

## Task 2: Applying Styling and Form Creation to an HTML Webpage

### Objective:

Enhance your HTML webpage by integrating text formatting elements, color attributes, and understanding the difference between block and inline elements. Additionally, incorporate HTML input fields and forms to practice creating interactive user interfaces. This task aims to develop skills in styling and form handling in HTML, essential for creating more dynamic and visually appealing web pages.

### Pre-requisites:

- Basic understanding of HTML elements and attributes
- Familiarity with a code editor like Visual Studio Code
- Completion of Task 1

### Concepts Covered:

- Text Formatting Elements
- Color Attributes
- Block vs Inline Elements
- HTML Input Fields and Forms

### Concepts:

#### 1. Text Formatting Elements:

Use tags like `<strong>`, `<em>`, `<blockquote>`, and `<hr>` to format text and emphasize content.

```
<p>This is a <strong>bold</strong> text.</p>
<p>This is an <em>italic</em> text.</p>
<blockquote>This is a blockquote.</blockquote>
<hr>
```

#### 2. Color Attributes:

Use HTML color attributes to set text color and background color directly in elements.

```
<p style="color: red;">This text is red.</p>
<p style="background-color: yellow;">This text has a yellow background.</p>
```

#### 3. Block vs Inline Elements:

Differentiate between block-level elements ( `<div>`, `<p>`, `<h1>` ) and inline elements ( `<span>`, `<a>`, `<img>` ). Block elements occupy the full width available, while inline elements only take up as much width as necessary.

```
<div>This is a block element.</div>
<span>This is an inline element.</span>
```

#### 4. HTML Input Fields and Forms:

Create a form using the `<form>` tag with various input fields like text, password, radio, checkbox, and submit. Use `<fieldset>` and `<legend>` to group related fields and `<label>` for accessibility.

```
<form action="/submit" method="post">
  <fieldset>
    <legend>Personal Information</legend>
    <label for="name">Name:</label>
    <input type="text" id="name" name="name"><br><br>
    <label for="password">Password:</label>
    <input type="password" id="password" name="password"><br><br>
    <input type="radio" id="male" name="gender" value="male">
    <label for="male">Male</label><br>
    <input type="radio" id="female" name="gender" value="female">
    <label for="female">Female</label><br><br>
    <input type="checkbox" id="subscribe" name="subscribe"
value="newsletter">
    <label for="subscribe">Subscribe to newsletter</label><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

#### Setup:

##### 1. Install Visual Studio Code (VS Code):

Download and install VS Code from [Visual Studio Code](#).

##### 2. Web Browsers:

Use Google Chrome or Mozilla Firefox for viewing your webpage and utilizing their developer tools for debugging.

#### Tasks:

##### 1. Text Formatting Elements (15 minutes):

- Use HTML tags like `<strong>`, `<em>`, `<blockquote>`, and `<hr>` to format text.
- Understand how these tags change the appearance and emphasis of text content.

##### 2. Exploring Color Attributes (15 minutes):

- Apply different color styles using HTML attributes for text and backgrounds.
- Learn about color codes (like hexadecimal, RGB, and color names) in HTML.

##### 3. Block vs Inline Elements (15 minutes):

- Differentiate between block-level elements (like `<div>`, `<p>`, `<h1>`) and inline elements (like `<span>`, `<a>`, `<img>`).
- Apply attributes to see how they occupy space and interact with each other on the webpage.



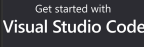

#### 4. HTML Input Fields and Forms (30 minutes):

- Create a form using the `<form>` tag with various input fields like text, password, radio, checkbox, and submit.
- Practice grouping input fields with `<fieldset>` and `<legend>` for better organization.
- Implement labels for each input field using `<label>` for accessibility.


#### Instructions:

1. Write the required code in `index.html`.
2. Open the file in your web browser to ensure the code displays correctly.
3. Use the browser's developer tools to debug and inspect the elements.

#### Resources:

-  **HTML: HyperText Markup Language | MDN**  
HTML (HyperText Markup Language) is the most basic building block of the Web. It define...  
 <https://developer.mozilla.org/en-US/docs/Web/HTML>
-  **Documentation for Visual Studio Code**  
Find out how to set-up and get the most from Visual Studio Code. Optimized for building and d...  
 <https://code.visualstudio.com/docs>

#### Videos:

- 

#### GitHub Instructions:

1. **Open in Visual Studio Code:**

After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, VSCode will open the repository directly. If prompted, select "Open" or "Allow" to open the repository in VSCode.

2. **Open the Terminal in VSCode:**

In VSCode, open a terminal by selecting Terminal > New Terminal from the top menu.

3. **Complete the Task:**

In VSCode, write your solution in the `index.html` file.

4. **Run and Test Your Code:**

Open your `index.html` file in a web browser to ensure it works correctly. Use the following command:

```
open index.html
```

5. **Commit Your Changes:**

In the VSCode terminal, add your changes to git:

```
git add index.html
```

Commit your changes with a meaningful message:

```
git commit -m "Completed task 2"
```

6. **Push Your Changes to Your Repository:**

Push your changes to your forked repository:

```
git push origin main
```

7. **Create a Pull Request:**

Go to your repository on GitHub.

Click on the "Pull Requests" tab.

Click the "New Pull Request" button.

Ensure the base repository is the original template repository and the base branch is `main`.

Ensure the head repository is your forked repository and the compare branch is `main`.

Click "Create Pull Request".

Add a title and description for your pull request and submit it.

**Summary of Commands:**

```
# Open in Visual Studio Code

# Open terminal in

VSCode

# Complete the task by editing index.html

# Navigate to the directory containing index.html
cd path/to/your/index.html

# Run your code
open index.html

# Add, commit, and push your changes
git add index.html
git commit -m "Completed task 2"
git push origin main

# Create a pull request on GitHub
```