**Roll No.: 76**

**Name: Raj Snajay Koli**

**Assignment No: 1.1**

**Title: Create a basic HTML document 1.Create a simple webpage using correct HTML syntax and document structure, including<!DOCTYPE html> <html>,<head>,<title> and<body> tags.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>My first webpage with attribute</title>

</head>

<body bgcolor="lightgreen" text="black" link="blue" vlink="purple" alink="red" >

    <h1 id="mainheading" title="welcome heading">Welcome to my first webpage</h1>

    <p id="intro" title="introduction paragraph">

        This is a <b>Basic HTML Document</b> of multiple attributes.

        It shows how to use attributes like <i>id</i>,<i>title</i>,<i>lang</i> and more.

    </p>

    <a href="https://imr.ac.in" target="\_blank" title="Go to example website">This is example of website</a>

    <img src="nature.jpg"

    alt="placeholder image" align="center" height="200" width="200" id="sampleimage" title="Sample image">

    <hr>

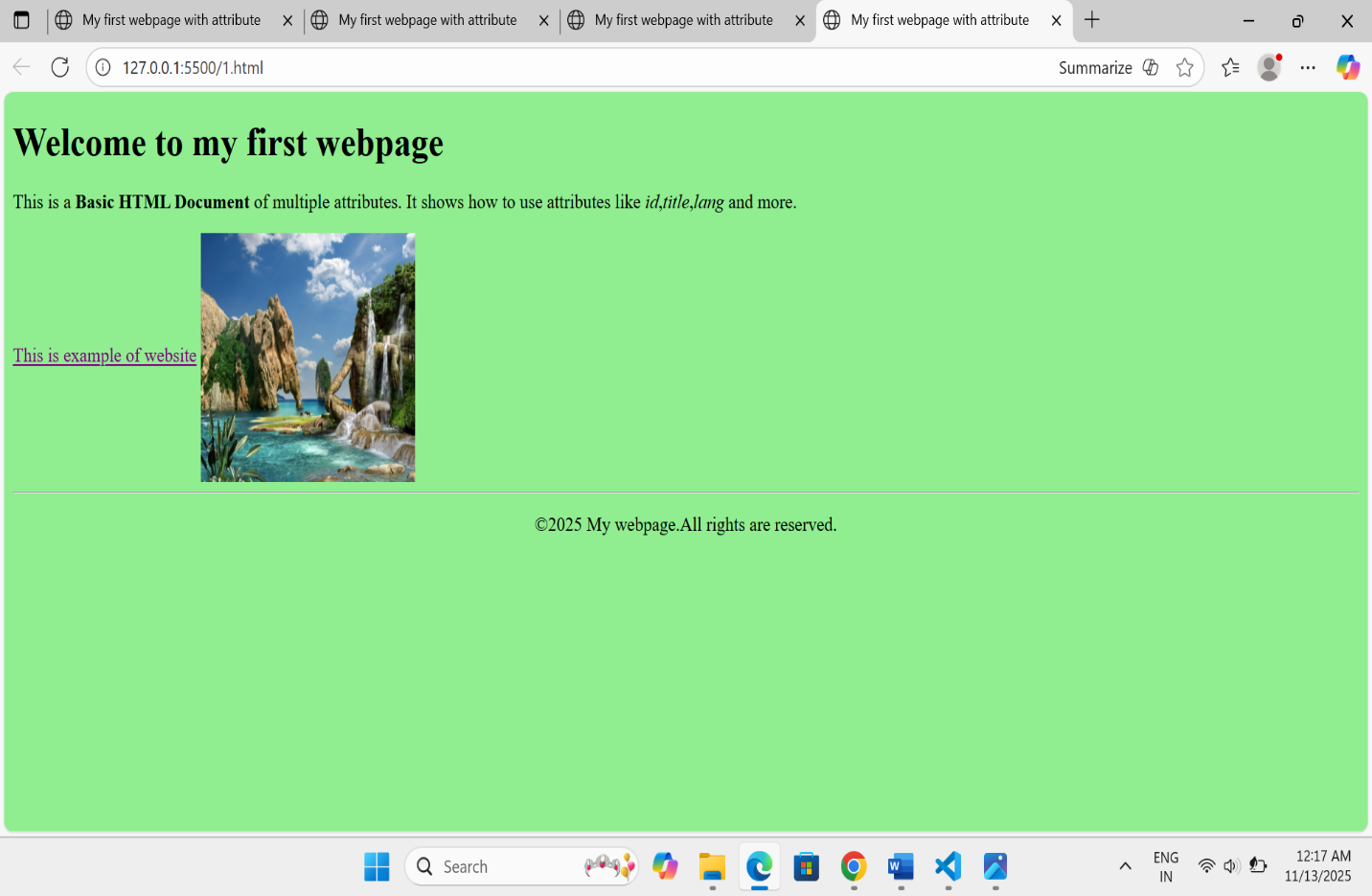
    <p align="center" id="footer" title="footer paragraph">

        ©2025 My webpage. All rights are reserved.

    </p>

</body>

</html>

**Output:**

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 1.2**

**Title: 2. Add placeholder text in the body section and display a title in the browser’s title bar using the <title> tag**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Basics of Web Development</title>

</head>

<body>    <h1>Hello, Welcome To My Webpage</h1>

    <h2> What is Web Development</h2>

    <p> Web development is creating user-friendly websites and web applications to accomplish business or personal goals.

         Every single business wants to build their web presence in today's time.

         There is an increase in the number of websites and applications.

    <h2> Who's a Web Developer? </h2>

    <p> A web developer works with the front-end and back-end of a website or an application.

        The full-stack developer handles projects anywhere from database design, to user interface design, and development.</p>

    <h3> Website and Web Servers</h3>

    <p> Obviously, before building anything, we must know what we are building.

        So, before learning to develop any website or web application, you must know the fundamentals of a website and servers.</p>

    <h3> Front-End Web Development</h3>

    <p> It is also referred to as client-side development. It means building a website or online application's User Interface (UI).

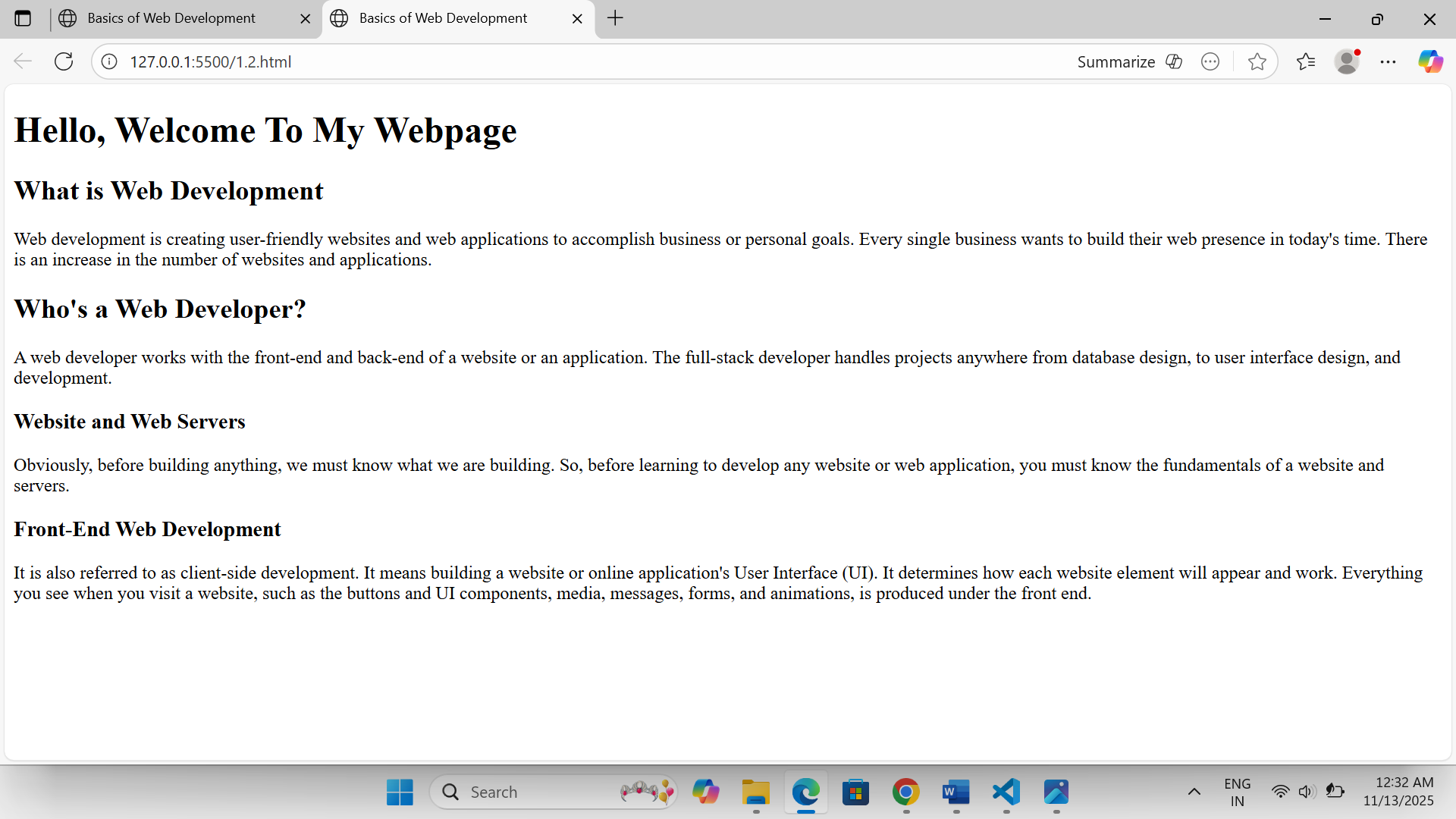
        It determines how each website element will appear and work.

        Everything you see when you visit a website, such as the buttons and UI components, media, messages,

        forms, and animations, is produced under the front end.</p>

</body>

</html>

**Output** **:**

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 2.1**

**Title: Content Structuring: 1. Use <div>tags to structure the webpage into sections (e.g., a header, content area, and footer).**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>structure of web page</title>

</head>

<body>

        <h1>MEGA SALE-Up to 50%OFF!</h1>

        <p><strong>Limited time only.</strong> Grab your favorites before they are gone.</p>

    </div>

    <div id="content">

        <div id="banner">

            <img src="sale.jpeg" alt="Sale banner" height="150px" width="200px">

        </div>

        <div id="products">

            <h2> Featured Products</h2>

          <div class="product">

             <h3>Laptop X100</h3>

             <img src="https://www.brilliantnews.com/wp-content/uploads/2016/02/laptop.jpg" alt="Smartphone X100" height="150px" width="150px">

             <p>Now only <strong>₹49,999</strong> -was<del>₹60,000</del></p>

             <p>Key features: 6.7" display, 128GB storage, 50MP camera.</p>

             <form action="#" method="post">

                <input type="hidden" name="product id" value="X100">

                <input type="submit" value="Buy Now">

              </form>

         </div>

       </div>

       <div id="contact">

        <h2>Have Questions?</h2>

        <p>Call us: <a href="Teliphone: +911234567890">+91 1234567890</a></p>

        <p>Email: <a href="mail to: abc@gmail.com"> abc@gmail.com</a></p>

       </div>

    </div>

    <div id="footer">

        <p>&copy;2025 ShopEasy Pvt.Ltd.All rights reserved.</p>

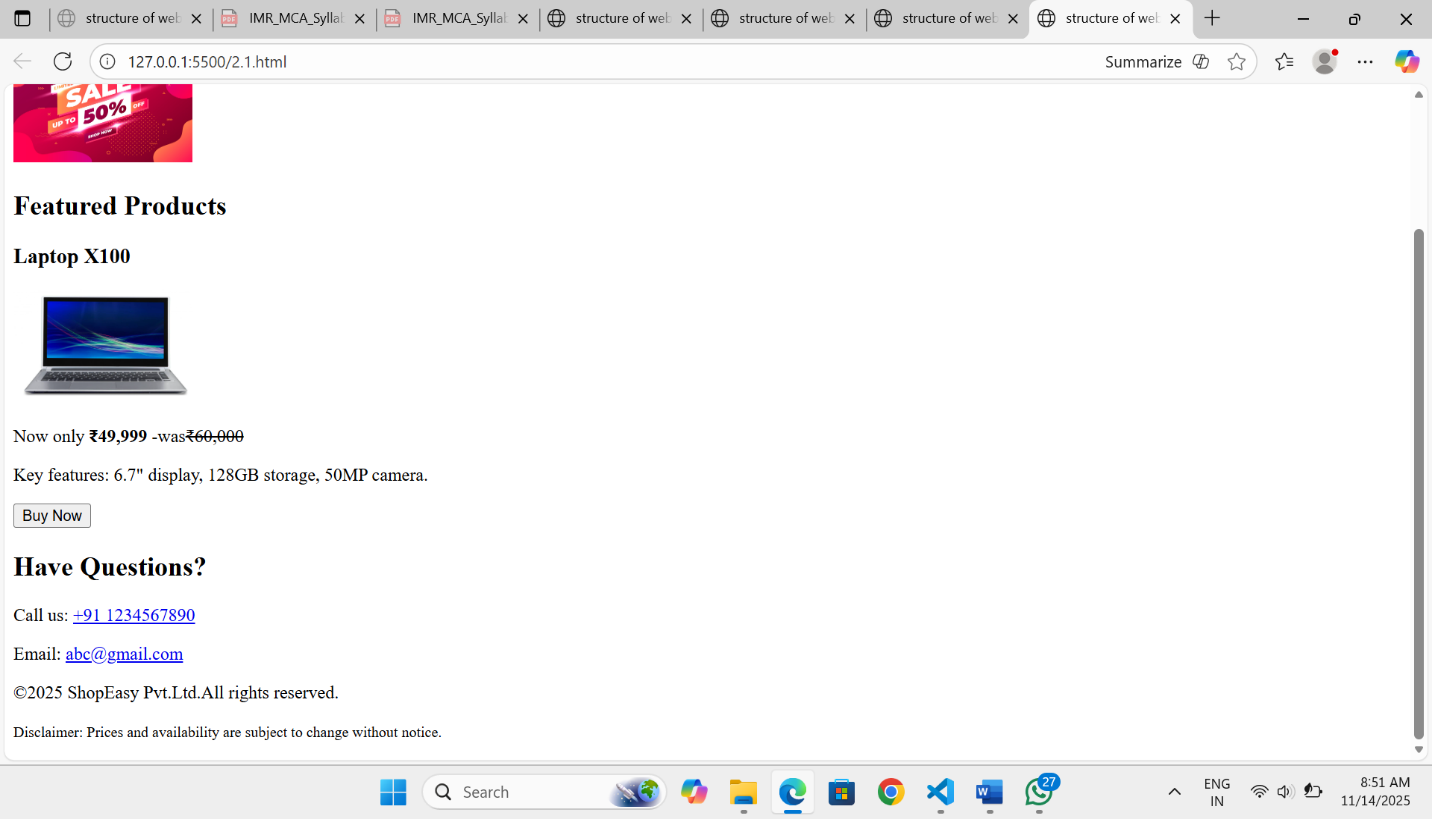
        <p><small>Disclaimer: Prices and availability are subject to change without notice.</small></p>

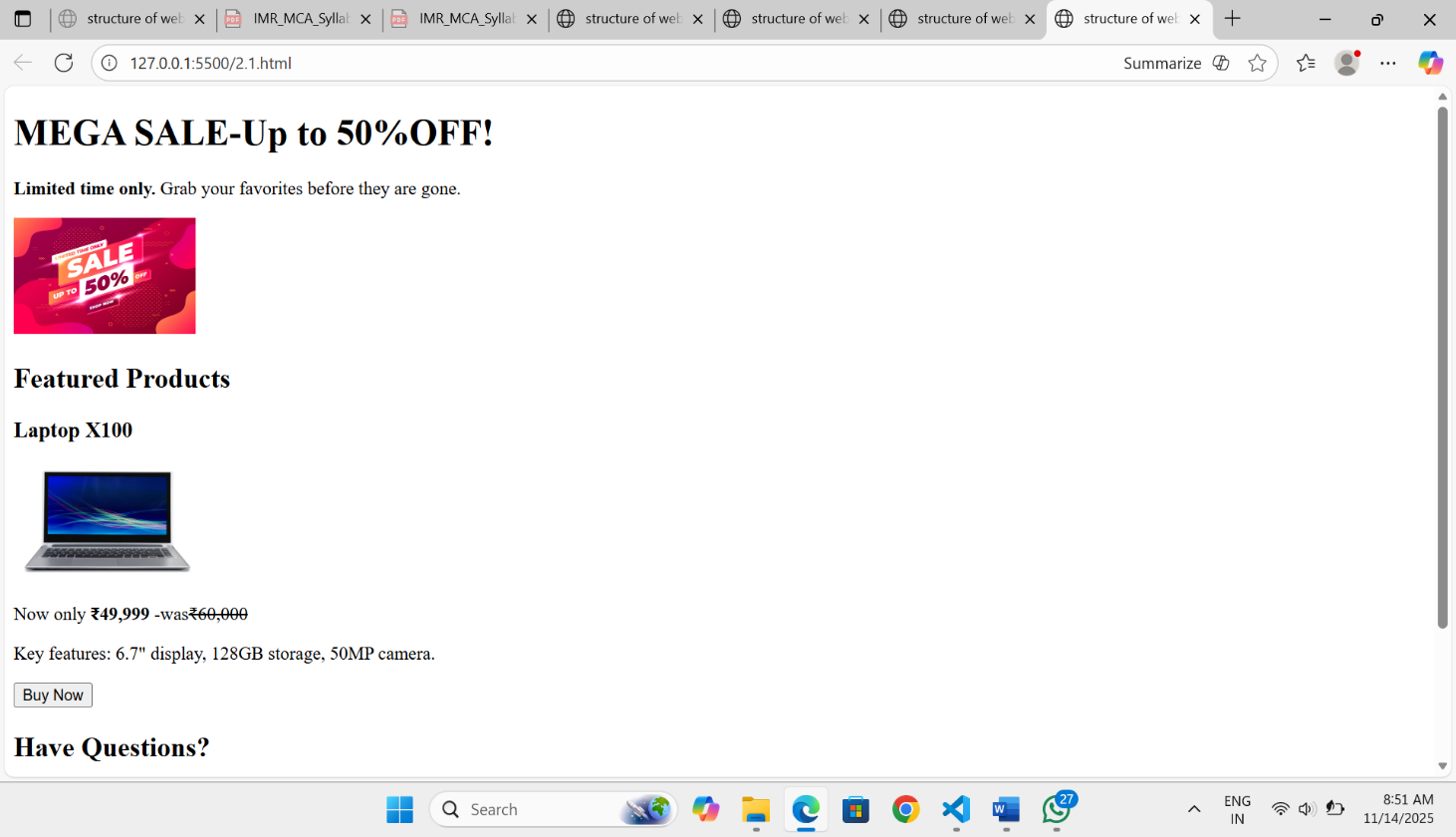
    </div>

</body>

</html>

**Output:**





**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 2.2**

**Title: 2. Use <span> tags to highlight or style inline elements (e.g., a bolded or colored word in a paragraph)**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title> SALE 2.2</title>

<link rel="stylesheet" href="style2.css">

</head>

<body>

<div id="header">

<h1>MEGA SALE \_ Up to 60% OFF!</h1>

<p><strong>Limited thime only.</strong> Grad your favorites before they're gone.</p>

</div>

<div id="content">

<div id="banner">

<img src="/SALE.jpeg" alt="sale Banner" height="50%" width="50%">

</div>

<div id="products">

<h2>Featured products</h2>

<div class="product">

<h3>Laptop x100</h3>

<img src="/lap.jpeg" alt="Laptop image" height="50%" width="50%">

<p>Now only <strong>₹49,999</strong> - Was <del>₹60,000</del></p>

<p><b>Key Features:<br> Display: 13.4",<br> Storage: 128Gb, <br> Camera: 50Mp.</b></p>

<form action="#" method="post">

<input type="hidden" name="product\_id" value="x100">

<input type="submit" value="Buy Now">

</form>

</div>

<p>

<b>A laptop is a portable personal computer that integrates a screen, keyboard, and touchpad

into a single, lightweight device.</b> <br> Designed for mobility, it allows users to perform various tasks

like <br> <span> browsing the internet, creating documents, <br> and running applications anywhere by

combining all necessary <br> components into a compact unit with a built-in battery. </span>

<br>Their portability and versatility have made them <br>essential for work, education, and

entertainment on the go, serving as a convenient and powerful tool for a wide range of users.

