Web Technologies

Experiment

Name : Mikhail Janli Purba

USN : 21BTRCS249

Code of Experiment 1

**WEB TECHNOLOGY**

1. Create a personal web site with minimum 4 web pages using basic HTML tags.

**Objective:**

To learn and create a personal website with a minimum of 4 web pages using basic HTML tags.

**Requirements:**

* A text editor (e.g., Notepad, Visual Studio Code, Sublime Text, etc.)
* A web browser (e.g., Chrome, Firefox, Safari, etc.)

*Step-by-Step Instructions:*

1. *Setting up the directory structure:*

* Create a folder named 'personal\_website'.
* Inside 'personal\_website', create four HTML files named index.html, about.html, portfolio.html, and contact.html.

1. *Creating the Home Page (index.html):*

* Open index.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Home - My Personal Website</title>

</head>

<body>

    <h1>Welcome to My Personal Website</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>This is the homepage of my personal website.</p>

</body>

</html>

1. *Creating the About Page (about.html):*

* Open about.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>About Me - My Personal Website</title>

</head>

<body>

    <h1>About Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>Here is some information about me...</p>

</body>

</html>

1. *Creating the Portfolio Page (portfolio.html):*

* Open portfolio.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Portfolio - My Personal Website</title>

</head>

<body>

    <h1>My Portfolio</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>Here are some of my projects...</p>

</body>

</html>

1. *Creating the Contact Page (contact.html):*

* Open contact.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Contact - My Personal Website</title>

</head>

<body>

    <h1>Contact Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>You can contact me at: myemail@example.com</p>

</body>

</html>

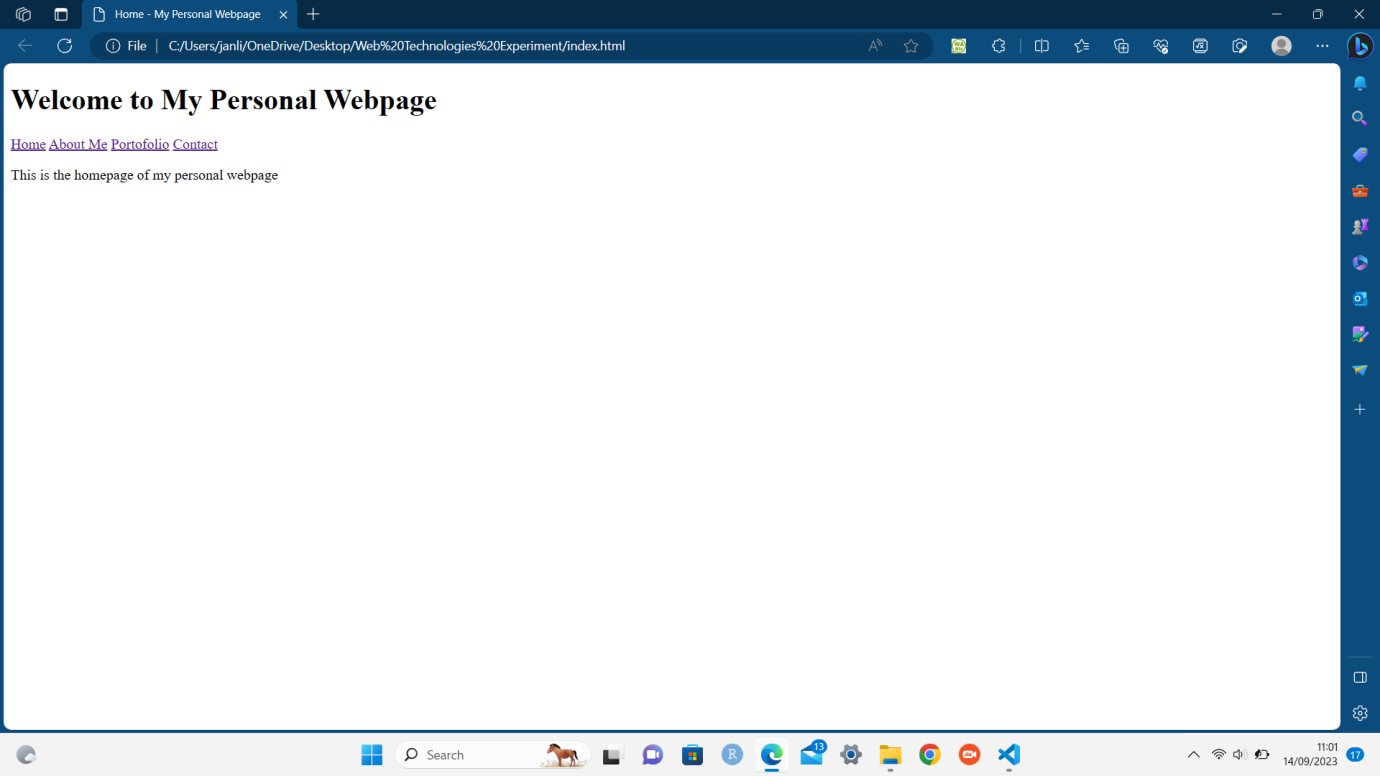
1. *Viewing the website:*

Open index.html using web browser. Navigate through personal website using the navigation links.

