

EXP 2:	a) Write JavaScript to validate the following fields of the Registration page.
Date:	b) First Name (Name should contains alphabets and the length Should not be less than 6 characters).
	c) Password (Password should not be less than 6 characters length).
	d) E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
	e) Mobile Number (Phone number should contain 10 digits only).
	f) Last Name and Address (should not be Empty).

AIM :

This project aims to design a registration form with robust validation for fields like First Name, Password, Email, Mobile Number, Last Name, and Address, using JavaScript for client-side validation and HTML/CSS for structure and styling.

ALGORITHM :

Start

1. **Setup Registration Form:**
 - Create an HTML form with fields: **First Name, Last Name, Password, Email, Mobile Number, and Address.**
 - Add an event listener to validate the form on submission.
2. **Capture User Input:**
 - Retrieve values entered in the form fields using JavaScript (`document.getElementById`).
3. **Validate First Name:**
 - Ensure the first name contains only alphabets using a **regular expression** (`/^[A-Za-z]+$`).
 - Ensure the **length is at least 6 characters.**
 - If validation fails, **show an alert message** and prevent form submission.
4. **Validate Password:**
 - Ensure the **password has at least 6 characters.**
 - If validation fails, **show an alert message** and prevent form submission.
5. **Validate Email:**
 - Ensure the email follows the **standard format** using a **regular expression** (`/^[a-zA-Z0-9._-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,6}$/`).
 - If validation fails, **show an alert message** and prevent form submission.
6. **Validate Mobile Number:**
 - Ensure the mobile number contains **exactly 10 digits** using a **regular expression** (`/^[0-9]{10}$/`).
 - If validation fails, **show an alert message** and prevent form submission.
7. **Validate Last Name:**
 - Ensure the last name **is not empty.**
 - If validation fails, **show an alert message** and prevent form submission.
8. **Validate Address:**
 - Ensure the address **is not empty.**
 - If validation fails, **show an alert message** and prevent form submission.
9. **Submit Form:**
 - If **all validations pass**, display **"Registration Successful!"** and allow form submission.

CODE :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Colorful Registration Form</title>
</style>
  body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background: linear-gradient(135deg, #f5f7fa 0%, #c3cfe2 100%);
    padding: 30px;
  }

  .form-container {
    max-width: 600px;
    margin: 0 auto;
    background: white;
    padding: 30px;
    border-radius: 15px;
    box-shadow: 0 10px 30px rgba(0,0,0,0.1);
  }

  h1 {
    text-align: center;
    color: #6a11cb;
    margin-bottom: 30px;
    font-size: 2.2em;
    text-shadow: 1px 1px 2px rgba(0,0,0,0.1);
  }

  .form-group {
    margin-bottom: 20px;
  }

  label {
    display: block;
    margin-bottom: 8px;
    font-weight: 600;
    color: #555;
  }

  input, textarea {
    width: 100%;
    padding: 12px;
    border: 2px solid #ddd;
    border-radius: 8px;
    font-size: 16px;
    transition: all 0.3s;
  }
  input:focus, textarea:focus {
    border-color: #6a11cb;
```

```

outline: none;
  box-shadow: 0 0 0 3px rgba(106,17,203,0.2);
}

.error {
  color: #ff3860;
  font-size: 14px;
  margin-top: 5px;
  display: none;
}

button {
  background: linear-gradient(to right, #6a11cb 0%, #2575fc 100%);
  color: white;
  border: none;
  padding: 14px 20px;
  border-radius: 8px;
  cursor: pointer;
  font-size: 16px;
  font-weight: 600;
  width: 100%;
  transition: all 0.3s;
}

button:hover {
  transform: translateY(-2px);
  box-shadow: 0 5px 15px rgba(106,17,203,0.3);
}

.success-icon {
  color: #2ecc71;
  display: none;
  margin-left: 10px;
}

.input-wrapper {
  position: relative;
}

.input-wrapper.success input,
.input-wrapper.success textarea {
  border-color: #2ecc71;
}

