1. **Create the HTML5 Base Template Scenario:**

You're setting up the base document that every page on the portal will use.
Objective: Ensure semantic structure and compatibility across browsers.
Task:

- **Add comments to label sections like "Navigation", "Main", "Footer"**
- **Save as index.html and open it in Chrome**
- **Inspect the document structure in Chrome Dev Tools**

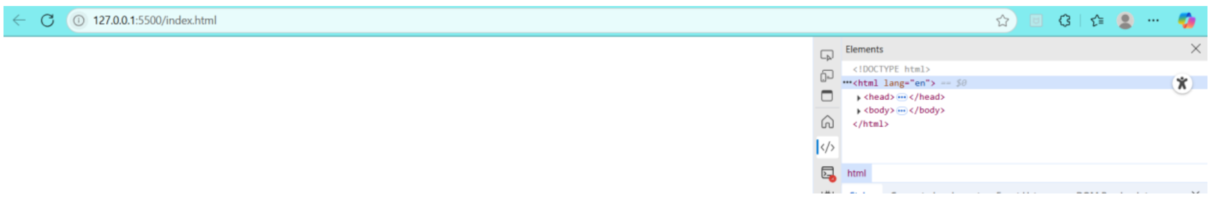
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Community Event Portal</title>
</head>
<body>

  <!-- Navigation -->
  <nav>
    <!-- Navigation links here -->
  </nav>

  <!-- Main -->
  <main>
    <!-- Page content -->
  </main>

  <!-- Footer -->
  <footer>
    <!-- Footer information -->
  </footer>

</body>
</html>
```



```
<!DOCTYPE html>
<html lang="en"> == $0
  <head> … </head>
  <body> … </body>
</html>
```

2. **<u>Navigation and Linking Scenario:</u>**

Users should navigate between "Home", "Events", and "Contact" sections.

Objective: Provide intuitive navigation and section-based references.

Task:

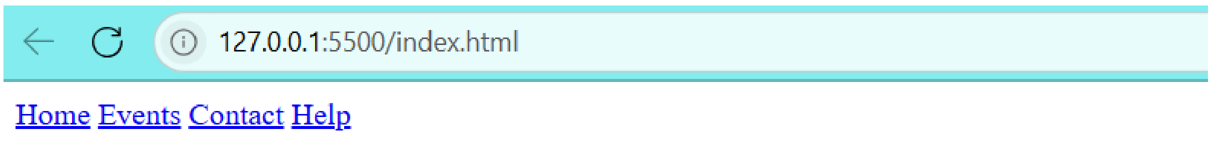
- **Use with anchor tags [Events]**
- **Define matching IDs for each section**
- **Add a link to an external help document**

```
<nav>
  <a href="#home">Home</a>
  <a href="#events">Events</a>
  <a href="#contact">Contact</a>
  <a href="help.html" target="_blank">Help</a>
</nav>

<section id="home">
  <!-- Home content -->
</section>

<section id="events">
  <!-- Events content -->
</section>

<section id="contact">
  <!-- Contact content -->
</section>
```



127.0.0.1:5500/index.html

Home Events Contact Help

3. **<u>Welcome Message with Styling and ID/Class Scenario:</u>**

Display a welcome banner styled uniquely for a logged-in user.

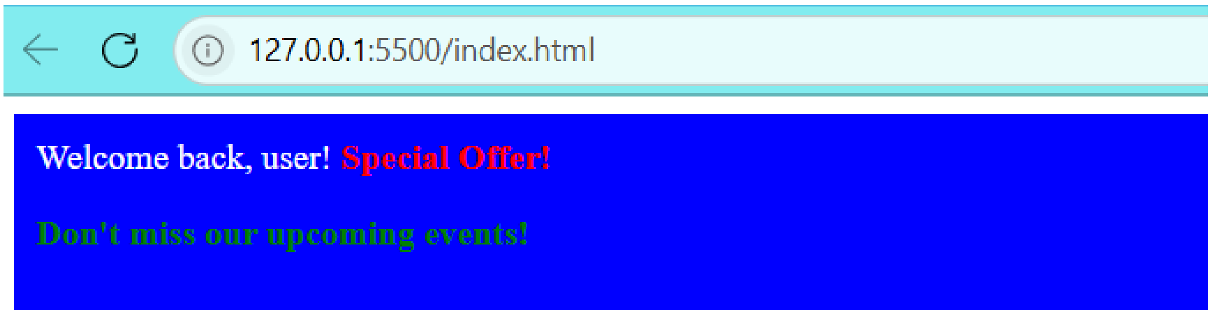
Objective: Practice block/inline tags and differentiate id and class

Task:

- **Use and apply a blue background via internal CSS**
- **Use inline styles for a special offer (e.g., color red, bold)**
- **Apply the .highlight class to certain elements for visual emphasis**

