**MCA-214: Web Technology Lab**

**How my work is different from other Students:-**

**1.)Professional Styling & UI Consistency**:  
I used consistent and responsive CSS across all pages. Inline, internal, and external stylesheets were carefully layered to demonstrate overriding precedence, which many students overlook.

**2.)Integrated Multimedia in Frames**:  
My nested frame structure goes beyond basic iframe usage. I implemented working **audio and video frames**, as well as **image mapping**, making navigation intuitive and dynamic.

**3.)** **User-Centric Forms with Full HTML5 Validation**:  
I used advanced HTML5 features such as **regex patterns**, **step validations**, **read-only/disabled fields**, and even included **formnovalidate** with JavaScript overrides—covering nearly all types of validations in a meaningful way.

**4.)React-Based Resume Project**:  
Unlike most students who build static resumes, I developed a fully functional **React-based resume**, demonstrating my knowledge of component structuring, JSX, and styling through CSS modules. It’s interactive, modern, and showcases real-time skills.

**5.)** **Backend Integration with Express and Cookie Handling**:  
I implemented an **Express.js server**, handling **cookies securely** and using **Fetch API** from the frontend. The Regex validation for mobile numbers is integrated into a real-world backend structure—something rarely done in basic practicals.

**6.)Clean Code & Commenting**:  
I kept all code well-indented and commented where necessary, making the project readable and professional—important traits in real-world development.

**7.)Project Relevance & Realism**:  
My work simulates actual web development practices, from frontend (React, HTML5, CSS3) to backend (Node.js, Express), focusing on **interoperability** and **user experience**, not just output.

Overall, my practical work reflects a sincere effort to go beyond the minimum and demonstrate both **technical depth and creativity**.

**MCA-214: Web Technology Lab**

**1. Implement an HTML program for a stylesheet consisting of the use of external, internal, and inline stylesheets, overriding sequences among them and the use of id and class.**

**HTML file :-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>CSS STYLES</title>

<style>

body {

font-family: Arial, sans-serif;

}

h1 {

text-align: center;

color: blue;

}

h2 {

text-align: center;

color: green;

}

h3 {

text-align: center;

color: red;

}

h4 {

text-align: center;

color: purple;

}

p {

text-align: center;

font-size: 18px;

color: black;

line-height: 1.5;

}

</style>

<link rel="stylesheet" href="styles.css" />

</head>

<body>

<h1 style="text-align: center; color: blue; font-family: Arial">

Name:- Armaan Vasal

</h1>

<h2 style="text-align: center; color: green; font-family: Verdana">

Roll No.:- 523110039

</h2>

<h3 style="text-align: center; color: red; font-family: Georgia">

Branch:- MCA

</h3>

<h4 style="text-align: center; color: purple; font-family: Tahoma">

Year:- 2nd

</h4>

<p

style="

text-align: center;

font-size: 18px;

color: black;

line-height: 1.5;

"

>

I am a second-year student of M.C.A. branch in NIT Kurukshetra. My CGPA is

9.0667. My objective is to become an SDE in a well-renowned company with a

whooping package.

</p>

</body>

</html>

**CSS file :-**

body {

font-family: Arial, sans-serif;

}

h1 {

text-align: center;

color: aqua;

}

h2 {

text-align: center;

color: black;

}

h3 {

text-align: center;

color: red;

}

h4 {

text-align: center;

color: yellow;

}

p {

text-align: center;

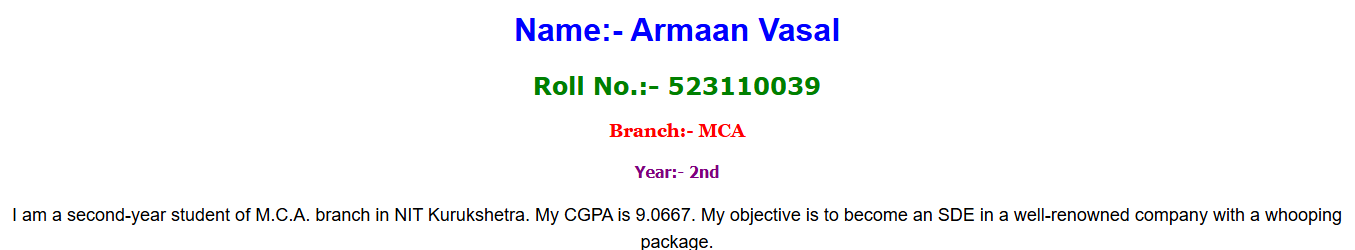
font-size: 18px;

color: black;

line-height: 1.5;

}

**Output :-**



**2. Implement an HTML program for a nested frame structure in TLB fashion consisting of targeting between frames and audio, video features, and image mapping.**

**Index.html :-**

<!DOCTYPE html>

<html>

<head>

<title>Iframe Example</title>

</head>

<body>

<!-- Header Section -->

<iframe

src="header.html"

style="width: 100%; height: 70px; border: none"

></iframe>

<!-- Main Layout with Navigation and Content -->

<div style="display: flex">

<!-- Left Navigation -->

<iframe

src="navigation.html"

style="width: 25%; height: 500px; border: none"

></iframe>

<!-- Main Content Area -->

<iframe

src="main.html"

name="mainFrame"

style="width: 75%; height: 500px; border: none"

></iframe>

</div>

<!-- Footer Section -->

<iframe

src="footer.html"

style="width: 100%; height: 50px; border: none"

></iframe>

</body>

</html>

**Header.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Header portion</title>

</head>

<body style="background-color: rgb(225, 60, 60); text-align: center">

<a href="./main.html" target="mainFrame">

<img

src="logo.png"

style="border-radius: 50%"

align="left"

height="50px"

width="50px"

/></a>

<h2>Welcome to NIT KKR</h2>

</body>

</html>

**Navigation.html :-**

<!DOCTYPE html>

<html>

<head>

<title>Navigation</title>

</head>

<body style="background-color: lightgray">

<h2>Navigation</h2>

<img

src="./img\_mapping.webp"

usemap="#image-map"

height="100%"

width="100%"

