Module 1

Q. 1 Explain basic Internet Protocols which are essential for transferring data and sending emails.

1. **Internet Protocol (IP)**:
   - IP is the fundamental protocol that enables the transfer of data from one device to another over the internet.
   - It defines how data packets are sent from a source to a destination, ensuring that each packet gets to the right location.
   - Uses unique IP addresses (IPv4 or IPv6) to identify each device.
2. **Transmission Control Protocol (TCP)**:
   - TCP is responsible for ensuring that data packets are transmitted reliably between devices.
   - It works alongside IP (forming the TCP/IP model) to guarantee that data is delivered accurately and in the correct order.
   - Establishes a connection using a "handshake" before transmitting data.
3. **User Datagram Protocol (UDP)**:
   - A faster, connectionless protocol for sending data without guarantees of reliability.
   - Does not establish connections or perform error checking, making it suitable for real-time applications.
   - Used in scenarios where speed is more important than accuracy (e.g., live streaming, online gaming).
4. **Simple Mail Transfer Protocol (SMTP)**:
   - SMTP is the protocol used for sending emails from a client to a server or from one server to another.
   - It defines how email messages are sent and routed between mail servers.
5. **Hypertext Transfer Protocol (HTTP/HTTPS)**:
   - HTTP is used for transferring web pages and data between browsers and servers.
   - Follows a request-response model, where the client requests data and the server responds.
   - **HTTPS** is a secure version that encrypts communication using SSL/TLS to protect data.
   - Stateless by nature, meaning each request is independent of the previous one.

6. **File Transfer Protocol (FTP)**

- FTP is used for transferring files between a client and a server over a network.
- It allows users to upload, download, and manage files on a remote server.

7. **Post Office Protocol (POP3)**

- POP3 is used to retrieve emails from a mail server to a client.
- It downloads emails from the server to the user’s local device and then (typically) deletes them from the server.

Q. 2 What is HTTP? Describe structure of HTTP request and response message

1. **HTTP** (Hypertext Transfer Protocol) is a protocol used for communication between web browsers (clients) and web servers.
2. It is the foundation of data communication for the World Wide Web
3. HTTP provides a standard between a web browser and a web server to establish communication.
4. It is a set of rules for **transferring data** from one computer to another.
5. Data such as text, images, and other multimedia files are shared on the World Wide Web.
6. **Stateless**: Each HTTP request from a client to server is independent. The server does not store any state information between requests.
7. The client (browser) sends a HTTP request to the server, and the server sends a response back to the client.
8. Whenever a web user opens their web browser, the user indirectly uses HTTP.
9. It is an application protocol that is used for distributed, collaborative, hypermedia information systems.

Structure of HTTP request and response message

HTTP is a request-response protocol, which means that for every request sent by a client (typically a web browser), the server responds with a corresponding response. The basic flow of an HTTP request-response cycle is as follows:

- **Client sends an HTTP request**: The client (usually a web browser) initiates the process by sending an HTTP request to the server. This request includes a request method (GET, POST), the target URI,headers, and an optional request body.
- **Server processes the request**: The server receives the request and processes it based on the requested method and resource. This may involve retrieving data from a database, executing server-side scripts, or performing other operations.

- **Server sends an HTTP response:** After processing the request, the server sends an HTTP response back to the client.
- **Client processes the response**: The client receives the server's response and processes it accordingly.