**Conclusion:**

Thus, successfully created a basic personal website with four separate pages using HTML.

Output of Experiment 1



Code of Experiment 2

**WEB TECHNOLOGY**

1. Create a static E-commerce website using Advanced HTML tags.

**Objective:** To create a basic static e-commerce website using advanced HTML tags.

**Tools Required:**

* A text editor (Notepad)
* A web browser (Chrome, or Firefox)

**Procedure:**

Setting Up the Environment:

* Open your preferred text editor.
* Create a new HTML file named index.html.

Creating the Header:

* Add a doctype declaration for HTML5 at the top of the file.
* Create a header for the website with a site title and navigation links.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>E-Shop</title>

</head>

<body>

    <header>

        <h1>E-Shop</h1>

        <nav>

            <ul>

                <li><a href="#home">Home</a></li>

                <li><a href="#products">Products</a></li>

                <li><a href="#about">About</a></li>

                <li><a href="#contact">Contact</a></li>

            </ul>

        </nav>

    </header>

</body>

</html>

Adding Products Section:

Create a products section with product cards using the <article> tag.

<section id="products">

    <h2>Products</h2>

    <article class="product-card">

        <img src="path/to/image1.jpg" alt="Product Image 1">

        <h3>Product Name 1</h3>

        <p>Product Description 1</p>

        <p>Price: $10</p>

    </article>

    <!-- Add more product cards similarly -->

</section>

Adding About Section:

Create an about section with information about the store using the <section> tag.

<section id="about">

    <h2>About Us</h2>

    <p>Welcome to E-Shop. We offer a wide range of products...</p>

    <!-- More about content -->

</section>

Creating Contact Section:

Add a contact section with a form for users to send inquiries.

<section id="contact">

    <h2>Contact Us</h2>

    <form action="#">

        <label for="name">Name:</label>

        <input type="text" id="name" name="name" required>

        <label for="email">Email:</label>

        <input type="email" id="email" name="email" required>

        <label for="message">Message:</label>

        <textarea id="message" name="message" rows="4" required></textarea>

        <button type="submit">Submit</button>

    </form>

</section>

Styling (Optional):

Add styles to your website using CSS either by linking an external stylesheet or using inline styles.

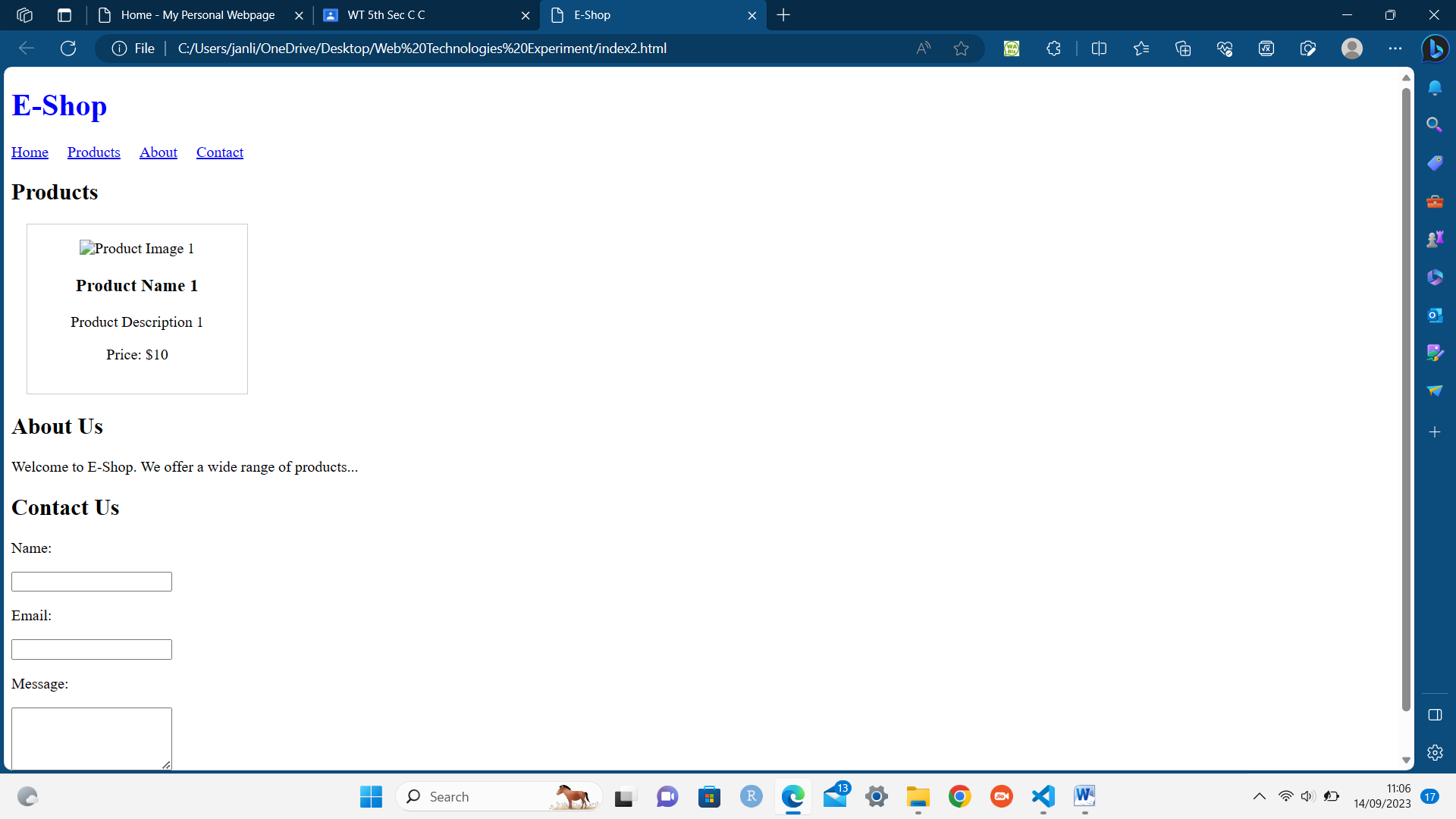
Testing the Website:

