**Program 1**

<!DOCTYPE html>

<html>

<head>

<title>Documents</title>

</head>

<body>

<ol type="1">

<li>Animals</li>

<ul>

<li>cow</li>

<li>cat</li>

<li>lion</li>

</ul>

<li>Colours</li>

<ul>

<li>red</li>

<li>green</li>

<li>blue</li>

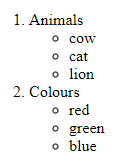
</ul>

</ol>

</body>

</html>

**Output:**

****

**Program 2**

<html>

<head>

<title>College</title>

</head>

<body>

<table border="1"width="600">

<tr>

<td rowspan="2">S.no</td>

<td rowspan="2">Dep name</td>

<td colspan="2" align="center">Courses</td>

</tr>

<tr>

<td align="center">UG</td>

<td align="center">PG</td>

</tr>

<tr>

<td>1</td><td><a href="CS.html">Computer Science</a></td>

<td>Yes</td><td>yes</td>

</tr>

<tr>

<td>2</td><td><a href="B.com.html">Commerce</a></td>

<td>Yes</td><td>yes</td>

</tr>

<tr>

<td>3</td><td><a href="Maths.html">Marhs</a></td>

<td>No</td><td>yes</td>

</tr>

<tr>

<td>4</td><td><a href="Chemistry.html">Chemistry</a></td>

<td>yes</td><td>No</td>

</tr>

<tr>

<td>5</td><td><a href="Zoology.html">Zoology</a></td>

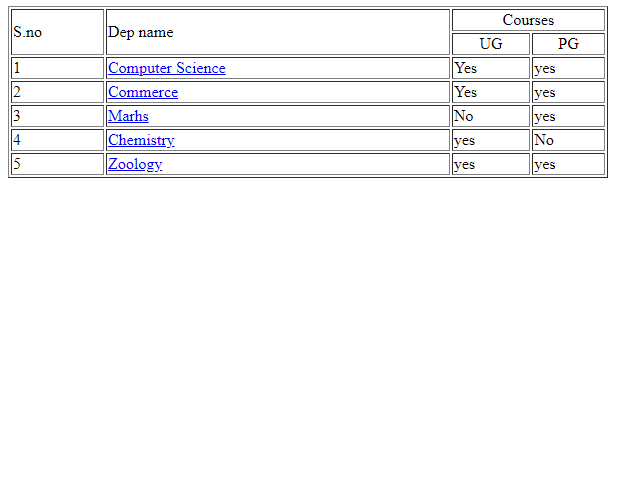
<td>yes</td><td>yes</td>

</tr>

</body>

</html>

**Output:**

****

**Program 3**

<html>

<head>

<title>validation</title>

</head>

<body>

<center>

<form action="webnew.html"method="post"enctype="multipart/form\_data">

Name<input type="text" id="name" placeholder="Enter name">

<br>

<br>

Email<input type="email" id="email"placeholder="Enter email">

<br>

<br>

<input type="file" id="fileUpload" accept="image/\*">

<br>

<br>

Female<input type="radio" id="female" name="male-female">

Male<input type="radio" id="male" name="male-female">

<br>

<br>

Password<input type="password" id="password" placeholder="Enter password">

<br>

<br>

<input type="submit" onclick="return f()">

</center>

</form>

</body>

<script>

function f()

{

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

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

var filePath=document.getElementById("fileUpload").value;

var allowedExtensions= /(\.jpg|\.jpeg|\.png|\.gif)$/i;

var female=document.getElementById("female");

var male=document.getElementById("male");

var pattern=document.getElementById("password").value;

if (name=="")

{

alert("please enter your name");

return false;

}

else if(email=="")

{

alert("please enter the email");

return false;

}

else if(!allowedExtensions.exec(filePath))

{

alert("Please choose image file");

return false;

}

else if(!male.checked && !female.checked){

alert("Please select gender");

return false;

}

else if(pattern.search(/[A-Z!@#$%^&\*\_=+-]/)<0)