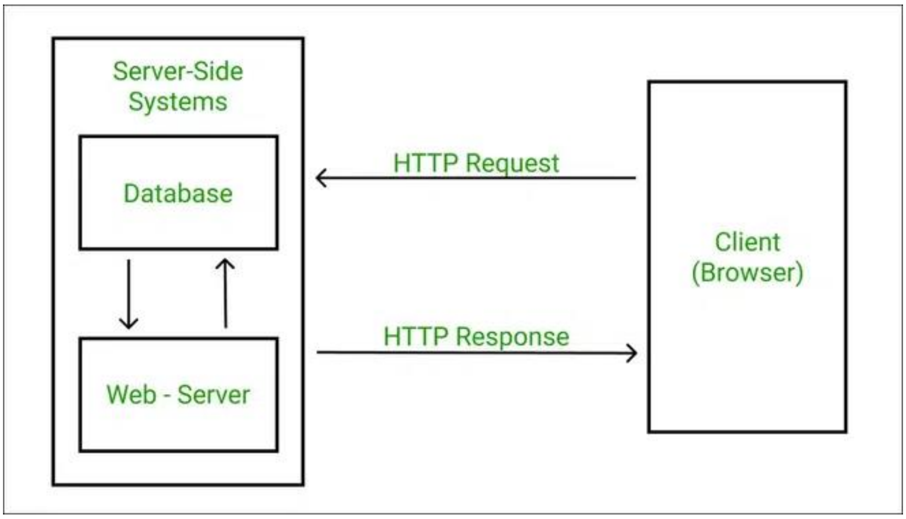

Fig : Structure of HTTP request and response message

Q. 3 Difference between HTML and HTML5.

Sr No.	HTML	HTML5
1	It didn't support audio and video without the use of flash player support.	It supports audio and video controls with the use of <audio> and <video> tags.
2	It uses cookies to store temporary data.	It uses SQL databases and application cache to store offline data.
3	Does not allow JavaScript to run in the browser.	Allows JavaScript to run in the background. This is possible due to JS Web worker API in HTML5.
4	It does not allow drag and drop effects.	It allows drag and drop effects.
5	Not possible to draw shapes like circle, rectangle, triangle etc.	HTML5 allows to draw shapes like circle, rectangle, triangle etc.
6	It works with all old browsers.	It supported by all new browser like Firefox, Mozilla, Chrome, Safari, etc.
7	<HTML>,<Body> , and <Head> tags are mandatory while writing a HTML code.	These tags can be omitted while writing HTML code.

8	Older version of HTML are less mobile-friendly.	HTML5 language is more mobile-friendly.
9	Doctype declaration is too long and complicated.	Doctype declaration is quite simple and easy.
10	Elements like nav, header were not present.	New element for web structure like nav, header, footer etc.
11	It can not handle inaccurate syntax.	It is capable of handling inaccurate syntax.
12	Being an older version , it is not fast , flexible , and efficient as compared to HTML5.	It is efficient, flexible and more fast in comparison to HTML.
13	Attributes like charset, async and ping are absent in HTML.	Attributes of charset, async and ping are a part of HTML 5.

Q. 4 Explain <audio> and <video> controls of HTML5 with appropriate example.

1. HTML <audio> Element

- The <audio> element is used to embed audio content in a webpage.
- The controls attribute adds audio controls, like play, pause, and volume.
- The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.
- The text between the <audio> and </audio> tags will only be displayed in browsers that do not support the <audio> element.

Key Attributes for <audio>:

- **controls**: Adds built-in playback controls (play, pause, volume).
- **autoplay**: Automatically starts playing the audio once the page is loaded.
- **loop**: Repeats the audio after it ends.
- **muted**: Starts the audio with no sound (muted).

Ex :

```
<audio controls autoplay loop muted>
    <source src="audiofile.mp3" type="audio/mp3">
    </audio>
```

2. HTML <video> Element

- The <video> element allows embedding video content into a webpage.
- The controls attribute adds video controls, like play, pause, and volume.
- It is a good idea to always include width and height attributes. If height and width are not set, the page might flicker while the video loads.

- The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.
- The text between the <video> and </video> tags will only be displayed in browsers that do not support the <video> element.

Key Attributes for <video>:

- **controls**: Displays default video controls such as play, pause, volume, seek bar, and fullscreen toggle.
- **autoplay**: Automatically plays the video when the page loads.
- **loop**: Repeats the video after it ends.
- **muted**: Starts the video with the sound turned off.
- **poster**: Sets a preview image before the video starts playing.

Ex :

```
<video width="640" height="360" controls autoplay loop muted>
<source src="videofile.mp4" type="video/mp4">
</video>
```

Q. 5 Explain the Document Object Model in detail with an example.

1. When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.
2. The **HTML DOM** model is constructed as a tree of **Objects.**

Key Concepts of the DOM:

1. **Document**: The DOM represents the entire web page as a document object.
2. **Objects**: Every element in the HTML document (such as <div>, <p>, or <img>) is represented as an object in the DOM.
3. **Nodes**: The DOM is structured as a tree of nodes. There are different types of nodes, including element nodes (representing HTML tags), text nodes (representing the content inside tags), and attribute nodes (representing the attributes of HTML tags).
4. **Tree Structure**: The DOM represents the HTML structure as a tree. The HTML document is the root, and every element is a branch of this tree. For example, <body>, <header>, and <footer> are children of the <html> element.

