

Lab Assignment-14

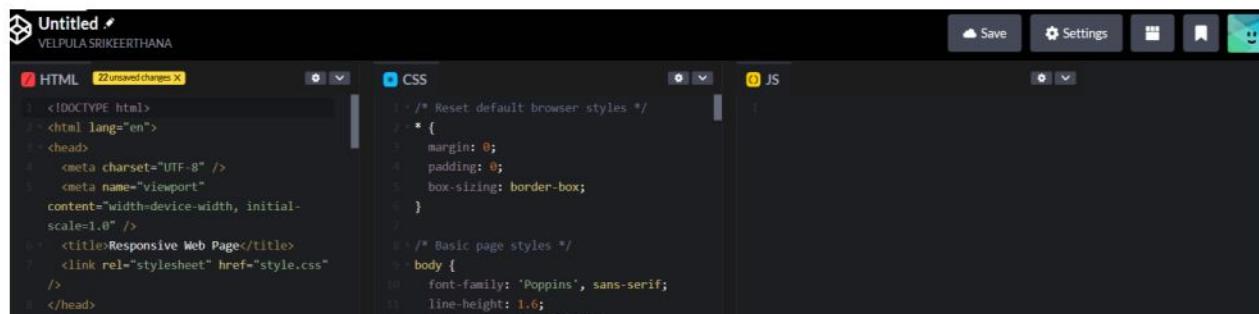
Task 1: Create a Responsive Web Page Layout

Instructions:

- Design a basic web page layout with a **header, main content area, and footer** using HTML and CSS.
- Use AI to assist in generating **responsive CSS** for different screen sizes.
- Ensure the layout is clean and visually organized.

Expected Output:

- A responsive web page with:
 - Header with navigation links
 - Main section with placeholder text/images
 - Footer with copyright or contact info
- Layout adapts correctly to desktop, tablet, and mobile screen sizes.



The screenshot shows a web development interface with three tabs: HTML, CSS, and JS. The HTML tab contains the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Responsive Web Page</title>
  <link rel="stylesheet" href="style.css" />
</head>
<body>
```

The CSS tab contains the following reset and basic styles:

```
/* Reset default browser styles */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