```
<style>
  #welcomeBanner {
    background-color: blue;
    color: white;
    padding: 10px;
  }
  .highlight {
    font-weight: bold;
    color: green;
  }
</style>

<div id="welcomeBanner">
  Welcome back, user! <span style="color: red; font-weight: bold;">Special
Offer!</span>
  <p class="highlight">Don't miss our upcoming events!</p>
</div>
```



Welcome back, user! **Special Offer!**

Don't miss our upcoming events!

4. <u>**Image Gallery for Community Events Scenario:**</u>

Show images from past events in a table layout. Objective: Work with Error! Filename not specified., tables, and formatting tags.
Task:

• Use a with 2 rows and 3 columns of Error! Filename not specified.tags

• Include alt, title, and style each image with borders using a class

• Add a caption to describe each event

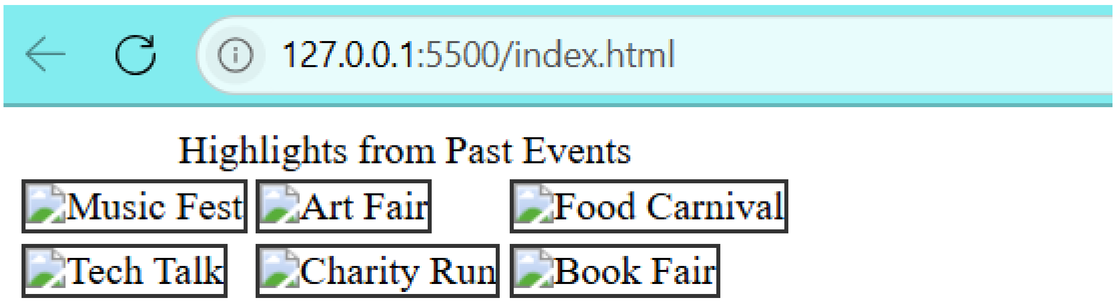
```
<table>

  <caption>Highlights from Past Events</caption>

  <tr>
```

```
          <td><img src="event1.jpg" alt="Music Fest" title="Music Fest" class="gallery-
img"></td>
          <td><img src="event2.jpg" alt="Art Fair" title="Art Fair" class="gallery-img"></td>
          <td><img src="event3.jpg" alt="Food Carnival" title="Food Carnival" class="gallery-
img"></td>
         </tr>
         <tr>
          <td><img src="event4.jpg" alt="Tech Talk" title="Tech Talk" class="gallery-
img"></td>
          <td><img src="event5.jpg" alt="Charity Run" title="Charity Run" class="gallery-
img"></td>
          <td><img src="event6.jpg" alt="Book Fair" title="Book Fair" class="gallery-
img"></td>
         </tr>
        </table>