/>

<map name="image-map">

<!-- Left side for audio -->

<area

shape="rect"

coords="0,0,300,100"

href="audio.html"

target="mainFrame"

/>

<!-- Right side for video -->

<area

shape="rect"

coords="300,0,600,100"

href="video.html"

target="mainFrame"

/>

</map>

</body>

</html>

**Main.html :-**

<!DOCTYPE html>

<html>

<head>

<title>Main Content</title>

</head>

<body style="background-color: white;">

<h2>Welcome to NIT KKR</h2>

<p>Click the links in the navigation menu to change this content.</p>

</body>

</html>

**Footer.html :-**

<!DOCTYPE html>

<html>

<head>

<title>Footer</title>

</head>

<body style="background-color: rgb(115, 187, 235); text-align: center;">

<p>&copy; 2024 Shubham Kumar Shaw. All Rights Reserved.</p>

</body>

</html>

**Audio.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Audio Frame</title>

</head>

<body>

<h2>Audio Frame</h2>

<audio controls>

<source src="audio.mp3" type="audio/mp3" />

Your browser does not support the audio element.

</audio>

</body>

</html>

**Video.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Video Frame</title>

</head>

<body>

<h2>Video Frame</h2>

<video width="100%" height="100%" controls>

<source src="12931114\_1080\_1920\_30fps.mp4">

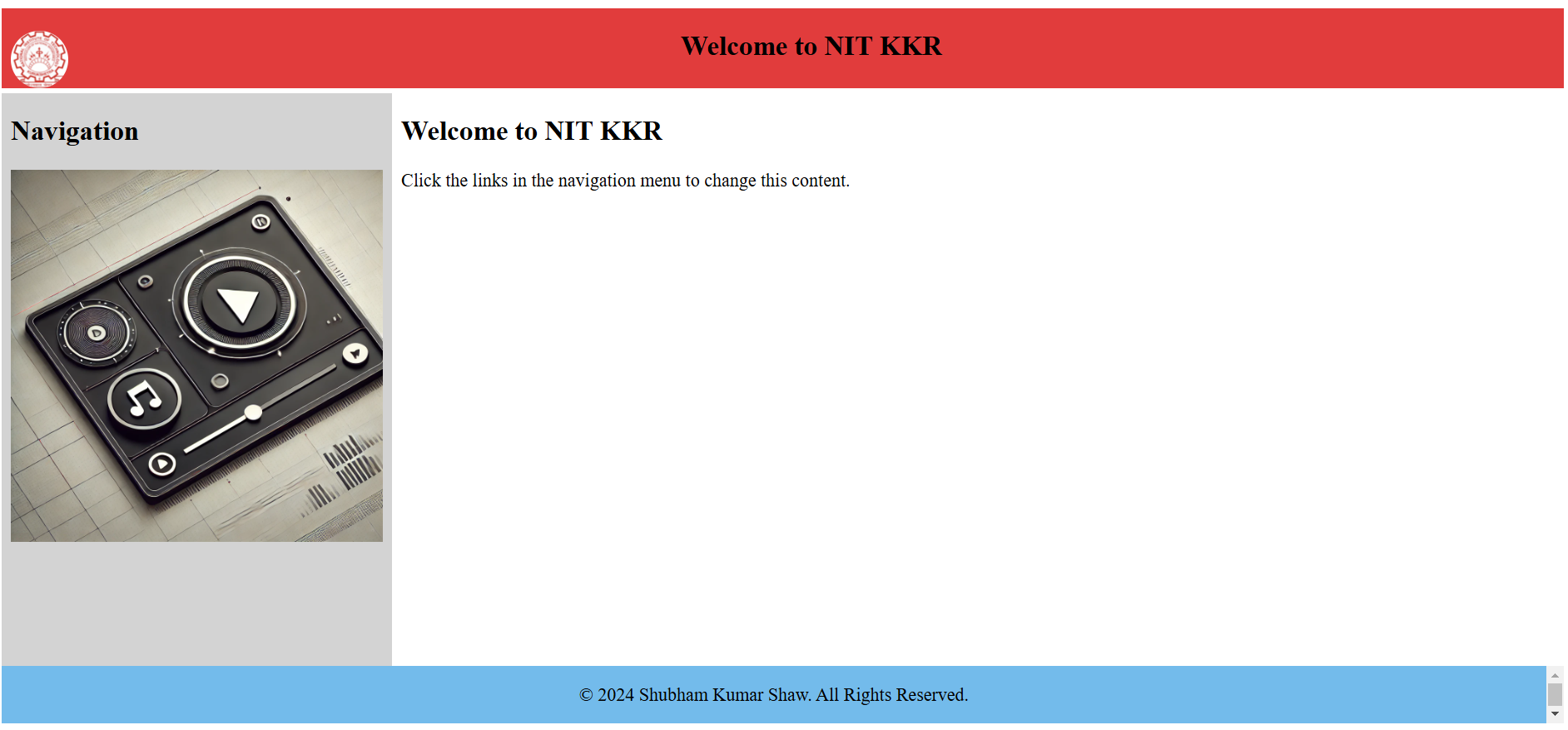
Your browser does not support the video tag.