{

alert("Password should contain special character");

return false;

}

else{

return true;

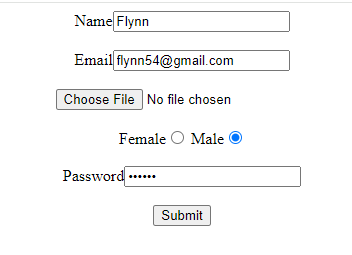
}

}

</script>

</html>

**Output:**

****

**Program 4**

<html>

<head>

<script src="https://cdnjs.cloudflare.com/ajax/libs/axios/1.2.3/axios.min.js"></script>

</head>

<body>

<h2>Using the<i>Axios NPM package</i>to fetch data from API</h2>

<div id="output"></div>

<button onclick="fetchData()">Fetch data using Axios</button>

<script>

let ouput=document.getElementById('output');

function fetchData(){

axios.get("https://jsonplaceholder.typicode.com/todos/1")

.then((response)=>{

output.innerHTML += "userId:" + response.data.userId + "</br/>";

output.innerHTML += "id:" + response.data.id + "</br/>";

output.innerHTML += "title:" + response.data.title + "</br/>";

output.innerHTML += "completed:" +response.data.completed + "</br/>";

})

.catch((er) => {

output.innerHTML += "The error is-" +err + "</br/>";

})

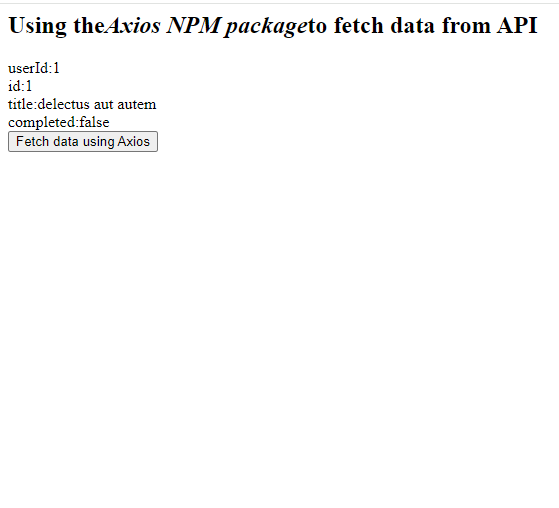
}

</script>

</body>

</html>

**Output:**

****

**Program 5**

**(index.html)**

<html>

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title>Login Form </title>

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

</head>

<body>

<div class="container">

<h2 class="t-center">Login</h2>

<form action="">

<div class="inputDiv">

<label for="name" class="username">Username</label>

<div class="userArea">

<i class="fa-solid fa-user"></i><input type="text" name="name" id="name"

placeholder="Type your username">

</div>

<hr>

</div>

<div class="inputDiv">

<label for="password" class="username">Password</label>

<div class="userArea">

<i class="fa-solid fa-key"></i><input type="password" name="password" id="password"

placeholder="Type your password">

</div>

<hr>

</div>

<button class="btn">Login</button>

</form>

</div>

</body>

</html>

**(style.css)**

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Ubuntu', sans-serif;

}

body {

width: 100%;

height: 100vh;

display: flex;

justify-content: center;

align-items: center;

background-color: #8fd5d5;

}

.container {

background-color: white;

justify-content: center;

align-items: center;

flex-direction: column;

padding: 1rem 3rem;

border-radius: .5rem;

}

.container form {

display: flex;

justify-content: center;

flex-direction: column;

position: relative;

}

.container h2 {

font-size: 1.5rem;

margin: 2rem;

}

.container form input {

font-weight: lighter;

border: none;

}

input:focus {

outline: none;

}

.container form .username {

margin-bottom: .7rem;

}

.container form i {

font-size: 1rem;

margin-right: 0.5rem;

}

.container form input {

font-size: 1rem;

}

.userArea {

margin-bottom: .7rem;

}

.inputDiv {

display: flex;

flex-direction: column;

padding-bottom: .8rem;