* Save the index.html file.
* Open it in a web browser to view the static e-commerce site.

**Conclusion**:

A basic static e-commerce website is created using advanced HTML tags.

Output of Experiment 2



Code of Experiment 3

**WEB TECHNOLOGY**

3. Enhance Exercise-1 with style properties using Inline, internal and external CSS.

**Objective:**

To learn and create a personal website with a minimum of 4 web pages using basic HTML tags and various CSS styling methods (Inline, Internal, and External).

**Requirements:**

* A text editor (e.g., Notepad, Visual Studio Code, Sublime Text, etc.)
* A web browser (e.g., Chrome, Firefox, Safari, etc.)

**Step-by-Step Instructions:**

***Part A:*** Setting up the Directory Structure

* Create a folder named personal\_website on your computer.
* Inside personal\_website, create four HTML files named index.html, about.html, portfolio.html, and contact.html.
* Create a CSS file named styles.css.

***Part B:*** Creating the Home Page (index.html)

* Open index.html in your text editor.
* Type the following HTML code, enhanced with inline and internal CSS:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <title>Home - My Personal Website</title>

    <style>

        body {

            background-color: lightgray;

        }

    </style>

</head>

<body>

    <h1 style="color: red;">Welcome to My Personal Website</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>This is the homepage of my personal website.</p>

</body>

</html>

***Part C:*** Creating the About Page (about.html)

* Open about.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <title>About Me - My Personal Website</title>

</head>

<body>

    <h1 style="color: green;">About Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>Here is some information about me...</p>

</body>

</html>

***Part D:*** Creating the Portfolio Page (portfolio.html)

* Open portfolio.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <title>Portfolio - My Personal Website</title>

</head>

<body>

    <h1 style="color: orange;">My Portfolio</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>Here are some of my projects...</p>

</body>

</html>

***Part E:*** Creating the Contact Page (contact.html)

* Open contact.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <title>Contact - My Personal Website</title>

</head>

<body>

    <h1 style="color: purple;">Contact Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a>

    </nav>

    <p>You can contact me at: <a href="mailto:myemail@example.com">myemail@example.com</a></p>

</body>

</html>

***Part F:*** Adding External CSS (styles.css)

* Open styles.css in your text editor.
* Type the following CSS code:

/\* styles.css \*/

body {

    font-family: Arial, sans-serif;

}

nav {

    margin-bottom: 20px;

    background-color: lightblue;

    padding: 10px;

}

a {

    text-decoration: none;

    margin: 0 15px;

}

p {

    font-size: 16px;

    line-height: 1.6;

