

PROGRAM-1

Write a program to develop a static website using HTML tables.

Program:

index.html

```
<!DOCTYPE html>

<html>

<head>

    <title>Home</title>

</head>

<body>

<center>

</center>

<br><br>

<center>

<table border="1" width="70%" cellpadding="10" cellspacing="5">

    <tr align="center">

        <td><a href="index.html">Home</a></td>

        <td><a href="about.html">About</a></td>

        <td><a href="vision.html">Vision</a></td>

        <td><a href="mission.html">Mission</a></td>

        <td><a href="register.html">Register</a></td>

    </tr>

</table>

</center>

<br><br>

<center>

    
```

```
<h1>CSE Department</h1>
<p>Welcome to the Computer Science And Engineering Department.</p>
<br>
<p>Designed by 1604-22-733-121</p>
</center>
</body>
</html>
```

about.html

```
<!DOCTYPE html>
<html>
<head>
  <title>About</title>
</head>
<body>
<center><h1>About Us</h1></center>
<center>
<table border="1" width="70%" cellpadding="10" cellspacing="5">
  <tr align="center">
    <td><a href="index.html">Home</a></td>
    <td><a href="about.html">About</a></td>
    <td><a href="vision.html">Vision</a></td>
    <td><a href="mission.html">Mission</a></td>
    <td><a href="register.html">Register</a></td>
  </tr>
</table>
</center>
<br><br>
```

<center>

<p>Established in 1980, Muffakham Jah College of Engineering and Technology is a premier institute of its kind, offering four year B.E. degree courses in 7 Engineering branches, namely, Civil Engineering, Computer Science Engineering, Computer Science Engineering (Artificial Intelligence), Computer Science Engineering (Artificial Intelligence & Machine Learning), Computer Science Engineering (Data Science), Electronics and Communication Engineering, Mechanical Engineering and four post graduate courses in M.E. (CAD/CAM), M.E. (Structural Engineering), M.E. (Embedded Systems & VLSI Design), M. Tech. (Computer Science) of two years duration. The College is a Minority Autonomous Educational Institution affiliated to Osmania University and is approved by the AICTE and accredited by the NAAC with Grade A+ and four UG programs Civil Engineering, Computer Science and Engineering, Electronics and Communication Engineering and Mechanical Engineering are accredited by NBA for a period of 3 years from 2024-25 to 2026-27 up to 30-6-2027.</p>

</center>

</body>

</html>

vision.html

<!DOCTYPE html>

<html>

<head>

<title>Vision</title>

</head>

<body>

<center><h1>Vision</h1></center>

<center>

<table border="1" width="70%" cellpadding="10" cellspacing="5">

<tr align="center">

<td>Home</td>

<td>About</td>

```

        <td><a href="vision.html">Vision</a></td>

        <td><a href="mission.html">Mission</a></td>

        <td><a href="register.html">Register</a></td>

    </tr>
</table>
</center>
<br><br>
<center>

    <p>To empower the faculty and students in the area of Research & Development by
    providing seed funds for implementing their innovative research and product development
    ideas.</p>

</center>
</body>
</html>

```

mission.html

```

<!DOCTYPE html>

<html>

<head>

    <title>Mission</title>

</head>

<body>

<center><h1>Mission</h1></center>

<center>

<table border="1" width="70%" cellpadding="10" cellspacing="5">

    <tr align="center">

        <td><a href="index.html">Home</a></td>

        <td><a href="about.html">About</a></td>

```

<td>Vision</td>

<td>Mission</td>

<td>Register</td>

</tr>

</table>

</center>

<center>

<p>

To motivate faculty and students to undertake Research and Development activities as a means of nourishing innovative thought process.

</p>

<p>

To encourage interdisciplinary Research and Development projects leading to solutions for real world problems through the synthesis of diverse ideas and skills.

</p>

<p>

To serve as a medium for three-way interaction between the Institute, Research and Development organizations, and Industry.

</p>

<p>

To assist faculty, Ph.D. scholars, and students to apply for funding under various Government, Professional Chapter, or Private Sector schemes.

</p>

<p>

To encourage publication of technical papers in National and International refereed journals and conferences based on research carried out by faculty

and students.

</p>

<p>

To register outcomes of Research and Development work under Intellectual Property Rights such as patents and copyrights.