}

.formBottom {

display: flex;

justify-content: center;

align-items: center;

flex-direction: column;

}

.t-center {

text-align: center;

}

.btn {

border: none;

border-radius: 0.8rem;

cursor: pointer;

padding: 0.7rem 0.8rem;

background: linear-gradient(to right, #8d68c5, #e34494);

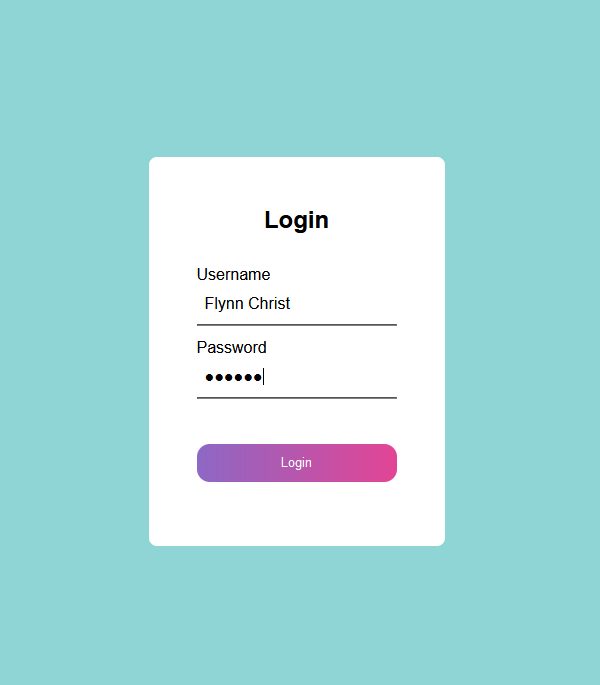
margin-top: 2rem;

margin-bottom: 3rem;

color: #fff;

}

**Output:**

****

**Program 6**

(index.html)

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title> Contact Form </title>

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

</head>

<body>

<div class="container">

<form>

<h2> GET IN TOUCH </h2>

<input type="text" id="name" placeholder="Enter Your Name" required>

<input type="email" id="email" placeholder="Enter Your Email" required>

<input type="phone" id="phone" placeholder="Enter Your Phone Number" required>

<input type="text" id="coupon" placeholder="Enter Coupen Code" required>

<textarea id="message" rows="4" placeholder="Write Something"></textarea>

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

</form>

</div>

</body>

</html>

(style.css)

\* {

margin: 0px;

padding: 0px;

font-family: "Poppins", sans-serif;

}

.container {

background-color: #eecbed;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

height: 100vh;

}

form {

background-color: white;

display: flex;

flex-direction: column;

padding: 2vw 4vw;

width: 60%;

max-width: 600px;

border-radius: 10px;

}

form h2 {

text-align: center;

color: #e65b00;

margin-bottom: 20px;

}

form input, textarea {

border: 0;

margin: 10px 0px;

padding: 20px;

outline: none;

background: #f5f5f5;

font-size: 16px;

border-radius: 10px;

resize: none;

}

form button {

background: white;

color: #e65b00;

border: 1px solid #e65b00;

padding: 15px;

font-size: 1rem;

outline: none;

cursor: pointer;

width: 100%;

font-weight: bold;

margin: 20px auto 0;

border-radius: 30px;

transition: all .5s ease-in;