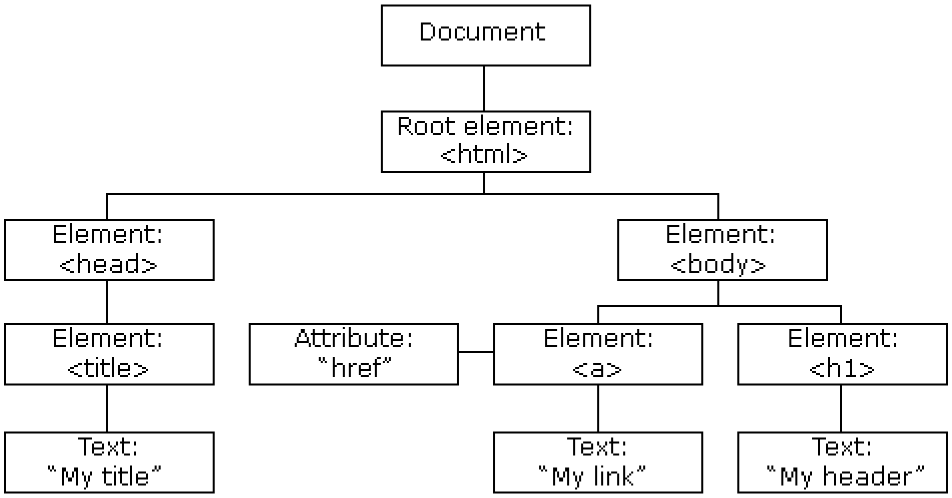

Fig : Structure of HTML DOM

- The Document node is the root of the DOM tree.
- The <html> element is the root element.
- Inside <html>, there are child nodes like <head> and <body>.
- Text nodes (like the text inside the <h1>, <p>, and <a> tags) are child nodes of their respective elements.

Ex :

```
<html>
<head>
<title>My title</title>
</head>
<body>
<h1 id="header">Welcome to the DOM</h1>
<a href="https://example.com" target="_blank">Click here</a>
</body>
</html>
```

Q. 6 Explain the working of rowspan and colspan of table when used in HTML with suitable example.

1. A table in HTML consists of table cells inside rows and columns.
2. Each table cell is defined by a <td> and a </td> tag.
3. Everything between <td> and </td> are the content of the table cell.
4. Each table row starts with a <tr> and ends with a </tr> tag.
5. Sometimes you want your cells to be table header cells. In those cases use the <th> tag instead of the <td> tag.

6. In HTML, many attributes help in developing the web page, including "colspan" and "rowspan", which are HTML attributes used inside the <th> and <td> tags.
7. These attributes help in creating more complex table layouts by merging cells either horizontally (columns) or vertically (rows).

HTML Table with Colspan

The "colspan" attribute is used to specify the number of columns a cell should span horizontally, meaning it merges multiple columns in the same row into one cell.

Syntax:

```
<td colspan="number">...</td>
```

HTML Table with Rowspan

The "rowspan" attribute is used to define the number of rows that a cell should span vertically, which means it merges multiple rows into one cell in the same column.

Syntax:

```
<td rowspan="number">...</td>
```

Ex :

```
<!DOCTYPE html>
<html>
<head>
<title>Table with Rowspan and Colspan Example</title>
</head>
<body>
<table border="1">
        <tr>
                <th colspan="2">Header 1</th>
                <th>Header 2</th>
        </tr>
        <tr>
                <td rowspan="2">Row 1, Cell 1<td>
                <td>Row 1, Cell 2</td>
                <td>Row 1, Cell 3</td>
        </tr>
        <tr>
                <td>Row 2, Cell 2</td>
                <td>Row 2, Cell 3</td>
        </tr>
```

```
</table>
</body>
</html>
```

Q. 7 Explain any 5 semantic tags of HTML5 with example.

- In HTML5, **semantic tags** are elements that give meaning to the structure of a web page.
- HTML5 Semantics refers to the use of specific tags like <header>, <footer>, <nav>, <article>, <section>, etc., to provide clearer structure and meaning to web content.
- Using semantic HTML tags enhances website accessibility, improves search engine optimization (SEO), and provides clearer structure and meaning to content.

1. <header>

The <header> tag defines the introductory content or navigational links for a section or page. It is typically used to group introductory elements like logos, headings, and navigation menus.