}

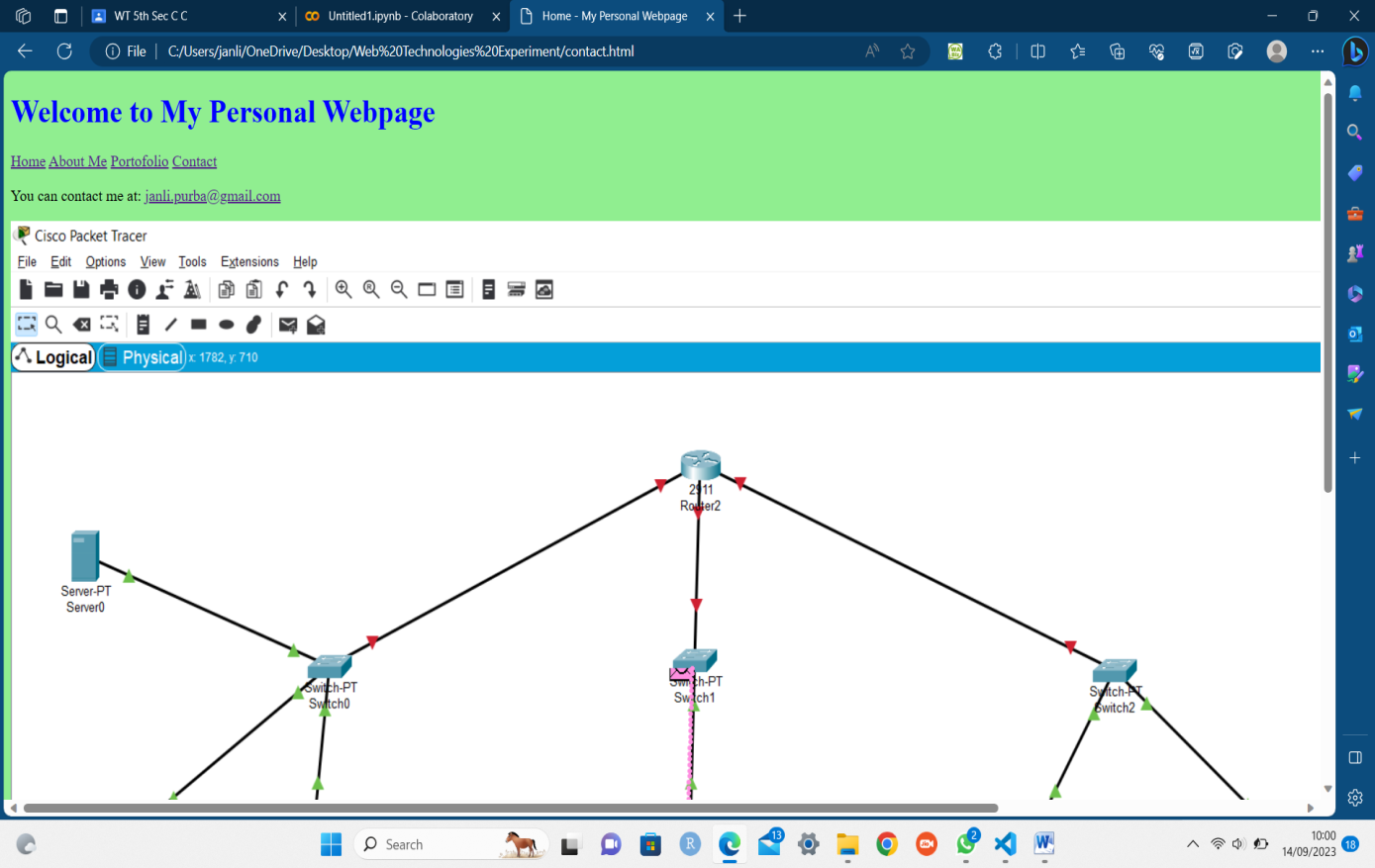
***Part G:*** Viewing the Website

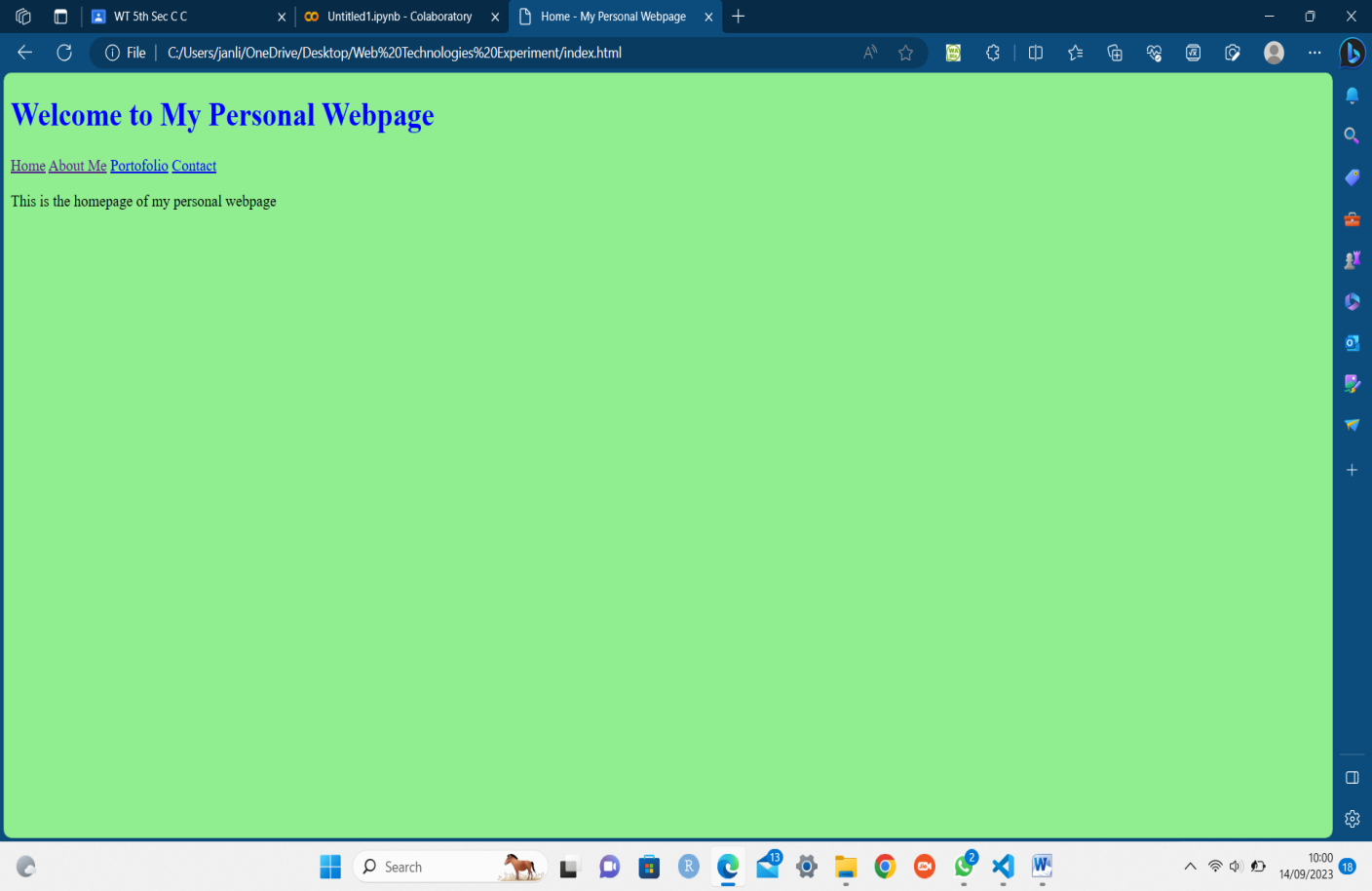
* Open index.html using your web browser.
* Navigate through your personal website using the navigation links.