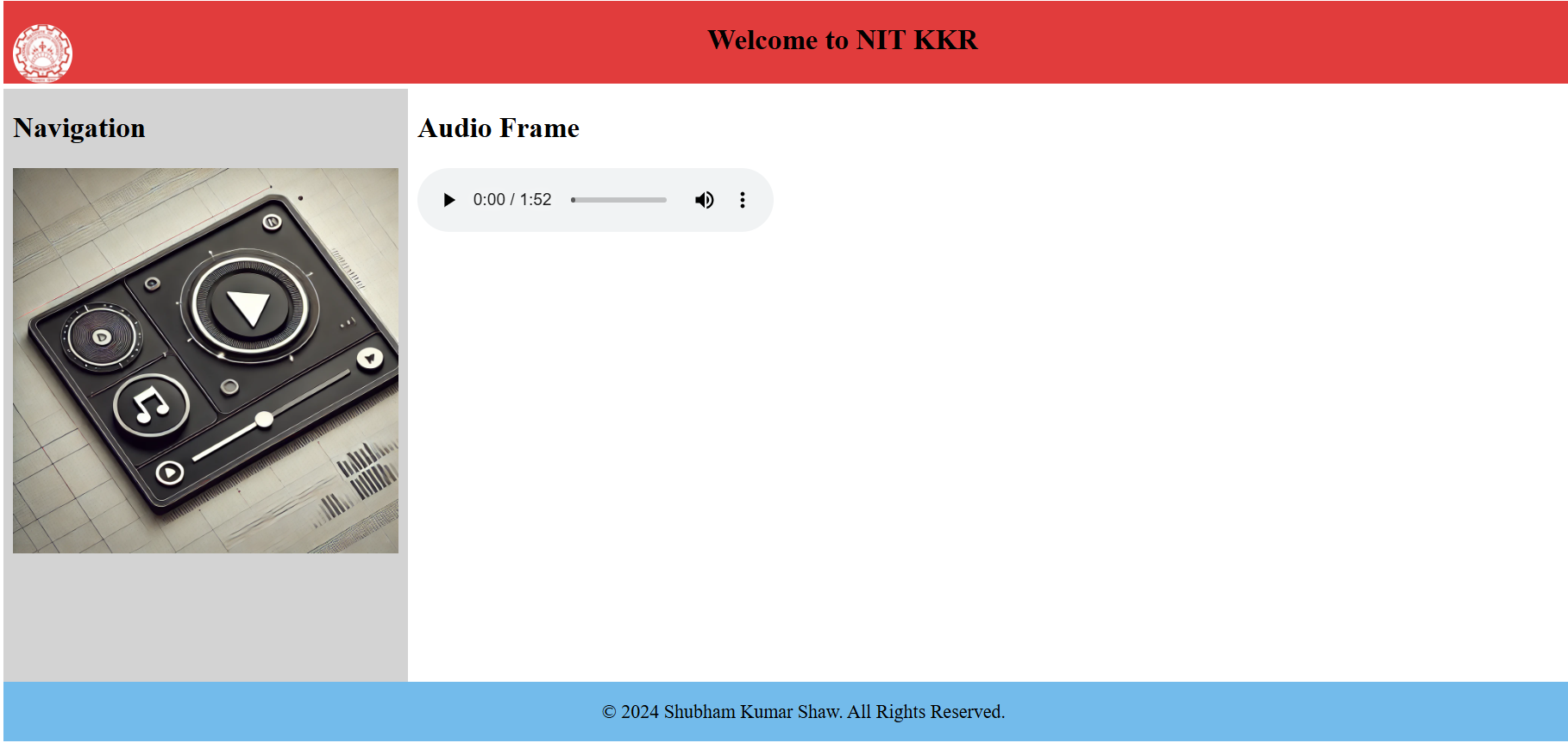
</video>

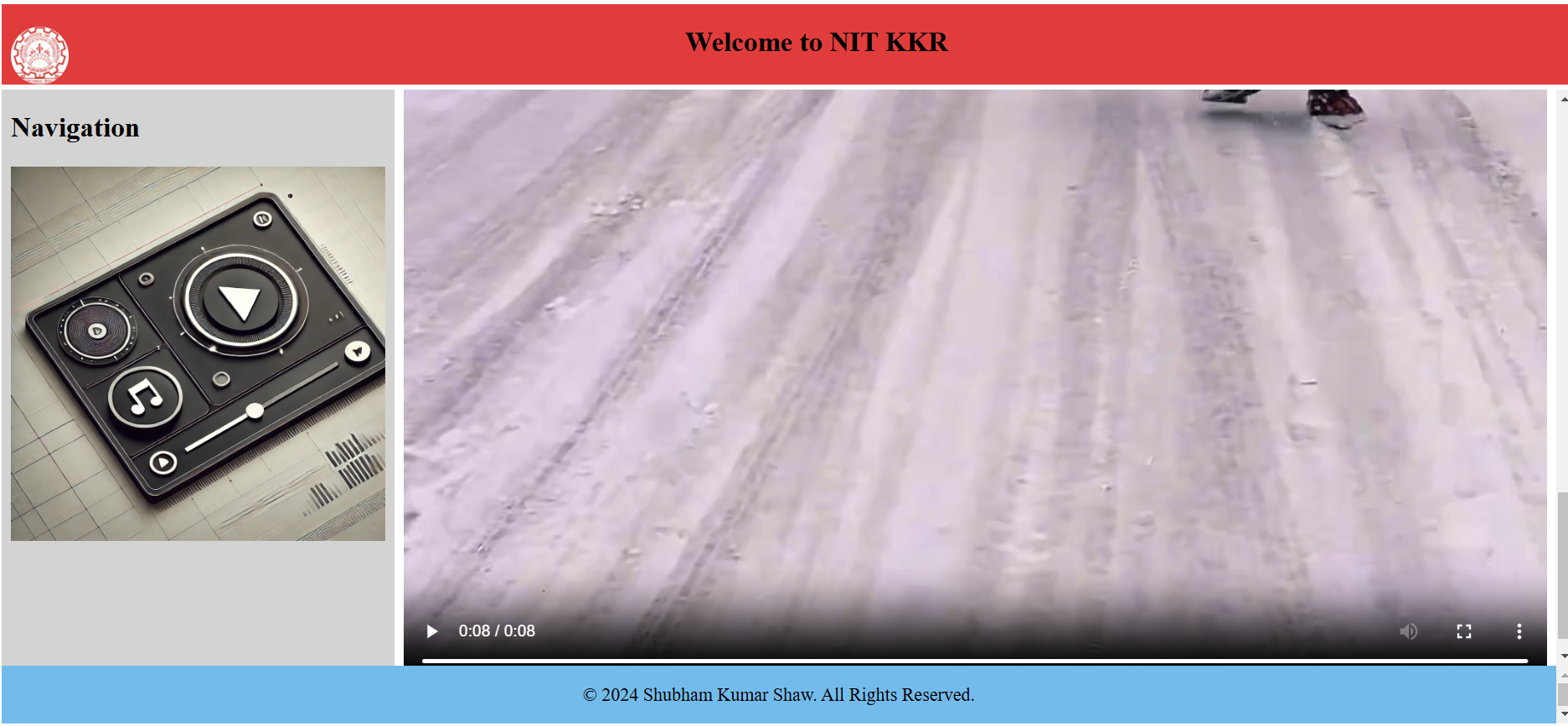
</body>

</html>

**Output :-**

****





**3. Implement an HTML program for form consisting of controls (single-line text field with size, value, max length; Dropdown list and scrolled list; Checkbox and radio controls; Text area with different wrapping options).**