Example :

```
<header>
  <h1>My Website</h1>
  <nav>
    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#">About</a></li>
      <li><a href="#">Contact</a></li>
    </ul>
  </nav>
</header>
```

2. <nav>

The <nav> tag is used to define a block of navigation links. It typically contains a list of links that helps users navigate different sections or pages of the website.

Example :

```
<nav>
  <ul>
    <li><a href="#home">Home</a></li>
    <li><a href="#services">Services</a></li>
    <li><a href="#contact">Contact</a></li>
  </ul>
</nav>
```

3. <section>

The <section> tag defines a section of content, typically with a thematic grouping. It is used for grouping related content within a web page, such as a group of articles, a set of services, or any content related to a particular theme.

Example :

```
<section>
```

```
  <h2>Our Services</h2>
  <p>We offer web development, graphic design, and SEO services.</p>
</section>
```

4. <article>
The <article> tag represents independent content that can stand alone or be distributed outside of the website, such as blog posts, news articles, or forum posts. Each <article> element should be self-contained.

Example :

```
<article>
  <h2>Understanding Semantic HTML</h2>
  <p>Semantic HTML5 introduces new elements to describe the content of web
pages...</p>
</article>
```

5. <footer>
The <footer> tag defines the footer section of a webpage or a section. Typically, it contains information like copyright details, links to privacy policies, or contact information.

Example :

```
<footer>
  <p>&copy; 2024 My Website. All Rights Reserved.</p>
  <nav>
    <a href="#">Privacy Policy</a> |
    <a href="#">Terms of Service</a>
  </nav>
</footer>
```

Q. 8 Explain how Shadow effect can be applied on Text using CSS with suitable example.

1. In CSS, the **shadow effect** can be applied to text using the text-shadow property.
2. This property allows you to add one or more shadows to the text of an element, creating various visual effects.
3. In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px)

Syntax of text-shadow:

```
text-shadow: h-shadow v-shadow blur-radius color;
```

- **h-shadow**: The horizontal shadow (required). Positive values shift the shadow to the right, negative values shift it to the left.
- **v-shadow**: The vertical shadow (required). Positive values move the shadow down, and negative values move it up.
- **blur-radius**: The blur distance (optional). The higher the value, the more blurred the shadow will be. If omitted, the shadow is sharp.

- **color**: The color of the shadow (optional). It can be specified in any valid CSS color format (name, hex, rgb, rgba, etc.).

Applying a simple shadow effect to text:

Ex :

```
<html>
<head>
  <style>
    h1 {
      text-shadow: 2px 2px 4px gray;
    }
  </style>
</head>
<body>
  <h1>Shadow Effect on Text</h1>
</body>
</html>
```

Explanation:

2px 2px 4px gray:

1. **2px**: The shadow is 2 pixels to the right of the text (horizontal shadow).
2. **2px**: The shadow is 2 pixels below the text (vertical shadow).
3. **4px**: The shadow has a blur radius of 4 pixels, creating a soft, blurred shadow.
4. **gray**: The color of the shadow is gray.

Q. 9 what is inheritance in CSS.

1. Inheritance, a fundamental concept in object-oriented programming (OOP), mirrors real-life inheritance where children inherit traits from parents.
2. In OOP, a child class inherits properties from a parent class, establishing a hierarchical relationship between them.
3. CSS Inheritance: In CSS inheritance, the child element will naturally inherit properties from its parent element.

Syntax:

```
<style>
    #parentclass {
        color: red;
    }
</style>
<div id="parentclass">
    Parent Div
    <div id="div1Child">Child Div 1</div>
    <div id="div2Child">Child Div 2</div>
</div>
```

Example :

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #parentclass {
      padding: 30px;
      color: red;
    }
    #Child {
      padding: inherit;
    }
  </style>
</head>
<body>
  <div id="parentclass">
    Parent
    <div id="Child">Child</div>
  </div>
</body>
</html>
```

Q. 10 Explain CSS3 Animation with example.

- **CSS3 Animations** allow you to create smooth transitions and animations between the styles of HTML elements.
- Unlike CSS transitions, which are triggered by an event such as hovering or clicking, animations can be set to run automatically or in response to other triggers, and they can involve more complex, multi-step sequences.
- CSS Animations are essential for controlling the movement and appearance of elements on web pages.
- They consist of defining animation sequences using *@keyframes* and applying these sequences to elements using various animation properties.

Key Concepts of CSS3 Animations

1. **@keyframes**: This rule defines the animation by specifying a sequence of keyframes, or "snapshots" of the styles at various points in the animation.
2. **animation** properties: These are the properties used to apply and control the animation on a specific element.

Syntax of @keyframes :