        <style>
         .gallery-img {
          border: 2px solid #333;
          width: 150px;
          height: 100px;
         }
        </style>
```


127.0.0.1:5500/index.html

Highlights from Past Events

Music Fest Art Fair Food Carnival

Tech Talk Charity Run Book Fair

5. **<u>Event Registration Form Scenario:</u>**

Residents need to register for events.

Objective: Practice input types, validation, placeholder, autofocus, and output

Task:

• Include fields: name (text), email (email), date (date), event type (select), message (textarea)

• Add placeholder, required, and autofocus

• Display a confirmation message using when the form is submitted

• Style the form using CSS

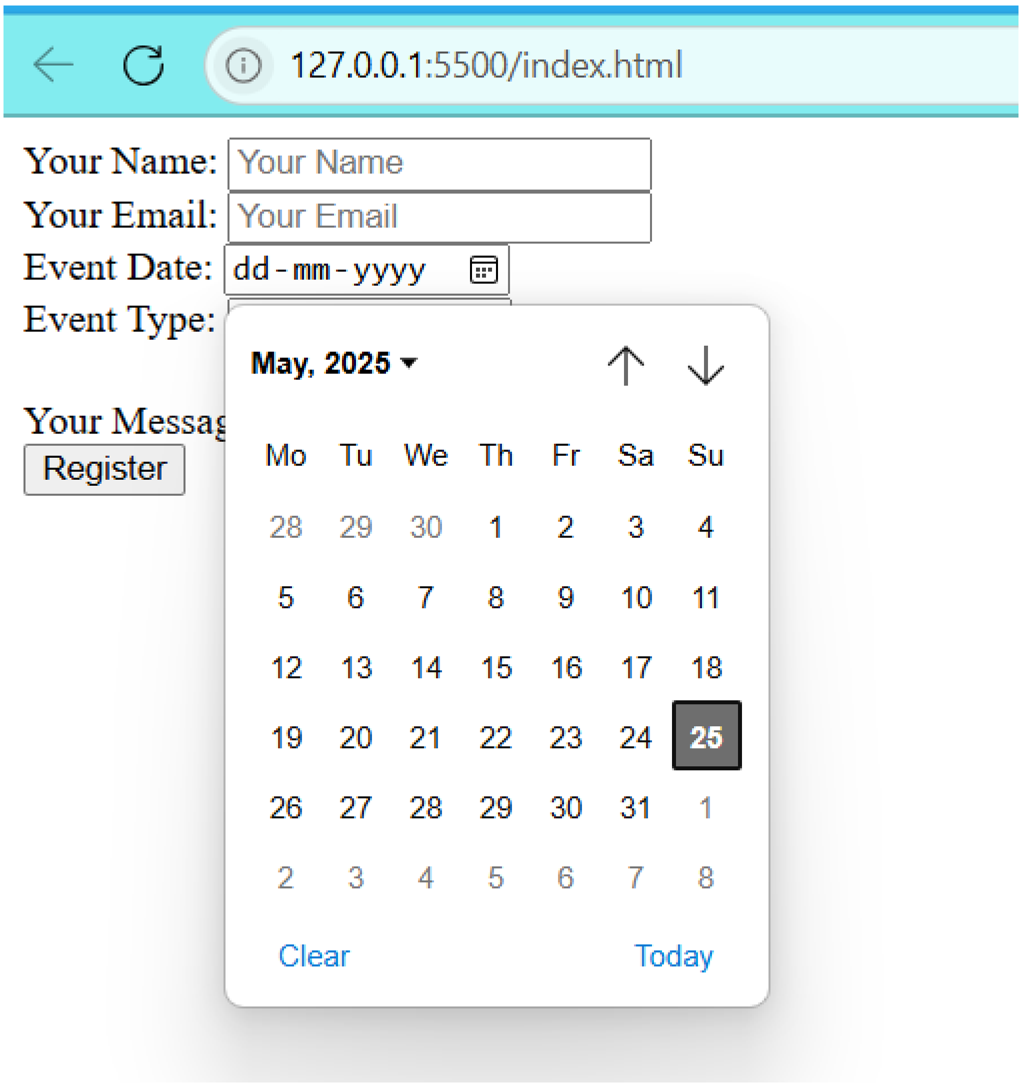
```
<body>
<form onsubmit="showConfirmation(); return false;">
  <label for="name">Your Name:</label>
  <input type="text" id="name" name="name" placeholder="Your Name" required
autofocus><br>
  <label for="email">Your Email:</label>
  <input type="email" id="email" name="email" placeholder="Your Email"
required><br>
  <label for="date">Event Date:</label>
  <input type="date" id="date" name="date" required><br>
  <label for="eventType">Event Type:</label>
  <select id="eventType" name="eventType" required>
    <option value="" disabled selected>Select an event</option>
    <option value="music">Music</option>
    <option value="art">Art</option>
  </select><br>
  <label for="message">Your Message:</label>
  <textarea id="message" name="message" placeholder="Your
message"></textarea><br>
```

```
  <input type="submit" value="Register"><br>
  <output id="confirmation"></output>
</form>

<script>
  function showConfirmation() {
    document.getElementById("confirmation").value = "Registration successful!";
  }
</script>



