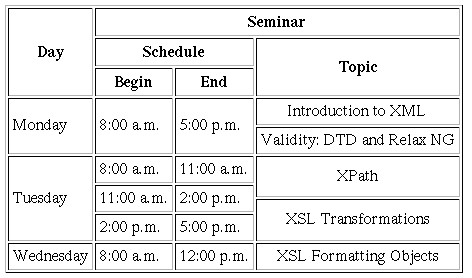
**Questions**

**1. Use <iframe> tag:-**

* Implement <iframe> tag by embedding the following elements from another webpage: <header>, <nav>, <section>,

<article>, <footer>. Add a navigation bar inside <nav> with links to different sections of the page using the <a> tag.

* Embed a table from another webpage using <iframe>. The structure of the table should be as:-



* Embed <map> and <area> from another web page using <iframe>. Create a clickable map of a school or college campus.
* Use an <img> tag to display a campus map.
* Use a <map> with multiple <area> tags to create clickable zones for different buildings (e.g., Library, Cafeteria, Sports Complex).
* Each <area> should link to a new page with information about that building. Use alt attributes to describe each area.

Iframe Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Seminar Schedule & Campus Map</title>

<style> body { font-family: Arial, sans-serif; text-align: center; margin: 0; padding: 0;

} nav { background-color: #333; padding: 10px;

} nav a { color: white; text-decoration: none; margin: 10px; font-size: 18px;

} iframe { width: 40%; height: 300px; border: 2px solid black; margin: 20px 0;

}

</style>

</head>

<body>

<header>

<h1>Seminar Schedule & Campus Map</h1>

</header>

<nav>

<a href="#table">Seminar Schedule</a>

<a href="#map">Campus Map</a>

</nav>

<section>

<h2 id="table">Embedded Seminar Schedule</h2>

<iframe srcdoc='

<html>

<head><title>Seminar Schedule</title></head>

<body>

<table border="1" width="100%" cellpadding="5" cellspacing="0">

<tr>

<th rowspan="2">Day</th>

<th colspan="2">Schedule</th>

<th rowspan="2">Topic</th>

</tr>

<tr>

<th>Begin</th>

<th>End</th>

</tr>

<tr>

<td rowspan="2">Monday</td>

<td rowspan="2">8:00 a.m.</td> <td rowspan="2">5:00 p.m.</td>

<td>Introduction to XML</td>

</tr>

<tr>

<td colspan="2">Validity: DTD and Relax NG</td>

</tr>

<tr>

<td rowspan="3">Tuesday</td>

<td>8:00 a.m.</td>

<td>11:00 a.m.</td>

<td>XPath</td>

</tr>

<tr>

<td>11:00 a.m.</td>

<td>2:00 p.m.</td>

<td rowspan="2">XSL Transformations</td>

</tr>

<tr>

<td>2:00 p.m.</td>

<td>5:00 p.m.</td>

</tr>

<tr>

<td>Wednesday</td>

<td>8:00 a.m.</td>

<td>12:00 p.m.</td>

<td>XSL Formatting Objects</td>

</tr>

</table> </body>

</html>' title="Seminar Table"></iframe>

<h2 id="map">Embedded Campus Map</h2>

<!-- Google Map of GEHU Bhimtal -->

<h3>Graphic Era Hill University - Bhimtal Campus</h3>

<iframe src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d1364.405374216063!

2d79.5282716!3d29.375376!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!

1s0x39a0987622d2a7f3%3A0x8a1cf97c55e9e156!2sGraphic%20Era%20Hill%20University

%20-%20Bhimtal%20Campus!5e0!3m2!1sen!2sin!4v1710512000000" allowfullscreen="" loading="lazy">