</p>

<p>

To facilitate signing of Memorandum of Understanding (MoU) with industries and Research and Development organizations for research and product development.

</p>

</center>

</body>

</html>

register.html

<!DOCTYPE html>

<html>

<head>

<title>Registration</title>

</head>

<body>

<center><h1>Student Registration</h1></center>

<center>

<table border="1" width="70%" cellpadding="10" cellspacing="5">

<tr align="center">

<td>Home</td>

<td>About</td>

```

<td><a href="vision.html">Vision</a></td>

<td><a href="mission.html">Mission</a></td>

<td><a href="register.html">Register</a></td>

</tr>

</table>

</center>

<br><br> <!-- GAP HERE -->

<center>

<form>

    Name: <input type="text"><br><br>

    Email: <input type="text"><br><br>

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

</form>

</center>

</body>

</html>

```

Output:



Home	About	Vision	Mission	Register
----------------------	-----------------------	------------------------	-------------------------	--------------------------



CSE Department

Welcome to the Computer Science And Engineering Department.

Designed by 1604-22-733-121

About Us

Home	About	Vision	Mission	Register
----------------------	-----------------------	------------------------	-------------------------	--------------------------



Established in 1980, Muffakham Jah College of Engineering and Technology is a premier institute of its kind, offering four year B.E. degree courses in 7 Engineering branches, namely, Civil Engineering, Computer Science Engineering, Computer Science Engineering (Artificial Intelligence), Computer Science Engineering (Artificial Intelligence & Machine Learning), Computer Science Engineering (Data Science), Electronics and Communication Engineering, Mechanical Engineering and four post graduate courses in M.E. (CAD/CAM), M.E. (Structural Engineering), M.E. (Embedded Systems & VLSI Design), M. Tech. (Computer Science) of two years duration. The College is a Minority Autonomous Educational Institution affiliated to Osmania University and is approved by the AICTE and accredited by the NAAC with Grade A+ and four UG programs Civil Engineering, Computer Science and Engineering, Electronics and Communication Engineering and Mechanical Engineering are accredited by NBA for a period of 3 years from 2024-25 to 2026-27 up to 30-6-2027.

Vision

Home	About	Vision	Mission	Register
----------------------	-----------------------	------------------------	-------------------------	--------------------------



To empower the faculty and students in the area of Research & Development by providing seed funds for implementing their innovative research and product development ideas.

Mission

Home	About	Vision	Mission	Register
----------------------	-----------------------	------------------------	-------------------------	--------------------------



- To motivate faculty and students to undertake Research and Development activities as a means of nourishing innovative thought process.
- To encourage interdisciplinary Research and Development projects leading to solutions for real world problems through the synthesis of diverse ideas and skills.
- To serve as a medium for three-way interaction between the Institute, Research and Development organizations, and Industry.
- To assist faculty, Ph.D. scholars, and students to apply for funding under various Government, Professional Chapter, or Private Sector schemes.
- To encourage publication of technical papers in National and International refereed journals and conferences based on research carried out by faculty and students.
- To register outcomes of Research and Development work under Intellectual Property Rights such as patents and copyrights.
- To facilitate signing of Memorandum of Understanding (MoU) with industries and Research and Development organizations for research and product development.

Student Registration

Home	About	Vision	Mission	Register
----------------------	-----------------------	------------------------	-------------------------	--------------------------

Name:

Email:

Program-2

Develop a static website using DIV and CSS

Index.html

```
<!DOCTYPE html>

<html>

<head>

    <title>Home Page</title>

    <!-- External CSS -->

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

    <!-- Internal CSS -->

    <style>

        #mainTitle {

            color: darkblue;

            text-decoration: underline;

        }

    </style>

</head>

<body>

<div class="content">

</div>

<br>

<div class="nav">

    <div><a href="index.html">Home</a></div>

    <div><a href="about.html">About</a></div>

    <div><a href="vision.html">Vision</a></div>

    <div><a href="mission.html">Mission</a></div>

    <div><a href="register.html">Register</a></div>
```

```
</div>

<br>

<div class="content">

    <h1 id="mainTitle">Computer Science Engineering</h1>

    <p style="color: darkgreen;">

        Welcome to the Computer Science Engineering Department.

    </p>

    <p>

        Designed by <b>1604-22-733-121</b>

    </p>

</div>

</body>

</html>
```