</body>
```



127.0.0.1:5500/index.html

Your Name: Your Name

Your Email: Your Email

Event Date: dd-mm-yyyy

Event Type:

Your Messag

Register

May, 2025

Mo	Tu	We	Th	Fr	Sa	Su
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Clear Today

6. **Event Feedback with Events Handling Scenario:**

Collect real-time feedback and interactions from the user.
Objective: Handle blur, change, click, double-click, and keyboard events.
Task:

- **Use onblur to validate a phone number field**
- **Use onchange on a dropdown to display the selected event fee**
- **onclick on a submit button to show a confirmation**
- **ondblclick on an image to enlarge it**
- **Capture key events in the feedback textarea and count characters**

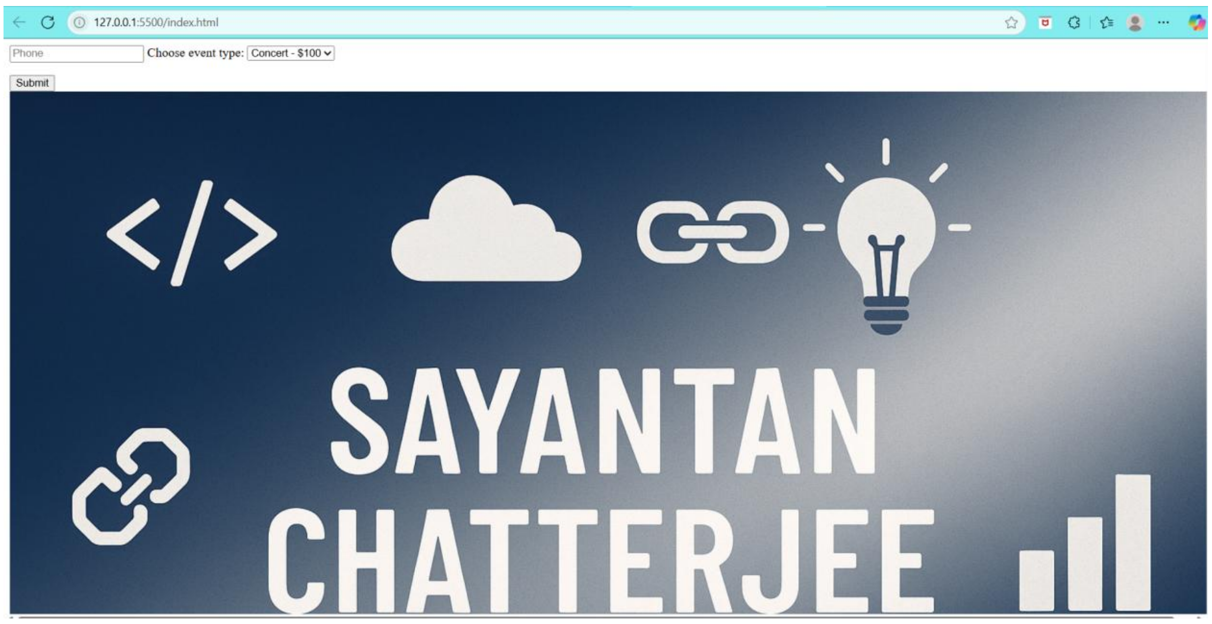
```
<input type="tel" placeholder="Phone" onblur="validatePhone(this)">
<label for="eventType">Choose event type:</label>
<select id="eventType" onchange="showFee(this.value)">
  <option value="100">Concert - $100</option>
  <option value="50">Art - $50</option>
</select>
<p id="feeDisplay"></p>

<button onclick="alert('Thank you for your feedback!')">Submit</button>

<img src="Sayantan.png" alt="Sayantan's photo"
ondblclick="this.style.width='400px';">

<textarea onkeyup="countChars(this)" placeholder="Enter your feedback
here"></textarea>
<p id="charCount">0 characters</p>

<script>
  function validatePhone(input) {
    if (!/^\d{10}$/.test(input.value)) {
      alert("Invalid phone number!");
    }
  }
  function showFee(fee) {
    document.getElementById("feeDisplay").innerText = "Fee: $" + fee;
  }
  function countChars(textarea) {
    document.getElementById("charCount").innerText = textarea.value.length + "
characters";
  }
</script>
```


127.0.0.1:5500/index.html

Phone Choose event type: Concert - $100

Submit

7. <u>Video Invite with Media Events Scenario:</u>
   Show a short event promo video.
   Objective: Work with and oncanplay event
   Task:
   • Insert a element with source and controls
   • Use oncanplay to display a message like "Video ready to play"
   • Use onbeforeunload to warn users if they try to leave the form page unfinished

```
<video width="320" height="240" controls oncanplay="readyMsg()">
  <source src="invite.mp4" type="video/mp4">
  Your browser does not support video.
</video>

