

Create a HTML form that accept the student data like enrolment number, name, email id, password, confirm password, phone number, address, gender, choices of branch and perform client-side validation using JavaScript.

practical_12.html

```
<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>Student Registration Form</title>

<style>

:root{

--bg:#f4f7fb;

--card:#fff;

--accent:#4f46e5;

--danger:#dc2626;

--muted:#6b7280 }

*{



  box-sizing:border-box

}

body{

  font-family:Inter,system-ui,Segoe UI,Roboto,'Helvetica Neue',Arial;

  background:var(--bg);

  padding:40px;

}

.container{



  max-width:900px;

  margin:0 auto}

```

```

.card{
  background:var(--card);
  padding:28px;
  border-radius:12px;
  box-shadow:0 6px 20px rgba(16,24,40,0.06)
}

h1{
  margin:0 0 8px;
  font-size:22px
}

p.lead{
  color:var(--muted);
  margin:0 0 20px
}

form .grid{
  display:grid;
  grid-template-columns:repeat(2,1fr);
  gap:14px
}

label{
  display:block;
  font-weight:600;
  margin-bottom:6px;
  font-size:13px
}

input[type=text], input[type=email], input[type=password], textarea, select{
  width:100%;
  padding:10px 12px;
  border:1px solid #e6e9ef;
  border-radius:8px;
  font-size:14px
}

```

}

```
textarea{  
    min-height:80px;  
    resize:vertical  
}  
  
.full{  
    grid-column:1 / -1  
}  
  
.error{  
    color:var(--danger);  
    font-size:13px;  
    margin-top:6px  
}  
  
.btn{  
    background:var(--accent);  
    color:white;  
    padding:10px 14px;  
    border-radius:10px;  
    border:none;  
    font-weight:600;  
    cursor:pointer  
}  
  
.btn:disabled{  
    opacity:0.6;  
    cursor:not-allowed  
}  
  
.valid{  
    border-color: #10b981
```

```

}

.invalid{
    border-color:var(--danger)
}

small.note{
    display:block;
    margin-top:12px;
    color:var(--muted)
}

}

@media (max-width:720px){
    .grid{grid-template-columns:1fr}
}

</style>

</head>

<body>

<div class="container">
    <div class="card">
        <h1>Student Registration</h1>
        <p class="lead">Fill in the form below.</p>

        <form id="studentForm" novalidate>
            <div class="grid">

                <div>
                    <label for="enroll">Enrollment Number *</label>
                    <input id="enroll" name="enroll" type="text" placeholder="e.g. 2025CS101" required
                />
                    <div class="error" id="enrollError"></div>

```

</div>

<div>

 <label for="name">Full Name *</label>

 <input id="name" name="name" type="text" placeholder="Your full name" required />

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

</div>

<div>

 <label for="email">Email ID *</label>

 <input id="email" name="email" type="email" placeholder="you@example.com" required />

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

</div>

<div>

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

 <input id="phone" name="phone" type="text" placeholder="10-digit phone number" required />

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

</div>

<div>

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

 <input id="password" name="password" type="password" placeholder="Min 8 chars, mix of letters & numbers" required />

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

</div>

```
<div>

    <label for="confirmPassword">Confirm Password *</label>

    <input id="confirmPassword" name="confirmPassword" type="password"
placeholder="Repeat password" required />

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

</div>

<div class="full">

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

    <textarea id="address" name="address" placeholder="Street, city, postal code"
required></textarea>

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

</div>

<div>

    <label>Gender *</label>

    <div class="radio-group">

        <label><input type="radio" name="gender" value="Male" /> Male</label>
        <label><input type="radio" name="gender" value="Female" /> Female</label>
        <label><input type="radio" name="gender" value="Other" /> Other</label>

    </div>

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

</div>

<div>

    <label for="branch">Branch *</label>

    <select id="branch" name="branch" required>

        <option value="">-- Choose branch --</option>
```

```

<option value="CSE">Computer Science & Engineering</option>
<option value="ECE">Electronics & Communication</option>
<option value="ME">Mechanical Engineering</option>
<option value="CE">Civil Engineering</option>
<option value="EE">Electrical Engineering</option>
</select>

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

</div>

<div style="margin-top:18px;display:flex;gap:12px;align-items:center">
  <button type="submit" class="btn" id="submitBtn">Register</button>
  <div id="formMessage" style="font-weight:600"></div>
</div>

<small class="note">* Required fields.</small>
</form>
</div>
</div>

<script>
  const form = document.getElementById('studentForm');
  const submitBtn = document.getElementById('submitBtn');
  const formMessage = document.getElementById('formMessage');

  const validators = {
    enroll: value => {