style.css

```
body {

    font-family: Arial, sans-serif;

    background-color: #f5f5f5;

}

.nav {

    width: 75%;

    margin: auto;

    border: 2px solid black;

    text-align: center;

    background-color: #e6e6e6;

}

.nav div {
```

```
display: inline-block;

padding: 12px;

}

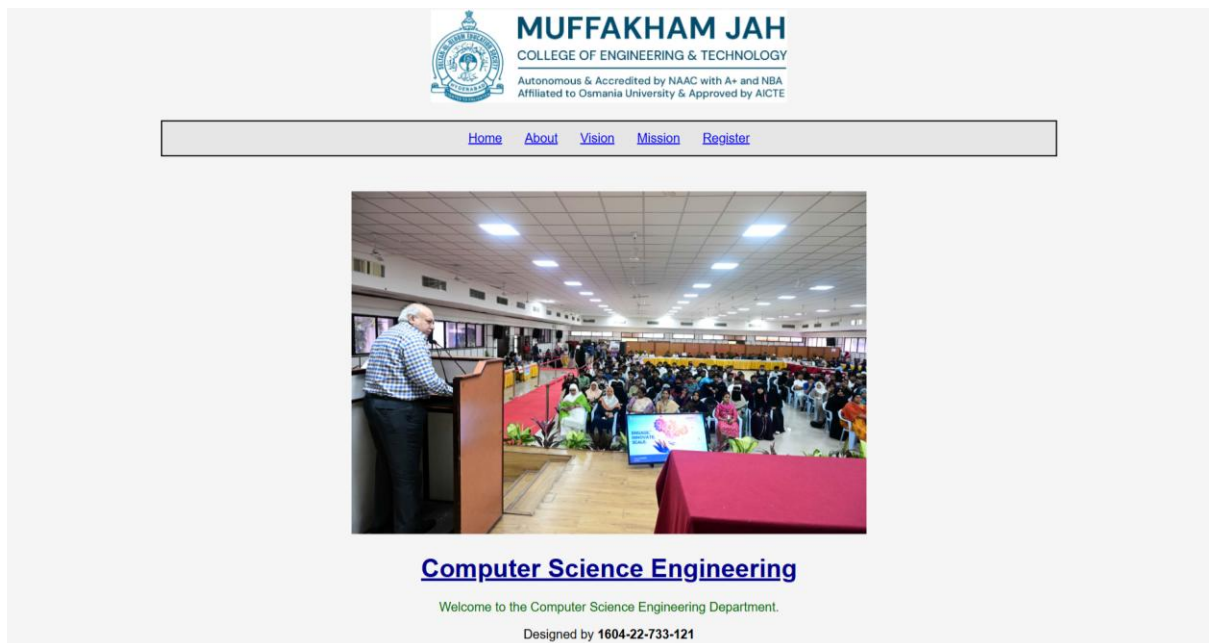
.content {

text-align: center;

margin-top: 25px;

}
```

Output:



Program-3

Develop a registration page using HTML forms

Index.html

```
<!DOCTYPE html>

<html>

<head>

    <!-- Title of the webpage -->

    <title>Registration Form</title>

    <!-- Linking external CSS file -->

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

</head>

<body>

<!-- Main container to center the form -->

<div class="container">

    <!-- Form heading -->

    <h2>Student Registration Form</h2>

    <!-- Form starts here -->

    <form>

        <!-- TEXT FIELD: Name -->

        <label>Name</label>

        <input type="text" placeholder="Enter your name">

        <!-- PASSWORD FIELD -->

        <label>Password</label>

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

        <!-- DROPDOWN LIST -->

        <label>Branch</label>

        <select>

            <option>Select Branch</option>
```

```
<option>CSE</option>
<option>ECE</option>
<option>MECH</option>
<option>CIVIL</option>
</select>

<!-- LIST BOX (Multiple selection using Ctrl key) -->
<label>Subjects (Hold Ctrl to select multiple)</label>
<select multiple size="4">
    <option>DSA</option>
    <option>DBMS</option>
    <option>OS</option>
    <option>AI</option>
</select>

<!-- RADIO BUTTONS (Select only one) -->
<label>Gender</label>
<div class="options">
    <input type="radio" name="gender"> Male
    <input type="radio" name="gender"> Female
</div>

<!-- CHECKBOXES (Select one or more) -->
<label>Skills</label>
<div class="options">
    <input type="checkbox"> Python
    <input type="checkbox"> Java
    <input type="checkbox"> C
</div>

<!-- TEXT AREA -->
<label>Address</label>
```

```
<textarea rows="4"></textarea>

<!-- SUBMIT & RESET BUTTONS (Inline CSS used here) -->

<div class="buttons">

    <input type="submit" value="Submit"

        style="background-color: navy; color: white;">

    <input type="reset" value="Reset"

        style="background-color: gray; color: white;">

</div>

</form>

<!-- Form ends here -->

</div>

<!-- Container ends here -->

</body>

</html>
```