**Index.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Interactive User Form</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 20px;

}

.container {

max-width: 700px;

background: white;

padding: 30px;

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

margin: auto;

}

h2 {

text-align: center;

color: #0cadfe;

}

label {

font-weight: bold;

margin-top: 10px;

margin-bottom: 15px;

display: block;

}

input,

select,

textarea {

width: 100%;

padding: 8px;

margin-top: 5px;

margin-bottom: 15px;

border-radius: 5px;

border: 1px solid #ccc;

}

textarea {

resize: vertical;

}

.checkbox-group,

.radio-group {

display: flex;

flex-direction: row;

align-items: center;

margin-bottom: 20px;

}

.checkbox-group label,

.radio-group label {

margin-right: 10px;

}

button {

width: 100%;

padding: 10px;

background: #28a745;

color: white;

border: none;

cursor: pointer;

margin-top: 15px;

border-radius: 5px;

}

button:hover {

background: #218838;

}

</style>

</head>

<body>

<div class="container">

<h2>Interactive User Form</h2>

<form action="#" method="POST">

<!-- Single Line Text Field -->

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

<input

type="text"

id="name"

name="name"

size="30"

value="John Doe"

maxlength="40"

required

/>

<!-- Dropdown List -->

<label for="country">Select Your Country:</label>

<select id="country" name="country">

<option value="india">India</option>

<option value="usa">United States</option>

<option value="uk">United Kingdom</option>

<option value="canada">Canada</option>

<option value="australia">Australia</option>

</select>

<!-- Multi-Select Scrolled List -->

<label for="hobbies">Select Your Hobbies:</label>

<select id="hobbies" name="hobbies[]" multiple size="2">

<option value="reading">Reading</option>

<option value="traveling">Traveling</option>

<option value="gaming">Gaming</option>

<option value="sports">Sports</option>

<option value="running">Running</option>

<option value="Internet Suffering">Internet Suffering</option>

<option value="Sketching">Sketching</option>

<option value="Other">Other</option>

</select>

<!-- Checkbox Controls -->

<label>Preferred Contact Method:</label>

<div class="checkbox-group">

<label

><input type="checkbox" name="contact\_method" value="email" />

Email</label

>

<label

><input type="checkbox" name="contact\_method" value="phone" />

Phone</label

>

<label

><input type="checkbox" name="contact\_method" value="sms" />

SMS</label

>

</div>

<!-- Radio Controls -->

<label>Gender:</label>

<div class="radio-group">

<label

><input type="radio" name="gender" value="male" checked />

Male</label

>

<label

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

>

<label

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

>

</div>

<!-- Text Area with Wrapping Options -->

<label for="comments">Comments (Soft Wrap):</label>

<textarea

id="comments"

name="comments"

rows="4"

wrap="soft"

placeholder="Enter your comments here..."

></textarea>

<label for="comments-hard">Comments (Hard Wrap):</label>

<textarea

id="comments-hard"

name="comments-hard"

rows="4"

wrap="hard"

placeholder="Text will wrap at specified width..."

></textarea>

<!-- Submit Button -->

<button type="submit">Submit</button>

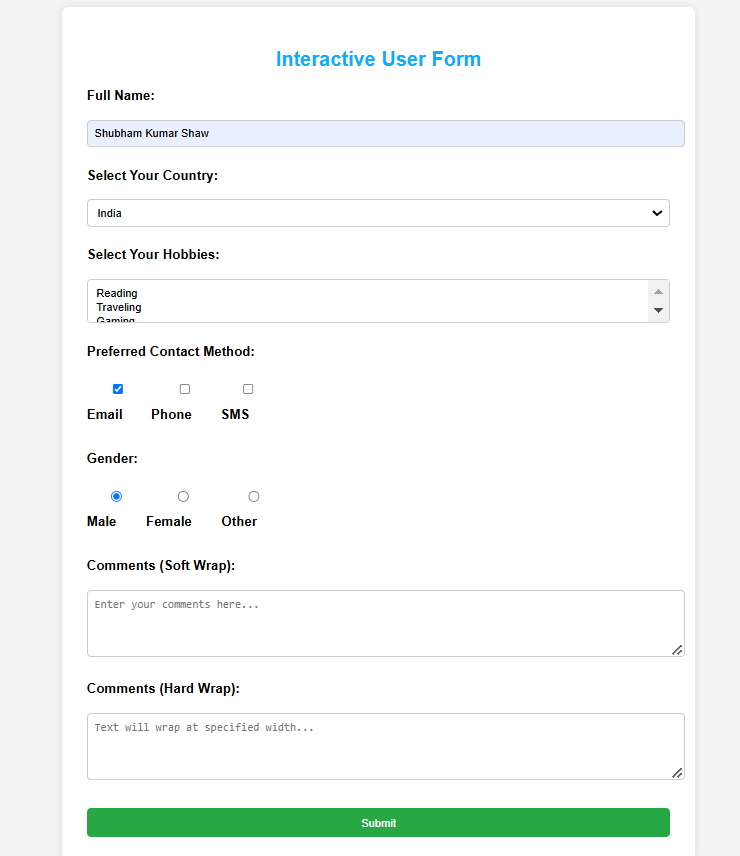
</form>

</div>

</body>

</html>

**Output :-**



**4.** **Apply validations (not null, min, max, multiple of n, read-only and disabled, Pattern length, domain, Overriding submit, form novalidate) on form controls.**

**Index.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Validated User Form</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 20px;

}

.container {

max-width: 700px;

background: white;

padding: 40px;

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

margin: auto;