**Conclusion:**

Thus, successfully created a basic personal website with four separate pages using HTML and various CSS styling methods (Inline, Internal, and External).

Output of Experiment 3





Code of Experiment 4

**WEB TECHNOLOGY**

1. Enhance Exercise-2 with style properties using Inline, internal and external CSS.

**1. Introduction**

This experiment provides an overview of the static E-commerce website, detailing its HTML structure, styling methods, and key code blocks. The experiment uses a mix of inline, internal, and external CSS for styling.

**2. File Structure**

index.html: The main HTML file containing the layout and content.

styles.css: The external stylesheet for global styles

**3. HTML Structure**

The main HTML structure consists of:

* Header: Contains the site title and navigation links.
* Products Section: Displays product cards with images, names, descriptions, and prices.
* About Section: Offers information about the store.
* Contact Section: Contains a form for users to send inquiries.

**4. Styling Methods**

Three types of CSS styling methods are used:

* Inline CSS: Applied directly to individual HTML elements.
* Internal CSS: Defined in the <head> section within a <style> tag.
* External CSS: Stored in an external .css file and linked to the HTML file.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

**<!-- External CSS -->**

    <title>E-Shop</title>

    <style>

**/\* Internal CSS \*/**

        nav ul {

            list-style-type: none;

            padding: 0;

            margin: 0;

            display: flex;

        }

        nav ul li {

            margin-right: 20px;

        }

        .product-card {

            border: 1px solid #ccc;

            padding: 16px;

            margin: 16px;

            width: 200px;

            text-align: center;

        }

        form label, form input, form textarea {

            display: block;

            margin-bottom: 16px;

        }

    </style>

</head>

<body>

    <header>

        <h1 style="color: blue;">E-Shop</h1> <!-- Inline CSS -->

        <nav>

            <ul>

                <li><a href="#home">Home</a></li>

                <li><a href="#products">Products</a></li>

                <li><a href="#about">About</a></li>

                <li><a href="#contact">Contact</a></li>

            </ul>

        </nav>

    </header>

    <section id="products">

        <h2>Products</h2>

        <article class="product-card">

            <img src="path/to/image1.jpg" alt="Product Image 1">

            <h3>Product Name 1</h3>

            <p>Product Description 1</p>

            <p>Price: $10</p>

        </article>

        <!-- Add more product cards similarly -->

    </section>

    <section id="about">

        <h2>About Us</h2>

        <p>Welcome to E-Shop. We offer a wide range of products...</p>

        <!-- More about content -->

    </section>

    <section id="contact">

        <h2>Contact Us</h2>

        <form action="#">

            <label for="name">Name:</label>

            <input type="text" id="name" name="name" required>

            <label for="email">Email:</label>

            <input type="email" id="email" name="email" required>

            <label for="message">Message:</label>

            <textarea id="message" name="message" rows="4" required></textarea>

            <button type="submit">Submit</button>

        </form>

    </section>

</body>

</html>

**External CSS**

To demonstrate external CSS, let's pretend we have an external file named **styles.css** that contains some global styles.