form.css

```
/* Style applied to entire page */

body {

    font-family: Arial, sans-serif;

    background-color: #f0f2f5;

}

/* Container div to center the form */

.container {

    width: 450px;

    background-color: white;

    padding: 20px;

    margin: 50px auto;

    border-radius: 8px;
```

```
    box-shadow: 0 0 10px gray;

    box-sizing: border-box;
}

/* Heading style */
h2 {
    text-align: center;
    color: navy;
}

/* Label styling */
label {
    font-weight: bold;
    display: block;
    margin-top: 10px;
}

/* Common style for inputs, dropdowns, textarea */
input, select, textarea {
    width: 100%;
    padding: 8px;
    margin-top: 5px;
    box-sizing: border-box;
}

/* Radio & checkbox input size correction */
.options input {
    width: auto;
    margin-right: 5px;
}

/* Space above option groups */
.options {
```

```

        margin-top: 5px;
    }

    /* Button container */
    .buttons {
        text-align: center;
        margin-top: 15px;
    }

    /* Button styling */
    .buttons input {
        padding: 8px 20px;
        margin: 5px;
        border: none;
        cursor: pointer;
    }

```

Output:

Student Registration Form

Name

Password

Branch

Subjects (Hold Ctrl to select multiple)

DSA

DBMS

OS

AI

Gender
☐ Male ☒ Female

Skills
☒ Python ☒ Java ☒ C

Address

Submit

Reset

Program-4

Develop a dynamic web page using JavaScript

```
<!DOCTYPE html>

<html>

<head>

    <title>Dynamic Web Page</title>

</head>

<body>

<center>

    <h2 id="greeting"></h2>

    <p id="date"></p>

    <h3>Your Lucky Number</h3>

    <p id="lucky"></p>

    <p id="modified"></p>

    <p> Modified by: 1604-22-733-121</p>

</center>

<script>

    // Date object

    var today = new Date();

    var hour = today.getHours();

    // Greeting based on time

    var greet;

    if (hour < 12) {

        greet = "Good Morning";

    } else if (hour < 18) {

        greet = "Good Afternoon";
```

```
} else {  
    greet = "Good Evening";  
}  
  
// Display greeting and date  
document.getElementById("greeting").innerHTML = greet;  
document.getElementById("date").innerHTML = "Today's Date: " + today.toDateString();  
  
// Math object - random lucky number  
var luckyNumber = Math.floor(Math.random() * 100) + 1;  
document.getElementById("lucky").innerHTML = luckyNumber;  
  
// Last modified information  
document.getElementById("modified").innerHTML =  
    "Last Modified: " + document.lastModified;  
</script>  
</body>  
</html>
```

Output:

