Unit no 08: Web Development with HTML, CSS and JavaScript

Long Question Answers:

1. **Discuss the fundamental differences between HTML, CSS, and JavaScript in the context of web development.**

HTML	CSS	JavaScript
HTML is the standard language used to create web pages.	CSS allows web developers to control the colours, fonts, layout, and overall design of HTML elements, separating the content from the presentation.	JavaScript is a programming language that is used to make websites interactive and engaging. It allows developers to create things like animations, games, and responsive features that react when you click buttons or move your mouse.
To create a basic HTML application that displays "a message" on a web page, follow these simple steps: **1. Open your text editor,** You can use Notepad, Notepad++, Sublime Text, or any other text editor. **2. Write the following HTML code** into your text editor. **3. Save your file** with a html extension, for example, My first website.html.	**Styling Ways** • Inline Styles • Internal Styles • External Styles	**Example**: when you see a pop-up message on a web page or when an image changes when you hover over it, that's JavaScript at work. Execution of JavaScript in a flowchart.
Example html <html> <head>	**selector { property: value; }** For example, h1 { color: red;	**Syntax:** html <!DOCTYPE html> <html> <head> <title>JavaScript Example</title> <html>

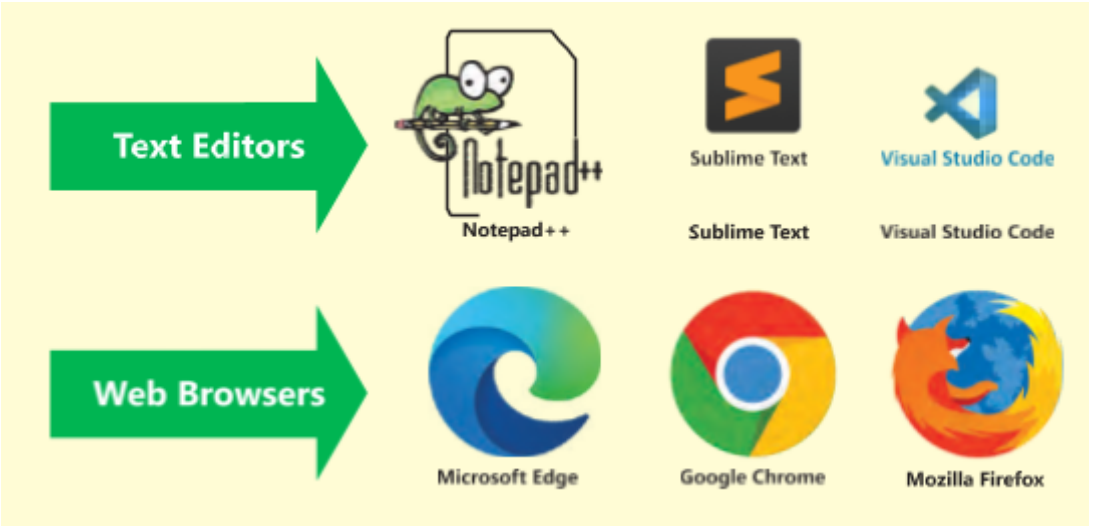
<title>Web Page</title> </head> <body> Welcome to My Website This is my first web page. I am learning HTML in the 9th class! </body> </html> >	font-size: 24px; } a simple CSS rule can change the color and size of all headings on a web page:	<body> Welcome JavaScript <script> alert("Hello, 9th Class Students"); </script> </body> </html>

2. **Explain the process of setting up a development environment for the web development. By discussing the necessary software and tools.**

   To start creating websites, you need a few basic tools and environments:

- **Text Editor:** This is where you write your HTML code. Popular text editors include Notepad++, Sublime Text, and Visual Studio Code.
- **Web Browser:** You will use this to view and test your HTML files. Common web browsers are Google Chrome, Mozilla Firefox, and Microsoft Edge.

  Start with a simple text editor and a web browser. Once you are comfortable with HTML, you can explore more advanced tools.



3. **Create a basic HTML page that includes a header, a paragraph, an image, and a hyperlink.**

```
<!DOCTYPE html>
<html lang=”en”>
<head>
  <title>Basic HTML page</title>
</head>
<body>

  <!-- Header -->
  <h1>Welcome to My Web Page</h1>

  <!-- Paragraph -->
  <p>This is a simple HTML page created to demonstrate basic elements like headers, paragraphs, images, and links.</p>

  <!-- Image -->
  <img src="https://www.w3schools.com/html/img_girl.jpg" alt="A sample image" width="300">

  <!-- Hyperlink -->
  <p>Visit <a href="https://www.example.com" target="_blank">Example.com</a> for more information.</p>

</body>
</html>
```

4. **How do you style a table using CSS? Create a sample table and apply styles to it.**