}

h2 {

text-align: center;

color: #0cadfe;

}

label {

font-weight: bold;

display: block;

margin-top: 15px;

margin-bottom: 10px;

}

input,

select,

textarea {

width: 100%;

padding: 10px;

margin-top: 5px;

margin-bottom: 20px;

border-radius: 5px;

border: 1px solid #ccc;

}

textarea {

resize: vertical;

}

.checkbox-group,

.radio-group {

display: flex;

flex-direction: row;

align-items: center;

margin-bottom: 20px;

}

.checkbox-group label,

.radio-group label {

margin-right: 15px;

}

button {

width: 100%;

padding: 10px;

background: #28a745;

color: white;

border: none;

cursor: pointer;

margin-top: 20px;

border-radius: 5px;

}

button:hover {

background: #218838;

}

.disabled {

background: #ddd;

}

</style>

</head>

<body>

<div class="container">

<h2>Validated User Form</h2>

<form id="validatedForm" onsubmit="return validateForm(event)">

<!-- Required Field -->

<label for="name"

>Full Name (Required, Only Letters, Min 3 characters):</label

>

<input

type="text"

id="name"

name="name"

pattern="[A-Za-z ]{3,}"

required

placeholder="Enter your full name"

/>

<!-- Min & Max Values -->

<label for="age">Age (Between 18 and 60):</label>

<input

type="number"

id="age"

name="age"

min="18"

max="60"

required

placeholder="Enter your age"

/>

<!-- Multiple of N -->

<label for="salary">Salary (Multiple of 1000):</label>

<input

type="number"

id="salary"

name="salary"

step="1000"

min="1000"

required

placeholder="Enter salary (1000, 2000, etc.)"

/>

<!-- Read-Only Field -->

<label for="country">Country (Read-Only):</label>

<input

type="text"

id="country"

name="country"

value="India"

readonly

class="disabled"

/>

<!-- Disabled Field -->

<label for="disabledField">Disabled Field:</label>

<input

type="text"

id="disabledField"

name="disabledField"

value="Cannot Edit"

disabled

class="disabled"

/>

<!-- Pattern for Specific Domains -->

<label for="email">Email (@gmail.com only):</label>

<input

type="email"

id="email"

name="email"

pattern="^[a-zA-Z0-9.\_%+-]+@gmail\.com$"

required

placeholder="Enter your Gmail"

/>

<!-- URL with Specific Domain -->

<label for="website">Personal Website (must be .com):</label>

<input

type="url"

id="website"

name="website"

pattern="https?://.\*\.com"

required

placeholder="https://example.com"

/>

<!-- Text Area with Length Constraints -->

<label for="comments">Comments (Min 10, Max 200 chars):</label>

<textarea

id="comments"

name="comments"

minlength="10"

maxlength="200"

required

placeholder="Write at least 10 characters"

></textarea>

<!-- Dropdown List -->

<label for="city">Select City:</label>

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

<option value="">--Choose City--</option>

<option value="Delhi">Delhi</option>

<option value="Mumbai">Mumbai</option>

<option value="Bangalore">Bangalore</option>

</select>

<!-- Multi-Select Scrolled List -->

<label for="hobbies">Select Your Hobbies:</label>

<select id="hobbies" name="hobbies[]" multiple size="4" required>

<option value="reading">Reading</option>

<option value="traveling">Traveling</option>

<option value="gaming">Gaming</option>

<option value="sports">Sports</option>

</select>

<!-- Checkbox Controls -->

<label>Preferred Contact Method:</label>

<div class="checkbox-group">

<label

><input

type="checkbox"

name="contact\_method"

value="email"

required

/>

Email</label

>

<label

><input type="checkbox" name="contact\_method" value="phone" />

Phone</label

>

<label

><input type="checkbox" name="contact\_method" value="sms" />

SMS</label

>

</div>

<!-- Radio Controls -->

<label>Gender:</label>

<div class="radio-group">

<label

><input type="radio" name="gender" value="male" checked />

Male</label

>

<label

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

>

<label

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

>

</div>

<!-- Submit Button -->

<button type="submit">Submit</button>

<!-- Form Submit Without Validation -->

<button type="submit" formnovalidate>Submit Without Validation</button>

</form>

</div>

<script>

function validateForm(event) {

const name = document.getElementById("name").value;

if (name.length < 3) {

alert("Name must be at least 3 characters long!");

return false;

}

alert("Form submitted successfully!");

return true;