```
@keyframes animation-name {
  0% { /* styles at the start */ }
  50% { /* styles halfway */ }
  100% { /* styles at the end */ }
}
```

Example :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>CSS3 Animation Example</title>
  <style>
    /* Keyframes for the animation */
    @keyframes moveAndChangeColor {
      0% {
        background-color: blue;
        left: 0;
      }
      50% {
        background-color: green;
        left: 50%;
      }
      100% {
        background-color: red;
        left: 100%;
      }
    }
    /* Applying the animation to the element */
    .box {
      width: 100px;
      height: 100px;
      position: absolute;
      background-color: blue;
      animation-name: moveAndChangeColor;
      animation-duration: 5s;
      animation-iteration-count: infinite;
      animation-timing-function: ease-in-out;
    }
  </style>
</head>
<body>
<div class="box"></div>
</body>
</html>
```

Q. 11 Explain CSS3 Animation Properties with example.

No.	Property	Description	Examples
1	@keyframes	The @keyframes rule in CSS is used to specify the animation rule.	@keyframes Demo { 0% {/* styles at the start */} 50% {/* styles halfway */} 100% {/* styles at the end */} }
2	animation-name	It is used to specify the name of the @keyframes describing the animation.	animation-name: Demo;
3	animation-duration	It is used to specify the time duration it takes animation to complete one cycle.	animation-duration: 5s;
4	animation-timing-function	It specifies how animations make transitions through keyframes. There are several presets available in CSS which are used as the value for the animation-timing-function like **linear**, **ease,ease-in**,**ease-out**, and **ease-in-out.**	animation-timing-function: ease-in-out;
5	animation-delay	It specifies the delay of the start of an animation.	animation-duration: 10s;
6	animation-iteration-count	This specifies the number of times the animation will be repeated.	animation-iteration-count: infinite;
7	animation-direction	It defines the direction of the animation. animation direction can be **normal**, **reverse**, **alternate**, and **alternate-reverse**.	animation-direction: normal;
8	animation-fill-mode	It defines how styles are applied before and after animation. The	animation-fill-mode: none;

		animation fill mode can be none, forwards, backwards, or both.	
9	animation-play-state	This property specifies whether the animation is running or paused.	animation-play-state: paused;

Q. 12 Create an External stylesheet and link it to an HTML form, the stylesheet should contain following:

i. **An header in text with red text color and yellow background color.**
ii. **A double lined(border) table.**
iii. **The table should have 5 rows and 3 columns.**
iv. **In the first column Sr. No. of the Product, Second column image with hyperlink and name of the product, and Third column the description of the product.**

CSS File name: style.css

```
header {
  color: red;
  background-color: yellow;
}
table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 20px;
}

table, th, td {
  border: 2px double black;
}
```

HTML File name : index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Product Table with External Stylesheet</title>
  <!-- Linking the external stylesheet -->
  <link rel="stylesheet" href="style.css">
</head>
<body>
```

```
<!-- i. Header -->
<header>
  Product List
</header>
<!-- iii. Table with 5 rows and 3 columns -->
<table>
  <!-- Table header row -->
  <tr>
    <th>Sr. No.</th>
    <th>Product Name & Image</th>
    <th>Description</th>
  </tr>
  <!-- Table rows with product details -->
  <tr>
    <td>1</td>
    <td>
      <a href="product1.html">
        <img src=" product1.png " alt="Product 1">
        Product 1
      </a>
    </td>
    <td>A brief description of product 1.</td>
  </tr>
  <tr>
    <td>2</td>
    <td>
      <a href=" product2.html ">
        <img src=" product2.png" alt="Product 2">
        Product 2
      </a>
    </td>
    <td>A brief description of product 2.</td>
  </tr>
  <tr>
    <td>3</td>
    <td>
      <a href=" product3.html ">
        <img src=" product3.png" alt="Product 3">
        Product 3
      </a>
    </td>
    <td>A brief description of product 3.</td>
  </tr>
  <tr>
```

