Experiment: Online Job Registration Page with JavaScript Validation

Course / Subject: Web Technologies / Front-end Development
Experiment No.: 02
Date: __________

Aim

To design and develop an online job registration web page using HTML and apply JavaScript form validation to ensure correct and complete user input before submission.

Apparatus / Software Required

- Text Editor (VS Code / Notepad++ / Sublime / Atom)
- Web Browser (Chrome / Firefox / Edge)
- Basic knowledge of HTML, CSS, and JavaScript

Theory / Background

Web forms are essential for collecting user information such as job applications. Validation ensures that the data entered by the user is accurate, complete, and secure. JavaScript provides client-side validation to prevent incorrect form submission.

Types of JavaScript Validation: 1. Required field check 2. Email format validation 3. Mobile number length check 4. Password strength/confirmation 5. Age validation 6. Dropdown and radio button validation

---

Procedure

1. Create a project folder named `job-registration`.
2. Inside the folder, create a file named `job_form.html`.
3. Add HTML form elements such as text fields, email field, password, gender, qualifications, resume upload, etc.
4. Write JavaScript validation functions and link them inside the `<script>` tag.
5. Apply CSS to make the form visually appealing.
6. Open the HTML file in the browser and test the validations.

Code: Online Job Registration Form with JavaScript Validation

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Online Job Registration</title>

  <style>
    body {
      font-family: Arial, sans-serif;
      background: #f3f5f7;
      padding: 20px;
    }
    .container {
      width: 450px;
      margin: auto;
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 0 10px rgba(0,0,0,0.1);
    }
    h2 {
      text-align: center;
      color: #333;
    }
    input, select {
      width: 100%;
      padding: 10px;
      margin: 8px 0;
      border: 1px solid #ccc;
      border-radius: 5px;
    }
    button {
      width: 100%;
      padding: 10px;
      background: #4CAF50;
      border: none;
      color: white;
      font-size: 16px;
      border-radius: 5px;
      cursor: pointer;
    }
    button:hover {
      background: #45a049;
    }
    .error {
      color: red;
      font-size: 14px;
    }
  </style>
</head>

<body>
  <div class="container">
    <h2>Online Job Registration</h2>
    <form onsubmit="return validateForm()">

      <label>Full Name:</label>
      <input type="text" id="name">

      <label>Email:</label>
      <input type="text" id="email">

      <label>Mobile Number:</label>
      <input type="text" id="mobile">

      <label>Password:</label>
      <input type="password" id="password">

      <label>Confirm Password:</label>
      <input type="password" id="cpassword">

      <label>Gender:</label>
      <select id="gender">
        <option value="">Select</option>
        <option value="Male">Male</option>
        <option value="Female">Female</option>
        <option value="Other">Other</option>
      </select>

      <label>Qualification:</label>
      <input type="text" id="qualification">

      <label>Upload Resume:</label>
      <input type="file" id="resume">

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

    <p id="error" class="error"></p>
  </div>


  <script>
    function validateForm() {

      let name = document.getElementById("name").value;
      let email = document.getElementById("email").value;
      let mobile = document.getElementById("mobile").value;
      let password = document.getElementById("password").value;
      let cpassword = document.getElementById("cpassword").value;
      let gender = document.getElementById("gender").value;
      let qualification = document.getElementById("qualification").value;
      let resume = document.getElementById("resume").value;
      let error = document.getElementById("error");

      error.innerHTML = "";

      if (name === "") {
        error.innerHTML = "Please enter your name";
        return false;
      }

      const emailPattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/;
      if (!email.match(emailPattern)) {
        error.innerHTML = "Enter a valid email address";
        return false;
      }

      if (mobile.length !== 10 || isNaN(mobile)) {
        error.innerHTML = "Mobile number must be 10 digits";
        return false;
      }

      if (password.length < 6) {
        error.innerHTML = "Password must be at least 6 characters";
        return false;
      }

      if (password !== cpassword) {
        error.innerHTML = "Passwords do not match";
        return false;
      }

      if (gender === "") {
        error.innerHTML = "Please select your gender";
        return false;
      }

      if (qualification === "") {
        error.innerHTML = "Enter your qualification";
        return false;
      }

      if (resume === "") {
        error.innerHTML = "Upload your resume";
```

```
        return false;
      }

      alert("Registration Successful!");
      return true;
    }
  </script>

</body>
</html>
```

Observations

1. The form prevents submission when required fields are empty.
2. Email validation ensures the correct format.
3. JavaScript blocks invalid inputs and displays the error message.
4. Password and confirm password must match.
5. Dropdown and file upload validations work correctly.

Result

An online job registration page was successfully created using HTML and styled with CSS. JavaScript validation ensures that users enter valid and complete information before submitting the form.

Conclusion

This experiment demonstrates how client-side validation improves user experience and prevents incorrect data submission. JavaScript plays a crucial role in ensuring that forms are validated before being sent to a server.

Viva Questions

1. What is the purpose of JavaScript validation?
2. Why is client-side validation important?
3. Difference between client-side and server-side validation?
4. What are regular expressions used for?
5. What happens if JavaScript is disabled in a browser?

End of Experiment Document