</iframe>

<!-- Interactive Image Map -->

</section>

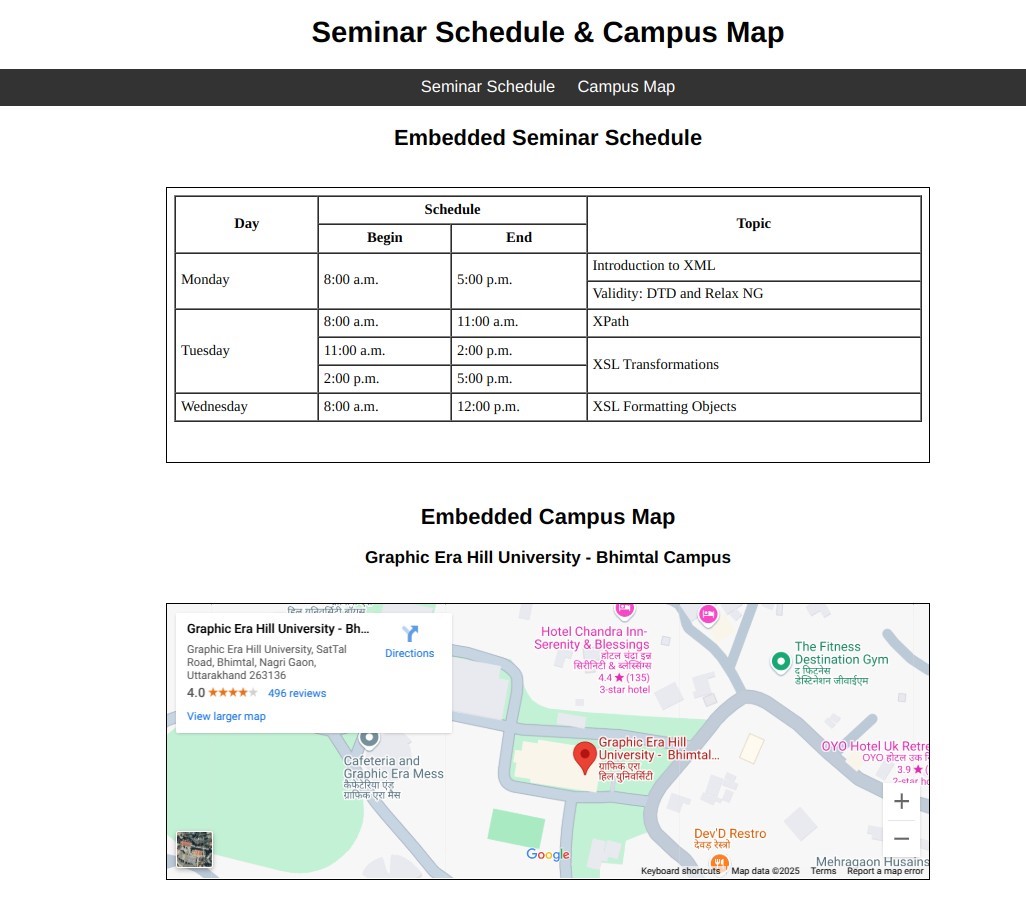
<footer>

<p>&copy; 2025 College Seminar Page</p>

</footer>

</body>

</htm>



**2. Implement the following using CSS3:**

* Style a webpage by changing the font, color, and size of headings and paragraphs.
* Create a button with **background color, padding, and rounded corners**. Add a **hover effect** that changes the background color.
* Display an **image** and add a **border-radius** for rounded corners and add a **shadow effect** to the image.
* Create an **animated gradient border** around a div using @keyframes and border-image-source.
  + Apply an **animated gradient border** that changes colors.
  + Use border-image-source instead of solid borders.
* Create a **parallax scrolling effect** where the background image moves at a different speed than the content.
  + Use background-attachment: fixed; for the parallax effect.
  + Ensure the content scrolls **over the background** smoothly.
* Use Flexbox to center a box in the middle of the page:-
  + Create a div box and center it both vertically and horizontally.
  + Add padding, background color, and a shadow to the box.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>CSS3 Practical Demo</title>

<style>

body {

font-family: 'Arial', sans-serif; margin: 0; padding: 0; color: #333; background-color: #f8f9fa;

}

h1, h2 { text-align: center; margin: 20px 0; color: #0056b3;

} p { text-align: center; font-size: 18px; color: #444; width: 80%; margin: auto;

} section { padding: 40px; text-align: center; border-bottom: 2px solid #ddd;

}

.btn { display: inline-block; padding: 12px 25px; font-size: 20px; color: white; background: linear-gradient(90deg, #007bff, #00b4d8); border: none; border-radius: 30px; cursor: pointer; transition: 0.4s ease-in-out; box-shadow: 0px 5px 15px rgba(0, 0, 0, 0.2);

}

.btn:hover { background: linear-gradient(90deg, #00b4d8, #007bff); transform: scale(1.1); }

.styled-img { width: 350px; height: auto; border-radius: 20px; box-shadow: 5px 5px 20px rgba(0, 0, 0, 0.3); transition: transform 0.3s ease-in-out;

}

.styled-img:hover { transform: scale(1.05); }

@keyframes gradientBorder {

0% { border-image-source: linear-gradient(45deg, #ff5733, #ffcc33);

}

50% { border-image-source: linear-gradient(45deg, #33ff57, #33ccff);

}

100% { border-image-source: linear-gradient(45deg, #ff33cc, #5733ff);

}

}

.gradient-border { width: 300px; height: 100px; margin: 30px auto; text-align: center; line-height: 100px; font-size: 22px; font-weight: bold; border: 12px solid; border-image-slice: 1; animation: gradientBorder 3s infinite alternate; }

.parallax { background-image: url('https://source.unsplash.com/1600x900/?nature,mountains');

height: 400px; background-attachment: fixed; background-position: center; background-repeat: no-repeat; background-size: cover; display: flex; align-items: center; justify-content: center; color: white; font-size: 28px; font-weight: bold; text-shadow: 2px 2px 10px rgba(0, 0, 0, 0.5); }

.center-box { display: flex; justify-content: center; align-items: center; height: 100vh;

}

.box { width: 320px; height: 200px; padding: 20px; text-align: center; background: linear-gradient(120deg, #ff9a9e, #fad0c4); box-shadow: 5px 5px 20px rgba(0, 0, 0, 0.3); border-radius: 15px; font-size: 20px; font-weight: bold; color: white;

}

</style>

</head>

<body>

<h1>CSS3 Practical Demonstration</h1>

<p>This page showcases various CSS3 features such as buttons, gradients, parallax effects, and Flexbox centering. </p>

<section>

<h2>Interactive Button</h2>

<p>Click the button to see a hover effect with a color transition.</p>

<button class="btn">Hover Me</button>

</section>

<section>

<h2>Styled Image with Effects</h2>

<p>The image below has rounded corners, a shadow effect, and a hover zoom effect.</p>