Good Evening

Today's Date: Tue Dec 16 2025

Your Lucky Number

84

Last Modified: 12/16/2025 23:00:31

Modified by: 1604-22-733-121

Program-5

Write a JavaScript program to validate the registration

```
<html>

<head>

<title>Registration Form Validation</title>

<script>

function validate() {

    /* Roll Number validation */

    var rn = document.getElementById("roll");

    if (rn.value == "") {

        alert("Roll Number cannot be empty");

        return false;

    }

    if (rn.value.search(/[0-9]{4}-[0-9]{2}-[0-9]{3}-[0-9]{3}$/) == -1) {

        alert("Invalid Roll Number");

        return false;

    }

    /* Name validation */

    var sn = document.getElementById("sname");

    if (sn.value == "") {

        alert("Name cannot be empty");

        return false;

    }

    if (sn.value.search(/[A-Z a-z]*$/ ) == -1) {

        alert("Invalid Name");

        return false;

    }

    /* Password match validation */
```

```
var pass = document.getElementById("pwd");
var cpass = document.getElementById("cpwd");
if (pass.value == "" || cpass.value == "") {
    alert("Password fields cannot be empty");
    return false;
}
if (pass.value != cpass.value) {
    alert("Passwords do not match");
    return false;
}
/* Phone number validation */
var pho = document.getElementById("ph");
if (pho.value == "") {
    alert("Phone number cannot be empty");
    return false;
}
if (pho.value.search(/[0-9]{10}$/) == -1) {
    alert("Invalid Phone Number");
    return false;
}
/* Email validation */
var em = document.getElementById("email");
if (em.value.search(/[A-Z a-z 0-9._]*@[A-Z a-z]*\.[A-Z a-z]*$/) == -1) {
    alert("Invalid Email Address");
    return false;
}
return true;
}
```

```
</script>
</head>
<body>
  <h1>Student Registration Form</h1>
  <form onsubmit="return validate();">
    <table border="1">
      <tr><td>Roll No</td><td><input type="text" id="roll"></td></tr>
      <tr><td>Name</td><td><input type="text" id="sname"></td></tr>
      <tr><td>Password</td><td><input type="password" id="pwd"></td></tr>
      <tr><td>Confirm Password</td><td><input type="password" id="cpwd"></td></tr>
      <tr><td>Phone</td><td><input type="text" id="ph"></td></tr>
      <tr><td>Email</td><td><input type="text" id="email"></td></tr>
      <tr><td>Address</td><td><textarea id="addr"></textarea></td></tr>
      <tr>
        <td></td>
        <td>
          <input type="submit">
          <input type="reset">
        </td>
      </tr>
    </table>
  </form>
</body>
</html>
```

Output:

Student Registration Form

Roll No	1604-22-733121
Name	Ameena Begum
Password	*****
Confirm Password	*****
Phone	7458765231
Email	ameenahsyed2003@gmail.
Address	2-3-5/354
	<input type="button" value="Submit"/> <input type="button" value="Reset"/>

This page says

Invalid Roll Number

OK

Student Registration Form

Roll No	1604-22-733-121
Name	Ameena Begum
Password	
Confirm Password	
Phone	7458765231
Email	ameenahsyed2003@gmail.
Address	2-3-5/354
	<input type="button" value="Submit"/> <input type="button" value="Reset"/>

This page says

Password fields cannot be empty

OK

Program-6

Create a dynamic web page using JavaScript event handling and DOM manipulation.

```
<!DOCTYPE html>

<html>

<head>

  <title>JavaScript Event Handling</title>

  <script>

    function showInstructions() {

      alert("Read the instructions carefully");

    }

    function validateForm() {

      alert("Form submitted successfully");

    }

    function clearForm() {

      alert("Form cleared");

    }

    /* onclick event + DOM manipulation */

    function copyAddress() {

      var contact = document.getElementById("cta").value;

      document.getElementById("pta").value = contact;

    }

    function focusPhone() {

      document.getElementById("phone").style.color = "red";

    }

    function blurPhone() {

      document.getElementById("phone").style.color = "black";

    }

  </script>
```

```
</head>

<body onload="showInstructions()">

<center>

  <h1>JavaScript Event Handling Demo</h1>

  <form onsubmit="validateForm()" onreset="clearForm()">

    Name:

    <input type="text" id="sname" onblur="alert('Name field left')">

    <br><br>

    Phone No:

    <input type="text" id="phone"

      onfocus="focusPhone()"

      onblur="blurPhone()">

    <br><br>

    Contact Address:

    <textarea id="cta"></textarea>

    <br><br>

    Permanent Address:

    <textarea id="pta"></textarea>

    <br><br>

    <input type="button"

      value="Same as Contact Address"

      onclick="copyAddress()">

    <br><br>

    <!-- Mouseover & Mouseout demo -->

    <p id="hoverText">Move mouse over this text</p>

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

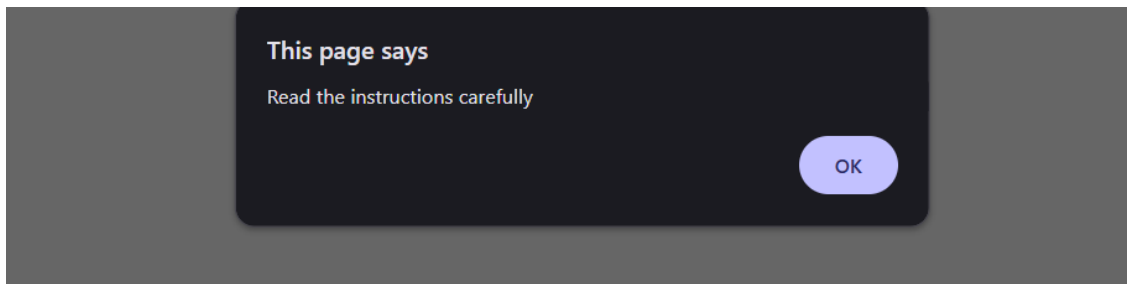
    <input type="reset" value="Reset">

    <br><br>
```



```
        <input type="button" value="Button-3" id="btn3">
    </form>
</center>
<!-- DOM handling using addEventListener -->
<script>
    /* mouseover & mouseout using DOM */
    var txt = document.getElementById("hoverText");
    txt.onmouseover = function () {
        txt.style.color = "blue";
        alert("Mouse Over Event");
    };
    txt.onmouseout = function () {
        txt.style.color = "black";
        alert("Mouse Out Event");
    };
    /* click event using addEventListener */
    var btn = document.getElementById("btn3");
    btn.addEventListener("click", function () {
        alert("Button-3 Clicked Listener 1");
    });
    btn.addEventListener("click", function () {
        alert("Button-3 Clicked Listener 2");
    });
</script>
</body>
</html>
```