}

form button:hover {

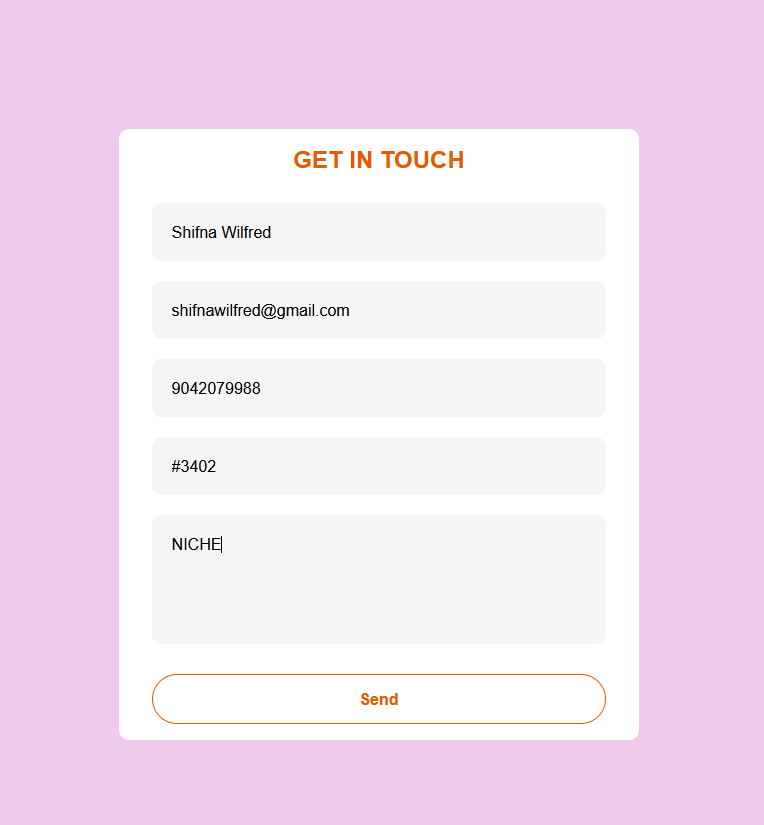
border: 1px solid #e65b00;

background: #e65b00;

color: white;

}

Output:



**Program 7**

(index.html)

<html>

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title> DropDown Menu</title>

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

</head>

<body>

<div class="navbar">

<div class="nav-left">

<h2>FLYNN CHRIST</h2>

</div>

<div class="nav-right">

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#pages">Pages</a></li>

<label for="service"></label>

<li class="services" id="dropdown">

<a href="#services">Services <i class="fa-solid fa-caret-down"></i></a>

<ul class="toggle">

<li><a href="#"> Content Writing </a></li>

<li><a href="#"> Web Development </a></li>

<li><a href="#"> YouTube Management </a></li>

<li><a href="#"> Social Media Management </a></li>

</ul>

</li>

<li><a href="#about">About Us</a></li>

<li><a href="#contact">Contact Us</a></li>

</ul>

</div>

</div>

</body>

<script src="script.js"></script>

</html>

(style.css)

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

.navbar {

background-color: rgb(11, 18, 54);

display: flex;

justify-content: space-between;

align-items: center;

padding: 1rem 2rem;

}

.nav-left h2 {

font-size: 2rem;

color: white;

font-family: 'Ubuntu', sans-serif;

}

.nav-right ul {

display: flex;

justify-content: center;

align-items: center;

position: relative;

}

.nav-right ul li {

list-style: none;

}

.nav-right ul li a {

color: white;

text-transform: capitalize;

text-decoration: none;

font-size: 1.1rem;

margin: 1rem;

font-family: 'Poppins', sans-serif;

}

.services {

position: relative;

transition: all .5s;

}

.nav-right .toggle {

display: none;

transition: all .5s linear;

}

.toggle.active {

display: flex;

position: absolute;

width: 20rem;

top: 100%;

left: 0;

margin-top: 1rem;

background-color: rgb(11, 18, 54);

flex-direction: column;

align-items: flex-start;

transition: all .5s linear;

}

.toggle li {

margin: 1rem 0;

}

.toggle li a {

font-size: 1.1rem;