<img src="https://resources.saylor.org/wwwresources/wp-content/uploads/

20231121110004/gehu-logo.jpeg" alt="GEHU Logo" class="styled-img"> </section>

<section>

<h2>Animated Gradient Border</h2>

<p>This box has a border that changes colors dynamically using CSS animations.</p>

<div class="gradient-border">Gradient Border</div>

</section>

<section class="parallax">

Smooth Parallax Scrolling

</section>

<section>

<h2>Centered Box using Flexbox</h2>

<p>This box is perfectly centered using CSS Flexbox.</p>

<div class="center-box">

<div class="box">

Centered Box with Gradient Background

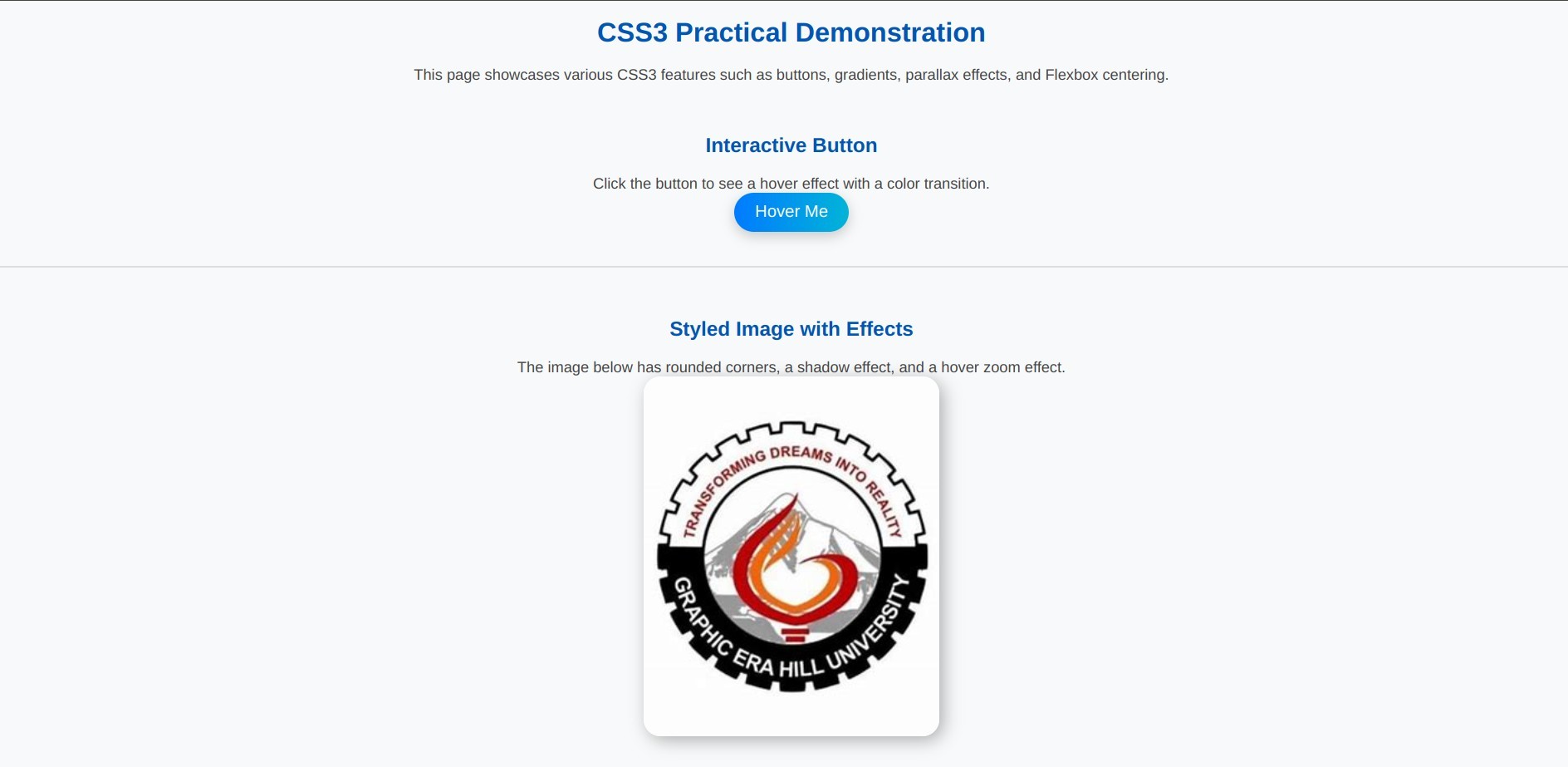
</div>

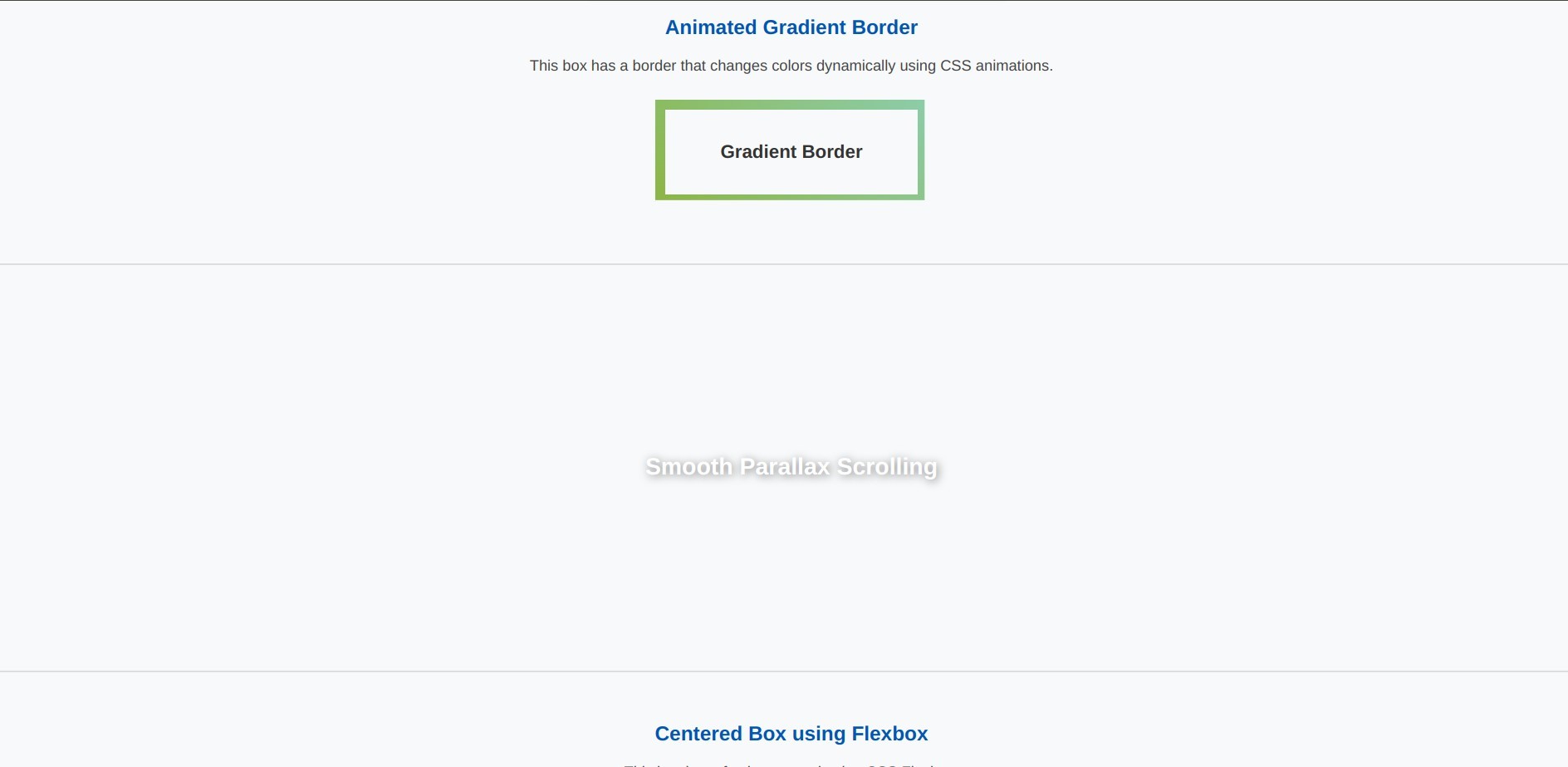
</div>

</section>

</body>

</html>





**3. Implement Javascript for following:-**

* Create a **Student Registration Form** using **HTML** that includes the following fields:

**Form Requirements:**

* + **Student Name** – (Text input, Required)
  + **Email** – (Email input, Required, Must be in a valid email format)
  + **Phone Number** – (Number input, Required, Must be exactly 10 digits)
  + **Date of Birth** – (Date input, Required)
  + **Gender** – (Radio buttons: Male, Female, Other, Required)
  + **Course Selection** – (Dropdown with at least 3 options, Required)
  + **Address** – (Textarea, Required)
  + **Password** – (Password input, Required, Minimum 6 characters)
  + **Confirm Password** – (Password input, Must match Password)
  + **Submit Button** – (Button to submit the form)

Validate a user registration form (e.g., check email format, password strength).

* **To-Do List** – Create a simple to-do list where users can add and remove tasks.
* **Digital Clock** – Display the current time that updates every second.
* **Color Changer** – Allow users to change the background color by clicking buttons.
* **Image Slider** – Create a basic image carousel that auto-plays or changes images on button clicks.

Form Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Student Registration Form</title>

<style> body { font-family: Arial, sans-serif; background-color: #f4f4f4; display: flex; justify-content: center; align-items: center; height: 100vh;

}

.container { background: white; padding: 20px; border-radius: 10px; box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1); width: 400px;

} h2 { text-align: center;

}

label { font-weight: bold;

} input, select, textarea { width: 100%; padding: 8px; margin: 5px 0 10px; border: 1px solid #ccc; border-radius: 5px;

}

.gender-group { display: flex; gap: 10px;

} button { background-color: #28a745; color: white; border: none; padding: 10px; border-radius: 5px; font-size: 16px; cursor: pointer; width: 100%;

}

button:hover { background-color: #218838;

}

.error { color: red; font-size: 14px;