</p>

</div>

<div id="cta">

<h2>Have questions?</h2>

<p>Call us: <a href="tel: +911234567890">+91 1234567890</a></p>

<p>Email: <a href="mailt:support@example.com">support@example.com</a></p>

</div>

</div>

<!-- end content -->

<!-- Footer section -->

<div id="footer">

<p>© 2025 ShopEasy Pvt. Ltd. All rigth reserved.</p>

<p><small>Disclaimer: prices and availability ae subject to change without notice.</small></p>

</div>

</body>

</html>

Css:

\*{

}

background-color: #F0A6E8;

text-align: center;

form input {

background-color: #fff;

font-size: 20px;

}

p{

}

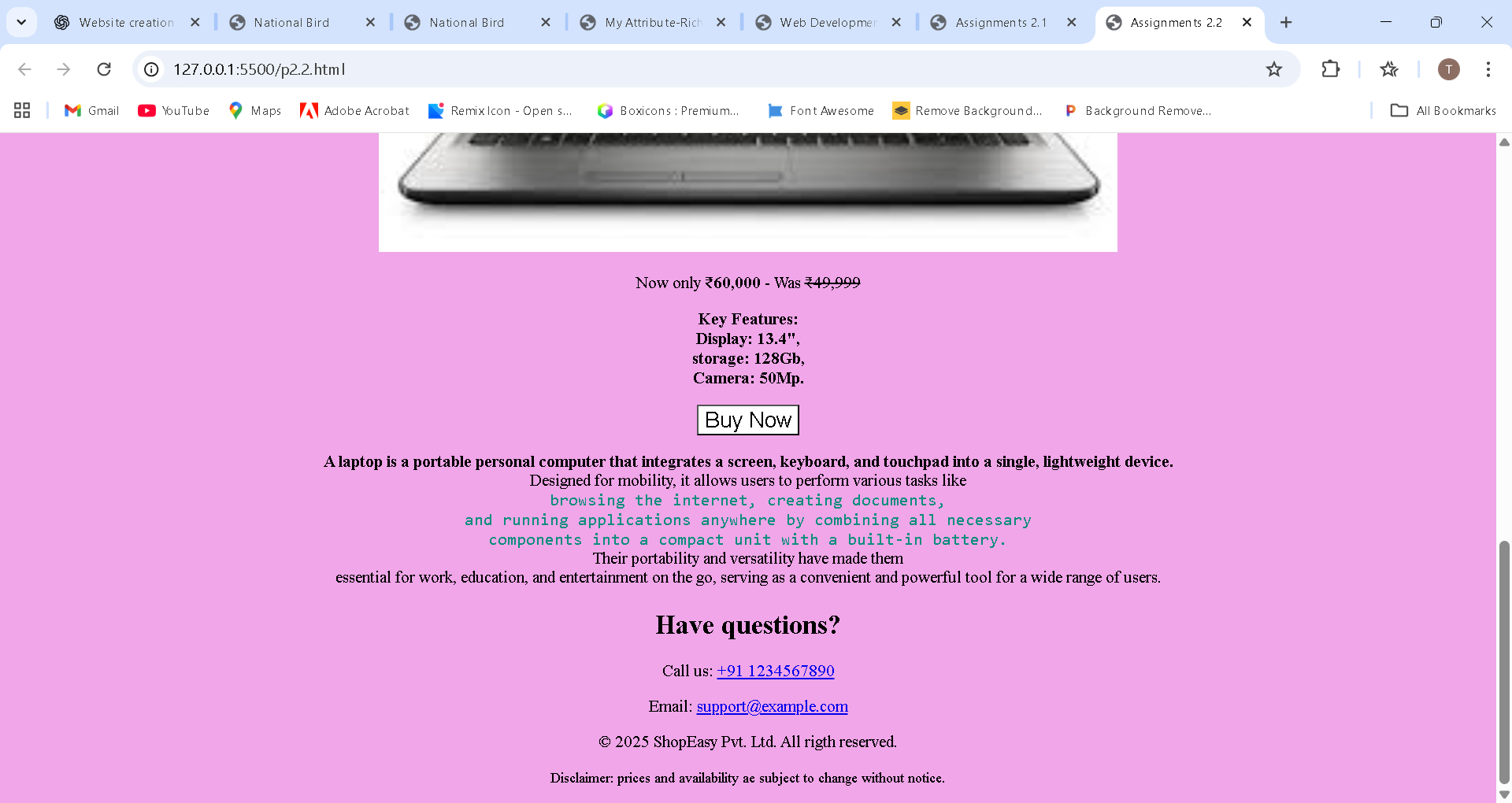
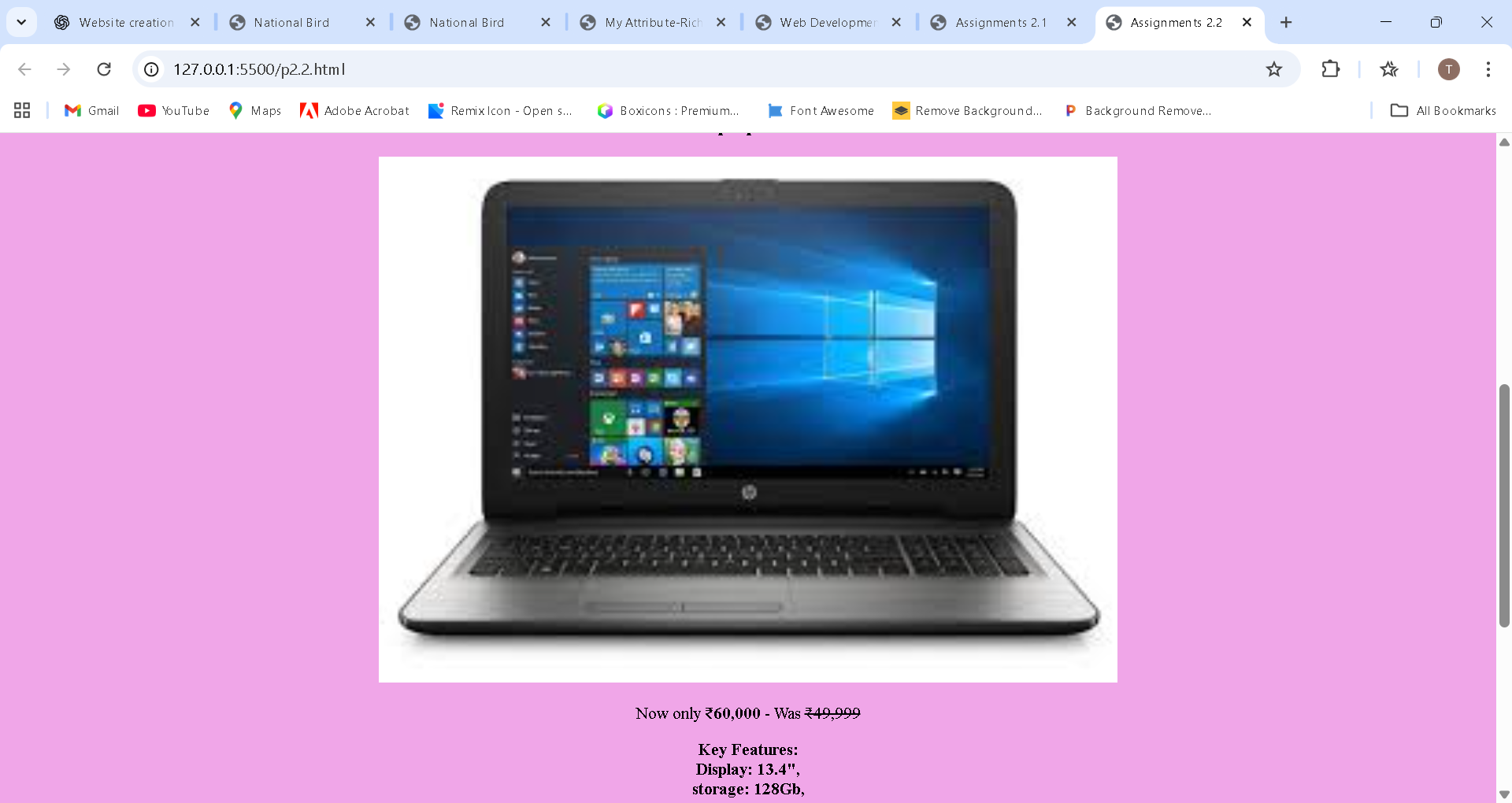
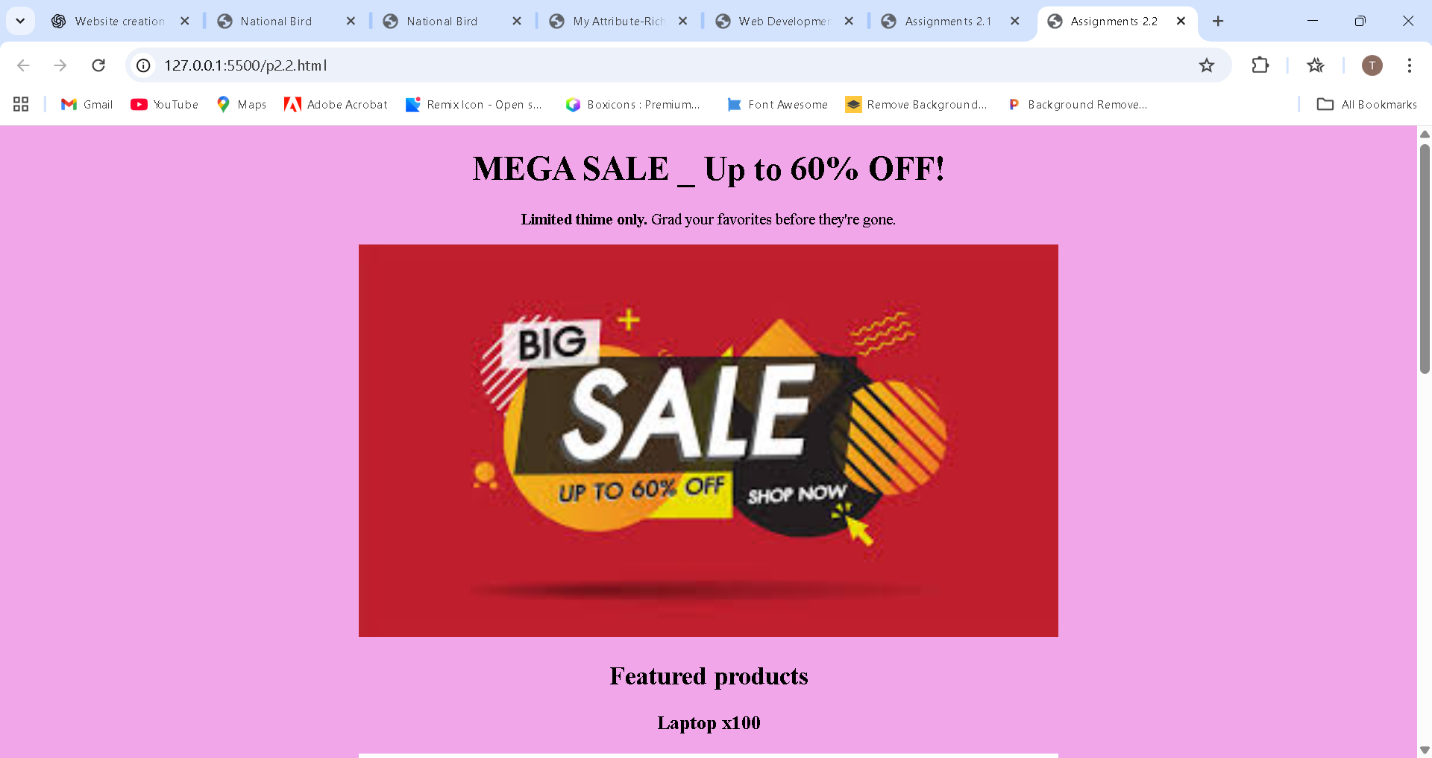
font-size: 15px;

samp{

color:#0c8e7b;

}

Output:



**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 3**

**Title: Create a webpage using Semantic HTML5 elements:**

**1. Design a webpage layout using the following semantic elements:**

**a. <header> for the page header, including a logo and a navigation menu.**

**b. <nav> for a navigation bar with links.**

**c. <section> for the main content sections.**

**d. <article> for individual pieces of content (e.g., a blog post or news item).**

**e. <aside> for sidebar content (e.g., advertisements or related links).**

**f. <footer> for the page footer with contact information and social media links.**

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>National Bird </title>

<link rel="stylesheet" href="style3.css">

</head>

<body>

<div class="wrap">

<!-- HEADER -->

<header>

<div class="brand">

<div class="logo" aria-hidden="true">

<img src="/p\_logo.jpg" alt="Peacock Logo" width="75" height="70">

</div>

<div class="brand-text">

<!-- <div class="title">Indian Peafowl</div> -->

<div class="subtitle">India’s National Bird • 2025</div>

</div>

</div>

<nav aria-label="Main navigation">

<ul>

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

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

<li><a href="#articles">Articles</a></li>

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

</ul>

</nav>

</header>

<!-- HERO SECTION -->

<main>

<section class="hero" id="home">

<div class="info">

<h1>Welcome to the Indian Peacock Information Portal</h1>

<p>

Discover India’s national bird — its vibrant plumage, cultural significance,

natural habitat, and conservation status for the year 2025.

</p>

<p class="learn-btn">

<a class="btn" href="#about">Learn more</a>

</p>

</div>

<div class="media" role="img" aria-label="Indian Peafowl">

<img src="/p.jpg" alt="Indian Peafowl" width="150" height="150">

</div>

</section>

<!-- CONTENT GRID -->

<div class="grid">

<!-- MAIN ARTICLE -->

<article>

<h2 id="about">About the Indian Peacock</h2>

<div class="meta">Origin • Habitat • Behaviour • Conservation</div>

<p class="excerpt">

The Indian Peacock (<em>Pavo cristatus</em>) is the national bird of India.

Known for its stunning iridescent feathers and majestic courtship dance,

the peacock holds deep cultural and ecological significance. It is commonly

found in forests, villages, and grasslands across the country.

</p>

<!-- Latest Articles Section -->

<section class="latest-articles">

<h3>Latest Articles</h3>

<div class="card-list">

<div class="post">

<strong>Peacock Conservation in 2025</strong>

<div class="meta">Oct 12, 2025 · 4 min read</div>

<div class="excerpt">

Understanding how habitat loss and climate change impact the peafowl,

and what communities are doing to protect it.

</div>

</div>

<div class="post">

<strong>The Peacock Courtship Dance</strong>

<div class="meta">Sep 18, 2025 · 3 min read</div>

<div class="excerpt">

A look at the science behind the stunning feather display and mate selection.

</div>

</div>

</div>

</section>

</article>

<!-- SIDEBAR -->

<aside aria-labelledby="related">

<h3 id="related">Quick Facts</h3>

<ul class="facts">

<li>Scientific Name: <strong>Pavo cristatus</strong></li>

<li>Habitat: Forests, grasslands, villages, farmlands</li>

<li>Conservation (2025): Least Concern, but requires habitat protection</li>

<li>Symbolism: Pride, beauty, grace, cultural heritage</li>

</ul>

<div class="connect">

<h4>Connect</h4>

<div class="socials">

<a href="#" aria-label="Twitter">Twitter</a>

<a href="#" aria-label="Instagram">Instagram</a>

<a href="#" aria-label="Facebook">Facebook</a>

</div>

</div>

<div class="contact-btn">

<a class="btn" href="#contact">Contact / Report Sighting</a>

</div>

</aside>

</div>

<!-- FEATURED ARTICLE -->

<section id="articles" class="featured-section">

<article>

<h2>Featured Article</h2>

<p class="meta">By Wildlife Research Team · Nov 6, 2025</p>

<p class="excerpt">

Explore how habitat restoration projects across India are helping rebuild

safe breeding grounds for peafowls, ensuring a protected future for the

national bird.

</p>

</article>

</section>

</main>

<!-- FOOTER -->

<footer>

<div>

© <strong>Indian Peacock</strong> — Built with ❤️ in 2025

</div>

<div>

<small id="contact">

Email: <a href="mailto:info@example.org">info@example.org</a> •

<a href="#">Privacy</a>

</small>

</div>

</footer>

</div>

</body>

</html>

Css:

:root {

--bg: #f7f8fb;

--card: #ffffff;

--accent: #2b6cb0;

--muted: #6b7280;

--max: 1100px;

--radius: 12px;

--glass: rgba(255, 255, 255, 0.6);

}

\* {

box-sizing: border-box;

}

html,

body {

height: 100%;

margin: 0;

font-family: Inter, system-ui, -apple-system,

'Segoe UI', Roboto, 'Helvetica Neue', Arial;

color: #0f172a;

background: linear-gradient(180deg, var(-

bg), #1c71d8);

}

a {

color: inherit;

text-decoration: none;

}

.wrap {

max-width: var(--max);

margin: 32px auto;

padding: 20px;

}

header {

display: flex;

align-items: center;

justify-content: space-between;

gap: 16px;

}

.brand {

display: flex;

align-items: center;

gap: 12px;

}

.logo {

width: 90px;

height: 90px;

border-radius: 10px;

background: linear-gradient(135deg, var(-

accent), #67b0ff);

display: flex;

align-items: center;

justify-content: center;

color: white;

font-weight: 700;

font-size: 20px;

}

nav ul {

display: flex;

gap: 14px;

list-style: none;

margin: 0;

padding: 0;

}

nav a {

padding: 8px 10px;

border-radius: 8px;

font-weight: 600;

color: var(--accent);

}

nav a:hover {

background: rgba(43, 108, 176, 0.08);

}

main {

margin-top: 20px;

}

.hero {

display: flex;

gap: 24px;

align-items: center;

background: var(--card);

padding: 20px;

border-radius: var(--radius);

box-shadow: 0 6px 18px rgba(13, 38, 63,

0.06);

}

.hero .info {

flex: 1;

}

.hero h1 {

margin: 0;

font-size: 28px;

}

.hero p {

margin: 8px 0 0;

color: var(--muted);

line-height: 1.5;

}

.hero .media {

width: 220px;

height: 160px;

border-radius: 12px;

background: linear-gradient(180deg, #e6f2ff,

#fff);

display: flex;

align-items: center;

justify-content: center;

color: var(--muted);

font-weight: 600;

}

.grid {

display: grid;

grid-template-columns: 1fr 320px;

gap: 20px;

margin-top: 20px;

}

article {

background: var(--card);

padding: 16px;

border-radius: 10px;

box-shadow: 0 6px 14px rgba(8, 20, 40, 0.04);

}

article>h2 {

margin-top: 0;

font-size: 18px;

}

.meta {

font-size: 13px;

color: var(--muted);

margin-bottom: 8px;

}

.excerpt {

color: var(--muted);

line-height: 1.5;

}

.card-list {

display: grid;

gap: 12px;

margin-top: 12px;

}

.post {

padding: 12px;

border-radius: 10px;

background: var(--glass);

border: 1px solid rgba(15, 23, 42, 0.03);

}

aside {

background: var(--card);

padding: 16px;

border-radius: 10px;

height: fit-content;

}

.socials {

display: flex;

gap: 8px;

margin-top: 12px;

}

.btn {

display: inline-block;

padding: 8px 12px;

border-radius: 8px;

background: var(--accent);

color: white;

font-weight: 600;

}

footer {

margin-top: 28px;

padding: 16px 0;

color: var(--muted);

