.</h1>
    <p>
      Web design refers to the process of designing the visual appearance and layout of websites. It involves creating a visual concept, planning the layout of web pages, selecting color schemes, and choosing typography and images.
      <br />
      Web designers use various tools and technologies to create visually appealing and user-friendly websites. These tools include graphic design software such as Adobe Photoshop, Adobe Illustrator, Sketch, and Figma. They also use programming languages like HTML, CSS, and JavaScript to bring their designs to life on the web.
    </p>
  </div>
</body>
```

webDevelopment.index

```
<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>
<h1>Web Development</h1>
<p>
    Web development refers to the process of creating websites and web applications. This involves a combination of different technologies and programming languages, such as HTML, CSS, and JavaScript, to create web pages and make them interactive.
    <br />
    Web development can be divided into two main categories: front-end development and back-end development. Front-end development deals with the visual aspects of a website or application, including the design, layout, and user interface. This involves working with HTML, CSS, and JavaScript to create web pages that can be viewed in a browser.
</p>
</div>
</body>
```

webDesign.index

```
<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>
<h1>Web Design.</h1>
<p>
    Web design refers to the process of designing the visual appearance and layout of websites. It involves creating a visual concept, planning the layout of web pages, selecting color schemes, and choosing typography and images.
    <br />
    Web designers use various tools and technologies to create visually appealing and user-friendly websites. These tools include graphic design software such as Adobe Photoshop, Adobe Illustrator, Sketch, and Figma. They also use programming languages like HTML, CSS, and JavaScript to bring their designs to life on the web.
</p>
</div>
</body>
```

Browser output

Home Page

[Home](#) | [Web Development](#) | [Web Design](#)

PW Skills

PW Skills is the result of a continual effort to exponentially increase the employability of every Indian, irrespective of their socioeconomic status. With accessibility and affordability being the support structure of high-quality, industry-relevant courses, PW Skills aims to empower professionals and students alike to either jumpstart their careers or leverage existing skills with new, future-driven upgrades that will help them realise their full potential.

Web Development Page

[Home](#) | [Web Development](#) | [Web Design](#)

Web Development

Web development refers to the process of creating websites and web applications. This involves a combination of different technologies and programming languages, such as HTML, CSS, and JavaScript, to create web pages and make them interactive.

Web development can be divided into two main categories: front-end development and back-end development. Front-end development deals with the visual aspects of a website or application, including the design, layout, and user interface. This involves working with HTML, CSS, and JavaScript to create web pages that can be viewed in a browser.

Web design page

[Home](#) | [Web Development](#) | [Web Design](#)

Web Design.

Web design refers to the process of designing the visual appearance and layout of websites. It involves creating a visual concept, planning the layout of web pages, selecting color schemes, and choosing typography and images.

Web designers use various tools and technologies to create visually appealing and user-friendly websites. These tools include graphic design software such as Adobe Photoshop, Adobe Illustrator, Sketch, and Figma. They also use programming languages like HTML, CSS, and JavaScript to bring their designs to life on the web.

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

Solution

```
<body>
  <h1>List of Tags.</h1>
  <ol>
    <li>h1: Defines a heading level 1.</li>
    <li>p: Defines a paragraph of text.</li>
    <li>div: Defines a container for HTML elements.</li>
    <li>table: Defines a table with rows and columns.</li>
    <li>ul: Defines an unordered list.</li>
  </ol>
</body>
```

Browser output

List of Tags.

1. h1: Defines a heading level 1.
2. p: Defines a paragraph of text.
3. div: Defines a container for HTML elements.
4. table: Defines a table with rows and columns.
5. ul: Defines an unordered list.

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

```
<body>
  <h1>Full Stack Web Development Tech Stack:</h1>
  <dl>
    <dt>HTML</dt>
    <dd>
      HyperText Markup Language is the standard markup language for creating web pages and other online documents.
    </dd>
    <dt>CSS</dt>
    <dd>
      Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in HTML or XML.
    </dd>
    <dt>JavaScript</dt>
    <dd>
```

JavaScript is a programming language used to create interactive effects within web browsers.

```
</dd>
<dt>Node.js</dt>
<dd>
  Node.js is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside of a browser.
</dd>
<dt>Express.js</dt>
<dd>
  Express.js is a web application framework for Node.js designed for building web applications and APIs.
</dd>
<dt>MongoDB</dt>
<dd>
  MongoDB is a NoSQL document-oriented database program that uses JSON-like documents with optional schemas.
</dd>
<dt>React.js</dt>
<dd>
  React.js is a JavaScript library for building user interfaces or UI components.
</dd>
</dl>
</body>
```

Browser output

Full Stack Web Development Tech Stack:

HTML

HyperText Markup Language is the standard markup language for creating web pages and other online documents.

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in HTML or XML.

JavaScript

JavaScript is a programming language used to create interactive effects within web browsers.

Node.js

Node.js is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside of a browser.

Express.js

Express.js is a web application framework for Node.js designed for building web applications and APIs.

MongoDB

MongoDB is a NoSQL document-oriented database program that uses JSON-like documents with optional schemas.

React.js

React.js is a JavaScript library for building user interfaces or UI components.

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 tech stack name, its primary use cases, and some key features or benefits. Below is a reference image.

Eg.

1. HTML

Primary Use Cases	Key Features/Benefits
Building the structure of web pages	<ul style="list-style-type: none">◦ 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	<ul style="list-style-type: none">◦ Allows for separation of content and presentation◦ Enables responsive design◦ Offers a wide range of styling options

Solution

```
<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>
</tr>
</table>
</li>
<li>
    <h2>JavaScript</h2>
    <table border="1">
        <tr>
            <th>Primary Use Cases</th>
            <th>Key Features/Benefits</th>
        </tr>
        <tr>
            <td>Adding interactivity and functionality to web pages</td>
            <td>
                <ul>
                    <li>Can manipulate and modify web page content in real-time</li>
                    <li>
                        Offers a wide range of functionality through libraries and
                        frameworks
                    </li>
                    <li>Allows for server-side scripting with Node.js</li>
                </ul>
            </td>
        </tr>
    </table>
</li>
</ol>
</body>

```

Solution

1. HTML

Primary Use Cases	Key Features/Benefits
Building the structure of web pages	<ul style="list-style-type: none"> ◦ 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	<ul style="list-style-type: none"> ◦ 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 functionality to web pages	<ul style="list-style-type: none"> ◦ Can manipulate and modify web page content in real-time ◦ Offers a wide range of functionality through libraries and frameworks ◦ Allows for server-side scripting with Node.js

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

Solution

```
<body>
    <h1>Table of Contents</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>
            </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 4: Conclusion</a></li>
  </ul>
</body>
```

Browser output

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](#)

Assignment Solution

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

Solution

```
<body>
    <h1>Conference Schedule</h1>
    <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>
```

Browser output

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