}

</style>

</head>

<body>

<div class="container">

<h2>Student Registration Form</h2>

<form id="registrationForm">

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

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

<span id="nameError" class="error"></span>

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

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

<span id="emailError" class="error"></span>

<label for="phone">Phone Number:</label>

<input type="number" id="phone" required>

<span id="phoneError" class="error"></span>

<label for="dob">Date of Birth:</label>

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

<span id="dobError" class="error"></span>

<label>Gender:</label>

<div class="gender-group">

<input type="radio" name="gender" value="Male" required> Male

<input type="radio" name="gender" value="Female" required> Female

<input type="radio" name="gender" value="Other" required> Other </div>

<span id="genderError" class="error"></span>

<label for="course">Course Selection:</label>

<select id="course" required>

<option value="">Select a Course</option>

<option value="B.Tech">B.Tech</option>

<option value="MBA">MBA</option>

<option value="B.Sc">B.Sc</option>

</select>

<span id="courseError" class="error"></span>

<label for="address">Address:</label>

<textarea id="address" required></textarea>

<span id="addressError" class="error"></span>

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

<input type="password" id="password" required>

<span id="passwordError" class="error"></span>

<label for="confirmPassword">Confirm Password:</label>

<input type="password" id="confirmPassword" required>

<span id="confirmPasswordError" class="error"></span>

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

</form>

</div> <script> document.getElementById("registrationForm").addEventListener("submit", function(event)

{ let valid = true; const name = document.getElementById("name").value.trim(); if (name === "") { document.getElementById("nameError").innerText = "Name is required!"; valid = false;

} else {

document.getElementById("nameError").innerText = "";

}

const email = document.getElementById("email").value.trim(); const emailPattern = /^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/; if (!emailPattern.test(email)) { document.getElementById("emailError").innerText = "Enter a valid email!"; valid = false;

} else { document.getElementById("emailError").innerText = "";

}

const phone = document.getElementById("phone").value.trim(); if (phone.length !== 10 || isNaN(phone)) { document.getElementById("phoneError").innerText = "Phone number must be 10 digits!"; valid = false;

} else {

document.getElementById("phoneError").innerText = "";

}

const dob = document.getElementById("dob").value; if (!dob) { document.getElementById("dobError").innerText = "Date of birth is required!"; valid = false;

} else { document.getElementById("dobError").innerText = "";

}

const gender = document.querySelector('input[name="gender"]:checked'); if (!gender) { document.getElementById("genderError").innerText = "Select a gender!"; valid = false;

} else { document.getElementById("genderError").innerText = "";

}

const course = document.getElementById("course").value; if (course === "") { document.getElementById("courseError").innerText = "Select a course!"; valid = false;

} else { document.getElementById("courseError").innerText = "";

}

const address = document.getElementById("address").value.trim(); if (address === "") { document.getElementById("addressError").innerText = "Address is required!"; valid = false;

} else {

document.getElementById("addressError").innerText = "";

}

const password = document.getElementById("password").value.trim(); if (password.length < 6) { document.getElementById("passwordError").innerText = "Password must be at least 6 characters!"; valid = false;