.input-wrapper.error input,
.input-wrapper.error textarea {
  border-color: #ff3860;
}
</style>
</head>

<body>
  <div class="form-container">

```

```
<h1> Register Now</h1>
<form id="registrationForm">
  <div class="form-group">
    <label for="firstName">First Name</label>
    <div class="input-wrapper">
      <input type="text" id="firstName" placeholder="At least 6 alphabets">
      <span class="success-icon">✓</span>
    </div>
    <div class="error" id="firstNameError"></div>
  </div>

  <div class="form-group">
    <label for="lastName">Last Name</label>
    <div class="input-wrapper">
      <input type="text" id="lastName" placeholder="Your last name">
      <span class="success-icon">✓</span>
    </div>
    <div class="error" id="lastNameError"></div>
  </div>

  <div class="form-group">
    <label for="email">Email Address</label>
    <div class="input-wrapper">
      <input type="email" id="email" placeholder="name@example.com">
      <span class="success-icon">✓</span>
    </div>
    <div class="error" id="emailError"></div>
  </div>

  <div class="form-group">
    <label for="password">Password</label>
    <div class="input-wrapper">
      <input type="password" id="password" placeholder="At least 6 characters">
      <span class="success-icon">✓</span>
    </div>
    <div class="error" id="passwordError"></div>
  </div>

  <div class="form-group">
    <label for="mobile">Mobile Number</label>
    <div class="input-wrapper">
      <input type="tel" id="mobile" placeholder="10 digits only">
      <span class="success-icon">✓</span>
    </div>
    <div class="error" id="mobileError"></div>
  </div>

  <div class="form-group">
    <label for="address">Address</label>
    <div class="input-wrapper">
      <textarea id="address" rows="3" placeholder="Your full address"></textarea>
      <span class="success-icon">✓</span>
    </div>
  </div>
</form>
```

```

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

    <button type="submit">Create Account</button>
</form>
</div>

<script>
    document.addEventListener('DOMContentLoaded', function() {
        const form = document.getElementById('registrationForm');

        // Real-time validation as user types
        document.querySelectorAll('input, textarea').forEach(input => {
            input.addEventListener('input', function() {
                validateField(this.id);
            });
        });

        // Form submission
        form.addEventListener('submit', function(event) {
            event.preventDefault();
            if (validateForm()) {
                // Form is valid - show success message
                alert('☑ Registration successful! Welcome aboard!');
                form.reset();
                document.querySelectorAll('.success-icon').forEach(icon => {
                    icon.style.display = 'none';
                });
            }
        });

        function validateField(fieldId) {
            const field = document.getElementById(fieldId);
            const value = field.value.trim();
            const wrapper = field.parentElement;
            const successIcon = wrapper.querySelector('.success-icon');
            const errorElement = document.getElementById(`${fieldId}Error`);

            // Reset state
            wrapper.classList.remove('success', 'error');
            errorElement.style.display = 'none';
            successIcon.style.display = 'none';

            // Skip validation if empty (except for showing required errors on submit)
            if (value === "") return true;

            let isValid = true;
            let errorMessage = "";

            switch(fieldId) {
                case 'firstName':

```

```

        isValid = /^[a-zA-Z]{6,}$/.test(value);
        errorMessage = 'Must contain only alphabets and be at least 6 characters long';
        break;
    case 'lastName':
        isValid = value.length > 0;
        errorMessage = 'Last name is required';
        break;
    case 'email':
        isValid = /^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(value);
        errorMessage = 'Please enter a valid email (e.g., name@example.com)';
        break;
    case 'password':
        isValid = value.length >= 6;
        errorMessage = 'Password must be at least 6 characters long';
        break;
    case 'mobile':
        isValid = /^\d{10}$/.test(value);
        errorMessage = 'Mobile number must contain exactly 10 digits';
        break;
    case 'address':
        isValid = value.length > 0;
        errorMessage = 'Address is required';
        break;
    }

    if (isValid) {
        wrapper.classList.add('success');
        successIcon.style.display = 'inline';
    } else {
        wrapper.classList.add('error');
        errorElement.textContent = errorMessage;
        errorElement.style.display = 'block';
    }

    return isValid;
}

function validateForm() {
    let isValid = true;