**/\* External CSS \*/**

body {

    font-family: Arial, sans-serif;

    margin: 0;

    padding: 0;

}

header {

    background-color: #f4f4f4;

    padding: 20px;

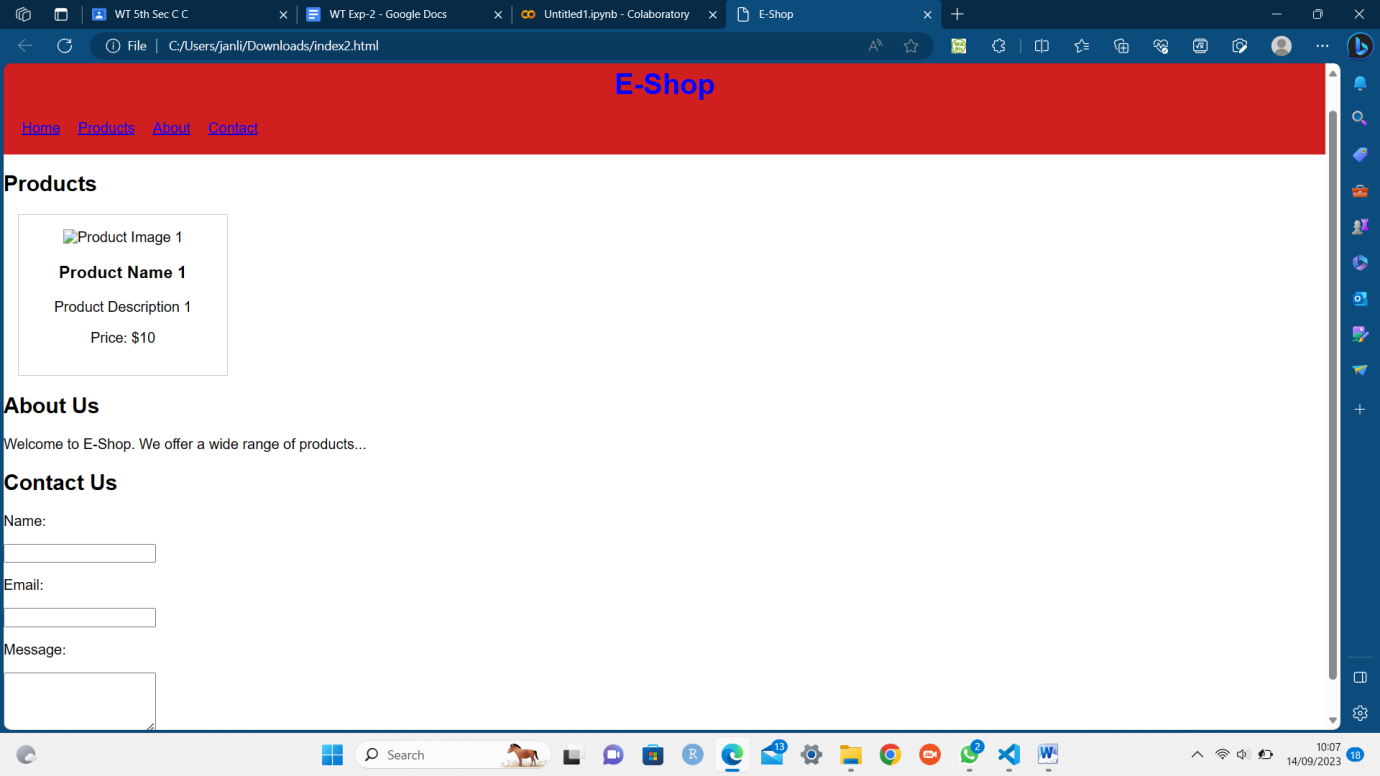
    text-align: center;

}

**Conclusion**:

Thus, exercise-2 is enhanced with style properties using Inline, internal and external CSS is created using advanced HTML tags.

Output of Experiment 4



Code of Experiment 5

**WEB TECHNOLOGY**

1. Add scientific calculator as a menu in your Exercise-1 using java script.

**Objective:**

To integrating a Scientific Calculator into a Personal Website using basic HTML tags and JavaScript.

**Requirements:**

* Basic understanding of HTML, CSS, and JavaScript.
* A text editor (e.g., Notepad, Visual Studio Code, Sublime Text, etc.)
* A web browser (e.g., Chrome, Firefox, Safari, etc.)

Setting up the Directory Structure:

* Create a folder named 'personal\_website'.
* Inside 'personal\_website', create five HTML files named index.html, about.html, portfolio.html, contact.html, and calculator.html.
* Also, create a style.css file inside 'personal\_website' for styling.

*Step-by-Step Instructions:*

1. *Setting up the directory structure:*

* Create a folder named 'personal\_website'.
* Inside 'personal\_website', create four HTML files named index.html, about.html, portfolio.html, and contact.html.