/* Basic page styles */
body {
  font-family: 'Poppins', sans-serif;
  line-height: 1.6;
  background-color: #f4f4f4;
```

The JS tab is currently empty. Below the editor, a preview window titled "My Website" shows a simple layout with a header, a main content area with a placeholder image, and a footer with navigation links.

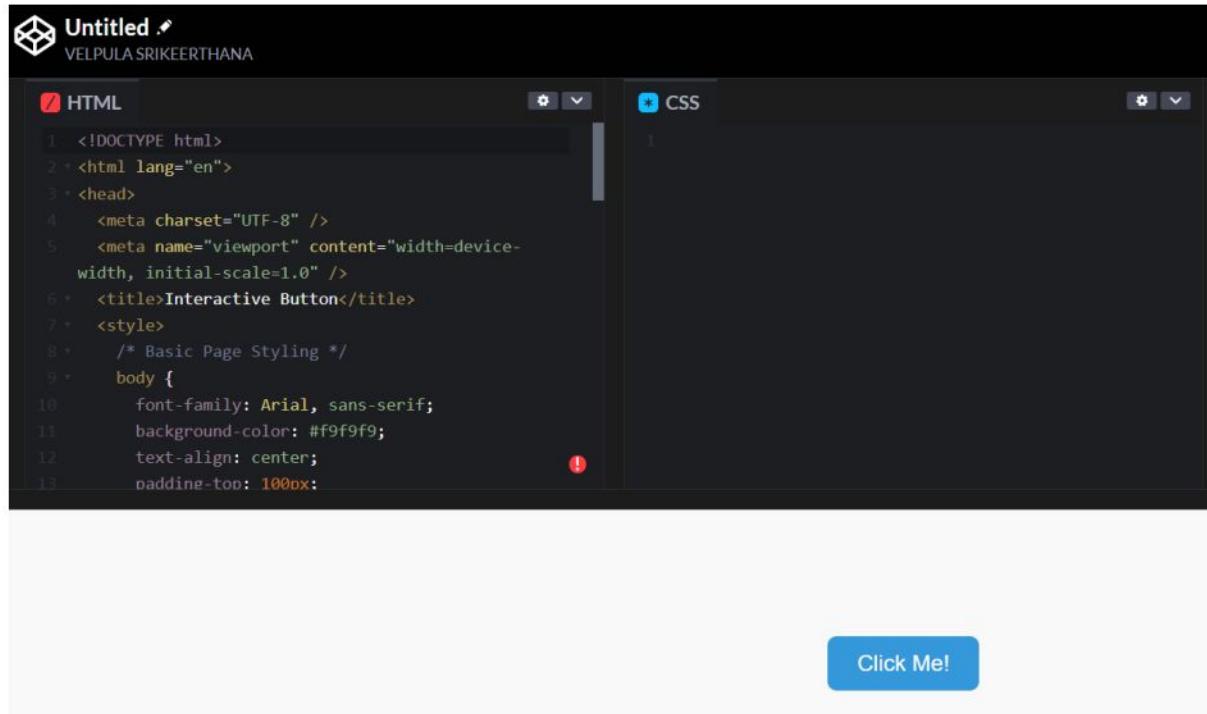
Task 2: Interactive Button with JavaScript

Instructions:

- Create a **button** on a web page.
- Use AI to generate JavaScript code that displays an **alert message** when the button is clicked.
- Ensure the code is clean and well-commented.

Expected Output:

- A web page with a button.
- Clicking the button shows a pop-up alert message, e.g., “Button clicked!”.



The screenshot shows a code editor interface with two tabs: "HTML" and "CSS". The "HTML" tab contains the following code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta name="viewport" content="width=device-
width, initial-scale=1.0" />
6   <title>Interactive Button</title>
7   <style>
8     /* Basic Page Styling */
9     body {
10       font-family: Arial, sans-serif;
11       background-color: #f9f9f9;
12       text-align: center;
13       padding-top: 100px;
```

The "CSS" tab is currently selected. Below the code editor is a preview area containing a single blue button with the text "Click Me!".

Task 3: Form with Validation

Instructions:

- Design a **contact form** with fields: Name, Email, Message.
- Use AI to generate JavaScript code for **form validation** (e.g., non-empty fields, valid email format).
- Add **inline error messages** if input is invalid.

Expected Output:

- A functional contact form where:
 - Submitting with empty fields shows error messages.
 - Submitting with valid input displays a “Form submitted successfully” message.

```
150     successMessage.textContent = "Form  
151     submitted successfully!";  
152     form.reset(); // Clear the form  
153   }  
154 };  
155 </script>  
156  
157 </body>  
158 </html>  
159 |
```

Enter your name
Email
Message
Write your message...

Task 4: Dynamic Content Generation

Instructions:

- Create a list of items (e.g., product names) using **HTML**.
- Use AI-generated JavaScript to **dynamically add or remove items** from the list when a button is clicked.

Expected Output:

- A web page with a list and buttons:
 - “Add Item” adds a new item to the list dynamically.
 - “Remove Item” deletes the last item.
- Updates occur without reloading the page.

```
count--;
Decrease counter
} else {
    alert("No more items to remove!");
    // Alert if list is empty
}
});
</script>
</body>
</html>
```

Product List



Task 5: Styled Modal Popup

Instructions:

- Use AI to generate a modal popup that opens when a button is clicked.
- Style the modal using CSS with a semi-transparent overlay.
- Include a close button that hides the modal.

Expected Output:

- A web page with a button labeled “Open Modal”.
- Clicking the button displays the modal popup with content.
- Modal can be closed by clicking the “X” button or overlay area.

Untitled ·
VELPULA SRIKEERTHANA

HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta name="viewport" content="width=device-
width, initial-scale=1.0" />
6   <title>Styled Modal Popup</title>
7   <style>
8     /* Basic Page Styling */
9    body {
10      font-family: 'Poppins', sans-serif;
11      background-color: #f4f4f4;
12      display: flex;
13      justify-content: center;
```

CSS

```
1
```

Open Modal