NAME – SAMEER KHATWANI

PRN – 22070126099

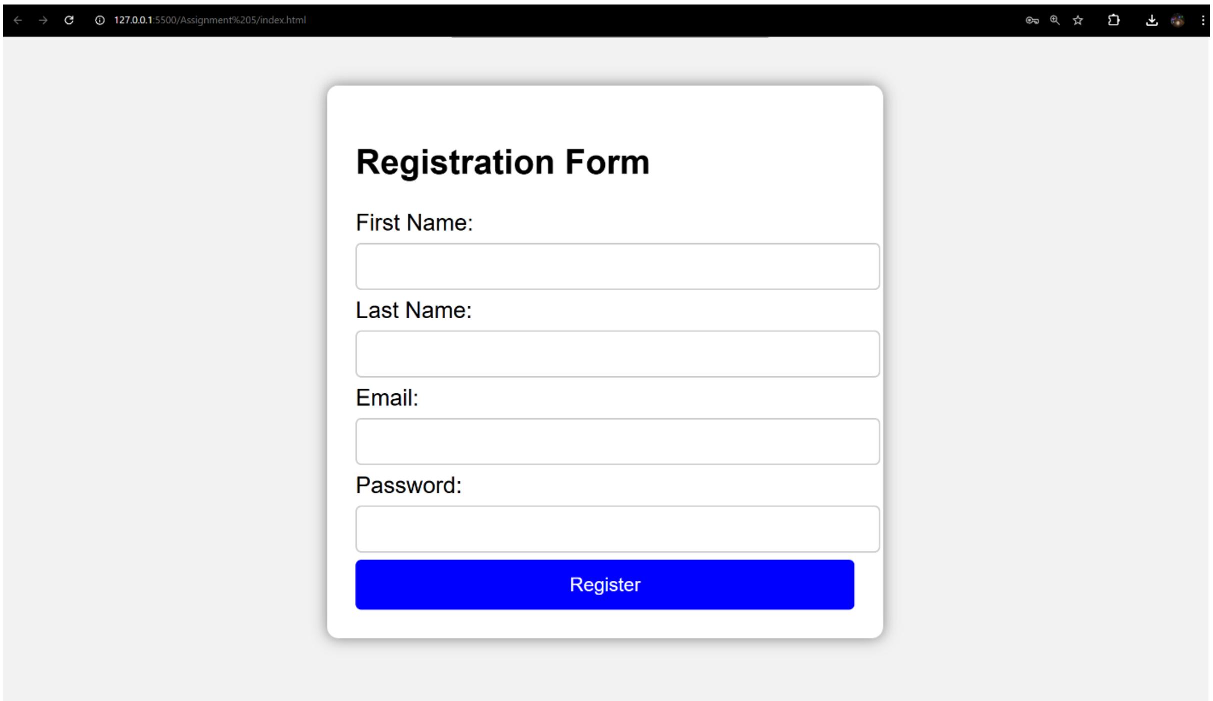
BATCH – AIML-B1 (FS3)

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f2f2f2;
        }
        .container {
            background: white;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0px 0px 10px gray;
            width: 350px;
        }
        .error {
            color: red;
            font-size: 12px;
        }
        input {
            width: 100%;
            padding: 8px;
            margin: 5px 0;
            border: 1px solid #ccc;
            border-radius: 4px;
        }
        .button {
            width: 100%;
            padding: 10px;
            background: blue;
            color: white;
            border: none;
            border-radius: 4px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <div class="container">
    <h2>Registration Form</h2>
    <form id="registrationForm">
      <label>First Name:</label>
      <input type="text" id="firstName">
      <span class="error" id="firstNameError"></span>

      <label>Last Name:</label>
      <input type="text" id="lastName">
      <span class="error" id="lastNameError"></span>

      <label>Email:</label>
      <input type="email" id="email">
      <span class="error" id="emailError"></span>

      <label>Password:</label>
      <input type="password" id="password">
      <span class="error" id="passwordError"></span>

      <button type="submit" class="button">Register</button>
    </form>
  </div>

  <script>
    document.getElementById("registrationForm").addEventListener("submit", function(event) {
      event.preventDefault();
      let isValid = true;

      let firstName = document.getElementById("firstName").value.trim();
      let lastName = document.getElementById("lastName").value.trim();
      let email = document.getElementById("email").value.trim();
      let password = document.getElementById("password").value.trim();

      document.getElementById("firstNameError").innerText = "";
      document.getElementById("lastNameError").innerText = "";
      document.getElementById("emailError").innerText = "";
      document.getElementById("passwordError").innerText = "";

      if (firstName === "") {
        document.getElementById("firstNameError").innerText = "First name is required";
        isValid = false;
      }
      if (lastName === "") {
        document.getElementById("lastNameError").innerText = "Last name is required";
        isValid = false;
      }
      if (email === "" || !email.includes("@")) {
        document.getElementById("emailError").innerText = "Valid email is required";
        isValid = false;
      }
      if (password.length < 8) {
        document.getElementById("passwordError").innerText = "Password must be at least 8 characters long";
        isValid = false;
      }
```