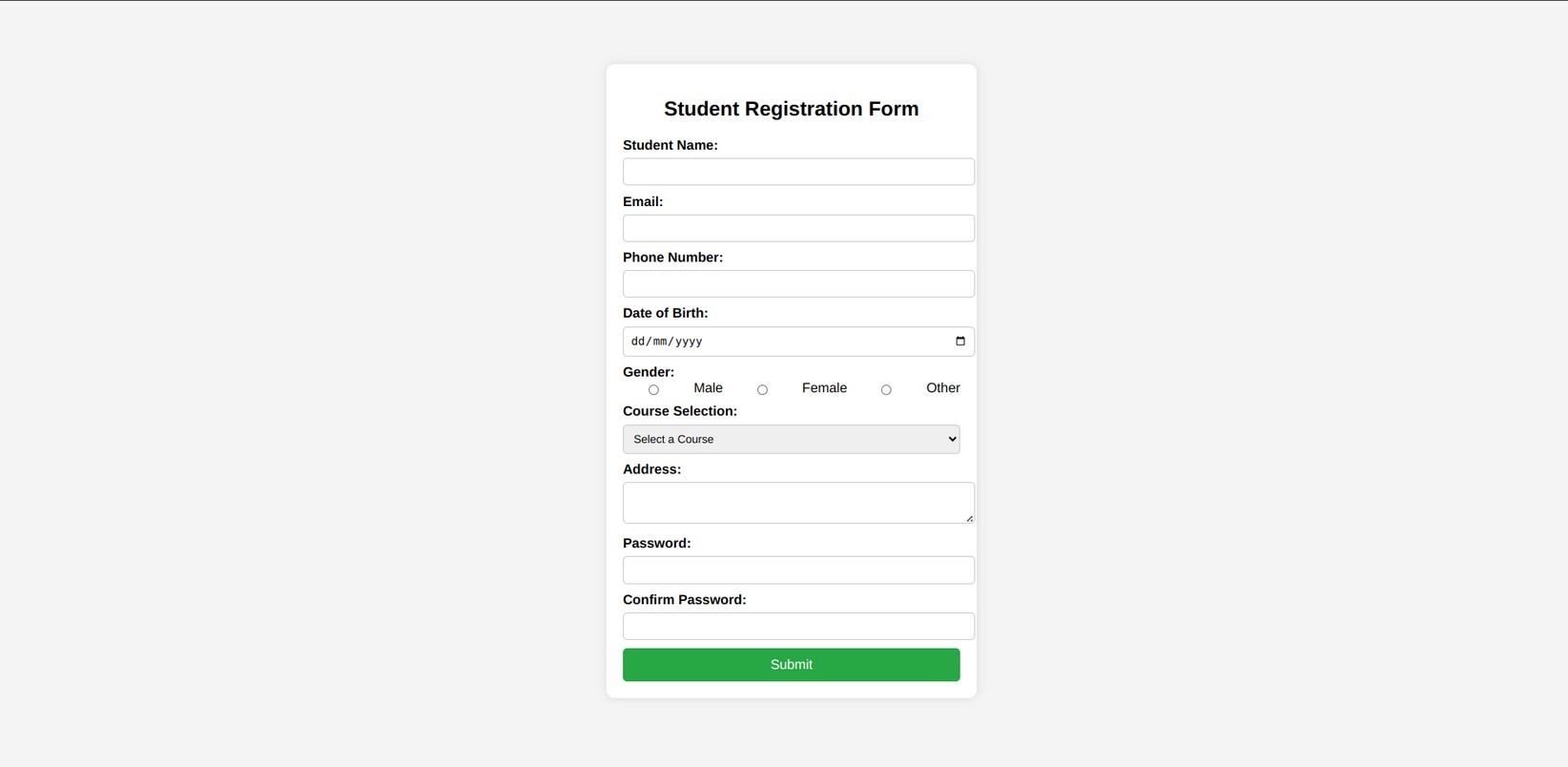
} else {

document.getElementById("passwordError").innerText = "";

}const confirmPassword = document.getElementById("confirmPassword").value.trim(); if (confirmPassword !== password) { document.getElementById("confirmPasswordError").innerText = "Passwords do not match!"; valid = false;

} else {

document.getElementById("confirmPasswordError").innerText = "";

} if (!valid) {

event.preventDefault();

}

});</script>

</body>

</html>

**TODO list code:**

**<!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: #f4f4f4; text-align: center; margin: 50px;

}

.container { max-width: 400px; background: white; padding: 20px; border-radius: 10px; box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); margin: auto;

} h2 { margin-bottom: 15px; color: #333;

}

input { width: 70%; padding: 10px; border: 1px solid #ccc; border-radius: 5px; font-size: 16px;

} button { background: #28a745; color: white; border: none; padding: 10px 15px; font-size: 16px; border-radius: 5px; cursor: pointer; margin-left: 5px;

}

button:hover { background: #218838;

} ul { list-style: none; padding: 0; margin-top: 20px;

}

li {

background: #fff; padding: 10px; border-radius: 5px; display: flex; justify-content: space-between; align-items: center; margin-bottom: 10px; box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1); transition: all 0.3s ease-in-out;

} li.fade-out { opacity: 0; transform: translateX(20px); transition: opacity 0.3s ease, transform 0.3s ease;

}

.remove-btn { background: red; color: white; border: none; padding: 5px 10px; cursor: pointer; border-radius: 5px; font-size: 14px;

}

.remove-btn:hover { background: darkred;

}

</style>

</head>

<body>

<div class="container">

<h2>To-Do List</h2>

<input type="text" id="task" placeholder="Enter Task">

<button onclick="addTask()">Add Task</button>

<ul id="taskList"></ul>

</div>

<script> document.addEventListener("DOMContentLoaded", loadTasks); document.getElementById("task").addEventListener("keypress", function(event) { if (event.key === "Enter") { addTask();

} });

function addTask() { let taskInput = document.getElementById("task"); let task = taskInput.value.trim(); if (task === "") { alert("Please enter a task"); return;

} let list = document.getElementById("taskList"); let li = document.createElement("li"); li.innerHTML = `${task} <button class="remove-btn" onclick='removeTask(this)'>Remove</button>`; list.appendChild(li);

saveTasks(); taskInput.value = ""; taskInput.focus();

}

function removeTask(element) { let li = element.parentNode; li.classList.add("fade-out"); setTimeout(() => { li.remove(); saveTasks();

}, 300); } function saveTasks() { let tasks = []; document.querySelectorAll("#taskList li").forEach(item => { tasks.push(item.firstChild.textContent);

});

localStorage.setItem("tasks", JSON.stringify(tasks));