1. *Creating the Home Page (index.html):*

* Open index.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Home - My Personal Website</title>

</head>

<body>

    <h1>Welcome to My Personal Website</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a> |

**<a href="calculator.html">Calculator</a>**

    </nav>

    <p>This is the homepage of my personal website.</p>

</body>

</html>

1. *Creating the About Page (about.html):*

* Open about.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>About Me - My Personal Website</title>

</head>

<body>

    <h1>About Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a> |

**<a href="calculator.html">Calculator</a>**

    </nav>

    <p>Here is some information about me...</p>

</body>

</html>

1. *Creating the Portfolio Page (portfolio.html):*

* Open portfolio.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Portfolio - My Personal Website</title>

</head>

<body>

    <h1>My Portfolio</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a> |

**<a href="calculator.html">Calculator</a>**

    </nav>

    <p>Here are some of my projects...</p>

</body>

</html>

1. *Creating the Contact Page (contact.html):*

* Open contact.html in your text editor.
* Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Contact - My Personal Website</title>

</head>

<body>

    <h1>Contact Me</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a> |

**<a href="calculator.html">Calculator</a>**

    </nav>

    <p>You can contact me at: myemail@example.com</p>

</body>

</html>

1. *Creating the Calculator Page (calculator.html):*

Open calculator.html in your text editor.

Type the following HTML code:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Calculator - My Personal Website</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <h1>Scientific Calculator</h1>

    <nav>

        <a href="index.html">Home</a> |

        <a href="about.html">About Me</a> |

        <a href="portfolio.html">Portfolio</a> |

        <a href="contact.html">Contact</a> |

        <a href="calculator.html">Calculator</a>

    </nav>

    <div class="calculator">

        <input type="text" id="display" disabled>

        <button onclick="clearDisplay()">C</button>

        <button onclick="appendToDisplay('7')">7</button>

        <button onclick="appendToDisplay('8')">8</button>

        <button onclick="appendToDisplay('9')">9</button>

        <button onclick="appendToDisplay('/')">/</button>

        <button onclick="appendToDisplay('4')">4</button>

        <button onclick="appendToDisplay('5')">5</button>

        <button onclick="appendToDisplay('6')">6</button>

        <button onclick="appendToDisplay('\*')">\*</button>

        <button onclick="appendToDisplay('1')">1</button>

        <button onclick="appendToDisplay('2')">2</button>

        <button onclick="appendToDisplay('3')">3</button>

        <button onclick="appendToDisplay('-')">-</button>

        <button onclick="appendToDisplay('0')">0</button>

        <button onclick="appendToDisplay('.')">.</button>

        <button onclick="calculate()">=</button>

        <button onclick="appendToDisplay('+')">+</button>

    </div>

    <script src="script.js"></script>

</body>

</html>

Now, create a new file named script.js inside 'personal\_website' for the calculator's JavaScript functionality.

1. *Adding CSS styles (style.css):*

Add the following code to style.css:

.calculator {

    display: grid;

    grid-template-columns: repeat(4, 1fr);

    gap: 10px;

    max-width: 300px;

    margin: 50px auto;

}

.calculator button {

    padding: 20px;

    font-size: 1.5rem;

}

.calculator input {

    grid-column: span 4;

    padding: 10px;

    font-size: 1.5rem;

}

1. *Adding JavaScript Functionality (script.js):*

Add the following code to script.js:

function appendToDisplay(value) {

    document.getElementById('display').value += value;

}

function clearDisplay() {

    document.getElementById('display').value = '';

}

function calculate() {

    try {

        let result = eval(document.getElementById('display').value);

        document.getElementById('display').value = result;

    } catch(e) {

        document.getElementById('display').value = 'Error';