}

(script.js)

const dropBtn = document.getElementById('dropdown')

dropBtn.addEventListener('click', function () {

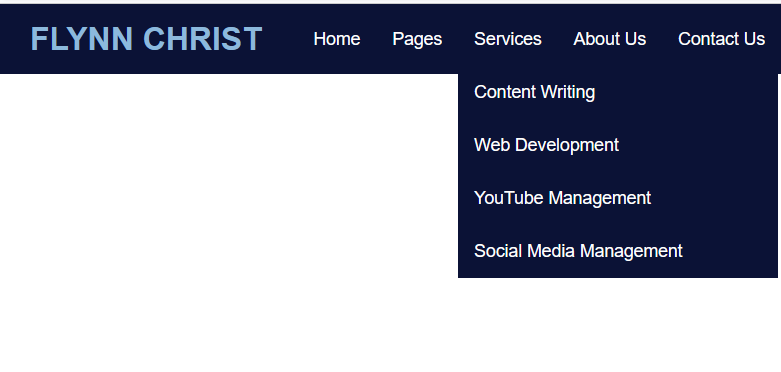
console.log('clicked');

const open = document.querySelector('.toggle')

open.classList.toggle('active')

})

**Output:**

****

**Program 8**

(index.JS)

var http=require('http');

var fs=require('fs');

http.CreateServer ((req,res)=>{

res.writeHead(200,{'content-Type':'text/html'});

var readStream =fs. createReadStream('index.html');

readStream.pipe(res);

}).listen(8080);

console.log("Server Running on port 8080");

(index.html)

<!DOCTYPE html>

<html lang=""en">

<head>

<meta charset=""UTF\_8">

<meta http-equiv="X\_UA\_Compatible"content="IE edge">

<meta name=""Viewport"Content"=width=device-width,initial-scale =1.0">

<title> HTML page </title>

</head>

<body>

    <h2>This is form html paqe <h2>

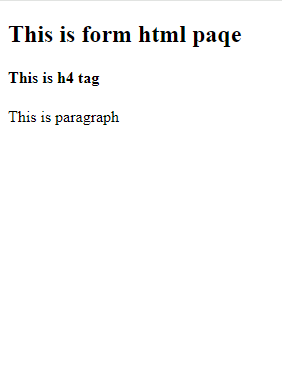
    <h4> This is h4 tag</h4>

    <p>This is paragraph</p>

</body>

</html>

**Output:**

****

**Program 9**

import React, { useState } from "react";

import "./App.css";

const App = () => {

const [counter, setCounter] = useState(0);

const handleClick1 = () => {

setCounter(counter + 1);

};

const handleClick2 = () => {

setCounter(counter - 1);

};