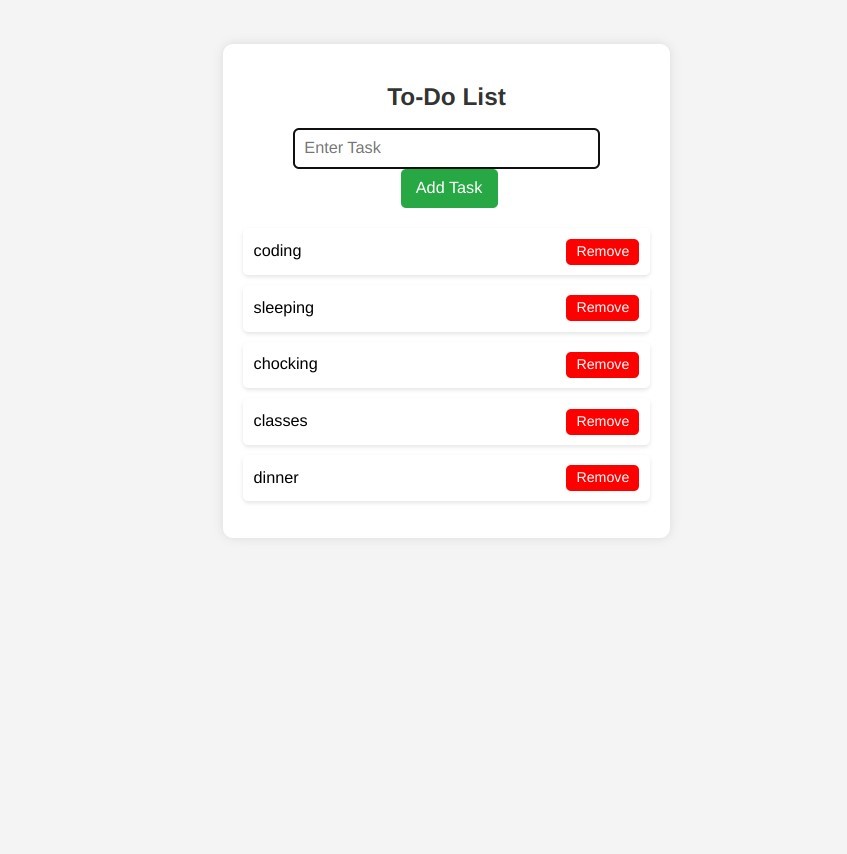
} function loadTasks() { let savedTasks = JSON.parse(localStorage.getItem("tasks")) || []; let list = document.getElementById("taskList"); savedTasks.forEach(task => { let li = document.createElement("li"); li.innerHTML = `${task} <button class="remove-btn" onclick='removeTask(this)'>Remove</button>`; list.appendChild(li);

});

}

</script> </body>

</html>



**Digital clock code:**

**<!DOCTYPE html>**

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Digital Clock</title>

<style> body { display: flex; justify-content: center; align-items: center; height: 100vh; background: #0e0e0e;

color: #fff; font-family: 'Arial', sans-serif; margin: 0;

}

.clock-container { text-align: center; background: rgba(255, 255, 255, 0.1); padding: 40px; border-radius: 15px; box-shadow: 0px 0px 20px rgba(0, 255, 255, 0.5); backdrop-filter: blur(10px);

}

.clock { font-size: 60px; font-weight: bold; letter-spacing: 2px; text-shadow: 0px 0px 10px rgba(0, 255, 255, 0.8); transition: all 0.2s ease-in-out;

}

.clock:hover { transform: scale(1.05);

} .date { font-size: 20px; margin-top: 10px; color: #ccc; text-shadow: 0px 0px 5px rgba(255, 255, 255, 0.5);

}

</style>

</head>

<body>

<div class="clock-container">

<div class="clock" id="clock">00:00:00</div>

<div class="date" id="date">Loading...</div>

</div> <script> function updateClock() { let now = new Date(); let hours = now.getHours().toString().padStart(2, '0'); let minutes = now.getMinutes().toString().padStart(2, '0'); let seconds = now.getSeconds().toString().padStart(2, '0'); let timeString = `${hours}:${minutes}:${seconds}`; document.getElementById("clock").innerText = timeString; let options = { weekday: 'long', month: 'long', day: 'numeric', year: 'numeric' }; document.getElementById("date").innerText = now.toLocaleDateString('en-US', options);

} setInterval(updateClock, 1000); updateClock();

</script>

</body>

</html>



Image slider & color changes code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Color Changer & Image Slider</title>

<style>

body { font-family: Arial, sans-serif; text-align: center; margin: 0; padding: 0; transition: background-color 0.5s ease-in-out;

} h1 { color: #333; margin-top: 20px;

}

.color-buttons { margin: 20px;

}

.color-btn { padding: 10px 20px; margin: 5px;

border: none; cursor: pointer; font-size: 16px; color: white; border-radius: 5px; transition: 0.3s; }

.color-btn:hover { opacity: 0.8; }

.slider-container { position: relative; width: 80%; max-width: 700px; margin: 30px auto; overflow: hidden; border-radius: 10px; box-shadow: 0px 4px 8px rgba(0, 0, 0, 0.2);

} .slider { display: flex; width: 400%; transition: transform 1s ease-in-out;

}

.slide {

width: 100%;

flex: 1 0 100%;

}

.slide img { width: 100%; height: 250px; object-fit: cover; border-radius: 10px;

}

.prev, .next { position: absolute; top: 50%; transform: translateY(-50%); background-color: rgba(0, 0, 0, 0.5); color: white; padding: 10px; border: none; cursor: pointer; border-radius: 50%; font-size: 20px; transition: 0.3s;

}

.prev:hover, .next:hover { background-color: black;

}

.prev { left: 10px; }

.next { right: 10px; }

</style>

</head>

<body>

<h1>Color Changer & Image Slider</h1>

<div class="color-buttons">

<button class="color-btn" style="background-color: #ff5733;" onclick="changeColor('#ff5733')">Red</button>

<button class="color-btn" style="background-color: #33ff57;" onclick="changeColor('#33ff57')">Green</button>

<button class="color-btn" style="background-color: #3357ff;" onclick="changeColor('#3357ff')">Blue</button>

<button class="color-btn" style="background-color: #f4c242;" onclick="changeColor('#f4c242')">Yellow</button>

<button class="color-btn" style="background-color: #8e44ad;" onclick="changeColor('#8e44ad')">Purple</button> </div>

<div class="slider-container">

<div class="slider" id="slider">

<div class="slide"><img src="https://gehu.ac.in/assets/images/GEHU-BTL-b80ef66e.webp" alt="GEHU Bhimtal"></div>