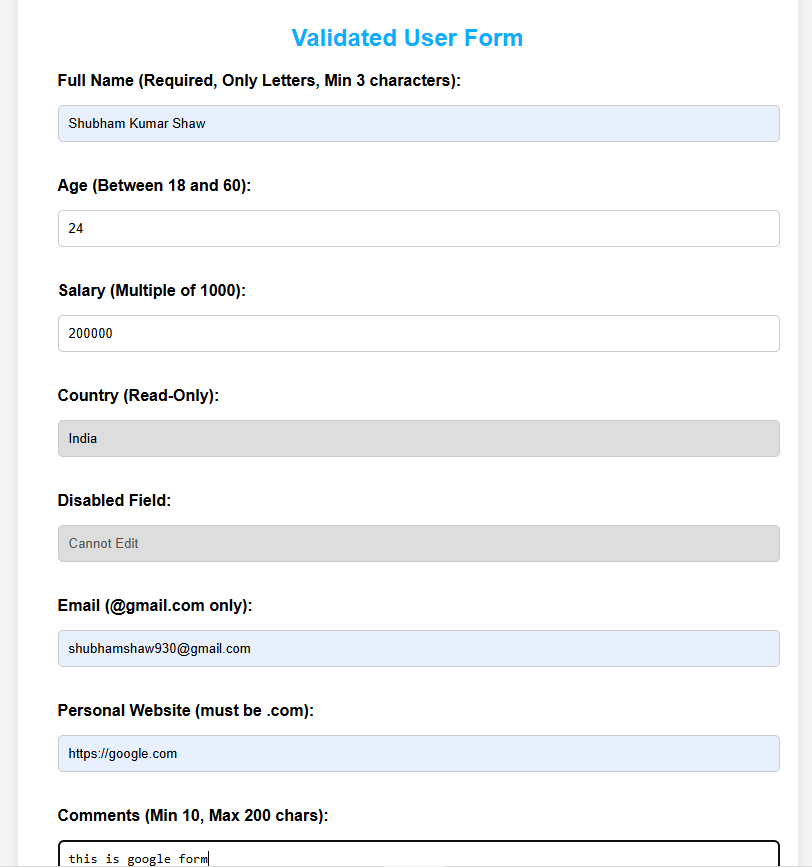
}

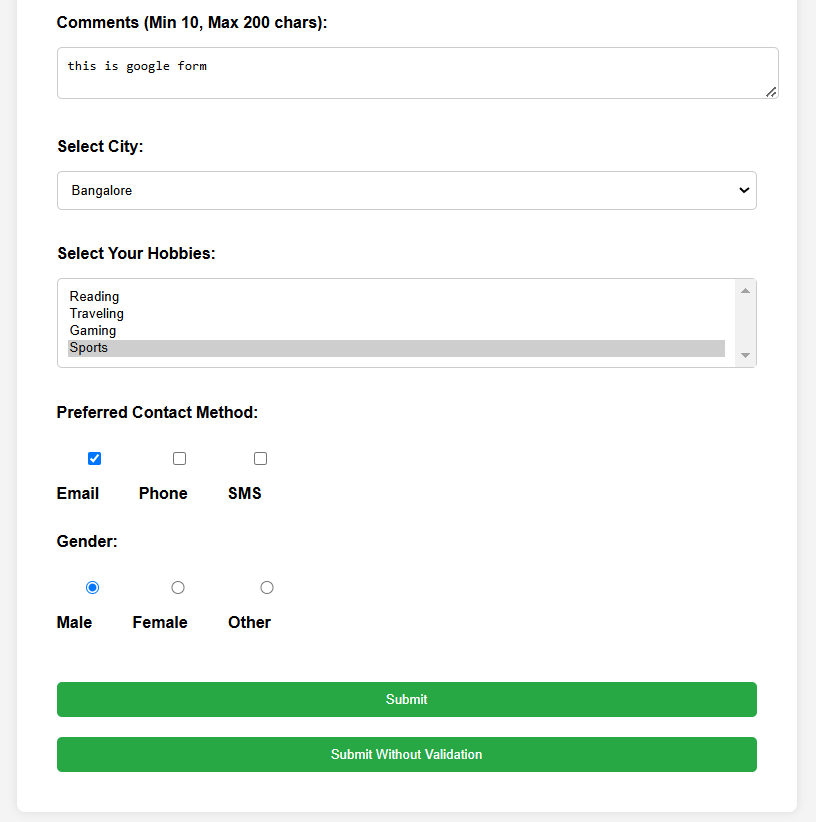
</script>

</body>

</html>

**Output :-**





**5. Implement of XML, XSD, XSLT.**

**Index.html :-**

<!DOCTYPE html>

<html>

<body>

<object data="data.xml" type="text/xml" width="500" height="200"></object>

</body>

</html>

**Data.xml :-**

<?xml version="1.0"?>

<?xml-stylesheet type="text/xsl" href="style.xsl" ?>

<employees xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:noNamespaceSchemaLocation="schema.xsd">

<employee id="1">

<name>JohnDoe</name>

<position>Software Engineer</position>

</employee>

</employees>

**style.xml :-**

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">

<html>

<body>

<h2>Employee List</h2>

<table border="1">

<tr><th>Name</th><th>Position</th></tr>

<xsl:for-each select="employees/employee">

<tr>

<td><xsl:value-of select="name"/></td>

<td><xsl:value-of select="position"/></td>

</tr>

</xsl:for-each>

</table>

</body>

</html>

</xsl:template>

</xsl:stylesheet>

**Schema.xsd :-**

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

<xsd:element name="employees">

<xsd:complexType>

<xsd:sequence>

<xsd:element name="employee" maxOccurs="unbounded">

<xsd:complexType>

<xsd:sequence>

<xsd:element name="name" type="xsd:Name" minOccurs="1"/>

<xsd:element name="position">

<xsd:simpleType>

<xsd:restriction base="xsd:string">

<xsd:maxLength value="50"/>

</xsd:restriction>

</xsd:simpleType>

</xsd:element>

</xsd:sequence>

<xsd:attribute name="id" type="xsd:integer" use="required"/>

</xsd:complexType>

</xsd:element>

</xsd:sequence>

</xsd:complexType>

</xsd:element>

</xsd:schema>

**Validate.js :-**

const validator = require('xsd-schema-validator');

const xmlString = require('fs').readFileSync('data.xml', 'utf8');

const xsdPath = 'schema.xsd';

validator.validateXML(xmlString, xsdPath)

.then(result => {

if (result.valid) {

console.log("XML is valid!");

} else {

console.error("XML is invalid:", result.errors);

}

})

.catch(error => {

console.error("Validation error:", error);

});

**Output :-**

****

**6. Create resume using react.**

**Main.jsx :-**

import { StrictMode } from "react";

import { createRoot } from "react-dom/client";

import "./index.css";

import App from "./App.jsx";

createRoot(document.getElementById("root")).render(

<StrictMode>

<App />

</StrictMode>

);

**App.jsx :-**

import { useState } from "react";

import reactLogo from "./assets/react.svg";

import viteLogo from "/vite.svg";

import "./App.css";

function App() {

const [count, setCount] = useState(0);

return (

<>

<div>

<img src="./logo.jpg" alt="logo" className="logo" />

</div>

<h1>Armaan Vasal</h1>

<div>

<header className="resume-header">

<div className="header-left">

<p>

<strong>Full Stack Developer</strong>

</p>

<p>

<strong>Phone:</strong> 6386238247

</p>

<p>

<strong>Location:</strong> Uttar Pradesh

</p>

</div>

<div className="header-right">

<p>

Email:{" "}

<a href="mailto:armaan7vasal@gmail.com">

armaan7vasal@gmail.com

</a>

</p>

<p>

<strong>GitHub:</strong>{" "}

<a href="https://github.com/armaanvasal007" target="\_blank">

github.com/armaanvasal007

</a>

</p>

<p>

<strong>LinkedIn:</strong>{" "}

<a href="https://www.linkedin.com/in/armaan-vasal-9a3696229/" target="\_blank">

linkedin.com/in/armaan-vasal

</a>

</p>

</div>

</header>

</div>

<section className="resume-section">

<h2>Summary</h2>

<p>

Eager to learn new things and hard working.To enhance my professional

skills and knowledge in an organisation which trust me with

responsibilies and challenges.