```

```

if (!value) return 'Enrollment number is required.';

if (!/^[A-Za-z0-9-]{4,20}$/.test(value))

    return 'Use 4-20 characters: letters, numbers or dashes.';

    return '';

},

name: value => {

    if (!value) return 'Name is required.';

    if (value.trim().length < 3) return 'Enter at least 3 characters.';

    return '';

},

email: value => {

    if (!value) return 'Email is required.';

    if (!/^\S+@\S+\.\S+$/.test(value)) return 'Enter a valid email address.';

    return '';

},

phone: value => {

    if (!value) return 'Phone number is required.';

    const digits = value.replace(/[^0-9]/g, "");

    if (!/\d{10}$/.test(digits)) return 'Enter a 10-digit phone number.';

    return '';

},

password: value => {

    if (!value) return 'Password is required.';

    if (value.length < 8) return 'Password must be at least 8 characters.';

    if (!/[0-9]/.test(value) || !/[A-Za-z]/.test(value))

        return 'Use a mix of letters and numbers.';

    return '';

},

```

```

confirmPassword: (value, allFields) => {
  if (!value) return 'Please confirm your password.';
  if (value !== allFields.password) return 'Passwords do not match.';
  return '';
},
address: value => {
  if (!value) return 'Address is required.';
  if (value.trim().length < 8) return 'Enter a more complete address.';
  return '';
},
gender: value => (!value ? 'Please select your gender.' : ''),
branch: value => (!value ? 'Please choose a branch.' : '')
};

const fields = [
  'enroll', 'name', 'email', 'phone', 'password', 'confirmPassword', 'address', 'gender', 'branch'
];

function getFieldValue(name){
  if (name === 'gender'){
    const g = form.querySelector('input[name="gender"]:checked');
    return g ? g.value : '';
  }
  const el = document.getElementById(name);
  return el ? el.value : '';
}

function setError(name, message){

```

```

const err = document.getElementById(name + 'Error');

const el = document.getElementById(name);

if (name === 'gender'){

  const radios = form.querySelectorAll('input[name="gender"]');

  radios.forEach(r=> r.parentElement.style.color = message ? 'var(--danger)' : 'inherit');

}

else if (el){

  el.classList.toggle('invalid', !!message);

  el.classList.toggle('valid', !message);

}

if (err) err.textContent = message;

}

function validateField(name, allFieldsValues){

  const raw = getFieldValue(name);

  const validator = validators[name];

  const error = validator ? validator(raw, allFieldsValues) : '';

  setError(name, error);

  return error === "";

}

function validateAll(){

  const values = {};

  fields.forEach(f => values[f] = getFieldValue(f));

  let ok = true;

  fields.forEach(f => { if (!validateField(f, values)) ok = false; });

  return {ok, values};

}

```

```
[enroll,'name','email','phone','password','confirmPassword','address','branch']

.forEach(id => {
  const el = document.getElementById(id);
  if (!el) return;
  el.addEventListener('input', () => validateField(id, {}));
  el.addEventListener('blur', () => validateField(id, {}));
});

const genderRadios = form.querySelectorAll('input[name="gender"]');
genderRadios.forEach(r => r.addEventListener('change', () => validateField('gender')));

form.addEventListener('submit', (e) => {
  e.preventDefault();
  formMessage.textContent = "";

  const {ok, values} = validateAll();

  if (!ok){
    formMessage.style.color = 'var(--danger)';
    formMessage.textContent = 'Please fix the errors above before submitting.';
    return;
  }

  formMessage.style.color = 'green';
  formMessage.textContent = 'Form looks good! (demo submission)';

  const safeCopy = {...values};
}
```

```
delete safeCopy.password;  
delete safeCopy.confirmPassword;  
console.log('Form submitted (demo):', safeCopy);  
  
submitBtn.disabled = true;  
setTimeout(()=> submitBtn.disabled = false, 1200);  
});  
</script>  
</body>  
</html>
```

Output:

Student Registration

Fill in the form below.

Enrollment Number * <input type="text" value="e.g. 2025CS101"/>	Full Name * <input type="text" value="Your full name"/>
Email ID * <input type="text" value="you@example.com"/>	Phone Number * <input type="text" value="10-digit phone number"/>
Password * <input type="password" value="Min 8 chars, mix of letters & numbers"/>	Confirm Password * <input type="password" value="Repeat password"/>
Address * <input type="text" value="Street, city, postal code"/>	
Gender * <input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other	Branch * <input type="text" value="-- Choose branch --"/>
Register	

* Required fields.