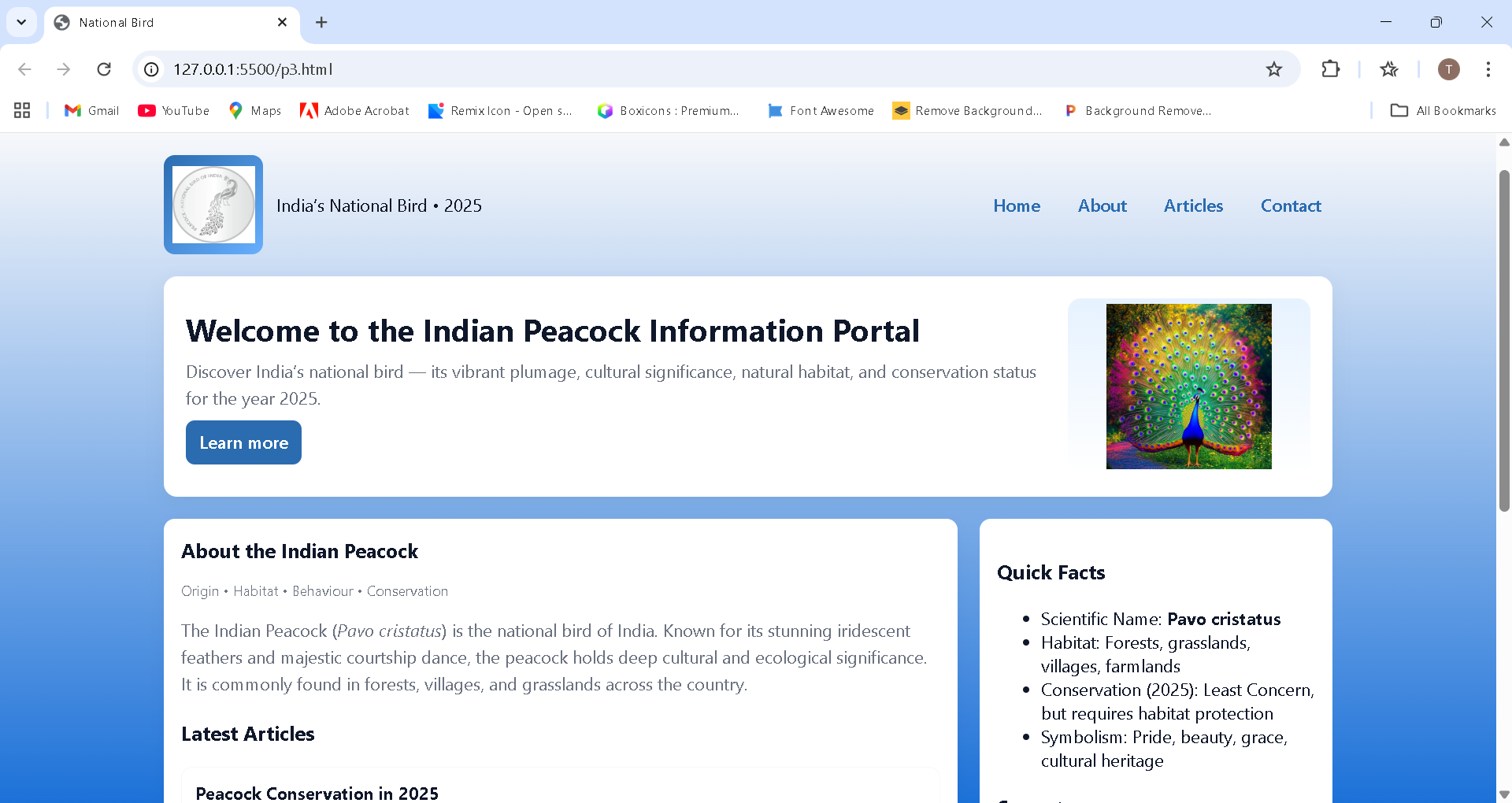
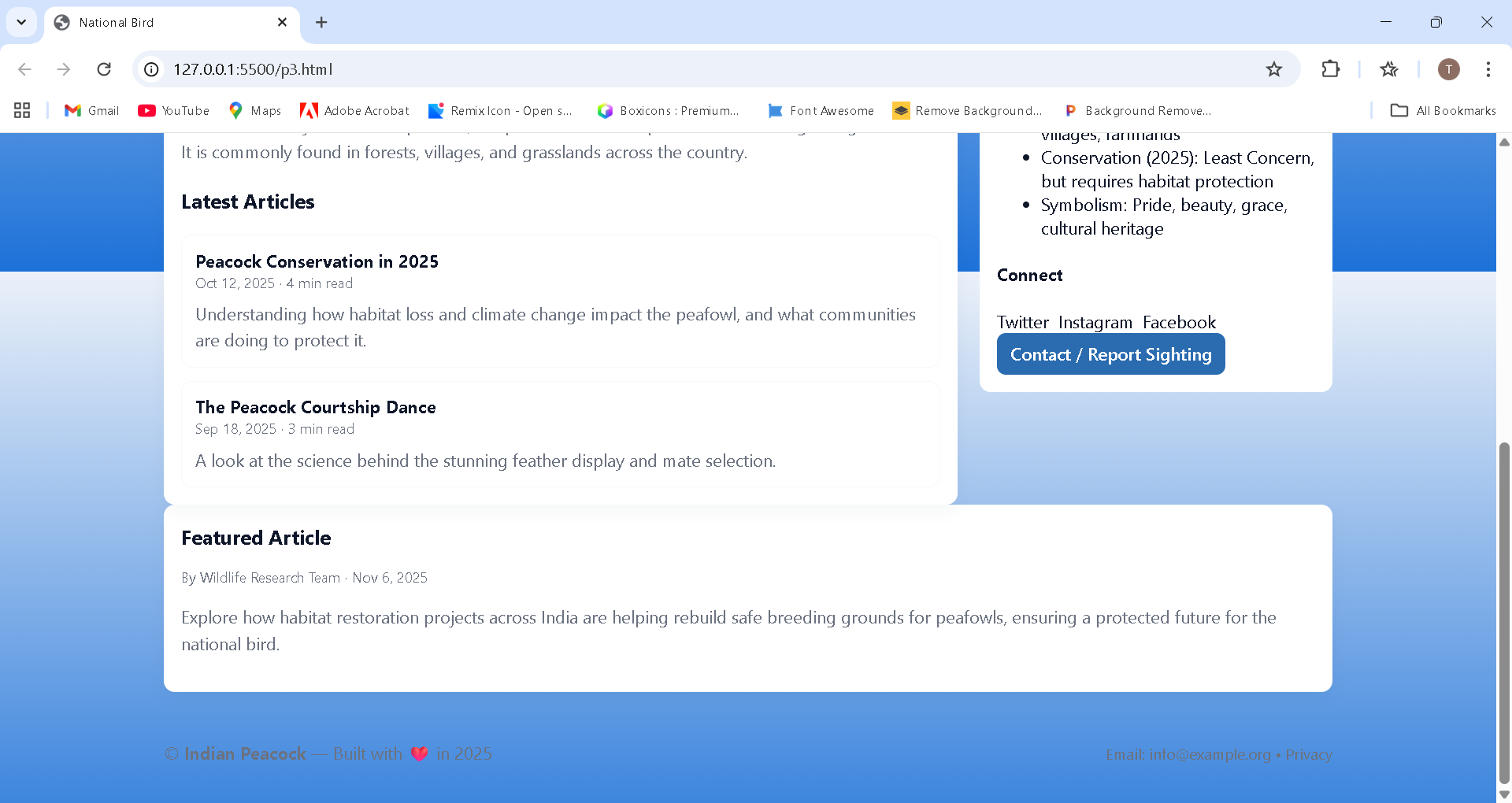
display: flex;

justify-content: space-between;

align-items: center;

}

**Output:**



**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 4.1**

**Title: Create a user registration form:**

**1. Build a form using the following elements:**

**a. <form> tag to define the form.**

**b. <input> elements for fields like name (text), email (email), and pass-**

**word (password).**

**c. <textarea> for a description or bio input.**

**d. <select> element with multiple <option> choices (e.g., country or role**

**selection).**

**e. <button> to submit the form.**

Html code

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>College Student Registration</title>

<link rel="stylesheet" href="style4.css">

</head>

<body>

<!-- Top Heading Bar -->

<div class="top-heading">

<h2>College Student Registration Form</h2>

</div>

<form id="studentForm">

<!-- Row 1 -->

<div class="row">

<div class="col">

<label>🎓 Full Name</label>

<input type="text" placeholder="Enter full name" required>

</div>

<div class="col">

<label>📧 Email Address</label>

<input type="email" placeholder="Enter email" required>

</div>

</div>

<!-- Row 2 -->

<div class="row">

<div class="col">

<label>📱 Mobile Number</label>

<input type="text" id="mobile" name="mobile" placeholder="Enter mobile number"

maxlength="10" required>

</div>

<div class="col">

<label>🆔 Student ID / Roll No</label>

<input type="text" placeholder="Enter student ID" required>

</div>

</div>

<!-- Row 3 -->

<div class="row">

<div class="col">

<label>🏫 Department</label>

<div class="multi-option">

<label><input type="checkbox" name="Department" value="MCA"> MCA</label>

<label><input type="checkbox" name="Department" value="BCA"> BCA</label>

<label><input type="checkbox" name="Department" value="BBA"> BBA</label>

<label><input type="checkbox" name="Department" value="MBA"> MBA</label>

<label><input type="checkbox" name="Department" value="IMCA"> IMCA</label>

</div>

</div>

<div class="col">

<label>📘 Course / Year</label>

<select required>

<option value="">Choose Year</option>

<option>1st Year</option>

<option>2nd Year</option>

<option>3rd Year</option>

<option>4th Year</option>

<option>5th Year</option>

</select>

</div>

</div>

<!-- Row 4 -->

<div class="row">

<div class="col password-box">

<label>🔑 Password</label>

<input type="password" id="password" placeholder="Enter password" minlength="8"

required>

</div>

<div class="col password-box">

<label>🔐 Confirm Password</label>

<input type="password" id="confirm\_password" placeholder="Re-enter password"

minlength="8" required>

</div>

</div>

<!-- Row 5 -->

<div class="row">

<div class="col">

<label>🎂 Birthdate</label>

<input type="date" id="birthdate" required>

</div>

<div class="col">

<label>⚧ Gender</label>

<div class="multi-option">

<label><input type="radio" name="Gender" value="male"> Male</label>

<label><input type="radio" name="Gender" value="female"> Female</label>

<label><input type="radio" name="Gender" value="other"> Other</label>

</div>

</div>

</div>

<!-- Bio -->

<label>📝 Short Bio</label>

<textarea rows="4" placeholder="Tell us about yourself..."></textarea>

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

</form>

</body>

</html>

Css code

/\* ---------- Global Styling ---------- \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: "Inter", sans-serif;

}

body {

background: linear-gradient(135deg, #6db3f2, #1e69de);

min-height: 100vh;

padding: 20px;

display: flex;

flex-direction: column;

align-items: center;

}

/\* ---------- Top Heading Bar ---------- \*/

.top-heading {

width: 100%;

text-align: center;

background: white;

padding: 12px 0;

margin-bottom: 20px;

box-shadow: 0 4px 10px rgba(0, 0, 0, 0.2);

}

.top-heading h2 {

font-size: 22px;

font-weight: 700;

color: #1e69de;

}

/\* ---------- Form Card ---------- \*/

form {

background: rgba(255, 255, 255, 0.90);

backdrop-filter: blur(18px);

padding: 30px 35px;

border-radius: 14px;

width: 100%;

max-width: 650px;

box-shadow: 0px 10px 40px rgba(0, 0, 0, 0.15);

}

/\* ---------- Labels ---------- \*/

label {

font-size: 15px;

font-weight: 600;

margin-bottom: 6px;

display: block;

color: #1c3c78;

}

/\* ---------- Inputs ---------- \*/

input, textarea, select {

width: 100%;

padding: 12px;

margin-bottom: 18px;

font-size: 14px;

border: 1px solid #d0d0d0;

border-radius: 8px;

background: #f8faff;

transition: all 0.25s ease;

}

input:focus, textarea:focus, select:focus {

border-color: #1e69de;

box-shadow: 0 0 8px rgba(30, 105, 222, 0.3);

}

/\* ---------- Side-by-side layout ---------- \*/

.row {

display: flex;

gap: 20px;

margin-bottom: 18px;

}

.col {

flex: 1;

display: flex;

flex-direction: column;

}

/\* ---------- Multi Option (Checkbox / Radio) ---------- \*/

.multi-option {

display: flex;

flex-wrap: wrap;

gap: 10px 20px;

margin-top: 8px;

}

.multi-option label {

display: flex;

gap: 6px;

align-items: center;

background: #eef3ff;

padding: 6px 12px;

border-radius: 8px;

border: 1px solid #c9d7ff;

cursor: pointer;

transition: 0.2s ease;

}

.multi-option label:hover {

background: #dce7ff;

border-color: #8bb0ff;

}

.multi-option input {

width: auto;

margin: 0;

}

/\* ---------- Button ---------- \*/

button {

width: 100%;

padding: 14px;

font-size: 16px;

border: none;

border-radius: 8px;

background: #1e69de;

color: white;

cursor: pointer;

font-weight: 600;

transition: 0.3s ease;

}

button:hover {

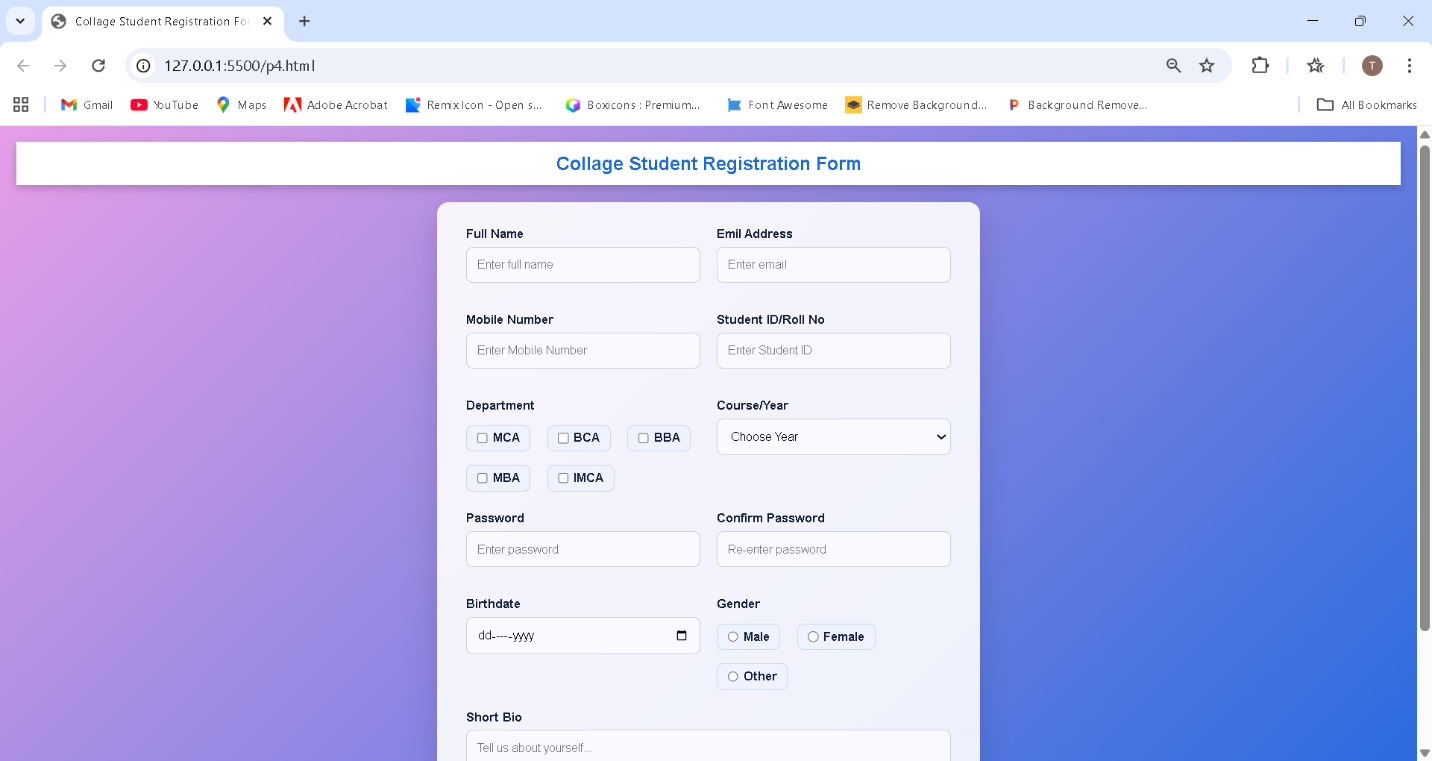
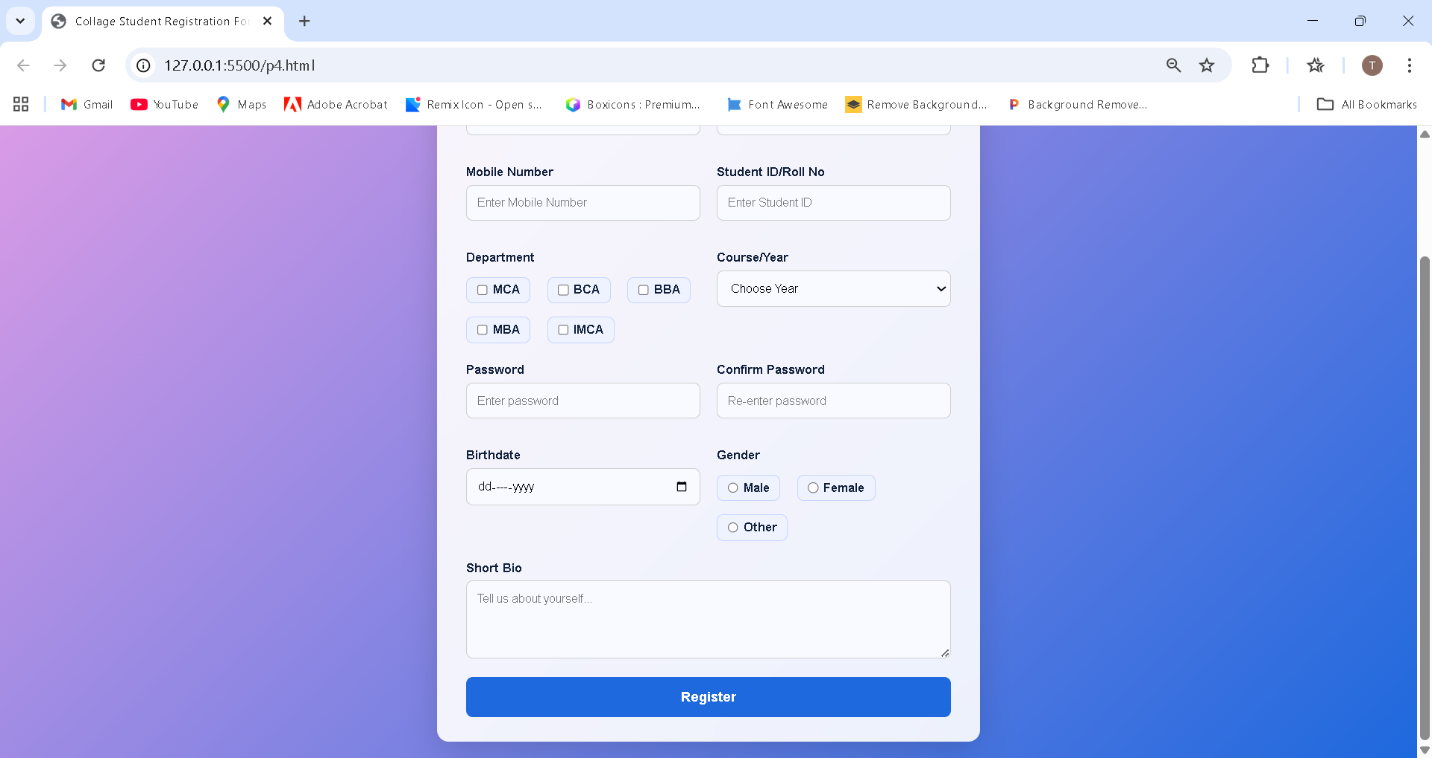
background: #1552b6;

transform: translateY(-2px);

box-shadow: 0 8px 18px rgba(0, 0, 0, 0.2);

}

output



**Roll No.: 76**

**Name :- Raj Sanjay Koli**

**Assignment No: 7**

**Title: Style a webpage with various CSS properties:**

**1. Set the text color of headings (<h1>, <h2>, etc.) using the color property.**

**2. Apply a background color or image to the body or a specific section using the background property.**

**3. Change the font family, size, and weight of paragraphs (<p>) using the font-family, font-size, and font weight properties.**

**4. Align text in different sections using the text-align property (e.g., center, left, right)**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>My Portfolio</title>

    <link rel="stylesheet" href="style5.css">

</head>

<body>

*<!-- Header Section -->*

    <header class="hero">

        <h1>Welcome to My Portfolio</h1>

        <h2>Web Developer</h2>

    </header>

*<!-- About Section -->*

    <section class="about">

        <h2>About Me</h2>

        <p>

            Hello! I am a <b>web developer </b>with a strong interest in designing beautiful

            and user-friendly websites. <br> I enjoy learning new technologies and applying them in

            creative a mini projects.

        </p>

    </section>

*<!-- Projects Section -->*

    <section class="projects">

        <h2>Live code Editor</h2>

        <img src="/p5 img.jpg" alt="project image">

        <p>

            This is my first <b> Mini </b> project.<br>

            I have used HTML for structure, CSS for styling, and JavaScript for interactivity.

        </p>

    </section>

*<!-- Contact Section -->*

    <footer class="contact">

        <h3>Contact Me</h3>

        <p>Email: 📧 <a href="t@gmail.com">t@gmail.com</a> <br>

            Phone: 📞 <a href="+91 9960995820 ">+91 9960995820 </a></p>

    </footer>

</body>

</html>

**CSS code**

h1, h2 {

    color: #9e71f1;

    text-align: center;

}

body {

    background-color: #f2f2f2;

}

*/\* Header section \*/*

.hero {

    background-color: #aed6f1;

    padding: 30px;

    text-align: center;

}

*/\* About section \*/*

.about {

    background-color: #ffffff;

    padding: 20px;

    text-align: center;

}

*/\* Projects section \*/*

.projects {

    background-color: #d5f5e3;

    padding: 20px;

    text-align: center;

}

.projects img{

    width: 250px;

    height: 150px;

}

*/\* Footer section \*/*

.contact {

    background-color: #f9e79f;

    padding: 20px;

    text-align: center;

}

*/\* 3. Paragraph font styling \*/*

p {

    font-family: "Arial", sans-serif;

    font-size: 18px;

    font-weight: 400;

    color: #333;

}

p a{

    text-decoration: none;

    font-family: "Arial", sans-serif;

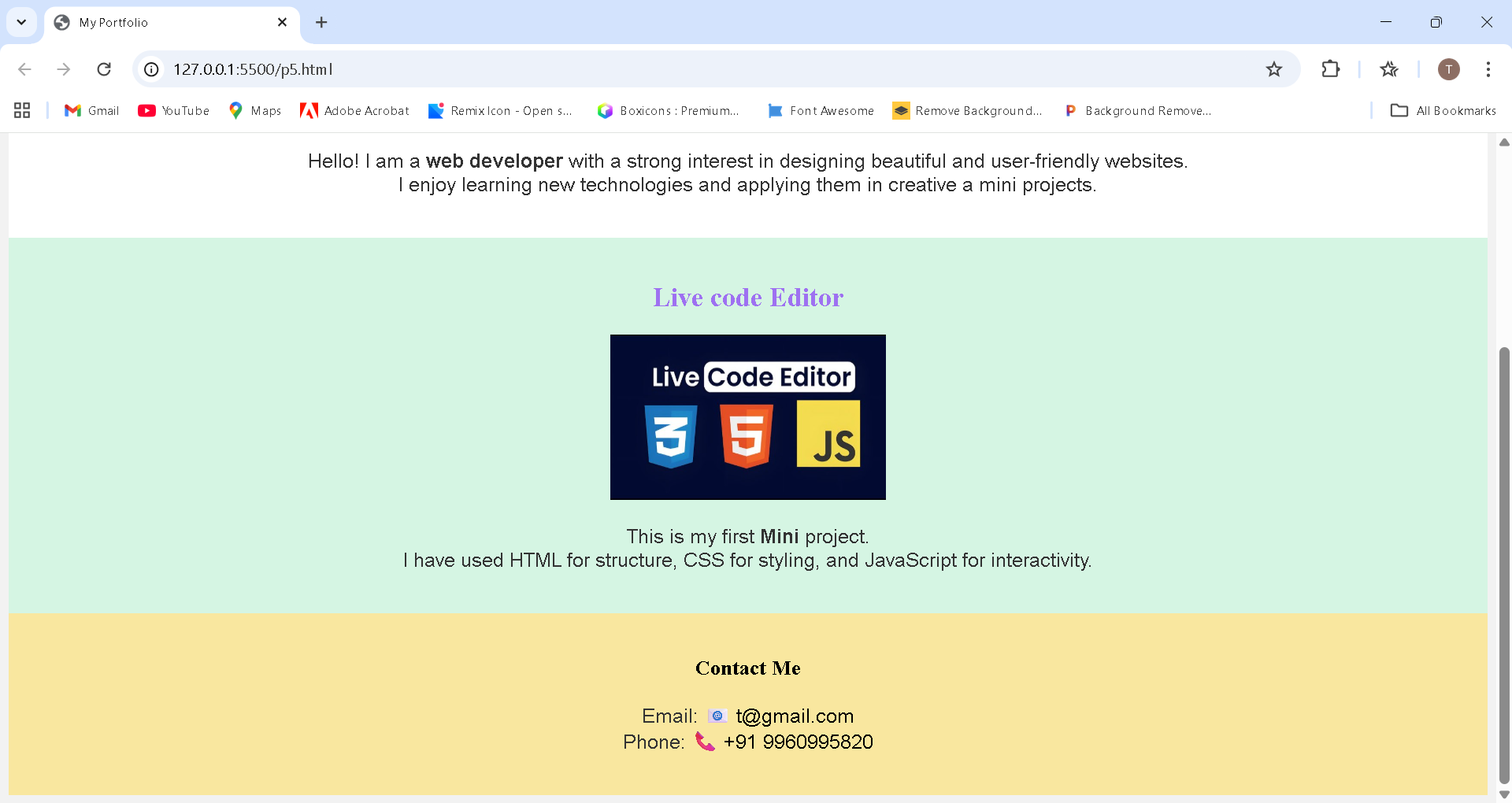
    font-size: 18px;

    color: #000;

}

**OUTPUT**

**Roll No.: 76**



**Name : Raj Sajay Koli**

**Assignment no :-06 Title: Create a webpage to demonstrate typography styling:**

**1. Use different font families (e.g., serif, sans-serif, monospace) on headings and paragraphs.**

**2. Set different font sizes for headings (<h1>, <h2>) and body text using font-size.**

**3. Apply different font weights (e.g., bold, normal, lighter) to various text elements.**

**4. Use the font-style property to italicize or emphasize specific text (e.g., using italic**

**for quotes or emphasis).**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Typography Styling Demo</title>

    <link rel="stylesheet" href="style6.css">

</head>

<body>

    <header class="page-header">

        <h1 class="serif-heading">Typography Styling in Web Design</h1>

       <h2 class="sans-heading">Understanding Fonts, Weights, and Styles</h2>

    </header>

    <section class="content">

        <p class="mono-text">

            Typography plays a crucial role in web design because it directly affects how users read

            and interpret information. In this paragraph, a <strong>monospace</strong> font-family is

            applied. Monospace fonts are commonly used in coding environments due to their equal

            letter spacing, which improves clarity when displaying technical content.