</p>

</section>

<section>

<h2>Education</h2>

<table class="education-table">

<thead>

<tr>

<th>University/College</th>

<th>Degree</th>

<th>Percentage/CGPA</th>

<th>Year</th>

</tr>

</thead>

<tbody>

<tr>

<td>NIT KKR</td>

<td>MCA</td>

<td>9.2647/10</td>

<td>2023-26</td>

</tr>

<tr>

<td>Dr.VSICS</td>

<td>BCA</td>

<td>82%</td>

<td>2020-23</td>

</tr>

<tr>

<td>UPKSSV</td>

<td>Intermediate</td>

<td>96%</td>

<td>2020</td>

</tr>

<tr>

<td>UPKSSV</td>

<td>Matriculation</td>

<td>90.33%</td>

<td>2018</td>

</tr>

</tbody>

</table>

</section>

<section className="resume-section">

<h2>Skills</h2>

<ul>

<li>JavaScript / ES6</li>

<li>React / Redux</li>

<li>Node.js / Express</li>

<li>HTML / CSS</li>

<li>SQL / MongoDB</li>

<li>Git / GitHub</li>

</ul>

</section>

<section>

<h2>Projects</h2>

<ul>

<li>Movie Recommendar System - ML Prtoject</li>

<li>Online food order System - Frontend project</li>

</ul>

</section>

<section>

<h2>Co-Curricular</h2>

<li>Playing individual Games and Team-Oriented</li>

<li>Exploring on Internet</li>

</section>

</>

);

}

export default App;

**App.css :-**

/\* Centered main title \*/

h1 {

text-align: center;

font-size: 36px;

color: #2c3e50;

margin-top: 20px;

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

.logo {

display: block;

margin: 20px auto;

/\* center the logo horizontally \*/

width: 120px;

/\* adjust size as needed \*/

height: auto;

border-radius: 12px;

/\* optional: rounded corners \*/

box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);

/\* subtle shadow \*/

transition: transform 0.3s ease;

}

.logo:hover {

transform: scale(1.05);

/\* zoom effect on hover \*/

}

/\* Resume header styling \*/

.resume-header {

display: flex;

justify-content: space-between;

align-items: flex-start;

padding: 20px 40px;

background-color: #f8f9fa;

border-bottom: 2px solid #ddd;

box-shadow: 0 2px 8px rgba(0, 0, 0, 0.05);

border-radius: 8px;

margin: 20px;

}

.header-left,

.header-right {

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

.header-left {

max-width: 60%;

}

.header-right {

max-width: 35%;

text-align: right;

}

.header-left p,

.header-right p {

margin: 6px 0;

font-size: 15px;

}

.header-right a {

color: #0073e6;

text-decoration: none;

transition: color 0.3s ease;

}

.header-right a:hover {

text-decoration: underline;

color: #005bb5;

}

/\* Section styling \*/

.resume-section {

padding: 20px 40px;

margin: 20px;

background-color: #ffffff;

border-radius: 8px;

border: 1px solid #eee;

box-shadow: 0 2px 8px rgba(0, 0, 0, 0.05);

}

.resume-section h2 {

border-bottom: 2px solid #0073e6;

padding-bottom: 5px;

margin-bottom: 15px;

color: #2c3e50;

}

/\* Education Table styling \*/

.education-table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

.education-table th,

.education-table td {

padding: 12px 15px;

text-align: left;

border: 1px solid #ddd;

transition: background-color 0.3s ease;

}

.education-table th {

background-color: #0073e6;

color: white;

font-size: 16px;

}

.education-table td {

font-size: 14px;

background-color: #fafafa;

}

.education-table tr:nth-child(even) td {

background-color: #f2f2f2;

}

.education-table tr:hover td {

background-color: #e6f3ff;

}

/\* Skills & Projects list \*/

ul {

list-style: disc inside;

padding-left: 40px;

line-height: 1.8;

}

li {

transition: transform 0.2s;

}

li:hover {

transform: translateX(5px);

color: #0073e6;

}

/\* Co-curricular section \*/

section>li {

margin-bottom: 8px;

}

/\* Responsive Design \*/

@media screen and (max-width: 768px) {

.resume-header {

flex-direction: column;

text-align: left;

}

.header-left,

.header-right {

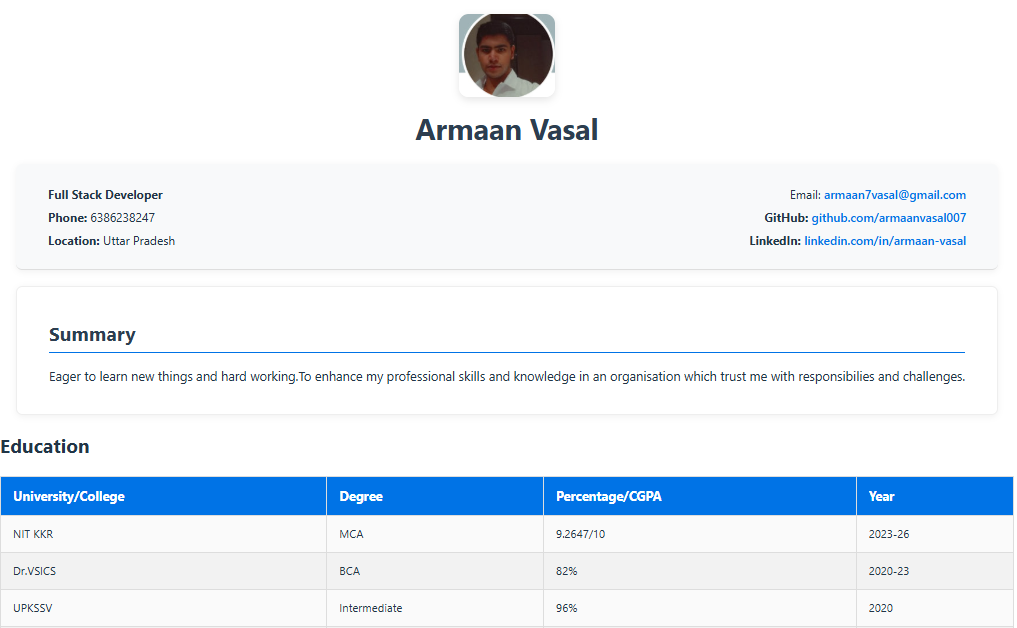
max-width: 100%;