Output:



JavaScript Event Handling Demo

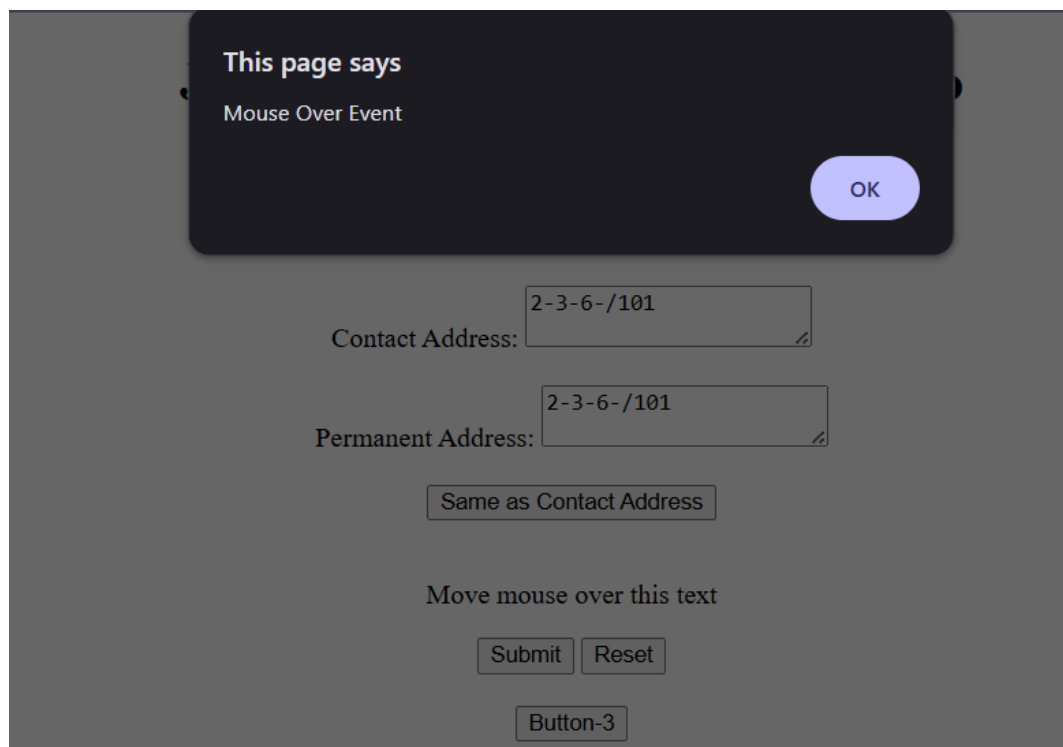
Name:

Phone No:

Contact Address:

Permanent Address:

Move mouse over this text



Program-7

Demonstrate working of built-in node.js modules and file system.