            In responsive design, typography must scale properly across different screen sizes to

            ensure readability, especially in code blocks viewed on mobile devices.

        </p>

        <p class="serif-text">

            This paragraph uses a <strong>serif</strong> font-family. Serif fonts include small decorative

            strokes that help guide the reader's eye, making them ideal for long passages of text. Good typography enhances visual hierarchy, improves user experience, and strengthens brand

            identity. In responsive web design, serif fonts are often used on larger screens like tablets

            and desktops because they maintain elegance and readability.

        </p>

        <p class="sans-text">

            This paragraph uses a <strong>sans-serif</strong> font-family. Sans-serif fonts are modern,

            clean, and highly readable on digital screens, making them the most widely used fonts

            in web interfaces. Effective typography ensures that font sizes, line spacing, and alignment are optimized for

            readability. In responsive design, sans-serif fonts maintain clarity even on smaller screens,

            supporting consistent user experience across devices.

        </p>

        <blockquote class="quote">

            “Good typography and responsive design work together to create readable, accessible,

            and visually appealing webpages.”

        </blockquote>

    </section>

    <footer>

        <p class="footer-text">Designed for Typography Demonstration — 2025</p>

    </footer>

</body>

</html>

**CSS**

body {

    margin: 0;

    padding: 0;

    background: #f4f4f4;

    font-family: Arial, sans-serif;

    line-height: 1.6;

    letter-spacing: 0.3px;

    word-spacing: 2px;

    direction: ltr;

}

.page-header {

    background: #da76f1;

    color: white;

    text-align: center;

    padding: 30px 20px;

    text-transform: uppercase;

}

.serif-heading {

    font-family: 'Times New Roman', serif;

    font-stretch: expanded;

    text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.3);

}

.sans-heading {

    font-family: Arial, sans-serif;

    font-variant: small-caps;

}

p {

    font-size: 18px;

    margin-bottom: 20px;

    text-indent: 25px;

    overflow-wrap: break-word;

    word-break: break-word;

}

.mono-text {

    font-family: "Courier New", monospace;

    font-style: normal;

}

.serif-text {

    font-family: Georgia, serif;

    text-align: justify;

}

.sans-text {

    font-family: Helvetica, sans-serif;

    text-align: left;

}

.serif-text strong {

    font-weight: bold;

}

.sans-text strong {

    font-weight: normal;

}

.mono-text strong {

    font-weight: 600;

}

.quote {

    font-style: italic;

    font-size: 20px;

    margin: 20px 0;

    padding: 15px;

    background: #ffffff;

    border-left: 5px solid #2c3e50;

    box-shadow: 0 0 5px rgba(0, 0, 0, 0.1);

    letter-spacing: 1px;

    white-space: normal;

}

.content {

    padding: 20px;

    margin: 20px;

    background: white;

    border-radius: 8px;

    box-shadow: 0 2px 6px rgba(0, 0, 0, 0.1);

}

h1 {

    font-size: 42px;

    font-weight: bold;

}

h2 {

    font-size: 26px;

    font-weight: 600;

}

footer {

    background: #6aa75e;

    padding: 15px;

    text-align: center;

}

.footer-text {

    color: #ddd;

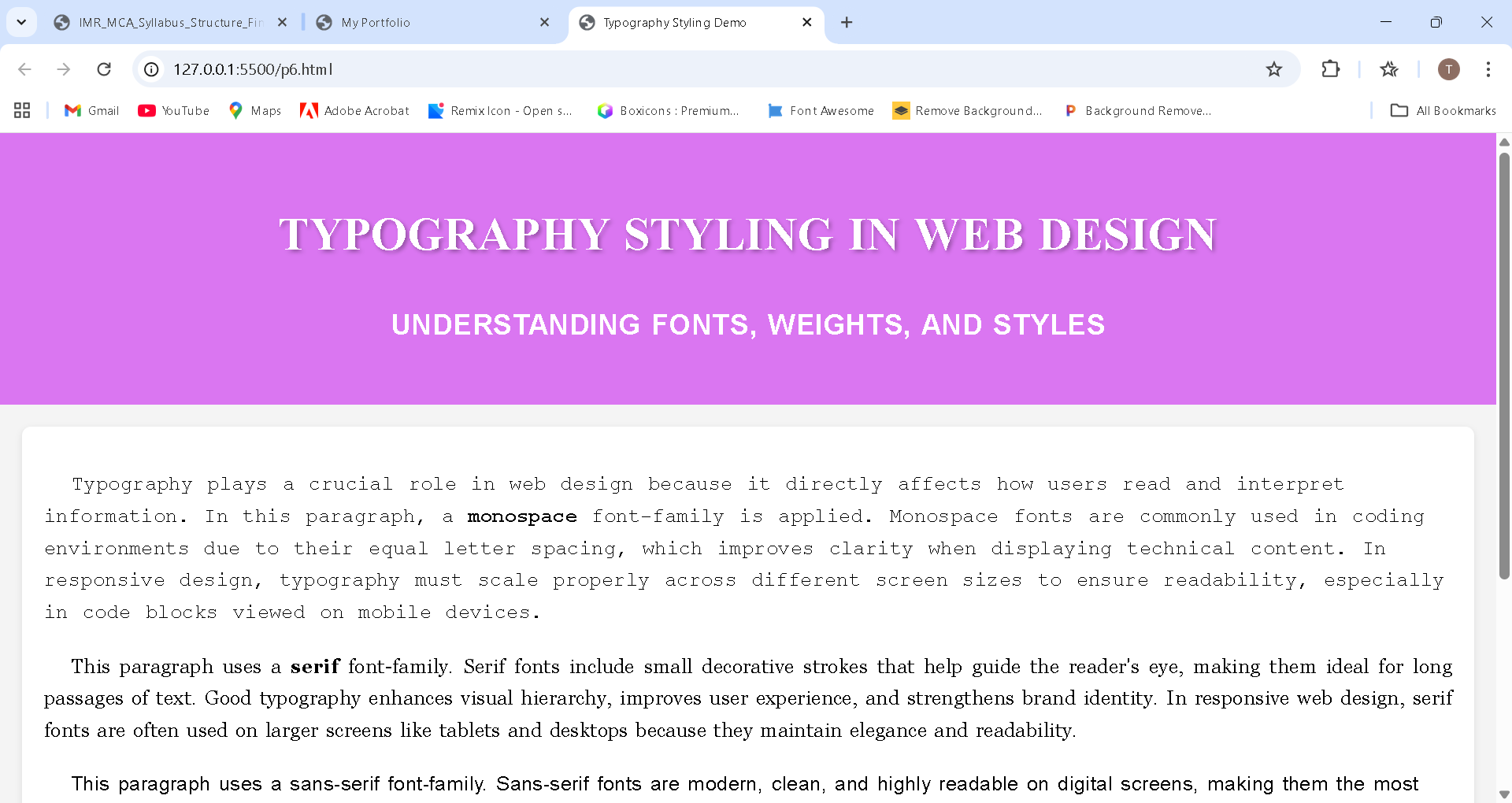
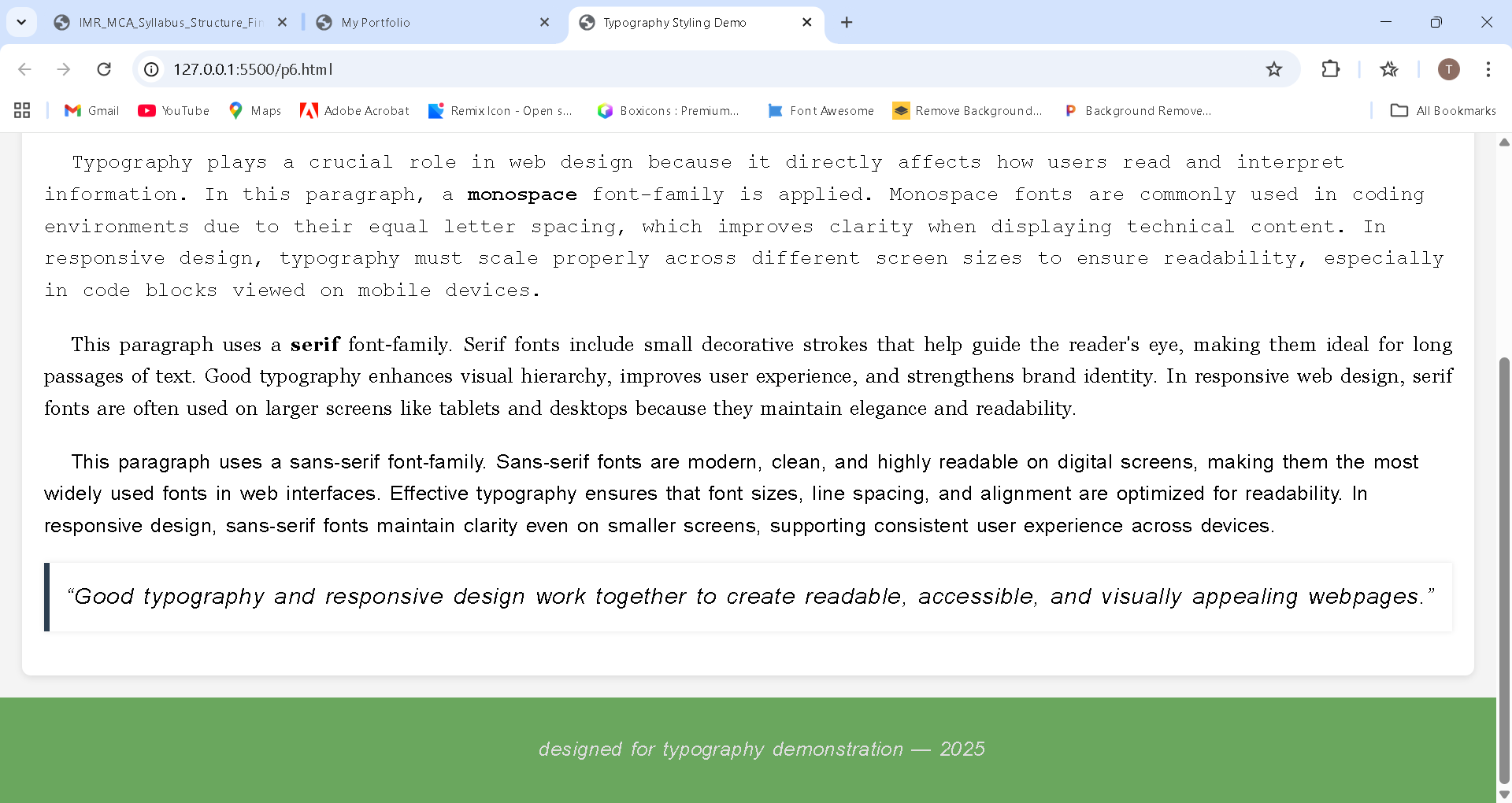
    font-style: italic;

    font-weight: lighter;

    text-transform: lowercase;

}

**OUTPUT**



**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 7**

**Title: Experiment with absolute and relative CSS units:**

**1. Create a webpage with sections that use px, em, rem, %, vh, and vw units to define**

**font sizes, widths, and heights.**

**2. Compare how absolute units (px) and relative units (em, rem) behave when**

**resizing the browser window or changing the font size of the parent element.**

**3. Set the width of containers using percentage values and adjust the viewport**

**height (vh) and viewport width (vw) to control the size of different sections.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Online Art Exhibition Portal</title>

    <link rel="stylesheet" href="style7.css">

</head>

<body>

    <header>

        <h1>Virtual Art & Creativity Exhibition 2025</h1>

    </header>

    <section class="px-section">

        <h2>About the Exhibition (px units)</h2>

        <p>

            Welcome to the <strong>Online Art & Creativity Exhibition 2025</strong>,

            a digital platform celebrating artistic evolution through time.

            This exhibition presents artworks that narrate the journey of human

            creativity — from ancient inspirations to contemporary expressions.

            Designed using <strong>px</strong> units, this section maintains fixed

            sizing to symbolize the permanence of history across eras.

        </p>

    </section>

    <section class="em-section">

        <h2>Theme of the Year: <em>“Echoes of Time”</em> (em units)</h2>

        <p>

            Using <strong>em</strong> units, this section scales with the parent element,

            reflecting how memories and ideas grow across generations.

            This year’s theme, <strong>“Echoes of Time”</strong>, explores the stories

            carried from the past, the heartbeat of the present, and the possibilities

            waiting in the future. Each artwork captures the essence of time —

            its rhythm, silence, transformation, and its powerful influence on human life.

        </p>

    </section>

    <section class="rem-section">

        <h2>Featured Artists (rem units)</h2>

        <p>

            Our exhibition brings together artists whose work beautifully resonates

            with the theme of time and transformation:

            <br><br>

            <strong>• Aaliya Verma</strong> – Blending ancient motifs with modern watercolor styles.<br>

            <strong>• Carlos Mendes</strong> – Known for surreal landscapes depicting timelines and transitions.<br>

            <strong>• Hana Saito</strong> – Minimalist artist capturing fleeting moments and eternal emotions.<br>

            <br>

            This section uses <strong>rem</strong> units, representing consistency across

            time — just like foundational human stories that remain unchanged.

        </p>

    </section>

    <section class="gallery">

        <div class="art-card">

            <img src="/p7 img1.jpg" alt="Abstract Art">

            <h3>Abstract Time Passage / Timelines</h3>

            <p>Abstract curves, timeline feel, futuristic + historical blend</p>

        </div>

        <div class="art-card">

            <img src="/p7 img2.jpg" alt="Modern Sculpture">

            <h3>Ancient Stone Texture / History Echo</h3>

            <p>Old stone wall texture – represents past + timelessness</p>

        </div>

        <div class="art-card">

            <img src="/p7 img3.jpg" alt="Classic Painting">

            <h3>Sand & Hourglass (Symbol of Time)</h3>

            <p>Beautiful hourglass showing passing of time</p>

        </div>

        <div class="art-card">

            <img src="/p7 img4.jpg" alt="Classic Painting">

            <h3>Minimalistic Moment / Stillness in Time</h3>

            <p>Calm, minimal, perfect for “moment suspended in time</p>

        </div>

    </section>

    <footer>

        &copy; 2025 Virtual Art Exhibition | Echoes of Time • Past • Present • Future

    </footer>

</body>

</html>

**CSS**

html {

font-size: 16px;

}

body {

margin: 0;

font-family: Arial, sans-serif;

background: linear-gradient(#f7f7f7, #ececec);

}

/\* Header \*/

header {

height: 40vh;

width: 100vw;

background: linear-gradient(rgba(0,0,0,0.5), rgba(0,0,0,0.5)), url('https://images.unsplash.com/photo-14100032628192-86f99bcd76bc') center/cover no-repeat;

display: flex;

justify-content: center;

align-items: center;

color: white;

}

header h1 {

font-size: 4vw;

background: rgba(0,0,0,0.6);

padding: 1rem 2rem;

border-radius: 10px;

}

/\* Sections \*/

.px-section,

.em-section,

.rem-section {

width: min(90%, 900px);

margin: 20px auto;

padding: 20px;

border-radius: 8px;

}

.px-section {

border-left: 5px solid #333;

background: #fff;

}

.em-section {

background: #e6ffe6;

font-size: 1.2em;

}

.rem-section {

background: #ffe6e6;

}

/\* Gallery \*/

.gallery {

width: min(92%, 1100px);

margin: auto;

display: grid;

grid-template-columns: repeat(auto-fit, minmax(260px, 1fr));

gap: 20px;

}

/\* Cards \*/

.art-card {

background: #fff;

padding: 15px;

border-radius: 12px;

text-align: center;

box-shadow: 0 0 12px rgba(0,0,0,0.15);

transition: 0.3s;

}

.art-card:hover {

transform: translateY(-8px);

box-shadow: 0 6px 18px rgba(0,0,0,0.25);

}

.art-card img {

width: 100%;

height: 25vh;

object-fit: cover;

border-radius: 10px;

}

.art-card h3 {

font-size: 1.3rem;

margin: 0.7rem 0;

}

/\* Footer \*/

footer {

background: #222;

color: white;