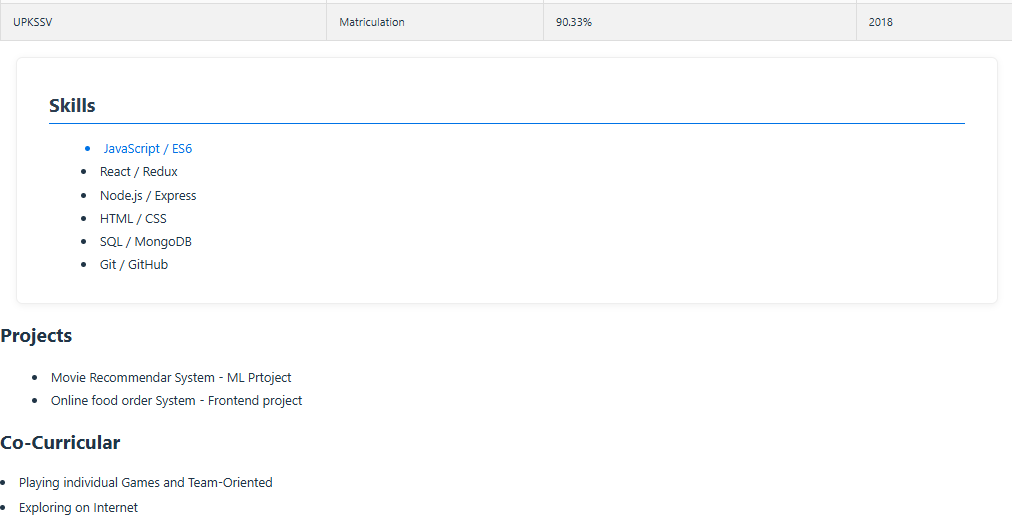
text-align: left;

}

}

**Output :-**

****

****

**7. Implement** **S-Regex , Debugging , Express Server , Cookies and fetch API**

**Index.html :-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>

<title>Cookies & Regex</title>

<link rel="stylesheet" href="style.css" />

<script>

function pingServer() {

fetch("http://localhost:8000/api", {

method: "GET",

credentials: "include",

})

.then((response) => response.json())

.then((data) => console.log(data))

.catch((error) => console.error("Ping Error:", error));

}

function checkMobileNumber() {

const mobileNumber = document.getElementById("mobileNumber").value;

fetch("http://localhost:8000/check-mobile", {

method: "POST",

headers: {

"Content-Type": "application/json",

},

body: JSON.stringify({ mobileNumber }),

credentials: "include",

})

.then((response) => response.json())

.then((data) => {

console.log(data);

})

.catch((error) => console.error("Validation Error:", error));

}

</script>

</head>

<body>

<h1>Cookies & Regex</h1>

<button onclick="pingServer()">Ping Server</button>

<br><br>

<input type="text" id="mobileNumber" placeholder="Enter mobile number" />

<button onclick="checkMobileNumber()">Check Mobile Number</button>

</body>

</html>

**Style.css :-**

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

text-align: center;

padding: 50px;

}

h1 {

color: #333;

}

button {

background-color: #007bff;

color: white;

border: none;

padding: 10px 20px;

font-size: 16px;

cursor: pointer;

margin: 10px;

border-radius: 5px;

transition: background 0.3s ease;

}

button:hover {

background-color: #0056b3;

}

input {

padding: 10px;

font-size: 16px;

border: 1px solid #ccc;

border-radius: 5px;

margin: 10px;

width: 250px;

}

input:focus {

border-color: #007bff;

outline: none;

box-shadow: 0 0 5px rgba(0, 123, 255, 0.5);

}

.container {

max-width: 400px;

margin: auto;

background: white;

padding: 20px;

border-radius: 10px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

}

**index.js :-**

const express = require("express");

const cookieParser = require("cookie-parser");

const cors = require("cors");

const path = require("path");

const app = express();

const PORT = 8000;

// Middleware

app.use(cookieParser());

app.use(express.json());

// CORS Setup for local testing

app.use(cors({

origin: "http://localhost:8000",

credentials: true,

}));

// Serve static files like index.html

app.use(express.static(path.join(\_\_dirname, "../")));

// Moved Cookie API to /api to avoid conflict with index.html

app.get("/api", (req, res) => {

res.cookie("userId", "1234", {

expires: new Date(Date.now() + 1000 \* 60 \* 60 \* 24 \* 7),

secure: false,

sameSite: "Lax",

});

res.cookie("sessionId", "1", {

secure: false,

sameSite: "Lax",

});

res.json({ message: "Welcome to Cookie API!" });

});

// Mobile validation route

app.post("/check-mobile", (req, res) => {

const { mobileNumber } = req.body;

const mobile = mobileNumber?.trim();

const mobileRegex = /^[0-9]{10}$/;

console.log("UserId from cookies: ", req.cookies.userId);

if (!req.cookies.userId) {

return res.status(401).json({ message: "Unauthorized!" });

}

if (!mobileRegex.test(mobile)) {

return res.status(400).json({ message: "Mobile number is invalid!" });

}

res.json({ message: "Mobile number is valid!" });

});

// Start server

app.listen(PORT, () => {

console.log(`Server running on http://localhost:${PORT}`);

});

**Output :-**