    }

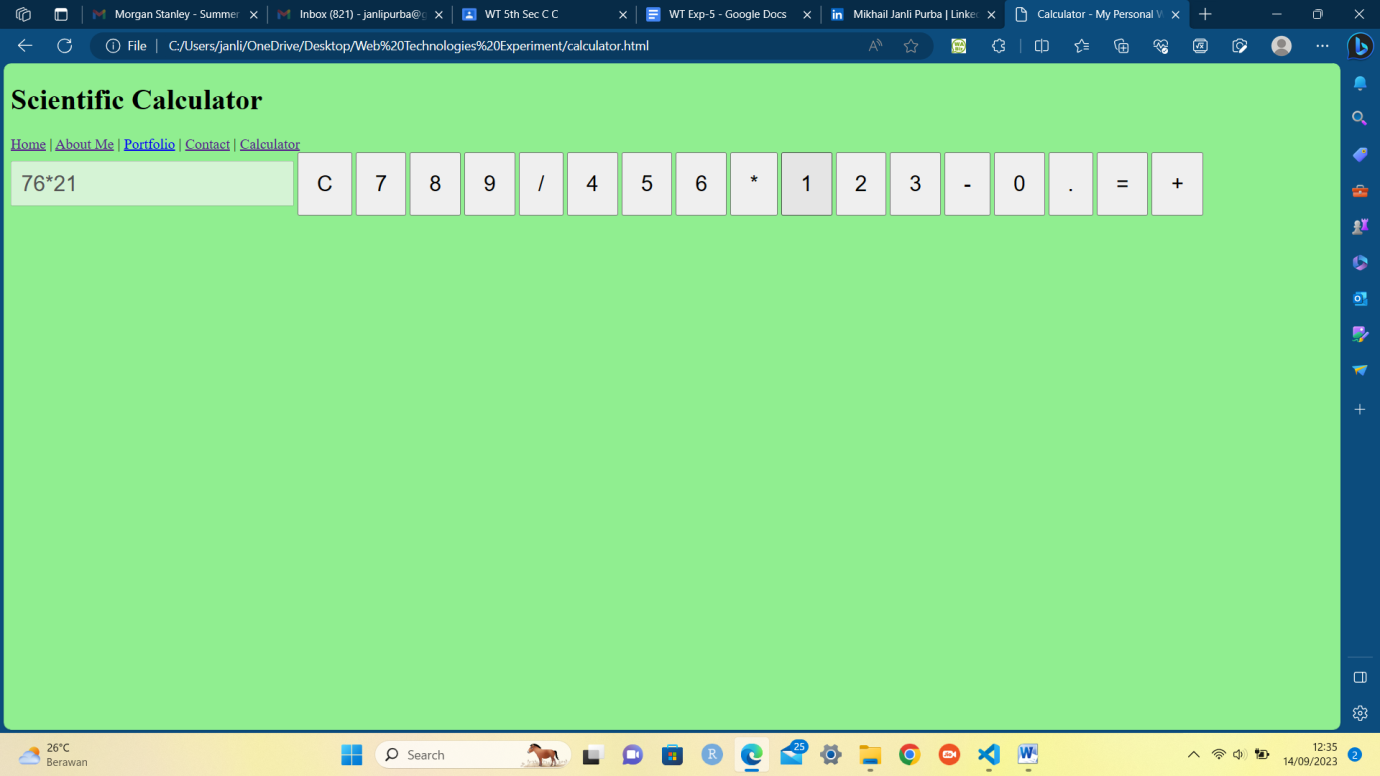
}

1. *Viewing the website:*

Open index.html using web browser. Navigate through personal website using the navigation links.

**Conclusion:**

The integration of a scientific calculator into the personal website offers enhanced functionality, catering to users who require mathematical computations, thus enriching the overall website experience.

Output of Experiment 5

**WEB TECHNOLOGY**

1. Create registration form, login form for Exercise-2 and validate using java script.

**Objective:** To create a registration and login form for an E-Shop and validate the forms using JavaScript.

**Tools Required:**

* A text editor (Notepad)
* A web browser (Chrome, or Firefox)

**Procedure:**

* Create a new HTML file.
* Copy the initial HTML code provided (from the E-Shop header down to the contact section).
* Add the registration form HTML code to the file below the contact section.
* Add the login form HTML code below the registration form section.
* Add the JavaScript validation code within a <script> tag before the closing </body> tag.
* Save the HTML file and open it in a web browser.
* Try entering different values into the registration and login forms and observe the validation.

**REGIDTRATION FORM:**

<section id="register">

    <h2>Register</h2>

    <form id="registrationForm" action="#">

        <label for="reg-username">Username:</label>

        <input type="text" id="reg-username" name="username" required>

        <label for="reg-email">Email:</label>

        <input type="email" id="reg-email" name="email" required>

        <label for="reg-password">Password:</label>

        <input type="password" id="reg-password" name="password" required>

        <label for="reg-confirm-password">Confirm Password:</label>

        <input type="password" id="reg-confirm-password" name="confirm-password" required>

        <button type="submit">Register</button>

    </form>

</section>

**LOGIN FORM:**

<section id="login">

    <h2>Login</h2>

    <form id="loginForm" action="#">

        <label for="login-username">Username:</label>

        <input type="text" id="login-username" name="username" required>

        <label for="login-password">Password:</label>

        <input type="password" id="login-password" name="password" required>

        <button type="submit">Login</button>

    </form>

</section>

**JAVA SCRIPT VALIDATION:**

document.addEventListener("DOMContentLoaded", function() {

    document.getElementById("registrationForm").addEventListener("submit", function(event) {

        let username = document.getElementById("reg-username").value;

        let email = document.getElementById("reg-email").value;

        let password = document.getElementById("reg-password").value;

        let confirmPassword = document.getElementById("reg-confirm-password").value;

        if (password !== confirmPassword) {

            alert("Passwords do not match!");

            event.preventDefault();

        }

    });

    document.getElementById("loginForm").addEventListener("submit", function(event) {

        let username = document.getElementById("login-username").value;

        let password = document.getElementById("login-password").value;

        if (username.trim() === "" || password.trim() === "") {

            alert("Both fields are required!");

            event.preventDefault();

        }

    });

});

**Conclusion**:

In this exercise, we created a registration and login form for an E-Shop and added validation using JavaScript to ensure data integrity and improve user experience.

Output of Experiment 6