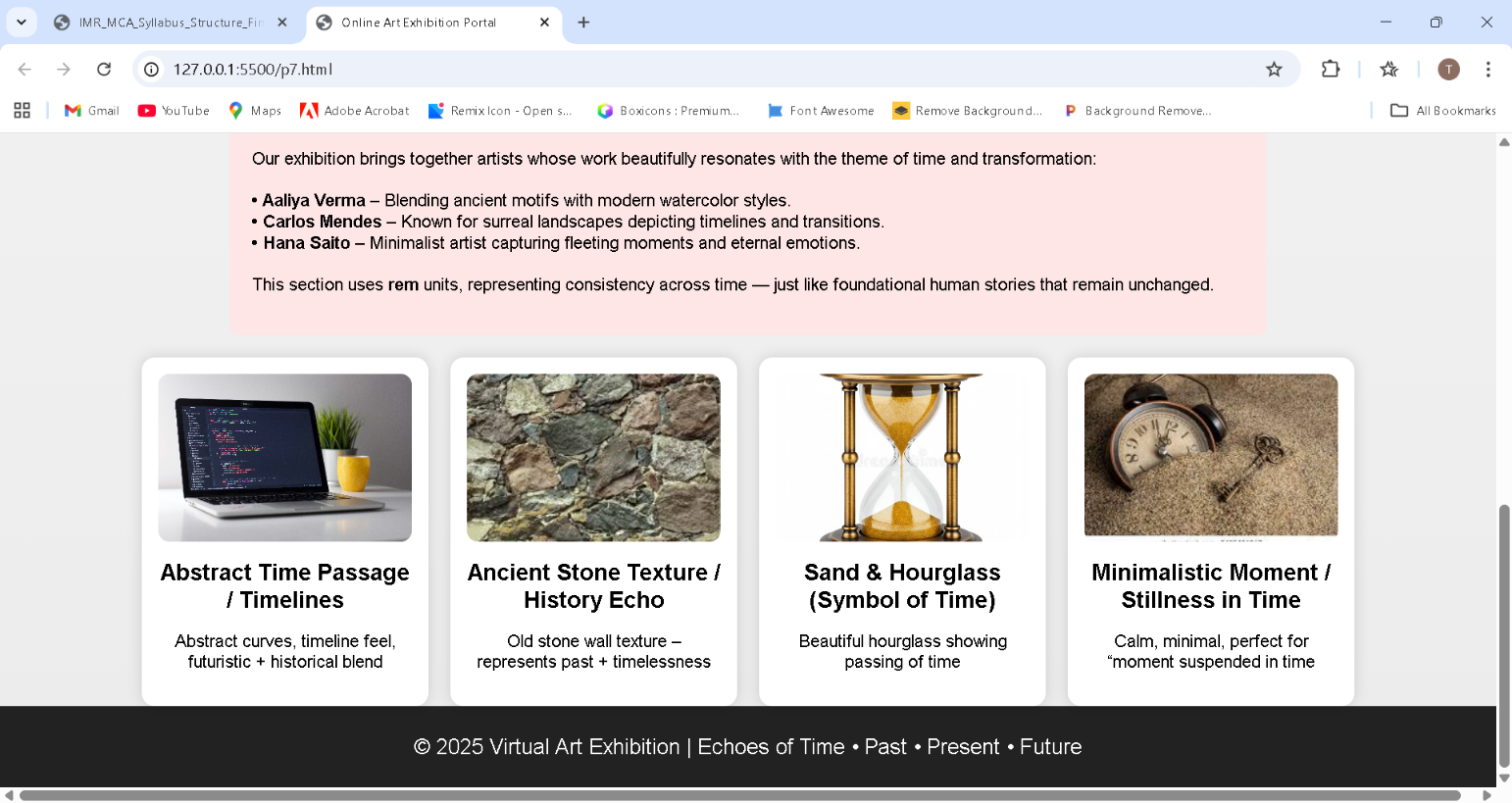
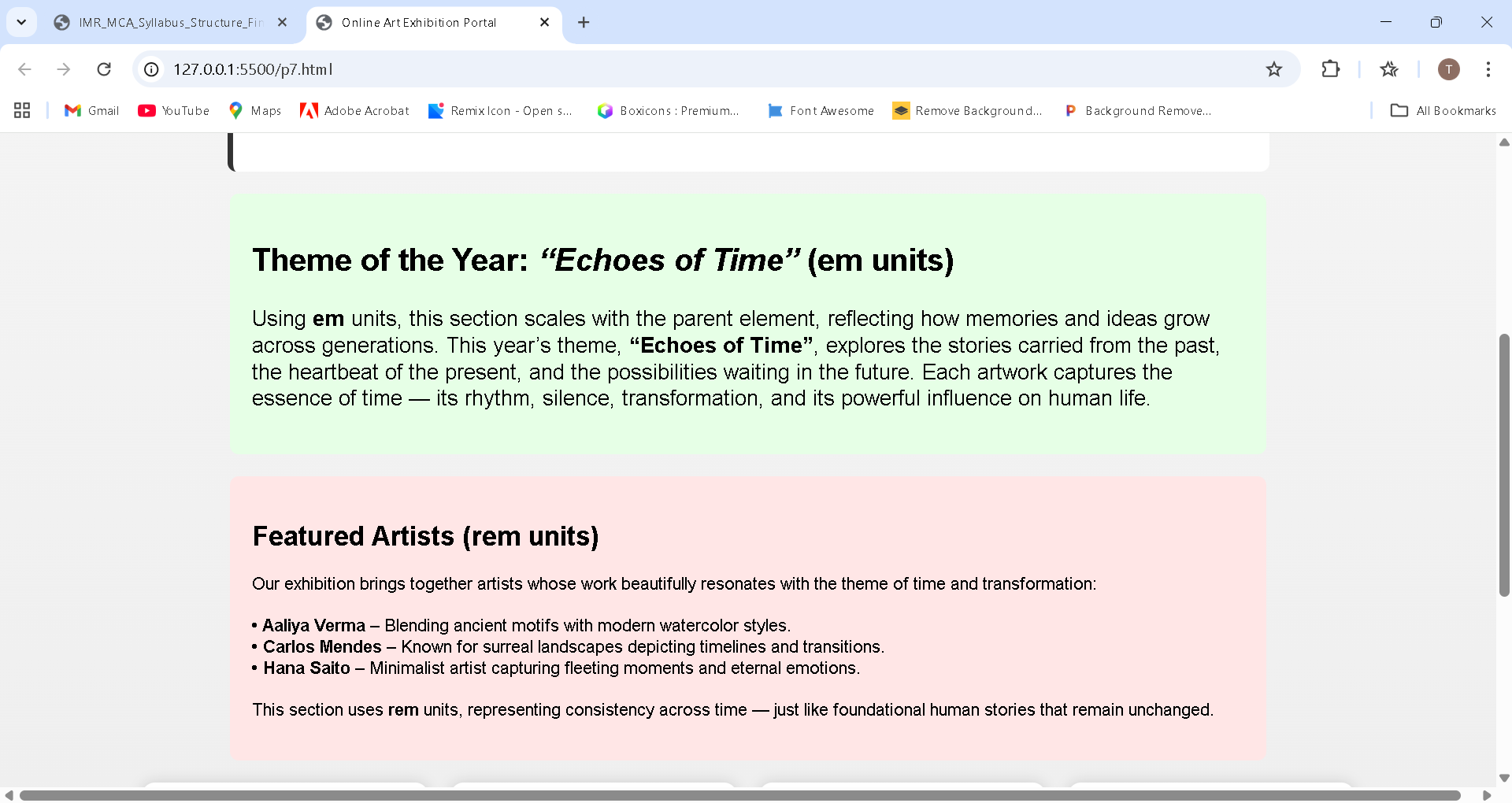
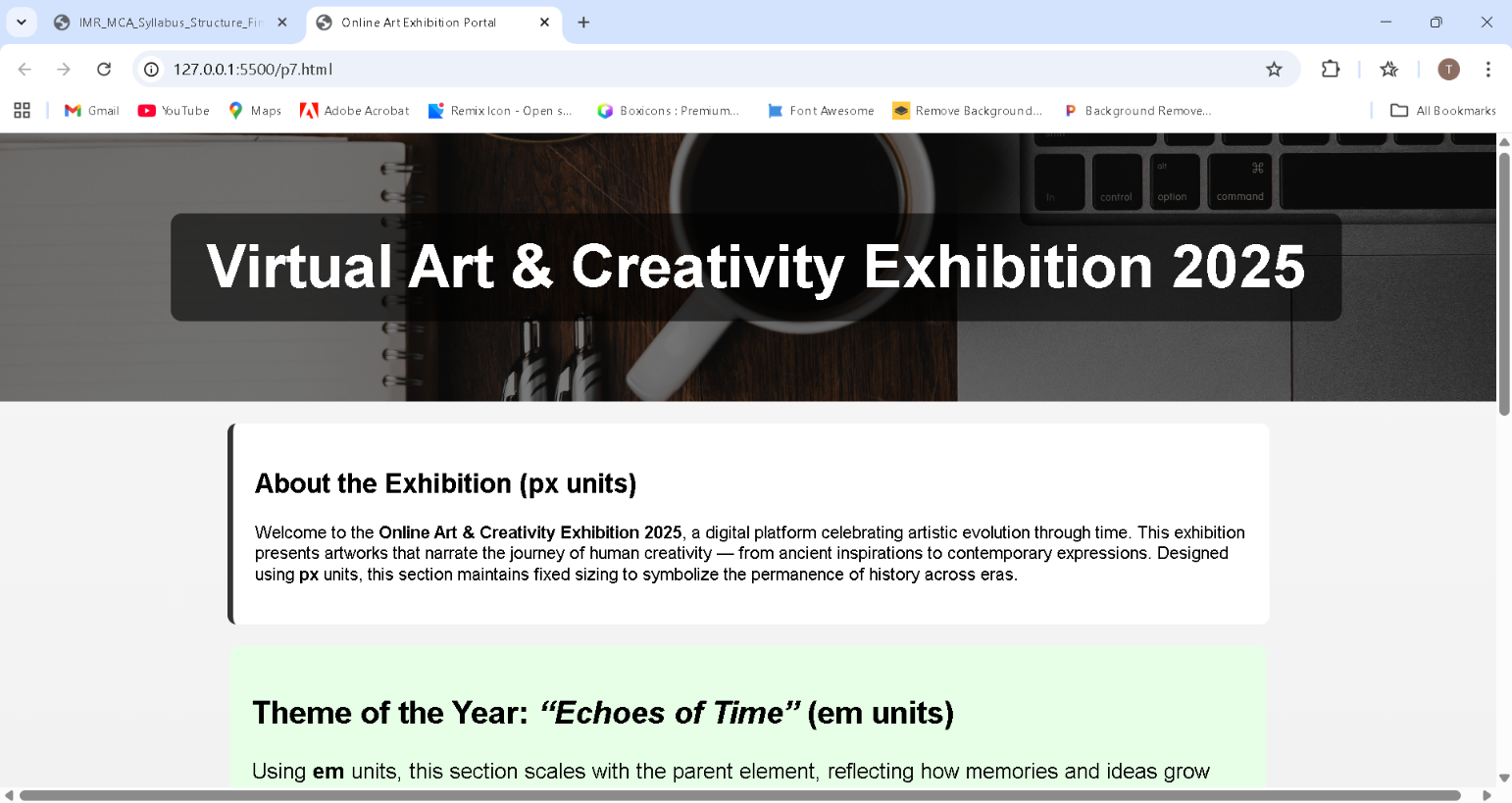
padding: 25px 10px;

text-align: center;

font-size: 1.2rem;

}

**OUTPUT**



**Roll No.:76**

**Name: Raj Sanjay Koli**

**Assignment No: 8**

**Title: Build a photo gallery using Flexbox.**

**1. The gallery should have the following features:**

**a. The images should automatically wrap to fit the screen size (use flexwrap).**

**b. Each image should maintain an equal width and height ratio, adapting**

**responsively across various screen sizes.**

**c. Use justify-content and align-items properties to center the images**

**when there is extra space.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Flexbox Photo Gallery</title>

<link rel="stylesheet" href="style8.css">

</head>

<body>

<h1 class="title">Beautiful Flexbox Photo Gallery</h1>

<div class="gallery">

<div class="box"><img src="/p8 img1.jpg" alt=""></div>

<div class="box"><img src="/p8 img2.jpg" alt=""></div>

<div class="box"><img src="/p8 img3.jpg" alt=""></div>

<div class="box"><img src="/p8 img4.jpg" alt=""></div>

<div class="box"><img src="/p8 img5.jpg" alt=""></div>

<div class="box"><img src="/p8 img6.jpg" alt=""></div>

<div class="box"><img src="/p8 img7.jpg" alt=""></div>

<div class="box"><img src="/p8 img8.jpg" alt=""></div>

</div>

</body>

</html>

**CSS**

body {

    font-family: Arial, sans-serif;

    background: linear-gradient(90deg, #2e8ff0, #5fcd99);

    margin: 0;

    padding: 20px;

}

.title {

    text-align: center;

    margin-bottom: 10px;

    color: #fff;

    font-size: 32px;

}

.gallery {

    display: flex;

    flex-wrap: wrap;

    justify-content: center;

    align-items: center;

    gap: 20px;

}

.box {

    width: 250px;

    aspect-ratio: 1 / 1; */\* Ensures square shape \*/*

    overflow: hidden;

    border-radius: 12px;

    box-shadow: 0 4px 10px rgba(0,0,0,0.2);

    transition: transform 0.3s ease, box-shadow 0.3s ease;

}

.box img {

    width: 100%;

    height: 100%;

    object-fit: cover;

}

.box:hover {

    transform: scale(1.05);

    box-shadow: 0 6px 15px rgba(0,0,0,0.3);

}

*/\* Responsive \*/*

@media(max-width: 600px) {

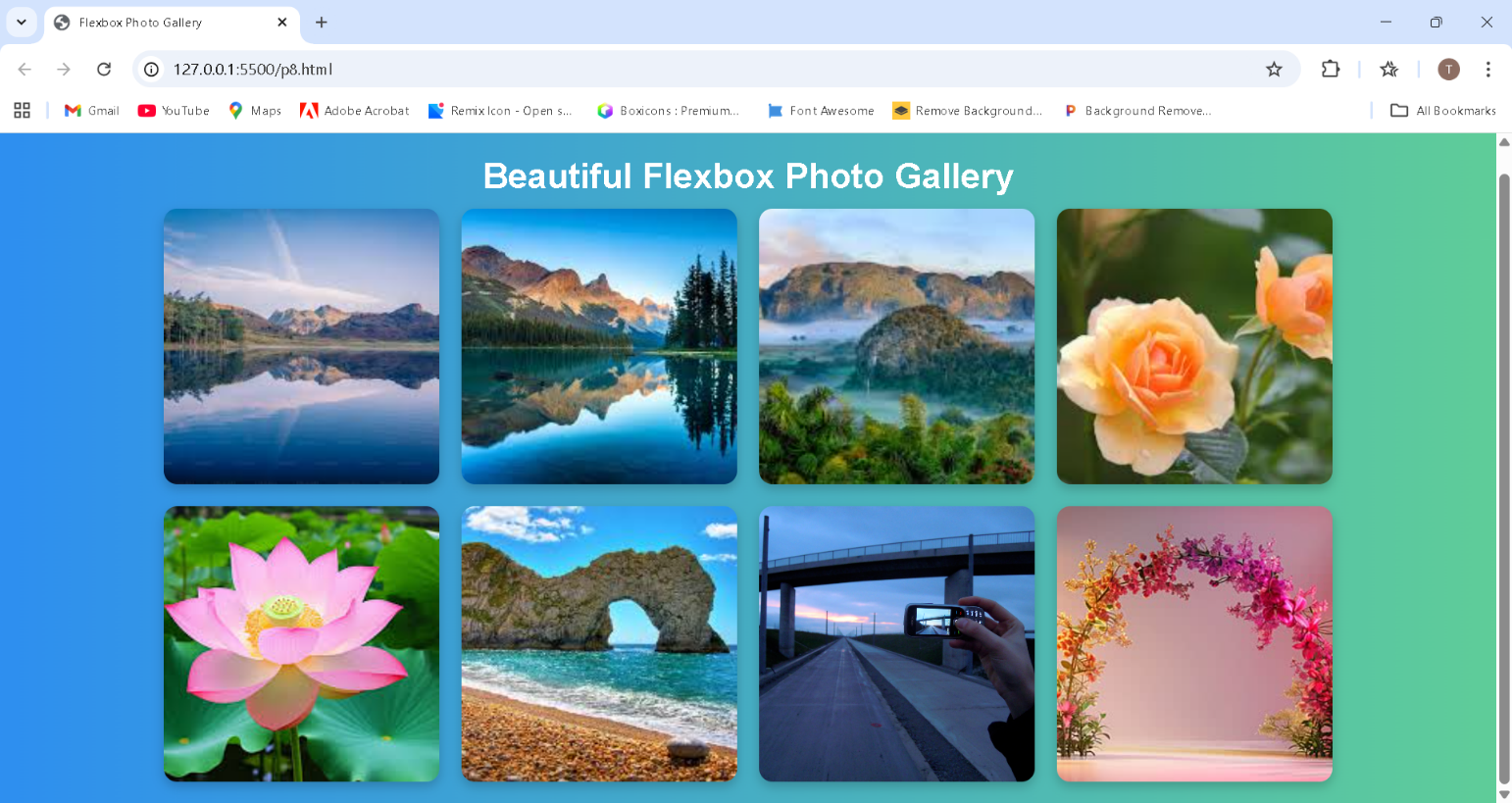
    .box {

        width: 150px;

    }

}

**OUTPUT**

****

**Roll No.:76**

**Name: Raj Sanjay Koli**

**Assignment No: 9**

**Title: Create an interactive button that changes color, size, and shadow on hover using**

**CSS transitions.**

**1. Add smooth animations for these properties with a duration of 0.5s and easing**

**effects like ease-in-out for smoother transitions.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<link rel="stylesheet" href="style9.css">

</head>

<body>

<div class="login-container">

<form class="login-form">

<div class="info-text"><b>Today is the Last Date of

Registration</b></div>

<h2>Login</h2>

<label for="username">Username</label>

<input type="text" id="username" placeholder="Enter username">

<label for="password">Password</label>

<input type="password" id="password"

placeholder="Enter password">

<button type="submit" class="login-btn">Login</button>

</form>

</div>

</body>

</html>

**CSS**

*/\* Center the form on screen \*/*

.login-container {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    background: #f0f0f5;

    font-family: Arial, sans-serif;

}

*/\* Form styling \*/*

.login-form {

    background: white;

    padding: 25px 35px;

    border-radius: 10px;

    box-shadow: 0 5px 20px rgba(0, 0, 0, 0.15);

    width: 300px;

}

.login-form h2 {

    margin-bottom: 20px;

    text-align: center;

    font-weight: 600;

}

*/\* Scrolling + fade animation text \*/*

.info-text {

    display: inline-block;

    font-size: 15px;

    color: #ff3b3b;

    margin-bottom: 15px;

    overflow: hidden;

    white-space: nowrap;

    animation: scrollText 8s linear infinite, fadeIn 2s ease-in;

}

*/\* Scroll animation (left to right) \*/*

@keyframes scrollText {

    0% {

        transform: translateY(120%);

    }

    100% {

        transform: translateY(-120%);

    }

}

*/\* Fade-in animation \*/*

@keyframes fadeIn {

    0% {

        opacity: 0;

    }

    100% {

        opacity: 1;

    }

}

*/\* Input fields \*/*

.login-form input {

    width: 100%;

    padding: 10px;

    margin: 10px 0 18px;

    border: 1px solid #ccc;

    border-radius: 5px;

}

.login-form input:focus {

    border-color: #007bff;

    outline: none;

    box-shadow: 0 0 5px rgba(0, 123, 255, 0.5);

}

*/\* Interactive Login Button \*/*

.login-btn {

    width: 100%;

    padding: 12px;

    background: #007bff;

    color: white;

    border: none;

    border-radius: 5px;

    font-size: 16px;

    cursor: pointer;

*/\* Smooth transitions for color, size, shadow \*/*

    transition:

        background-color 0.5s ease-in-out,

        transform 0.5s ease-in-out,

        box-shadow 0.5s ease-in-out;

}

.login-btn:hover {

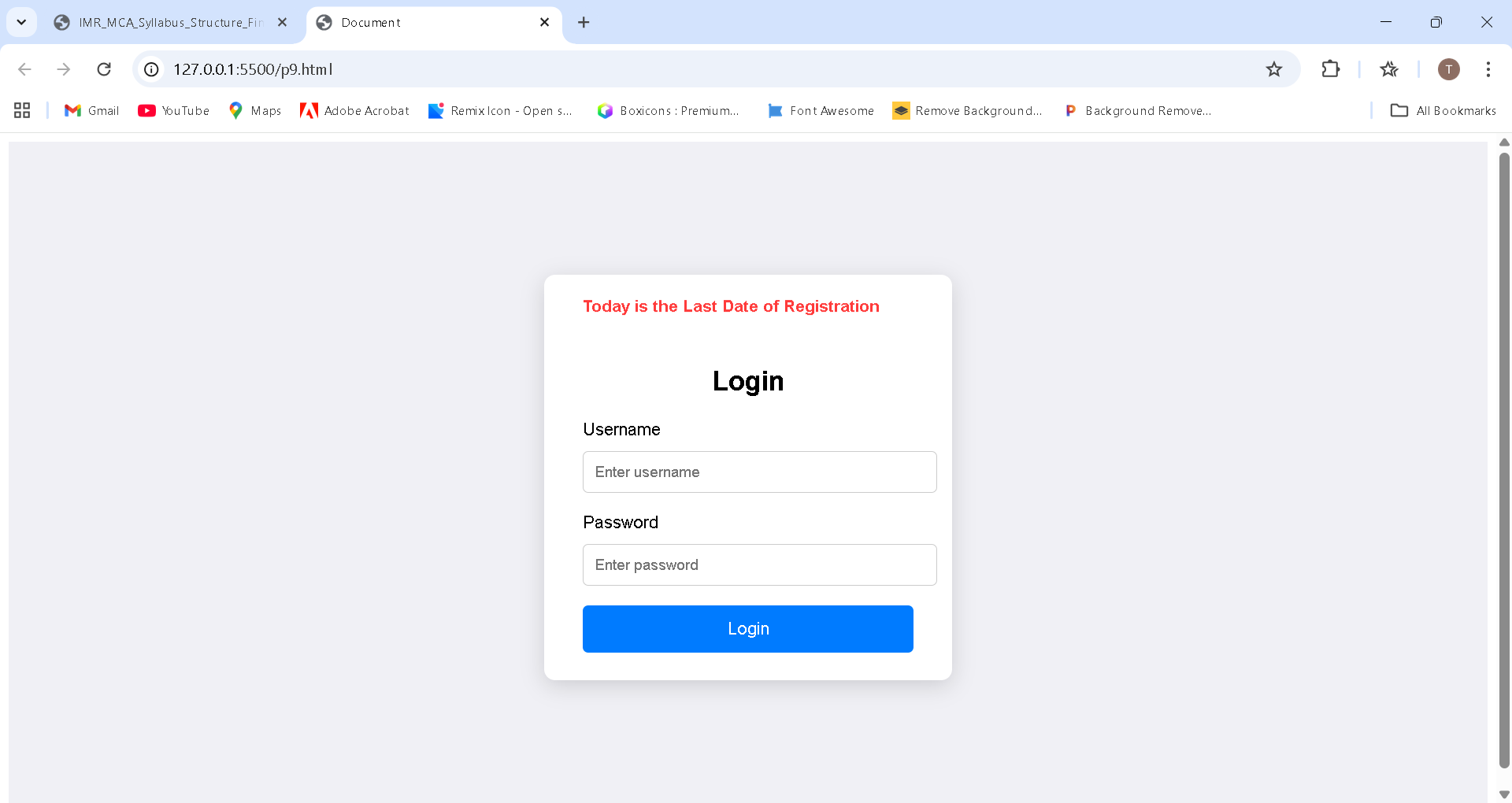
    background-color: #50ebd4;

