|  |  |
| --- | --- |
| **Exp No: 2** | **REGISTRATION FORM** |

**AIM:**

To write a JavaScript to validate the following fields of a registration page   
a) First Name ( Name should contain alphabets and the length should not be less than 6 characters

b)Password (Password should not be less than 6 characters length)

c)Email id ( should not contain any invlaid and must follow the standard pattern [name@domain.com](mailto:name@domain.com))

d)Mobile Number(phone number should contain 10 digits only )

e)Last Name and Address ( should not be Empty)

**PROGRAM:**

<html>

<head>

<title>Registration Form</title>

<style>

.error-message {

color: red;

font-size: 14px;

margin-top: 5px;

}

input{

padding: 5px;

border-radius: 6px;

display: block;

}

</style>

</head>

<body style="display:flex; flex-direction: column; justify-content: center; align-items: center;">

<h1>REGISTRATION FORM</h1>

<form name="RegForm" onsubmit="return validateForm()">

<p>

<label for="firstname">First Name:</label>

<input type="text" id="firstname" name="firstname" placeholder="Enter First Name" required/>

<span id="firstname-error" class="error-message"></span>

</p>

<p>

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

<input type="password" id="password" name="password" placeholder="Enter Password" required/>

<span id="password-error" class="error-message"></span>

</p>

<p>

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

<input type="text" id="email" name="email" placeholder="Enter Email" required/>

<span id="email-error" class="error-message"></span>

</p>

<p>

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

<input type="tel" id="phone" name="phone" placeholder="Enter Phone Number" required/>

<span id="phone-error" class="error-message"></span>

</p>

<p>

<label for="lastname">Last Name:</label>

<input type="text" id="lastname" name="lastname" placeholder="Enter Last Name" required/>

<span id="lastname-error" class="error-message"></span>

</p>

<p>

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

<textarea id="address" name="address" placeholder="Enter Address" required></textarea>

<span id="address-error" class="error-message"></span>

</p>

<p>

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

</p>

</form>

<script>

function validateForm() {

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

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

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

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

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

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

const firstnameError = document.getElementById("firstname-error");

const passwordError = document.getElementById("password-error");

const emailError = document.getElementById("email-error");

const phoneError = document.getElementById("phone-error");

const lastnameError = document.getElementById("lastname-error");

const addressError = document.getElementById("address-error");

firstnameError.textContent = "";

passwordError.textContent = "";

emailError.textContent = "";

phoneError.textContent = "";

lastnameError.textContent = "";

addressError.textContent = "";

let isvalid = true;

if(/[^a-zA-Z]/.test(firstname))

{

firstnameError.textContent="First Name only contains alphabet";

isvalid = false;

}

else if(firstname.length <= 6){

firstnameError.textContent="First Name length not less than 6 characters";

isvalid = false;

}

if (password.length !=8) {

passwordError.textContent = "Password must be between 8 and 30 characters.";

isvalid = false;

}

const emailPattern = /^[a-zA-Z0-9.\_-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;

if (!emailPattern.test(email)) {

emailError.textContent = "Please enter a valid email address.";

isvalid = false;

}

if(phone.length!=10){

phoneError.textContent = "Phone number length must be 10";

isvalid = false;

}

if(isvalid){

alert("Register successfully");

return true;

}

else{

return false;

}

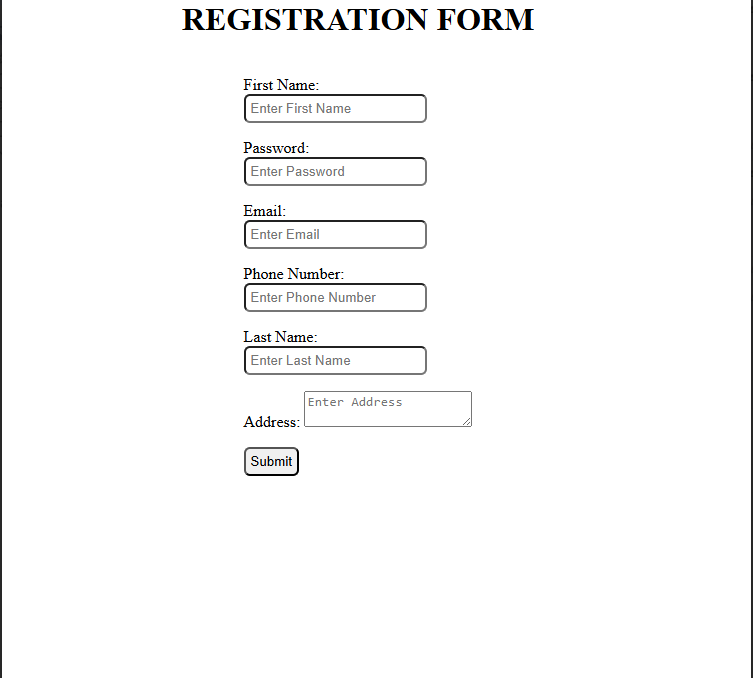
}

</script>

</body>

</html>

**OUTPUT:**

****

**RESULT :**Thus the experiment to create a web page that embeds a map with hotspots, frames, and links using embedded, external, and inline CSS has been successfully implemented and verified .