```
    <td>4</td>
    <td>
      <a href=" product4.html ">
        <img src=" product4.png" alt="Product 4">
        Product 4
      </a>
    </td>
    <td>A brief description of product 4.</td>
  </tr>
  <tr>
    <td>5</td>
    <td>
      <a href=" product5.html ">
        <img src=" product5.png" alt="Product 5">
        Product 5
      </a>
    </td>
    <td>A brief description of product 5.</td>
  </tr>
</table>
</body>
</html>
```

Q. 13 Create an External stylesheet and link it to an HTML form, the stylesheet should contain following :

- i. **The web page will have the background image(image.jpg).**
- ii. **The table heading have the red background color.**
- iii. **Background color of alternate paragraph are different.**
- iv. **The hyperlink on web page will not have underlined.**

CSS File Name : style.css

```
/* i. Set a background image for the entire web page */
body {
  background-image: url('image.jpg');
  background-size: cover;
  background-repeat: no-repeat;
  background-position: center;
}

/* ii. Style for the table heading with red background color */
table th {
  background-color: red;
  color: white;  /* Optional: Makes the text white for contrast */
  padding: 10px;
}

/* iii. Style for alternate paragraph background colors */
p:nth-child(odd) {
  background-color: #f0f0f0;  /* Light gray background for odd paragraphs */
}

p:nth-child(even) {
  background-color: #e0e0e0;  /* Slightly darker background for even paragraphs */
}

/* iv. Remove underline from hyperlinks */
a {
  text-decoration: none;
}
```

HTML File name : index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Styled Form with External CSS</title>
  <!-- Link to the external stylesheet -->
  <link rel="stylesheet" href="style.css">
</head>
<body>

  <!-- Heading for the form -->
  <h1>Product Information Form</h1>

  <!-- Form with a table structure -->
  <form>
    <table border="1">
      <thead>
        <tr>
          <th>Product Name</th>
          <th>Product Price</th>
          <th>Product Category</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td><input type="text" name="productName"></td>
          <td><input type="number" name="productPrice"></td>
          <td>
            <select name="category">
              <option value="electronics">Electronics</option>
              <option value="fashion">Fashion</option>
              <option value="groceries">Groceries</option>
            </select>
          </td>
        </tr>
      </tbody>
    </table>

    <br>

    <!-- Additional information in paragraphs -->
    <p>This is some additional information about the product.</p>
    <p>This product is available in multiple colors.</p>
    <p>Make sure to double-check the price before submitting.</p>
    <p>Refer to the product manual for detailed information.</p>
  </form>

  <!-- Hyperlink on the webpage -->
  <a href=" example.html">Visit the product page</a>

</body>
</html>
```

Q. 14 List and explain the 3 way to add style sheet(CSS) to an HTML web page with suitable example.

1. CSS is a style sheet language that describes the presentation of a document written in HTML.
2. CSS describes how HTML elements should be displayed on the screen

Include CSS In HTML

There are 3 different ways to include CSS in HTML.

1. Inline CSS - Inline CSS is the CSS code that is written inside the HTML tag.
2. Internal CSS - Internal CSS is the CSS code that is written inside the HTML file in the <style> tag.
3. External CSS - External CSS is written in a separate CSS file and is included in the HTML file using the <link> tag.

1. **Inline CSS**
   - Inline CSS is the CSS code that is written inside the HTML tag.
   - An inline style may be used to apply a unique style for a single element.
   - To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
   - Example :

```
<!DOCTYPE html>
<html>
<body>
<h1 style="color:blue;text-align:center;">This is a heading</h1>
<p style="color:red;">This is a paragraph.</p>
</body>
</html>
```

2. **Internal CSS**
   - The internal CSS is also known as the embedded CSS
   - Internal CSS is the CSS code that is written inside the HTML file in the <style> tag.
   - An internal style sheet may be used if one single HTML page has a unique style.
   - The internal style is defined inside the <style> element, inside the head section.
   - Example :

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}
h1 {
  color: maroon;
  margin-left: 40px;
}
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

3. **External CSS**
   - External CSS is written in a separate CSS file and is included in the HTML file using the <link> tag.
   - With an external style sheet, you can change the look of an entire website by changing just one file!
   - Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.
   - Example :

     HTML Code :

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

CSS Code : mystyle.css

```
body {
  background-color: lightblue;
}

h1 {
  color: navy;
  margin-left: 20px;
}
```