return (

<div

style={{

display: "flex",

flexDirection: "column",

alignItems: "center",

justifyContent: "center",

fontSize: "300%",

position: "absolute",

width: "100%",

height: "100%",

top: "-15%",

}}

>

Counter App

<div

style={{

fontSize: "120%",

position: "relative",

top: "10vh",

}}

>

{counter}

</div>

<div className="buttons">

<button

style={{

fontSize: "60%",

position: "relative",

top: "20vh",

marginRight: "5px",

backgroundColor: "green",

borderRadius: "8%",

color: "white",

}}

onClick={handleClick1}

>

Increment

</button>

<button

style={{

fontSize: "60%",

position: "relative",

top: "20vh",

marginLeft: "5px",

backgroundColor: "red",

borderRadius: "8%",

color: "white",

}}

onClick={handleClick2}

>

Decrement

</button>

</div>

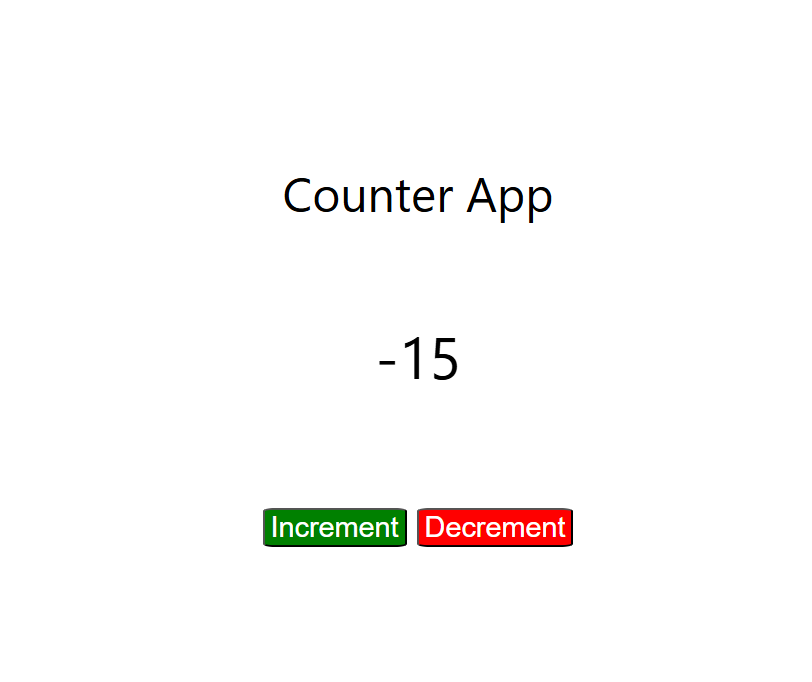
</div>

);

};

export default App;

**Output:**

****

**Program 10**

(app.js):

const express = require('express');

const bodyParser = require('body-parser');

const fs = require('fs-extra');

const path = require('path');

const { engine } = require('express-handlebars');

const app = express();

const PORT = process.env.PORT || 3000;

app.use(bodyParser.urlencoded({ extended: false }));

app.use(bodyParser.json());

app.engine('handlebars', engine({ extname: '.handlebars', defaultLayout: 'main' }));

app.set('view engine', 'handlebars');

app.set('views', path.join(\_\_dirname, 'views'));

app.use(express.static(path.join(\_\_dirname, 'public')));

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

res.render('form');

});

app.post('/submit', async (req, res) => {

const data = req.body;

const filePath = path.join(\_\_dirname, 'data', 'data.json');

try {

await fs.writeJson(filePath, data, { spaces: 2 });

res.redirect('/display');

} catch (err) {

console.error(err);

res.status(500).send('Internal Server Error');

}

});

app.get('/display', async (req, res) => {

const filePath = path.join(\_\_dirname, 'data', 'data.json');

try {

const data = await fs.readJson(filePath);

res.render('display', { data });

} catch (err) {

console.error(err);

res.status(500).send('Internal Server Error');

}

});

app.listen(PORT, () => {

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

});

Main.handlebar:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

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

<title>Express Form App</title>

</head>

<body>

{{{body}}}

</body>

</html>

From.handlebars:

<form action="/submit" method="POST">

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

<input type="text" id="name" name="name" required>

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

<input type="email" id="email" name="email" required>

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

</form>

Display.handlebars:

<h1>Stored Data</h1>

<p>Name: {{data.name}}</p>

<p>Email: {{data.email}}</p>

<a href="/">Back to Form</a>

Display.handlebar:

body {

font-family: Arial, sans-serif;

margin: 20px;

}

form {

display: flex;

flex-direction: column;

max-width: 300px;

}

label, input {

margin-bottom: 10px;

}

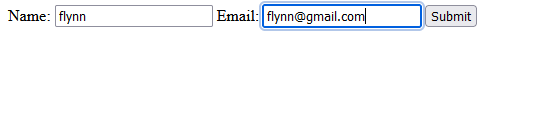
button {

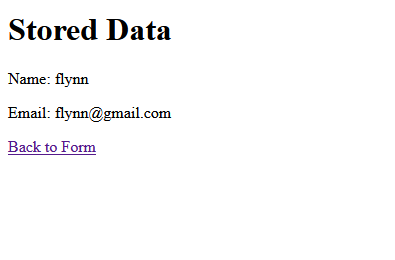
width: 100px;

padding: 5px;

}

**Output:**

****

****

**Program 11**

const express = require('express');

const mongoose = require('mongoose');

const bodyParser = require('body-parser');

const app = express();

app.use(bodyParser.urlencoded({ extended: true }));

app.set('view engine', 'ejs');

mongoose.connect('mongodb://localhost:27017/studentsDB')