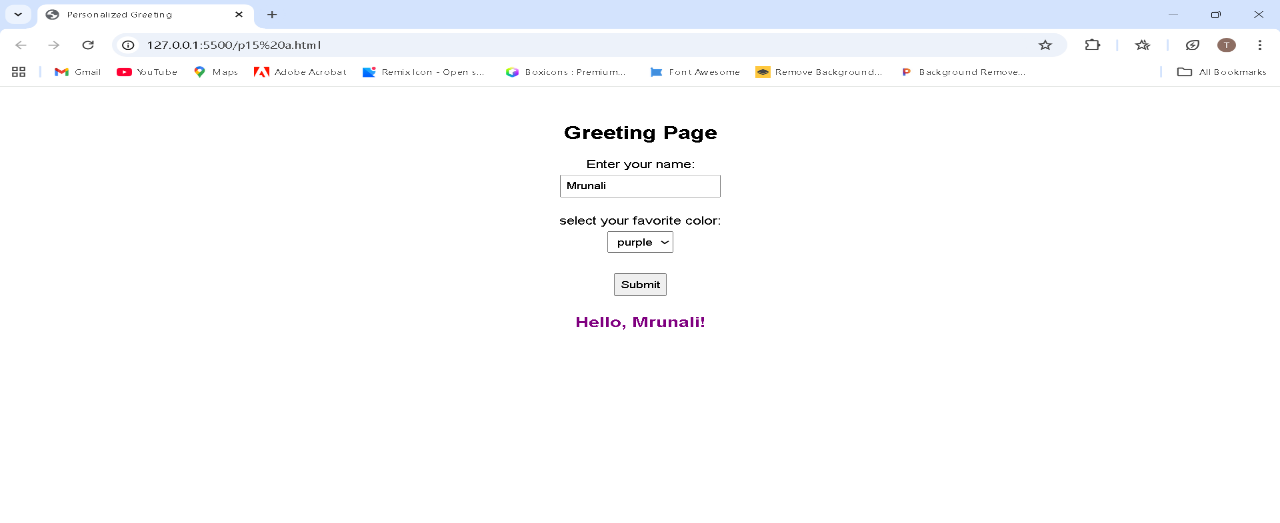
    transform: scale(1.08);

    box-shadow: 0 8px 20px rgba(0, 0, 0, 0.25);

}

**OUTPUT**

**Without Transitions.**

**With Transitions.**

**Name: Raj Sanjay Koli**

**Assignment No: 10**

**Title: Create a simple form with input fields (text, email, checkbox, radio buttons, etc.). Style different form elements using attribute selectors (i.e Child, Sibling and Descendant selectors)**

<!DOCTYPE html>

<html>

<head>

    <title>Form Styling with CSS Selectors</title>

    <link rel="stylesheet" href="style10.css">

</head>

<body>

<h2>Form Styled with CSS Selectors</h2>

<form>

    <div class="row">

        <label>Name:</label>

        <input type="text" placeholder="Enter your name">

    </div>

    <div class="row">

        <label>Email:</label>

        <input type="email" placeholder="Enter your email">

    </div>

    <div class="row">

        <label>Password:</label>

        <input type="password" placeholder="Enter your password">

    </div>

    <div class="row">

        <label>Newsletter:</label>

        <div>

            <input type="checkbox" id="subscribe">

            <label for="subscribe">Subscribe</label>

        </div>

    </div>

    <div class="row">

        <label>Gender:</label>

        <div>

            <input type="radio" name="g" value="male"> Male

            <input type="radio" name="g" value="female"> Female

        </div>

    </div>

    <input type="submit" value="Submit">

</form>

</body>

</html>

**CSS Code**

body {

    font-family: Arial, sans-serif;

    background: linear-gradient(135deg, #54dfd3, #49aff8);

}

h2 {

    text-align: center;

    color: #333;

}

form {

    width: 450px;

    margin: 50px auto;

    padding: 20px;

    border: 2px solid #ccc;

    border-radius: 10px;

    background-color: #f9f9f9;

}

.row {

    display: flex;

    align-items: center;

    margin-bottom: 12px;

}

.row label {

    width: 120px;

    font-weight: bold;

}

.row input[type="text"],

.row input[type="email"],

.row input[type="password"] {

    flex: 1;

    padding: 5px;

}

input[type="text"] {

    border: 2px solid blue;

}

input[type="email"] {

    border: 2px solid green;

}

input[type="password"] {

    border: 2px solid purple;

}

input[type="checkbox"] {

    width: 18px;

    height: 18px;

}

input[type="radio"] {

    accent-color: red;

}

input[type="submit"] {

    background-color: orange;

    color: white;

    border: none;

    padding: 8px 12px;

    cursor: pointer;

    margin-top: 10px;

}

*/\* Sibling Selector \*/*

label+input {

    color: rgba(179, 10, 240, 0.523);

}

*/\* Descendant Selector \*/*

div input[type="checkbox"] {

    margin-right: 6px;

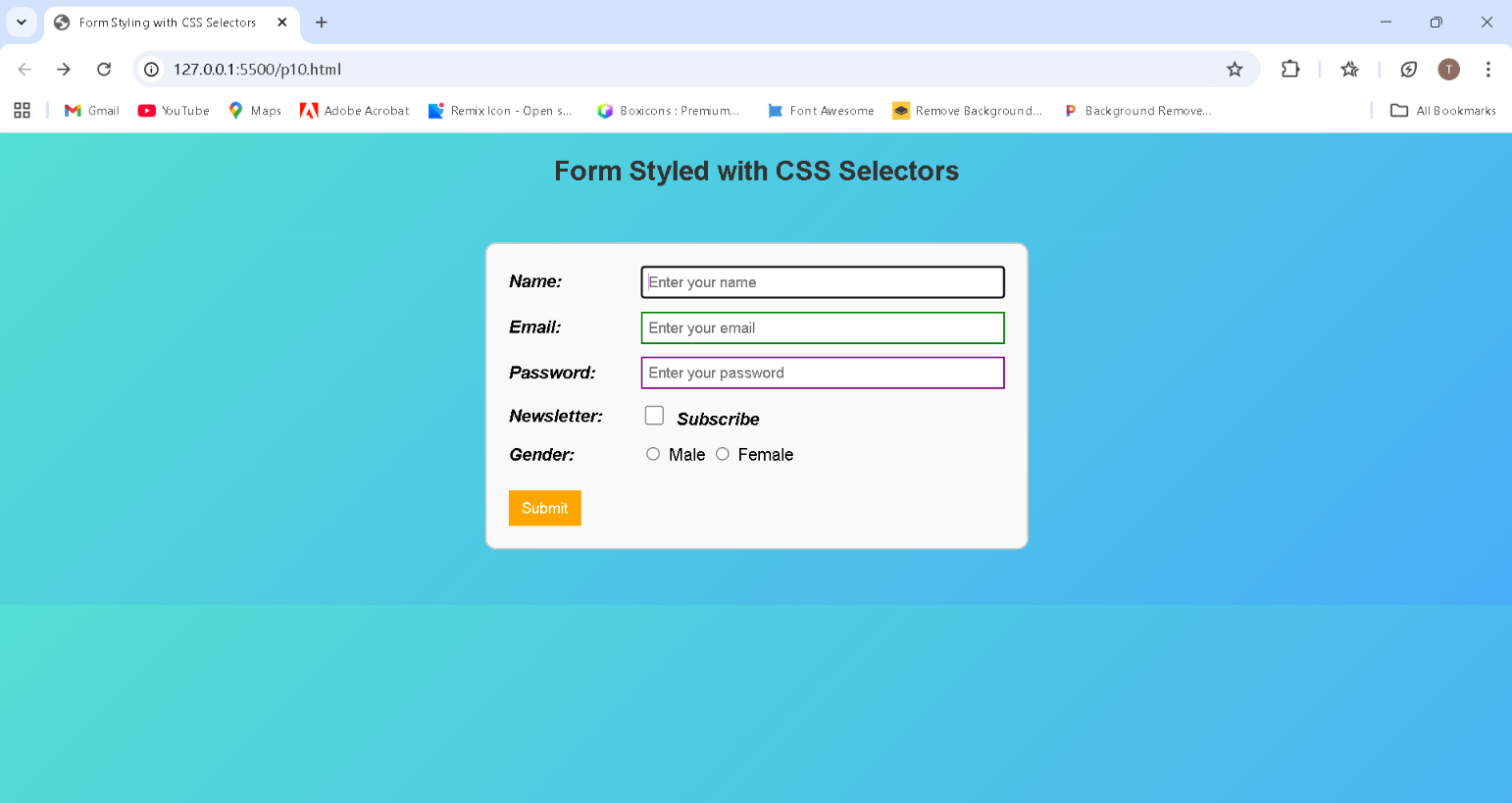
}

div label {

    font-style: italic;

}

**OUTPUT:**

****

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 11**

**Title: a: Create a simple webpage and include CSS variables for spacing (e.g., --padding, --margin) and typography (e.g., --font-size, --line-height).**

**1. Dynamically adjust the layout and typography by modifying the variables inside media queries for responsiveness.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width,initial-scale=1">

    <title>Responsive CSS Variables — Spacing & Typography</title>

    <link rel="stylesheet" href="style11.css">

</head>

<body>

    <div class="wrap">

        <header>

            <div class="brand">TypoSpace</div>

            <nav>

            <ul>

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

                <li><a href="#">Docs</a></li>

                <li><a href="#">Examples</a></li>

            </ul>

            </nav>

        </header>

        <main>

            <article class="card">

                <h1>Responsive layout using CSS variables</h1>

                <p class="meta">Resize the viewport to see spacing and typography change via media queries that update CSS variables.</p>

                <h2>Why use variables?</h2>

                <p>CSS variables let you centralize design tokens (spacing, font sizes, line heights). Changing a variable in a media query updates the whole UI without repeating many rules.</p>

                <div class="grid">

                    <div class="demo-box">

                        <strong>Base font-size:</strong>

                        <div class="muted">Uses <code>--font-size-base</code> (changes at breakpoints)</div>

                    </div>

                    <div class="demo-box">

                        <strong>Spacing:</strong>

                        <div class="muted">Uses <code>--padding</code>, <code>--card-padding</code>, and <code>--gutter</code></div>

                    </div>

                    <div class="demo-box">

                        <strong>Line-height:</strong>

                        <div class="muted">Uses <code>--line-height</code> to control readable paragraphs</div>

                    </div>

                </div>

                <h2>Example paragraph</h2>

                <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Cupiditate, pariatur? Doloremque error repellendus quasi, magni officia. Sint, voluptates. Exercitationem, tempora.</p>

            </article>

            <aside class="card">

                <h2>Controls</h2>

                <p class="muted">No JavaScript — the demonstration uses only CSS variables and media queries.</p>

                <div style="margin-top:var(--gutter);">

                    <p><strong>Current tokens</strong></p>

                    <ul class="muted">

                        <li><code>--padding</code></li>

                        <li><code>--gutter</code></li>

                        <li><code>--card-padding</code></li>

                        <li><code>--font-size-base</code></li>

                        <li><code>--line-height</code></li>

                    </ul>

                </div>

            </aside>

        </main>

        <footer>

            Built with CSS variables • Mobile-first responsive tokens

        </footer>

    </div>

</body>

</html>

**CSS Code**

:root {

    --padding: 1rem;

    --gutter: 0.75rem;

    --card-padding: 0.75rem;

    --margin: 1rem;

    --font-size-base: 16px;

    --line-height: 1.6;

    --scale-ratio: 1.15;

    --container-max-width: 900px;

}

@media (min-width: 600px) {

    :root {

        --padding: 1.5rem;

        --gutter: 1rem;

        --card-padding: 1rem;

        --font-size-base: 17px;

        --line-height: 1.65;

        --scale-ratio: 1.2;

        --container-max-width: 1100px;

    }

}

@media (min-width: 1024px) {

    :root {

        --padding: 2rem;

        --gutter: 1.25rem;

        --card-padding: 1.25rem;

        --font-size-base: 18px;

        --line-height: 1.75;

        --scale-ratio: 1.25;

        --container-max-width: 1200px;

    }

}

html {

    font-size: var(--font-size-base);

}

body {

    margin: 0;

    font-family: system-ui, -apple-system, "Segoe UI", Roboto, "Helvetica Neue", Arial;

    line-height: var(--line-height);

    color: #111827;

    background: linear-gradient(180deg, #1f78d1, #4beee0);

    -webkit-font-smoothing: antialiased;

}

.wrap {

    max-width: var(--container-max-width);

    margin: 0 auto;

    padding: var(--padding);

}

header {

    display: flex;

    gap: var(--gutter);

    align-items: center;

    justify-content: space-between;

    margin-bottom: calc(var(--margin) \* 0.5);

}

.brand {

    font-weight: 700;

    font-size: calc(var(--font-size-base) \* var(--scale-ratio));

    letter-spacing: -0.02em;

}

nav ul {

    list-style: none;

    padding: 0;

    margin: 0;

    display: flex;

    gap: calc(var(--gutter) \* 0.6);

    align-items: center;

}

nav a {

    text-decoration: none;

    color: #0f172a;

    padding: calc(var(--gutter) \* 0.5);

    border-radius: 6px;

    font-size: 0.95rem;

}

main {

    display: grid;

    grid-template-columns: 1fr;

    gap: var(--gutter);

}

@media (min-width: 900px) {

    main {

        grid-template-columns: 2fr 1fr;

    }

}

article.card,

aside.card {

    background: white;

    padding: var(--card-padding);

    border-radius: 12px;

    box-shadow: 0 6px 18px rgba(16, 24, 40, 0.06);

}

h1 {

    font-size: calc(var(--font-size-base) \* var(--scale-ratio) \* 1.15);

    margin: 0 0 .35rem 0;

}

h2 {

    font-size: calc(var(--font-size-base) \* var(--scale-ratio));

    margin: 0 0 .5rem 0;

}

p {

    margin: 0 0 1rem 0;

}

.meta {

    font-size: .9rem;

    color: #475569;

}

.grid {

    display: grid;

    gap: var(--gutter);

    grid-template-columns: repeat(auto-fit, minmax(180px, 1fr));

}

footer {

    margin-top: calc(var(--margin) \* 1.25);

    text-align: center;

    color: #475569;

    font-size: .9rem;

}

.demo-box {

    padding: calc(var(--card-padding));

    border-radius: 8px;

    background: linear-gradient(180deg, #f8fafc, #ffffff);

    border: 1px solid rgba(15, 23, 42, 0.03);

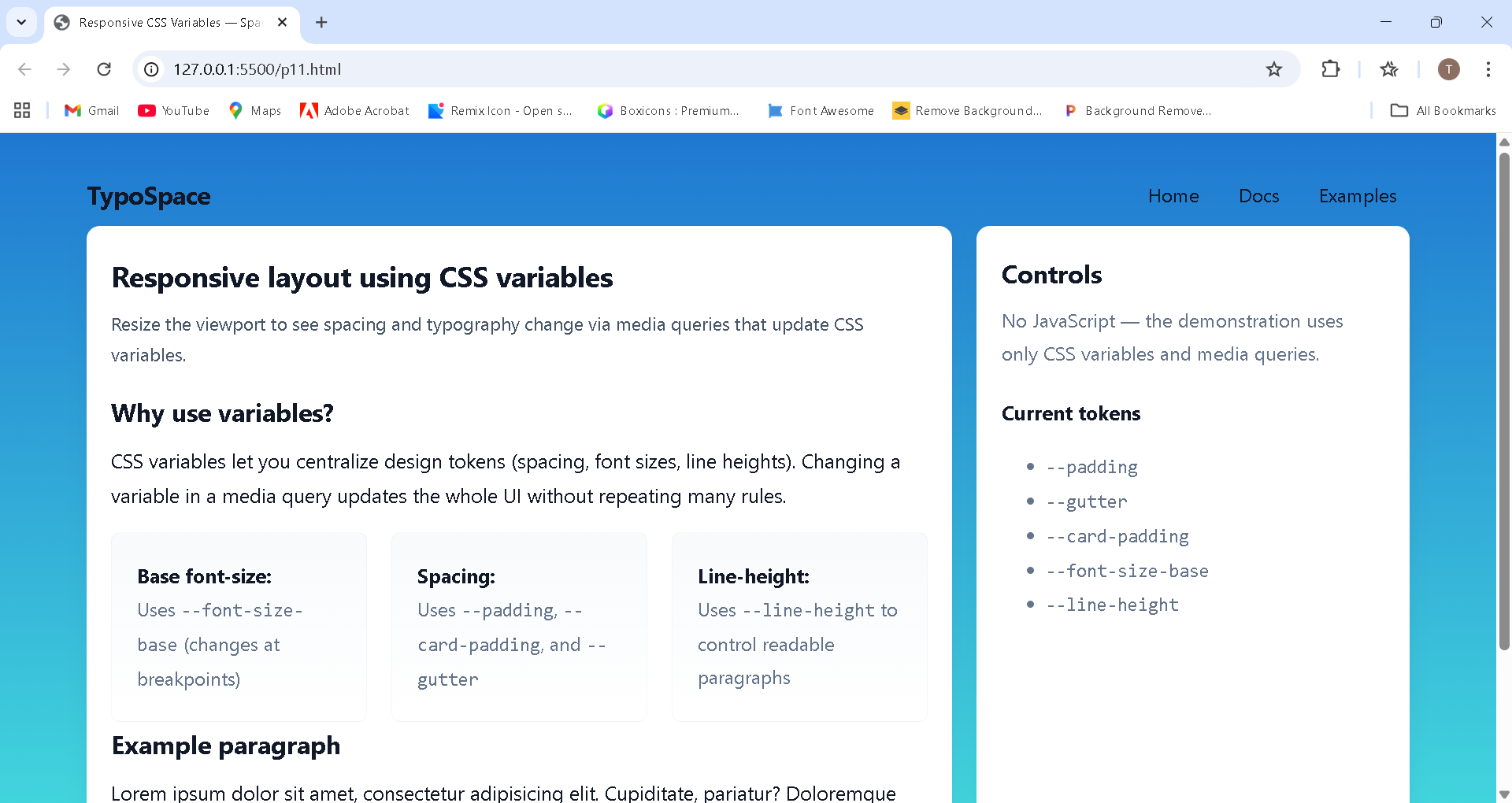
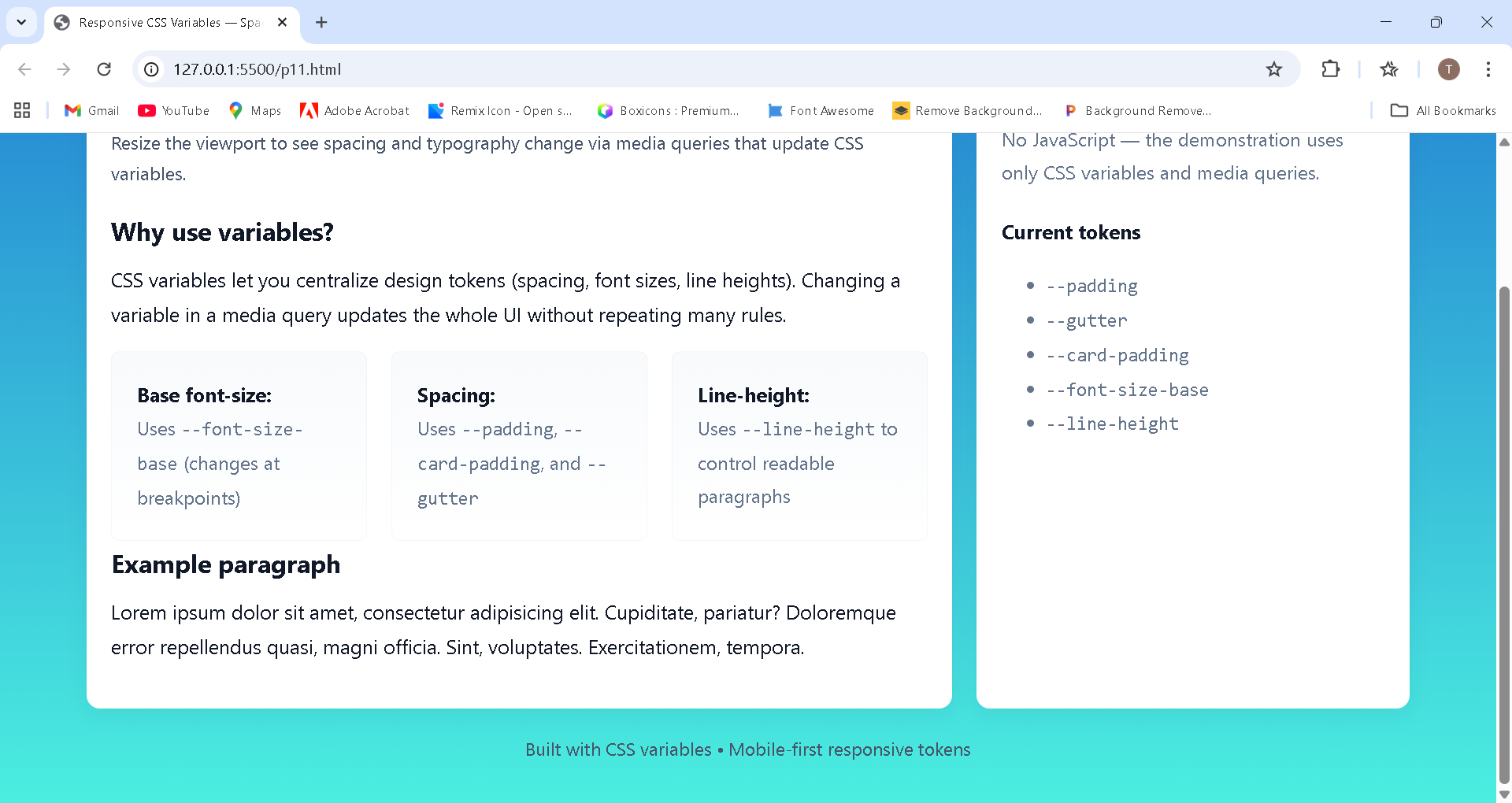
}

.muted {

    color: #64748b;

    font-size: .95rem;

}

**OUTPUT**

**Roll No.:76**

**Roll No.:76**

**Name: Raj Sanjay Koli**

**Assignment No: 12**

**Title: Install SASS and compile a basic SASS file into CSS.**

1. **Create a styles.scss file with basic styles and compile it into styles.css.**
2. **Use SASS features like variables, nesting, and partials to structure your CSS in a more modular way.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<link rel="stylesheet" href="/sass/main.css">