    // Validate all fields
    ['firstName', 'lastName', 'email', 'password', 'mobile', 'address'].forEach(fieldId => {
        const field = document.getElementById(fieldId);
        const value = field.value.trim();
        const wrapper = field.parentElement;
        const errorElement = document.getElementById(`${fieldId}Error`);

        // Reset state
        wrapper.classList.remove('success', 'error');
        errorElement.style.display = 'none';
        if (value === "") {
            wrapper.classList.add('error');

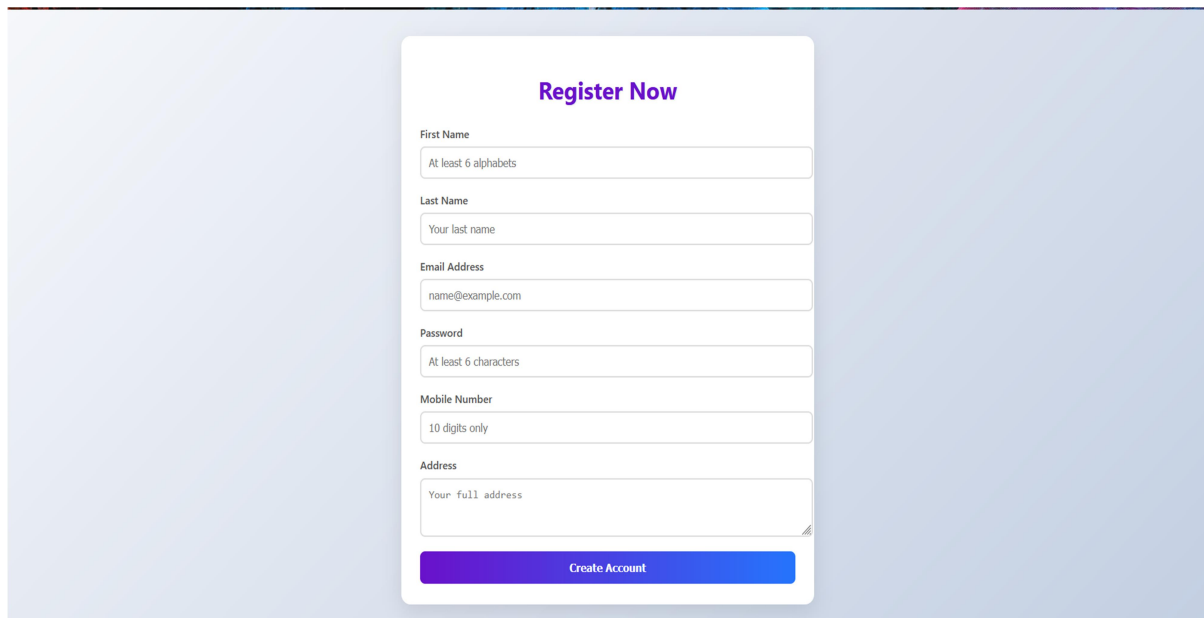
```

```

        errorElement.textContent = 'This field is required';
        errorElement.style.display = 'block';
        isValid = false;
    } else if (!validateField(fieldId)) {
        isValid = false;
    }
    });
    return isValid;
}
});
</script>
</body>
</html>

```

OUTPUT :



Register Now

First Name
At least 6 alphabets

Last Name
Your last name

Email Address
name@example.com

Password
At least 6 characters

Mobile Number
10 digits only

Address
Your full address

Create Account

RESULT :

Thus the given task is executed successfully.