```
        if (isValid) {
          alert("Form submitted successfully!");
        }
      });
   </script>
</body>
</html>
```

OUTPUT



127.0.0.1:5500/Assignment%205/index.html

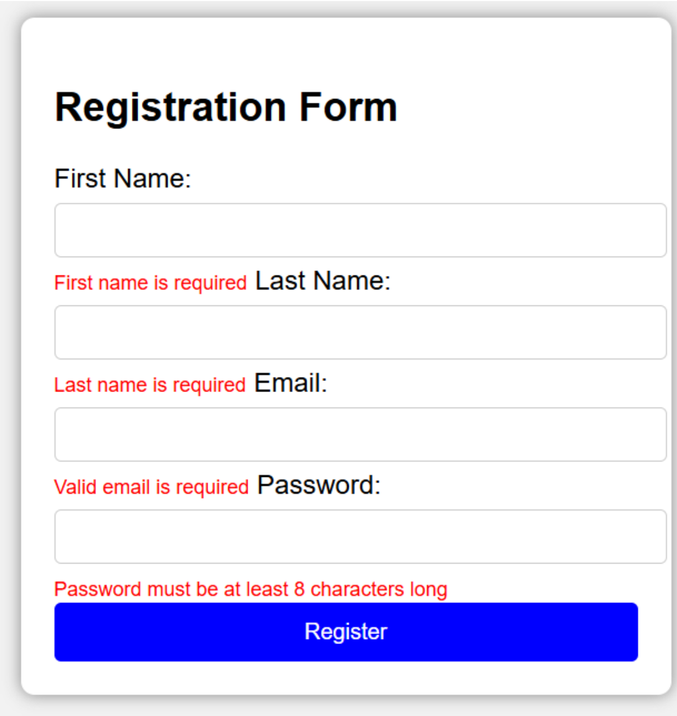
Registration Form

First Name:

Last Name:

Email:

Password:



Registration Form

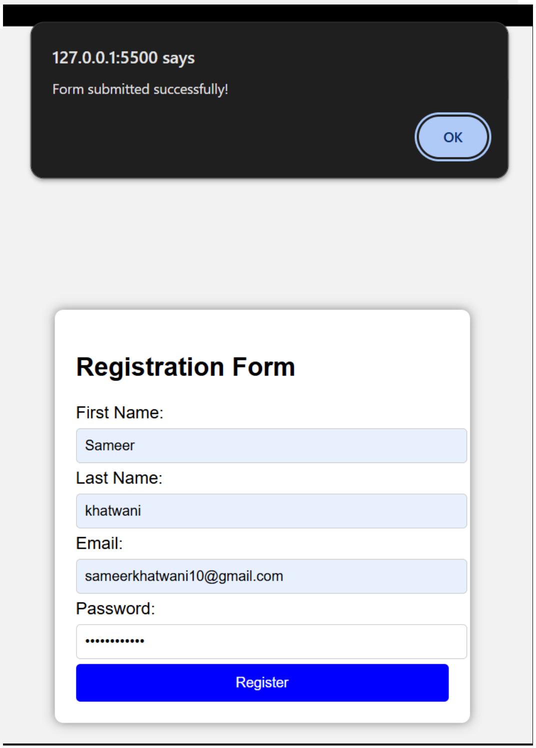
First Name:

First name is required Last Name:

Last name is required Email:

Valid email is required Password:

Password must be at least 8 characters long



127.0.0.1:5500 says

Form submitted successfully!

Registration Form

First Name: Sameer

Last Name: khatwani

Email: sameerkhatwani10@gmail.com

Password: ••••••••••••

Documentation:

Overview

This registration form is designed for user sign-up on a website. It includes fields for First Name, Last Name, Email, and Password. The form ensures that all fields are correctly filled before submission using JavaScript validation. Users receive real-time feedback on errors, and the form prevents submission if any field contains invalid data.

Instructions for Use

1. **Enter your First Name and Last Name** – These fields are required and cannot be left empty.
2. **Enter your Email** – The email must follow a valid format (e.g., user@example.com).
3. **Enter your Password** – The password must be at least **8 characters long**.
4. **Click the Register button** – If all inputs are valid, the form submits successfully. If there are errors, they are displayed in red next to the respective fields.

Features and Enhancements

- **Live validation** – Errors appear dynamically when the user enters invalid data.
- **Styled error messages** – Red-colored error text to indicate issues.
- **Prevents invalid submission** – The form cannot be submitted unless all fields meet the required criteria.
- **User-friendly interface** – The form is styled with CSS for better readability.

Explanation of Tags and Code Used

HTML Elements

- <form> – Creates the form container.
- <input> – Defines text fields for user input.
- <label> – Describes each input field.
- <button> – Submits the form.

CSS Styling

- border, padding, margin – Used for spacing and styling form elements.
- color: red; – Used to highlight error messages.
- box-shadow – Enhances the appearance of the form.

JavaScript Logic

- addEventListener('submit', function(e) {...}) – Prevents form submission if validation fails.
- document.getElementById('field').value – Retrieves user input.
- if (condition) { showError(); } – Checks input validity and displays an error message.

Testing Process

- **Empty Form Submission** – Ensured that required fields trigger error messages.

- **Invalid Email Format** – Tested with missing @ and incorrect domain formats.
- **Short Passwords** – Verified that passwords less than 8 characters were rejected.
- **Successful Submission** – Confirmed that a fully completed form submits without errors.

Challenges and Future Improvements

Challenges Faced

- Ensuring that the error messages dynamically disappear when corrected.
- Aligning form elements properly with CSS.

Possible Enhancements

- Add real-time password strength validation.
- Implement a "Show Password" toggle.
- Improve mobile responsiveness.