</head>

<body>

<div id="d1"> Welcome to My Web page

<ul>

<li>Algo</li>

<li>DS</li>

<li>language</li>

<li>interviews</li>

<li>CS subject</li>

</ul>

</div>

</body>

</html>

**main.scss**

$fs: 30px;

$bgcolor: Magenta;

$pd: 100px 350px; #d1{

font-family: $fs; color: $bgcolor; padding: $pd;

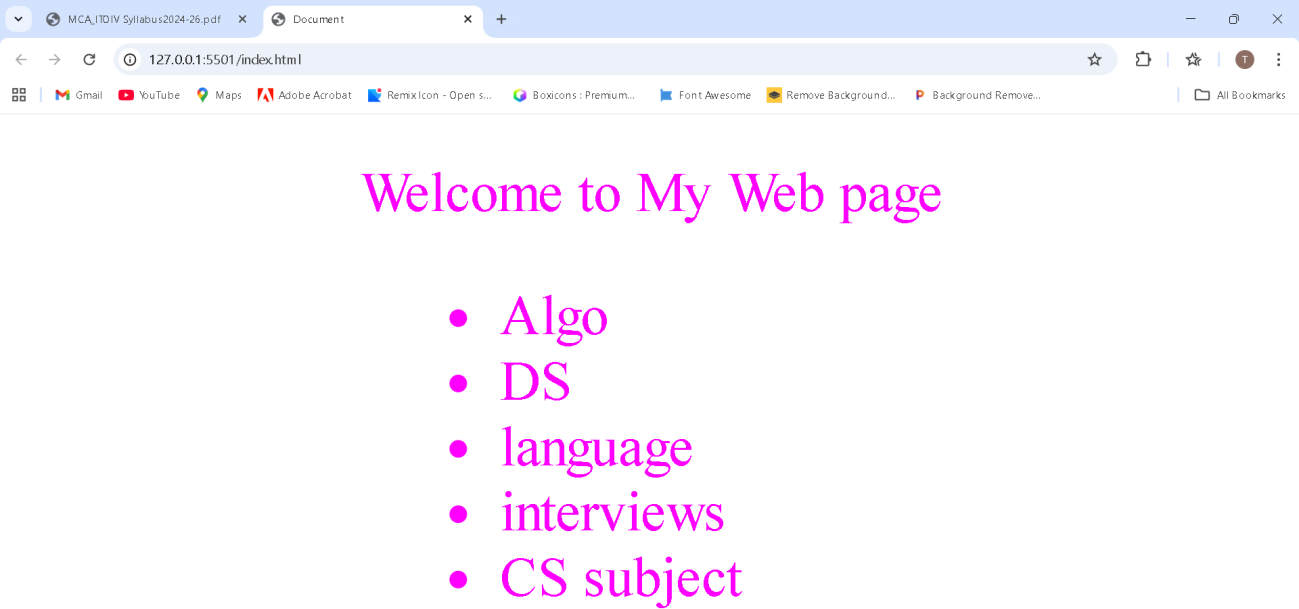
}

**main.css**

#d1 {

font-family: 30px; color: Magenta; padding: 100px 350px;

}

**OUTPUT**

**Roll No.:76**

**Name: Raj Sanjay Koli**

**Assignment No: 12**

**Title: Install SASS and compile a basic SASS file into CSS.**

1. **Create a styles.scss file with basic styles and compile it into styles.css.**
2. **Use SASS features like variables, nesting, and partials to structure your CSS in a more modular way.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>SASS Variables, Nesting and Partials</title>

<link rel="stylesheet" href="/sass/main.css">

*<!-- Compiled CSS file -->*

<link rel="stylesheet" href="css/main.css">

</head>

<body>

<h2>SASS Button Example</h2>

<button class="btn-primary">Primary Button</button>

<button class="btn-success">Success Button</button>

<button class="btn-danger">Danger Button</button>

</body>

</html>

\_**variables.scss**

$padding: 10px 20px;

$font-size: 15px;

$radius: 5px;

$primary-color: blueviolet;

$success-color: rgb(166, 115, 213);

$danger-color: rgb(188, 243, 35);

$hover-primary: aqua;

$hover-success: rgb(255, 0, 247);

$hover-danger: rgb(4, 66, 66);

\_**buttons.scss**

*/\* Placeholder selector \*/*

%button-base { display: inline-block; padding: $padding; font-size: $font-size; font-weight: bold; text-align: center; border: none;

border-radius: $radius; cursor: pointer;

transition: background-color 0.3s ease;

}

*/\* Primary Button \*/*

.btn-primary {

@extend %button-base;

background-color: $primary-color;

&:hover {

background-color: $hover-primary;

}

*/\* Success Button \*/*

.btn-success {

@extend %button-base;

background-color: $success-color;

&:hover {

background-color: $hover-success;

}

}

*/\* Danger Button \*/*

.btn-danger {

@extend %button-base;

background-color: $danger-color;

&:hover {

background-color: $hover-danger;

}

}

}

**main.scss**

@import 'variables'; @import 'buttons';

**main.css**

*/\* Placeholder selector \*/*

.btn-danger, .btn-success, .btn- primary {

display: inline-block; padding: 10px 20px; font-size: 15px;

font-weight: bold; text-align: center; border: none;

border-radius: 5px; cursor: pointer;

transition: background-color 0.3s ease;

}

*/\* Primary Button \*/*

.btn-primary {

background-color: blueviolet;

}

.btn-primary:hover {

background-color: aqua;

*/\* Success Button \*/*

.btn-success {

background-color: rgb(166, 115,

213);

}

.btn-success:hover {

background-color: rgb(255, 0, 247);

}

*/\* Danger Button \*/*

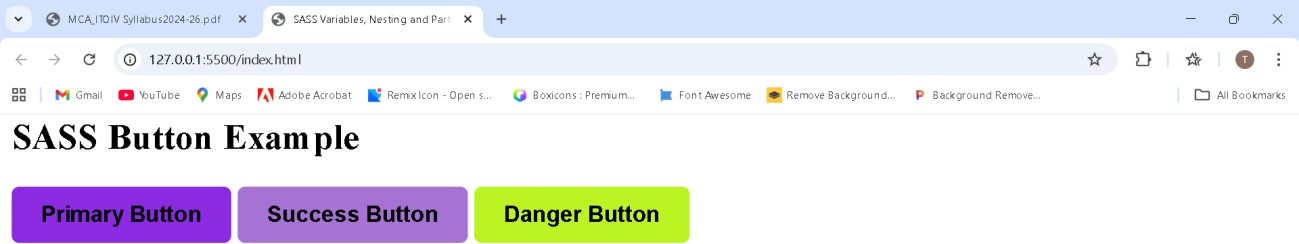
.btn-danger {

background-color: rgb(188, 243, 35);

}

.btn-danger:hover {

background-color: rgb(4, 66,

**OUTPUT**

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 13**

**Title: Create a basic webpage that displays a welcome message and uses JavaScript to:**

**1. Declare variables of different data types (string, number, boolean, array, object).**

**2. Display the variables' values on the webpage.**

**3. Write a function that uses typeof to log the data types of these variables in the browser console.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>JavaScript Data Types Example</title>

    <link rel="stylesheet" href="style13.css">

</head>

<body>

    <h1>Welcome to JavaScript Data Type Demo</h1>

    <div class="box">

        <p>Name: <span id="stringVal"></span></p>

        <p>Age: <span id="numberVal"></span></p>

        <p>Is it True/False:<span id="booleanVal"></span></p>

        <p>Students: <span id="arrayVal"></span></p>

        <p>Information: <span id="objectVal"></span></p>

    </div>

    <script>

*//1.Declare variables of different data types*

        let myString = "Tejas";

        let myNumber = 21;

        let myBoolean = true;

        let myArray = ["Rohit","Yash","Sai"];

        let myObject = {Name:"Tejas",Course:"MCA"};

*//2.Display values on the webpage*

        document.getElementById("stringVal").textContent = myString;

        document.getElementById("numberVal").textContent = myNumber;

        document.getElementById("booleanVal").textContent = myBoolean;

        document.getElementById("arrayVal").textContent = myArray.join(",");

        document.getElementById("objectVal").textContent = JSON.stringify(myObject);

*// 3.Function to log data types using typeof*

        function showTypes() {

            console.log("Type of myString:",typeof myString);

            console.log("Type of myNumber:",typeof myNumber);

            console.log("Type of myBoolean:",typeof myBoolean);

            console.log("Type of myArray:",typeof myArray);

            console.log("Type of myObject:",typeof myObject);

        }

*//Call the function*

        showTypes();

    </script>

</body>

</html>

**Css code**

body{

    font-family: Arial,sans-serif;

    background-color: #f2f2f2;

    padding: 20px;

}

h1{

    color: #333;

}

.box{

    background: white;

    padding: 15px;

    border-radius: 8px;

    width: 350px;

    box-shadow: 0px 0px 10px rgba(0,0,0,0.2);

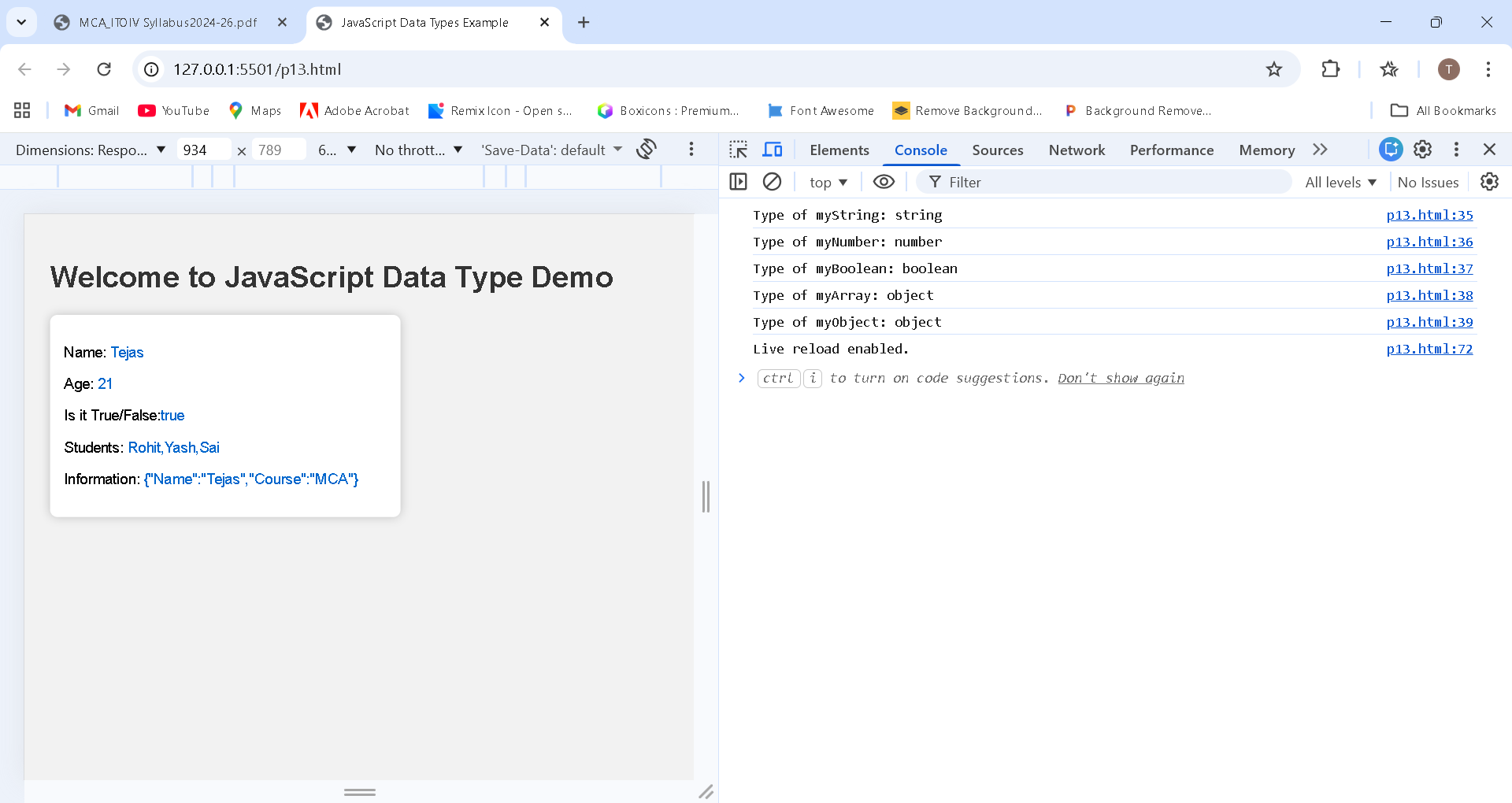
}

p span{

    font: weight bold;

    color: #0066cc;

}

**OUTPUT**

**Roll No: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 14 (A)**

**Title: Extend the webpage to include a simple calculator that performs basic arithmetic operations (+, -, \*, /):**

**1. Use operators to perform calculations based on user input.**

**2. Implement conditional statements to display error messages for invalid inputs (e.g., division by zero).**

**3. Use loops to display a multiplication table for a number entered by the user.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Simple Calculator</title>

</head>

<body>

    <h2>Simple Calculator</h2>

    Number 1:

    <input type="number" id="n1"><br><br>

    Number 2:

    <input type="number" id="n2"><br><br>

    <button onclick="add()">Add</button>

    <button onclick="sub()">Subtract</button>

    <button onclick="mul()">Multiply</button>

    <button onclick="div()">Divide</button>

    <p id="result"></p>

    <hr>

    <script>

        function add(){

            let a= Number(document.getElementById("n1").value);

            let b= Number(document.getElementById("n2").value);

            document.getElementById("result").innerHTML =

                "Result = " + (a + b);

        }

        function sub(){

            let a = Number(document.getElementById("n1").value);

            let b = Number(document.getElementById("n2").value);

            document.getElementById("result").innerHTML =

                "Result =" + (a - b);

        }

        function mul() {

            let a = Number(document.getElementById("n1").value);

            let b = Number(document.getElementById("n2").value);

            document.getElementById("result").innerHTML =

                "Result =" + (a \* b);

        }

        function div() {

            let a = Number(document.getElementById("n1").value);

            let b = Number(document.getElementById("n2").value);

            if (b == 0) {

                document.getElementById("result").innerHTML =

                "Error: Cannot divide by Zero";

            } else {

                document.getElementById("result").innerHTML =

                "Result =" + (a / b);

            }

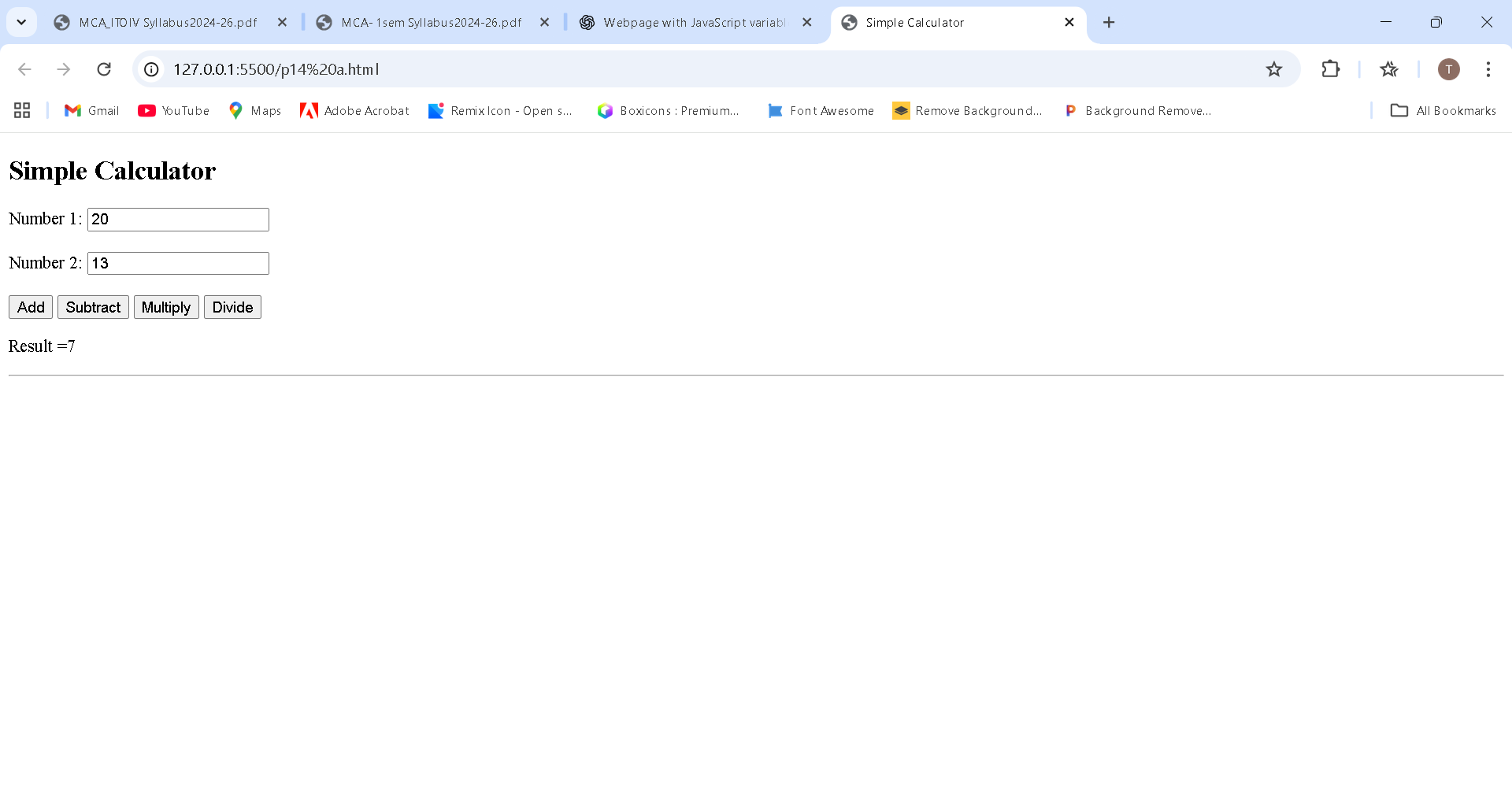
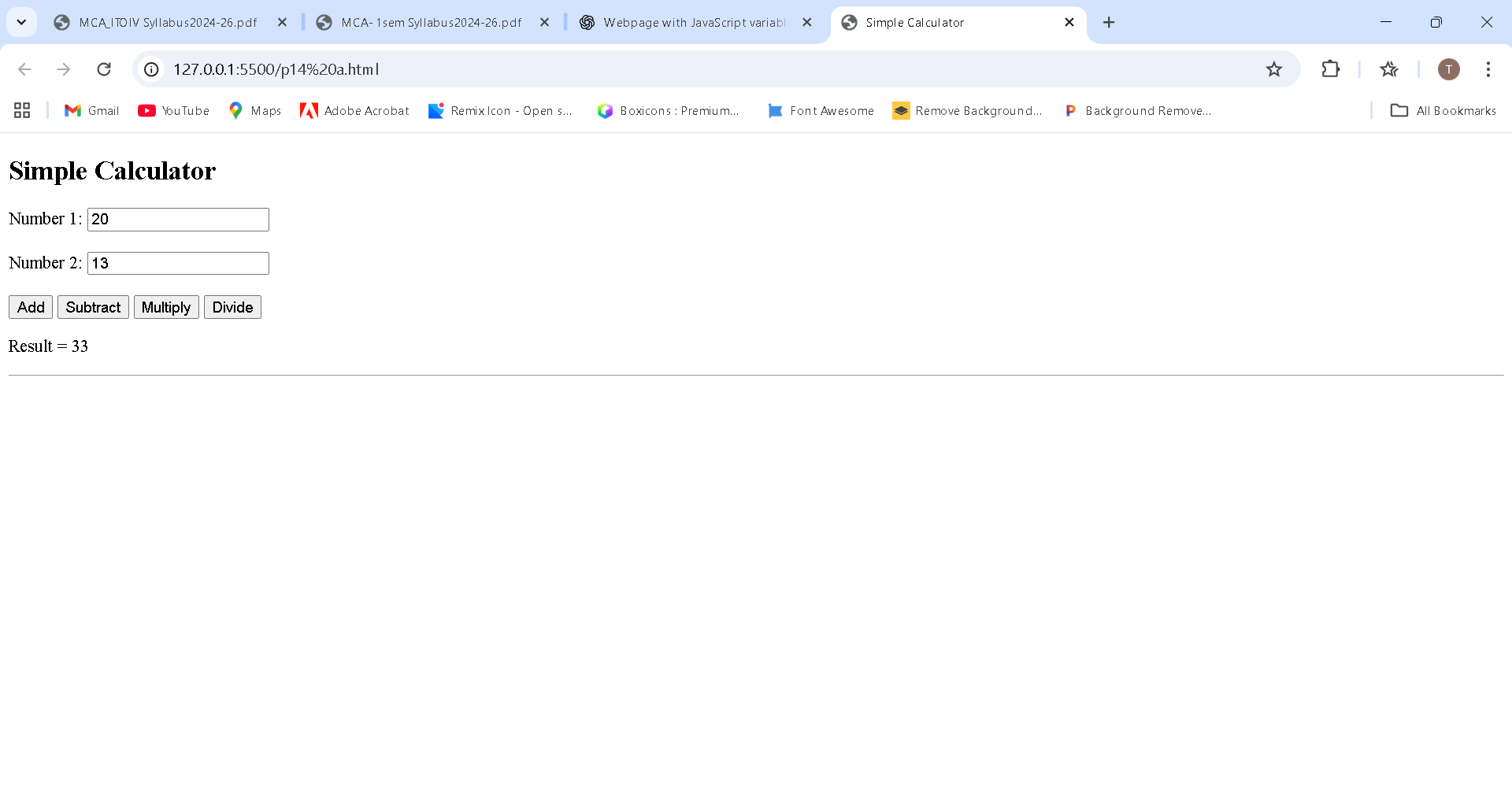
        }