<div class="slide"><img src="https://lookaside.fbsbx.com/lookaside/crawler/media/? media\_id=880431600782829" alt="GEHU Bhimtal"></div>

<div class="slide"><img

src="https://image-static.collegedunia.com/public/college\_data/images/appImage/ 57212\_cover2.jpg?h=260&w=360&mode=crop" alt="GEHU Bhimtal"></div>

<div class="slide"><img src="https://lookaside.instagram.com/seo/google\_widget/crawler/?

media\_id=3295975471895630683" alt="GEHU Bhimtal"></div>

</div>

<button class="prev" onclick="prevSlide()">❮</button>

<button class="next" onclick="nextSlide()">❯</button>

</div> <script>

function changeColor(color) { document.body.style.backgroundColor = color;

}

let currentSlide = 0; const slides = document.querySelectorAll(".slide"); const totalSlides = slides.length; function updateSlider() { document.getElementById("slider").style.transform = `translateX(-${currentSlide \* 100}%)`;

} function nextSlide() { currentSlide = (currentSlide + 1) % totalSlides; updateSlider();

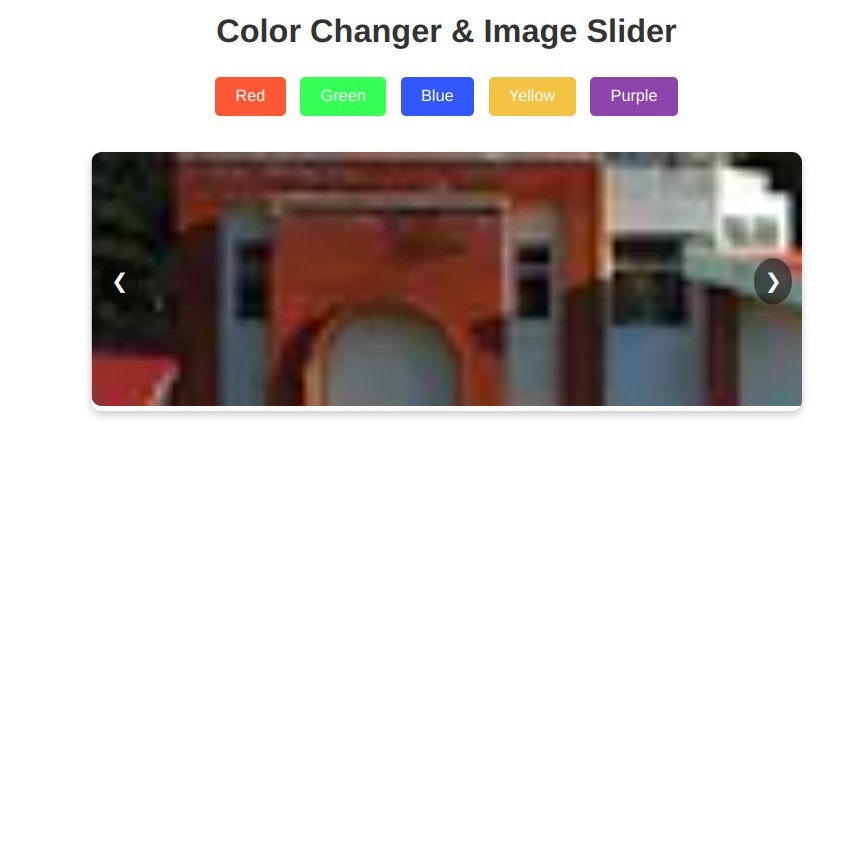
} function prevSlide() { currentSlide = (currentSlide - 1 + totalSlides) % totalSlides; updateSlider();

}

setInterval(nextSlide, 3000);

</script> </body>

</html>



3.Create an XML file to store information about students, books, or employees.

<?xml version="1.0" encoding="UTF-8"?>

<database>

<students>

<student id="S001">

<name>Ajay Rawat</name>

<age>25</age>

<gender>Male</gender>

<course>Computer Science</course>

<email>ajay@example.com</email>

<phone>9876543210</phone>

</student>

<student id="S002">

<name>Diya Bisht</name>

<age>22</age>

<gender>Female</gender>

<course>Data Science</course>

<email>diya@example.com</email>

<phone>9876543211</phone>

</student>

</students>

<!-- Book Information -->

<books>

<book id="B101">

<title>Introduction to Machine Learning</title>

<author>Tom Mitchell</author>

<publisher>McGraw-Hill</publisher>

<year>1997</year>

<isbn>978-0070428072</isbn>

</book>

<book id="B102">

<title>Artificial Intelligence: A Modern Approach</title>

<author>Stuart Russell, Peter Norvig</author>

<publisher>Pearson</publisher>

<year>2020</year>

<isbn>978-0134610993</isbn>

</book>

</books>

<employees>

<employee id="E001">

<name>Rahul Verma</name>

<position>Software Engineer</position>

<department>IT</department>

<email>rahul@example.com</email>

<phone>9876543212</phone>

<salary>90000</salary>

</employee>

<employee id="E002">

<name>Priya Singh</name>

<position>HR Manager</position> <department>Human Resources</department>

<email>priya@example.com</email>

<phone>9876543213</phone>

<salary>75000</salary>

</employee>

</employees>

</database>