index.js

```
var http = require('http');

var os = require('os');

var fs = require('fs');

// HTTP Server

var server = http.createServer(function (req, res) {

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

    res.write('Hello Web Technologies Lab <br>');

    res.write('Date: ' + Date() + '<br>');

    res.write('Platform: ' + os.platform() + '<br>');

    res.write('Architecture: ' + os.arch() + '<br>');

    res.write('Host Name: ' + os.hostname() + '<br>');

    res.write('OS Type: ' + os.type() + '<br>');

    var text = fs.readFileSync('content.txt', 'utf8');

    res.write('<br>File Content:<br>' + text);

    res.end();

});

server.listen(8080);

// File System Operations

fs.open('myfile1.txt', 'w', function (err, file) {

    if (err) throw err;

    console.log('File opened');

});

fs.writeFile('myfile3.txt', 'hello lab', function (err) {

    if (err) throw err;

    console.log('File written');
```

```

});

fs.appendFile('myfile1.txt', ' hello web', function (err) {

    if (err) throw err;

    console.log('File appended');

});

fs.rename('myfile3.txt', 'myfile4.txt', function (err) {

    if (err) throw err;

    console.log('File renamed');

});

```

Output:

```

PS C:\Users\Ameena\OneDrive\Desktop\7th SEM\WT LAB> cd 7
PS C:\Users\Ameena\OneDrive\Desktop\7th SEM\WT LAB\7> node index.js
File opened
File renamed
File written
File appended

```

```

JS index.js U  myfile1.txt U X
7 > myfile1.txt
1  hello web

```

```

JS index.js U X  myfile4.txt U X
7 > myfile4.txt
1  hello lab

```

```

<  >  ↻  ⓘ  localhost:8080

```

Hello Web Technologies Lab
 Date: Thu Dec 18 2025 22:20:01 GMT+0530 (India Standard Time)
 Platform: win32
 Architecture: x64
 Host Name: LAPTOP-0OBP2H49
 OS Type: Windows_NT

File Content:

Program-8

Demonstrate routes and parameter handling in Express.js

login.js

```
var express=require('express');  
var router=express.Router();  
router.get('/',(req,res)=>{  
  res.send('Login Page');  
});  
router.post('/',(req,res)=>{  
  res.send('Login Page');  
});  
module.exports=router;
```

signup.js

```
var express=require('express');  
var router=express.Router();  
router.get('/',(req,res)=>{  
  res.send('Sign Up Page');  
});  
router.post('/',(req,res)=>{  
  res.send('Signup page');  
});  
module.exports=router;
```

index.js

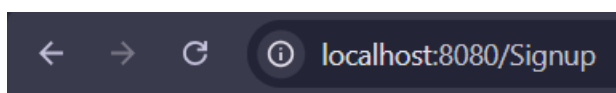
```
var express=require('express')  
var app=express();  
var loginroute=require('./Routes/login');
```

```
var signuproute=require('./Routes/signup');  
app.get('/',(req,res)=>{  
  res.send('Hello Lab');  
});  
app.use('/login',loginroute);  
app.use('/signup',signuproute);  
app.get('/display/:uname/:pwd',(req,res)=>{  
  var values=req.params;  
  console.log(values);  
  console.log(req.query);  
  res.send('Values Received :'+req.params.uname);  
});  
app.listen(8080);
```

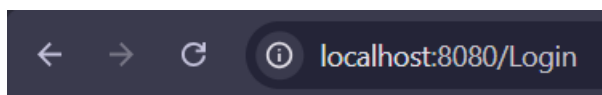
Output



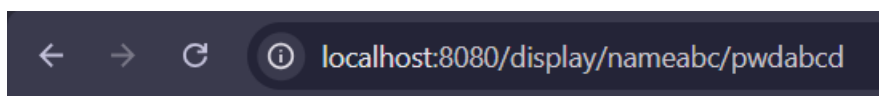
Hello Lab



Sign Up Page



Login Page



Values Received : nameabc

PROGRAM-9

Write a program to implement the MVC Architecture

Memo.ejs

```
<html>

<head>

<title>

Marks Memo

</title>

</head>

<body>

<center>

<h1>Marks Memo</h1>

<h2> 1604-22-733-121</h2>

<table width="60%" border="1">

<tr>

<th>S.No</th>

<th>Code</th>

<th>Subject</th>

<th>Internal Marks</th>

<th>External Marks</th>

</tr>

<% data.forEach(function(data){ %>

<tr align="center">

<td><%= data.id %></td>

<td><%= data.code %></td>

<td align="left"><%= data.sub %></td>

<td><%= data.imarks %></td>

<td><%= data.emarks %></td>
```

```
</tr>
<% }} %>
</table>
</center>
</body>
</html>
```