<p id="videoStatus"></p>

<script>
  function readyMsg() {
    document.getElementById("videoStatus").innerText = "Video ready to play!";
  }

  window.onbeforeunload = function () {
    return "Are you sure you want to leave without submitting the form?";
  }
</script>
```

8. <u>**Saving User Preferences Scenario:**</u>
Store preferred event type for returning users.
Objective: Work with localStorage, sessionStorage, and deletion
Task:
• Save selected event type in localStorage
• On reload, retrieve and pre-select it
• Add a "Clear Preferences" button that clears both localStorage and sessionStorage

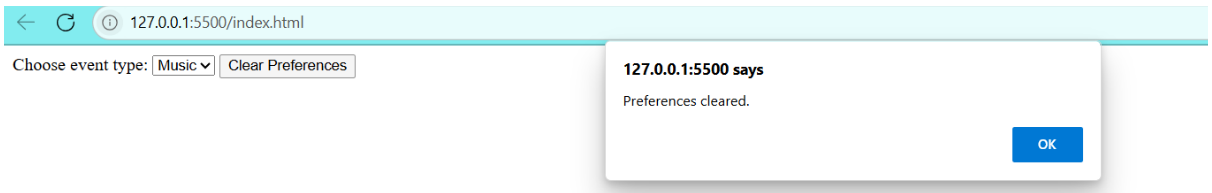
```
<body>
<label for="eventType">Choose event type:</label>
<select id="eventType" onchange="savePreference()">
  <option value="music">Music</option>
  <option value="art">Art</option>
</select>
<button onclick="clearPreferences()">Clear Preferences</button>

<script>
 window.onload = function() {
  let savedType = localStorage.getItem("preferredEvent");
  if (savedType) {
   document.getElementById("eventType").value = savedType;
  }
 };

 function savePreference() {
  const value = document.getElementById("eventType").value;
  localStorage.setItem("preferredEvent", value);
  sessionStorage.setItem("currentSelection", value);
 }

 function clearPreferences() {
  localStorage.clear();
  sessionStorage.clear();
  alert("Preferences cleared.");
 }
</script>
```

```
</body>
```



127.0.0.1:5500/index.html

Choose event type: Music Clear Preferences

127.0.0.1:5500 says

Preferences cleared.

OK

9. **<u>Geolocation for Event Mapping Scenario:</u>**

Locate the nearest event to the user.

Objective: Practice geolocation.getCurrentPosition, error handling, and options

Task:

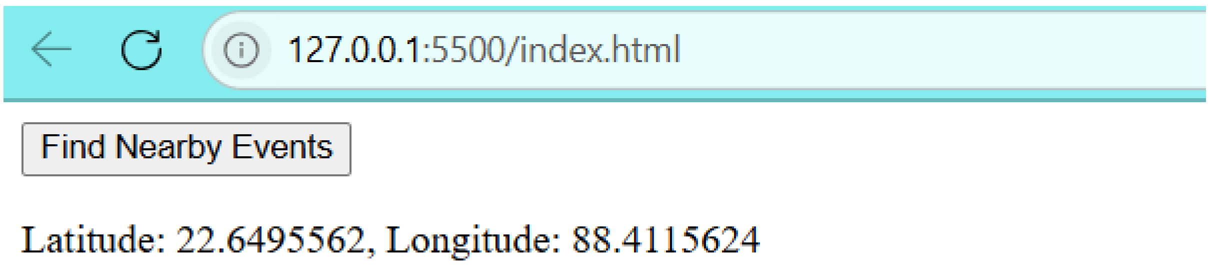
- **Create a button "Find Nearby Events"**
- **On click, use getCurrentPosition to get and display coordinates**
- **Handle permission denial and timeouts**
- **Use high accuracy options**

```
<button onclick="findEvents()">Find Nearby Events</button>
<p id="location"></p>

<script>
 function findEvents() {
  navigator.geolocation.getCurrentPosition(showPos, showError, {
   enableHighAccuracy: true,
   timeout: 10000
  });
 }

 function showPos(pos) {
  document.getElementById("location").innerText =
   "Latitude: " + pos.coords.latitude + ", Longitude: " + pos.coords.longitude;
 }

 function showError(error) {
  document.getElementById("location").innerText = "Location access denied or
unavailable.";
 }
</script>
```



```
Latitude: 22.6495562, Longitude: 88.4115624
```