.then(() => console.log('Connected to MongoDB'))

.catch(err => console.error('Could not connect to MongoDB', err));

const studentSchema = new mongoose.Schema({

name: String,

age: Number,

grade: String

});

const Student = mongoose.model('Student', studentSchema);

app.get('/', async (req, res) => {

try {

const students = await Student.find({});

res.render('index', { studentsList: students });

} catch (err) {

res.status(500).send(err);

}

});

app.get('/add', (req, res) => {

res.render('add');

});

app.post('/add', async (req, res) => {

const newStudent = new Student({

name: req.body.name,

age: req.body.age,

grade: req.body.grade

});

try {

await newStudent.save();

res.redirect('/');

} catch (err) {

res.status(500).send(err);

}

});

app.get('/edit/:id', async (req, res) => {

try {

const student = await Student.findById(req.params.id);

res.render('edit', { student: student });

} catch (err) {

res.status(500).send(err);

}

});

app.post('/edit/:id', async (req, res) => {

try {

await Student.findByIdAndUpdate(req.params.id, {

name: req.body.name,

age: req.body.age,

grade: req.body.grade

});

res.redirect('/');

} catch (err) {

res.status(500).send(err);

}

});

app.get('/delete/:id', async (req, res) => {

try {

console.log(`Deleting student with ID: ${req.params.id}`);

await Student.findByIdAndDelete(req.params.id);

console.log('Student deleted successfully');

res.redirect('/');

} catch (err) {

console.error(`Error deleting student: ${err}`);

res.status(500).send(err);

}

});

const PORT = 3001;

app.listen(PORT, () => {

console.log(`Server started on port ${PORT}`);

});

Index.ejs:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Students List</title>

</head>

<body>

<h1>Students List</h1>

<a href="/add">Add New Student</a>

<ul>

<% studentsList.forEach(student => { %>

<li>

<%= student.name %> - <%= student.age %> - <%= student.grade %>

<a href="/edit/<%= student.\_id %>">Edit</a>

<a href="/delete/<%= student.\_id %>">Delete</a>

</li>

<% }) %>

</ul>

</body>

</html>

Add.ejs:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Add Student</title>

</head>

<body>

<h1>Add New Student</h1>

<form action="/add" method="POST">

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

<input type="text" name="name" required><br>

<label for="age">Age:</label>

<input type="number" name="age" required><br>

<label for="grade">Grade:</label>

<input type="text" name="grade" required><br>

<button type="submit">Add Student</button>

</form>

</body>

</html>

Edit.ejs:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Edit Student</title>

</head>

<body>

<h1>Edit Student</h1>

<form action="/edit/<%= student.\_id %>" method="POST">

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

<input type="text" name="name" value="<%= student.name %>" required><br>

<label for="age">Age:</label>

<input type="number" name="age" value="<%= student.age %>" required><br>

<label for="grade">Grade:</label>

<input type="text" name="grade" value="<%= student.grade %>" required><br>

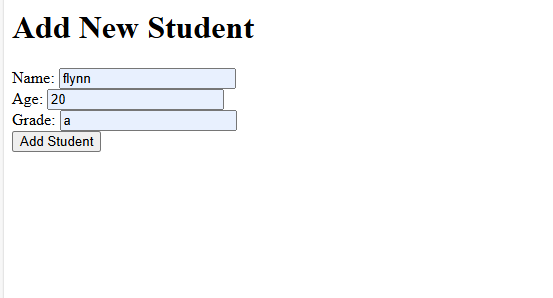
<button type="submit">Update Student</button>

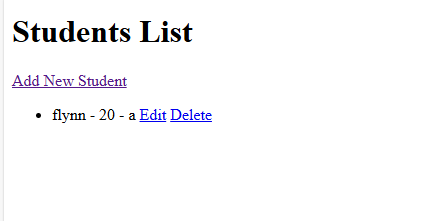
</form>

</body>

</html>

**Output:**

****

****