    </script>

</body>

</html>

**OUTPUT**



**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 14 (B)**

**Title: Extend the webpage to include a simple calculator that performs basic arithmetic operations (+, -, \*, /):**

**1. Use operators to perform calculations based on user input.**

**2. Implement conditional statements to display error messages for invalid inputs (e.g., division by zero).**

**3. Use loops to display a multiplication table for a number entered by the user.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Multiplication Table Generator</title>

    <style>

        body {

    font-family: Arial, sans-serif;

    background-color: #b486da;

    text-align: center;

    margin-top: 50px;

}

button {

    padding: 6px 12px;

    background-color: Tan;

    color: white;

    border: none;

    cursor: pointer;

}

button:hover {

    background-color: teal;

}

#tableResult {

    font-family: monospace;

    font-size: 18px;

    display: inline-table;

    padding: 15px;

    text-align: left;

}

    </style>

</head>

<body>

    <h2>Multiplication Table</h2>

    Enter a number:

    <input type="number" id="num">

    <button onclick="table()">Show Table</button>

    <div id="tableResult"></div>

    <script>

        function table() {

            let n = Number(document.getElementById("num").value);

            let output = "";

            for (let i = 1; i <= 10; i++) {

                output += n + " \* " + i + " = " + (n \* i) + "<br>";

            }

            document.getElementById("tableResult").innerHTML = output;

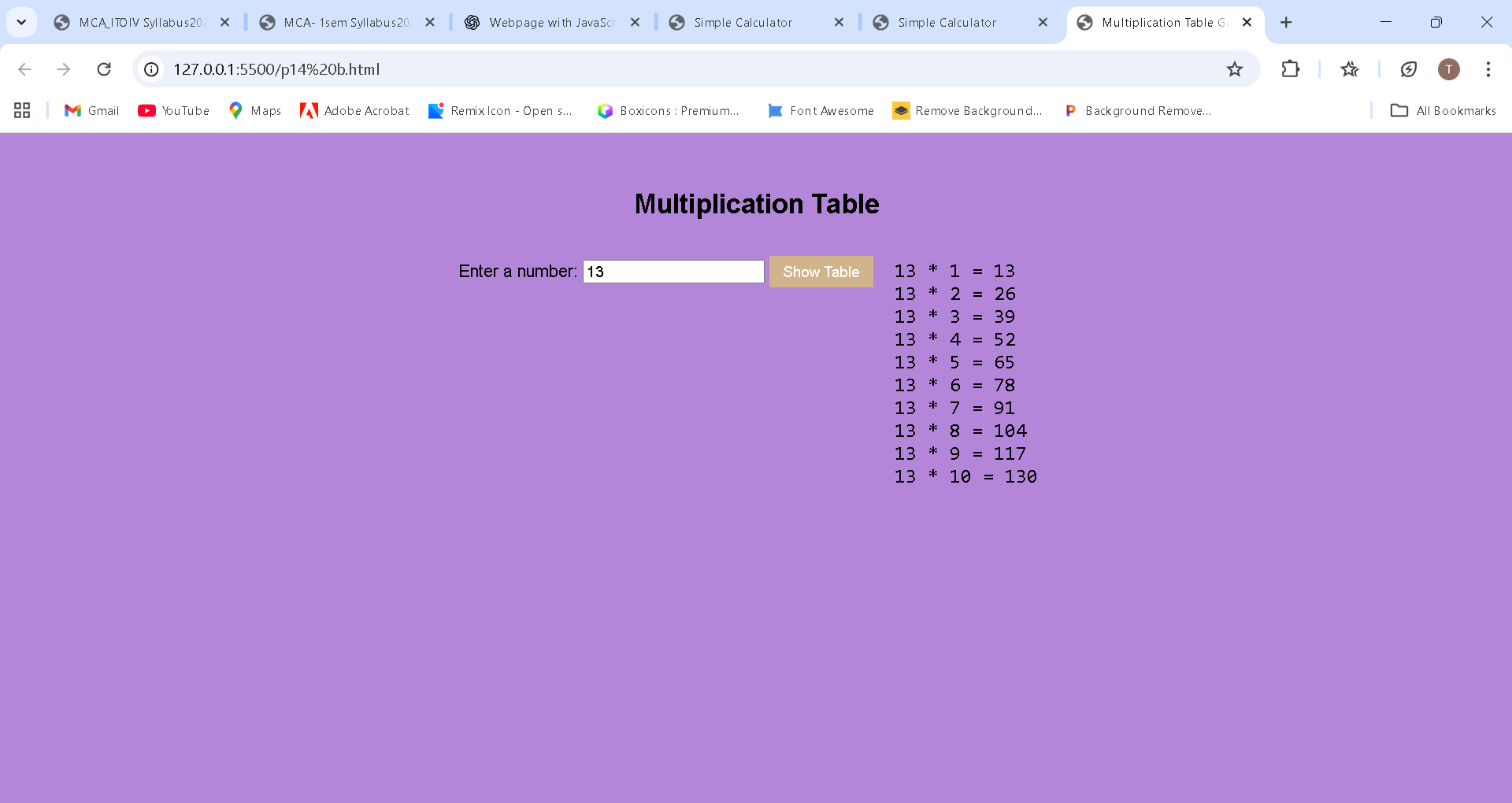
        }

    </script>

</body>

</html>

**OUTPUT**

****

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 15 (A)**

**Title: Build a simple webpage where users can enter their name and select their favorite color from a dropdown.**

**1. Write a JavaScript function to capture user input and display a personalized greeting (e.g., "Hello, [name]!").**

**2. Add an event listener to the color dropdown that changes the background color of the page based on the user's selection.**

**3. Use addEventListener to handle click and change events.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Personalized Greeting</title>

    <style>

        body{

            font-family: Arial, sans-serif;

            margin-top: 50px;

            text-align: center;

        }

        select, input, button{

            padding: 6px;

            margin: 5px;

        }

        #result{

            margin: 20px;

            font-size: 20px;

            font-weight: bold;

        }

    </style>

</head>

<body>

    <h2>Greeting Page</h2>

    Enter your name:

    <br>

    <input type="text" id="name">

    <br><br>

    select your favorite color:

    <br>

    <select id="color">

        <option value="red">Red</option>

        <option value="blue">Blue</option>

        <option value="green">Green</option>

        <option value="purple">purple</option>

        <option value="cyan">cyan</option>

        <option value="yellow">Yellow</option>

    </select>

    <br><br>

    <button onclick="showGreeting()">Submit</button>

    <p id="result"></p>

    <script>

        function showGreeting()

        {

            let userName = document.getElementById("name").value;

            let favColor = document.getElementById("color").value;

            if (userName === "")

            {

                document.getElementById("result").innerHTML =

                "Please enter your name.";

                return;

            }

            document.getElementById("result").innerHTML ="Hello, "+ userName + "!";

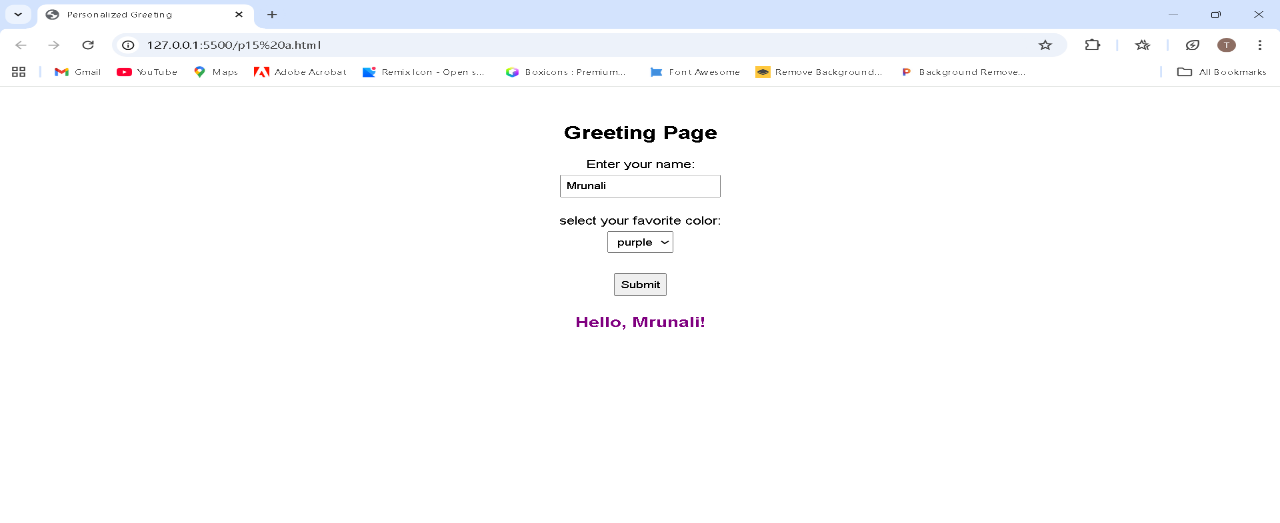
            document.getElementById("result").style.color = favColor;

        }

    </script>

</body>

</html>

**OUTPUT**

**Roll No.: 76**

**Name: Raj Sanjay Koli**

**Assignment No: 15 (B)**

**Title: Build a simple webpage where users can enter their name and select their favorite color from a dropdown.**

**1. Write a JavaScript function to capture user input and display a personalized greeting (e.g., "Hello, [name]!").**

**2. Add an event listener to the color dropdown that changes the background color of the page based on the user's selection.**

**3. Use addEventListener to handle click and change events.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Event Listener Example</title>

    <style>

        body{

            font-family: Arial, sans-serif;

            text-align: center;

            margin-top: 50px;

            transition: background-color 0.5s;

        }

        input, select, button {

            padding: 6px;

            margin: 5px;

        }

        #output {

            margin-top: 20px;

            font-size: 20px;

            font-weight: bold;

        }

    </style>

</head>

<body>

    <h2>Personalized Greeting</h2>

    Enter your name:

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

    Select your favorite color:

    <select id="color">

        <option value="white">Select</option>

        <option value="lightpink">Pink</option>

        <option vlaur="lightblue">Blue</option>

        <option vlaue="lightgreen">Green</option>

        <option value="lavender">Purple</option>

    </select>

    <br><br>

    <button id="greetBtn">Greet Me</button>

    <div id="output"></div>

    <script>

        document.getElementById("color").addEventListener(

            "change", function ()

            {

                let selectedColor = *this*.value;

                document.body.style.backgroundColor = selectedColor;

            }

        );

        document.getElementById("greetBtn").addEventListener(

            "click", function () {

                let name = document.getElementById("name").value;

                let favColor = document.getElementById("color").value;

                document.getElementById("output").innerHTML =

                name + " Your favourite color is "+ favColor + "!"

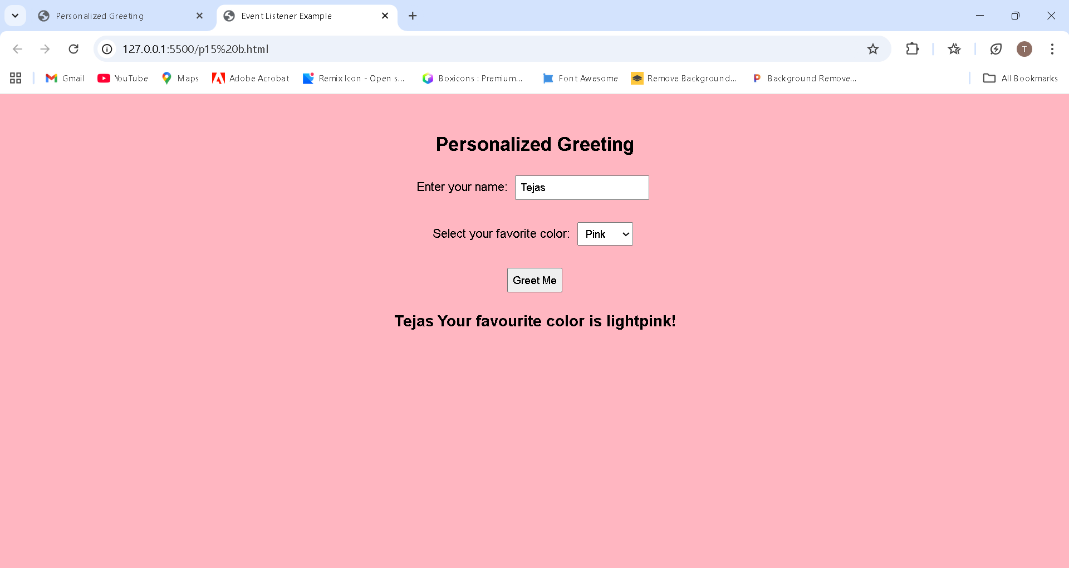
            }

        );

    </script>

</body>

</html>

**OUTPUT**

**Name: Raj Sanjay Koli**

**Roll No.:76**

**Assignment No.: 16**

**Title: Create a basic quiz application with multiple-choice questions:**

**1.Use conditional statements to validate the answers and give feedback to the user**

**(e.g., "Correct!" or "Try again").**

**2.Use loops to display the quiz questions and iterate through an array of answers.**

1. **Use functions to evaluate and display the user's score at the end of the quiz.**

<!DOCTYPE html>

<html>

<head>

<title>Basic Quiz Application</title>

<style> body {

font-family: Arial, sans-serif; margin: 40px;

}

.question {

margin-bottom: 15px;

}

.feedback {

font-weight: bold;

}

</style>

</head>

<body>

<h2>JavaScript Quiz</h2>

<div id="quiz"></div>

<button onclick="submitQuiz()">Submit Quiz</button>

<h3 id="score"></h3>

<script>

// Array of questions let quizData = [

{

question: "1. Which keyword is used to declare a variable in JavaScript?", options: ["let", "int", "float"],

correct: "let"

},

{

question: "2. JavaScript is a language.", options: ["Markup", "Programming", "Styling"], correct: "Programming"

},

{ question: "3. Which symbol is used for comments in JavaScript?", options: ["//", "<! >", "#"],

correct: "//"

}

];

let quizDiv = document.getElementById("quiz");

// Loop to display questions

for (let i = 0; i < quizData.length; i++) { quizDiv.innerHTML +=

"<div class='question'>" +

"<p>" + quizData[i].question + "</p>" +

"<input type='radio' name='q" + i + "' value='" + quizData[i].options[0] + "'> "

+ quizData[i].options[0] + "<br>" +

"<input type='radio' name='q" + i + "' value='" + quizData[i].options[1] + "'> "

+ quizData[i].options[1] + "<br>" +

"<input type='radio' name='q" + i + "' value='" + quizData[i].options[2] + "'> "

+ quizData[i].options[2] +

"<p class='feedback' id='f" + i + "'></p>" + "</div>";

}

// Function to evaluate answers and display score function submitQuiz() {

let score = 0;

for (let i = 0; i < quizData.length; i++) {

let selected = document.querySelector("input[name='q" + i + "']:checked");

if (selected) {

// Conditional statement for validation

if (selected.value === quizData[i].correct) { document.getElementById("f" + i).innerHTML = "Correct!"; score++;

} else {

document.getElementById("f" + i).innerHTML = "Try again!";

}

} else {

document.getElementById("f" + i).innerHTML = "No answer selected";

}

}

document.getElementById("score").innerHTML =

"Your Score: " + score + " / " + quizData.length;

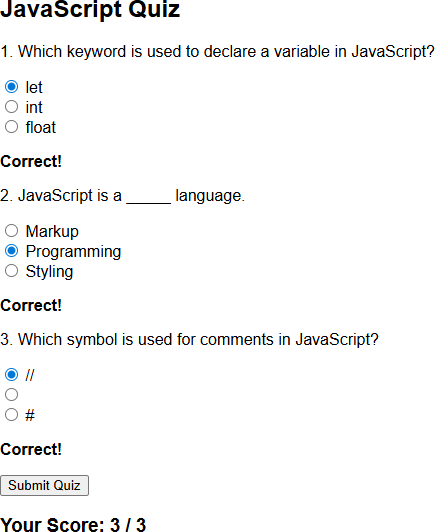
}

</script>

</body>

</html>

**OUTPUT:**

****

**Roll No.:76**

**Name : Raj Sanjay Koli**

**Assignment No.: 17**

**Title: Create a webpage with a list of items (e.g., to-do list or shopping list).**

**1. Use JavaScript to select specific elements from the DOM (e.g., getElementById, querySelectorAll).**

**2. Modify the list items dynamically using JavaScript (e.g., mark items as completed, change text content, or style).**

**3. Implement a button that changes the color of all the list items using classList or inline styles.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>To do list</title>

<style>

body{

font-family: Arial, sans-serif;

background-color: #f2f2f2;

padding: 30px;

}

h2{

color: #333;

}

ul{

list-style: none;

padding: 0;

max-width: 400px;

}

li{

background: #fff ;

margin: 8px 0;

padding: 10px;

border-radius: 5px;

display: flex;

align-items: center;

gap: 10px;

}

/\*completed task style\*/

.completed{

text-decoration: line-through;

color: gray;

}

.highlight{

background-color: rgb(175, 122, 218);

}

button{

margin-top: 15px;

padding: 8px 15px;

cursor: pointer;

}

</style>

</head>

<body>

<h2>My to do list</h2>

<ul id="todoList">

<li>

<input type="checkbox">

<span>Complete Javascript Assignment</span>

</li>

<li>

<input type="checkbox">

<span>Prepare for Web development Assignment</span>

</li>

<li>

<input type="checkbox">

<span> Practice of DOM Manipulation</span>

</li>

<li>

<input type="checkbox">

<span>Revise CSS basics</span>

</li>

</ul>

<button id="colorBtn">Done</button>

<script>

/\* Select elements from DOM \*/

const checkboxes= document.querySelectorAll("#todolist input[type='checkbox']");

const listItems= document.querySelectorAll("todolist li");

const colorButton= document.getElementById("colorBtn");

/\* Mark item as completed when checkbox is checked \*/

checkboxes.forEach(function(checkbox){

checkbox.addEventListener("change",function(){

const text= checkbox.nextElementSibling;

text.classList.toggle("completed");

});

});

/\* change color of all list items \*/

colorButton.addEventListener("click", function() {

listItems.forEach(function(item) {

item.classList.toggle("highlight");

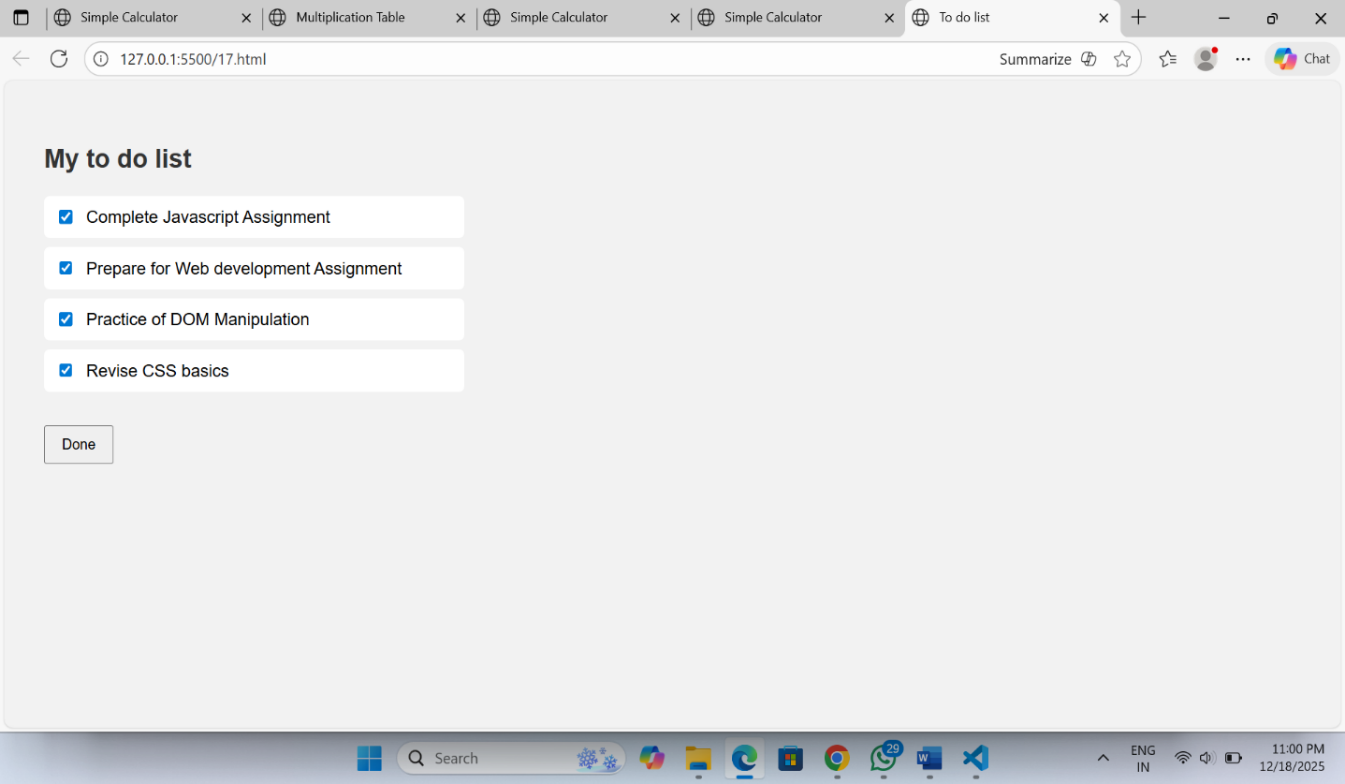
});

});

</script>

</body>

</html>

**Output: **