To style a table using CSS, you can apply properties like border, padding, background-color, text-align, and hover effects to elements such as <table>, <th>, and <td>. Below is a **sample HTML table** along with **CSS styles** applied to it.

```
<!DOCTYPE html>

<html>

<head>

 <style>

  table {

    width: 70%;
```

```
    border-collapse: collapse;
    margin: 20px auto;
    font-family: Arial, sans-serif;
    box-shadow: 0 0 10px rgba(0,0,0,0.1);
  }

  th, td {
    border: 1px solid #ccc;
    padding: 10px 15px;
    text-align: center;
  }

  th {
    background-color: #4CAF50;
    color: white;
  }

  tr:nth-child(even) {
    background-color: #f2f2f2;
  }

  tr:hover {
    background-color: #ddd;
  }
 </style>
</head>
```

```
<body>

 <h2 style="text-align:center;">Student Grades Table</h2>

 <table>
  <tr>
   <th>Name</th>
   <th>Subject</th>
   <th>Grade</th>
  </tr>
  <tr>
   <td>Ali</td>
   <td>Math</td>
   <td>A</td>
  </tr>
  <tr>
   <td>Ahmad</td>
   <td>Science</td>
   <td>B+</td>
  </tr>
  <tr>
   <td>Ayesha</td>
   <td>English</td>
   <td>A-</td>
  </tr>
 </table>
```

</body>

</html>

This Code Does:

- Adds a green header with white text.
- Alternates row colors using nth-child(even).
- Adds a hover effect to highlight a row when the mouse moves over it.
- Gives the table a neat shadow and spacing.

5. **Describe the different CSS selectors and provide examples of each.**
   Styling HTML Elements with Fonts, Colors, Backgrounds Styling Fonts:
   You can change the appearance of text on a web page using CSS. This includes changing the font family, size, weight, and style.
   Example of Styling Fonts

   Here is how you can style the font of a paragraph:

   P {

   font-family: Arial, sans-serif;
   font-size: 16px;

   font-weight:
   bold; font-style:
   italic;

   }

   In this example:

   - **font-family:** Arial, sans-serif; sets the font to Arial. If Arial is not available, it will use a generic sans-serif font.
   - **font-size:** 16px; sets the font size to 16 pixels.
   - **font-weight:** bold; makes the text bold.
   - **font-style:** italic; makes the text italic.

6. **Explain the process of creating a responsive web page using CSS with the help of examples and explanations.**

CSS Flexbox:

Flexbox is another layout tool that helps in arranging items in a flexible and responsive way. It is useful for aligning items in a row or column.

For Example:

.container

{ display:flex;

Justify-content:space-between;

}

.item

{ padding: 20px;

Background-color;lightgrey;

}

Positioning:

CSS positioning properties like position, top, left, right and bottom allow you to place elements exactly where you want them on webpage.

For example:

.box

{ position:absolute;

Top:50px;

Left:100px;

Width:200px;

Height:100px;

Background-color:lightblue;

}

7. **Write a JavaScript function that changes the background color of a web page when a button is clicked. Provide the complete code and explain how it works.**

```
<!DOCTYPE html>
<html>
<head>
  <title>Change Background Color</title>
</head>
<body>

  <h2>Click the Button to Change Background Color</h2>

  <button onclick="changeColor()">Change Color</button>

  <script>
    function changeColor() {
      document.body.style.backgroundColor = "lightblue";
    }
  </script>

</body>
</html>
```

How It Works:

<button> creates a clickable button.

onclick="changeColor()" tells the browser to run the changeColor function when the button is clicked.

The JavaScript function changeColor() changes the background color of the page to light blue.

8. **How do you add animations and transitions using CSS? Provide examples and explain the properties involved.**

CSS animations and transitions can make your web pages more engaging by adding movement and effects. Let us learn how to use them!

Adding Animations

CSS allows you to add animations to your web page to make it more interactive. Animations can change the way elements look or move over a period of time. Here are some basic steps to create animations with CSS:

- **Define Keyframes:** Keyframes are used to specify the start and end points of

an animation, as well as any intermediate steps. For example:

```
@keyframes example {

from {background-color: red;} to {background-color: yellow;}

}
```

This keyframe animation changes the background color from red to yellow.

- **Apply the Animation:** To apply the animation to an element, use the animation property.

For example:

```
.animated-box

{   width:  100px; height:
    100px;
    background-color:
    red;

    animation-name: example;
    animation-duration: 4s;

 }
```

This will change the background color of the box change from red to yellow for four seconds.

- **Loop and Timing:** You can also set how many times the animation should repeat and its timing function. For example:

```
.animated-box {

animation-iteration-count: infinite; /* Animation will repeat forever */-animation-
timing-function: linear; * /Animation will progress at a constant speed */

}
```

Adding Transitions

CSS allows you to add transitions to a web page to make changes between styles smooth and visually appealing. Transitions can change properties like color, size, or position gradually, instead of instantly. Here are some basic steps to create transitions with CSS:

- Set the Initial Style: First, define the initial style for the element you want to animate. For example:

.box {

width: 100px; height:
100px;
background-color:
red;

transition: background-color 2s, width 2s;

}

This sets the initial size and color of the box, and specifies that changes to the background color and width should transition over 2 seconds.

- **Define the Hover State:** Next, define the styles for the element when it is hovered over. For example:

.box:hover {

background-color: yellow; width: 200px;

}

This will change the background color to yellow and double the width of the box when the mouse hovers over it.