index.js

```
var express = require("express");
var app = express();
app.set("view engine", "ejs");
app.set("views", __dirname + "/views");
var data = [
  { id: 1, code: "CS401", sub: "DAA", imarks: 28, emarks: "S" },
  { id: 2, code: "CS402", sub: "CN", imarks: 30, emarks: "A" },
  { id: 3, code: "CS403", sub: "OS", imarks: 22, emarks: "B" },
  { id: 4, code: "CS404", sub: "Java", imarks: 25, emarks: "C" }
];
app.get("/", (req, res) => {
  res.render("memo", { data });
});
app.listen(8080, () => {
  console.log("Server running on port 8080");
});
```


Output:



Marks Memo

1604-22-733-121

S.No	Code	Subject	Internal Marks	External Marks
1	CS401	DAA	28	S
2	CS402	CN	30	A
3	CS403	OS	22	B
4	CS404	Java	25	C

PROGRAM 10

Demonstrate rendering HTML and JSX using React.js

```
import './App.css';

function App() {

  const collegeName = "Muffakham Jah College";

  const links = (

    <div>

      <a href="#">Faculty</a><br />

      <a href="#">Staff</a><br />

      <a href="#">Students</a><br />

    </div>

  );

  const today = new Date();

  return (

    <div className="App">

      <h1>{collegeName}</h1>

      <h2 style={{ color: "blue" }}>CSE Department</h2>

      {links}

      <p>{today.toDateString()}</p>

      <h3>Mission</h3>

      <p>

        To mentor students towards a successful professional career in a

        global environment through quality education.

      </p>

    </div>

  );

}
```

export default App;

Output:



Muffakham Jah College

CSE Department

[Faculty](#)
[Staff](#)
[Students](#)

Thu Dec 18 2025

Mission

To mentor students towards a successful professional career in a global environment through quality education.

Designed by: 1604-22-733-121

PROGRAM 11

Demonstrate use of props, events, lists, forms using React.js

```
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <GetUser name="ABC" uid={100} />
      <GetUser name="XYZ" uid={101} />
      <MyForm />
      <StudentsData />
      <EventDemo />
    </div>
  );
}

function GetUser(props) {
  return(
    <div>
      <p>{props.name}</p>
      <p>{props.uid}</p>
    </div> )
}

function MyForm() {
  return (
    <form>

    Name: <input type='text'></input><br></br>
    Email: <input type='text'></input><br></br>
    Password: <input type='text'></input><br></br>
```

```
<input type='Submit'></input><br></br>
<input type='Reset'></input><br></br>
</form>
)
}
function StudentsData()
{
var list=[
{roll:1,name:'AAA'},
{roll:2,name:'BBB'},
{roll:3,name:'CCC'},
{roll:4,name:'DDD'}
];
return(
<>
<h1>Students List</h1>
<ul>
{list.map((row) => <li key={row.roll}>{row.name}</li>)}
</ul>
</>
)
}
function EventDemo()
{
var display=() =>{
alert('Hello');
}
return(
```

```
<button onClick={display}>Click Here</button>

);

}

export default App;
```

Output:

A screenshot of a web browser window with the address bar showing 'localhost:3000'. The page content includes:

- Name: Ameen
- ID: 201
- Name: Sara
- ID: 202
- Username:
- Email:
- Password:
-
-
- Student Names**
- Tom
- Jerry
- Spike
- Tyke
-

A screenshot of the same web browser window after a button click. A dark modal dialog box is displayed with the text 'localhost:3000 says' and 'Button Clicked!', and an 'OK' button. The background form is dimmed and shows updated data:

- Username:
- Email:
- Password:
-
-
- Student Names**
- Tom
- Jerry
- Spike
- Tyke
-

PROGRAM-12

Create a Single Page Application (SPA) using REST service.

RestServices/Index.js

```
var express = require('express');

var cors = require('cors');

var fs = require("fs");

var app = express();

app.use(cors());

app.get('/students', function(req, res) {

  fs.readFile(__dirname + "/students.json", 'utf8', function(err, data) {

    res.end(data);

  });

});

app.get('/imarks', function(req, res) {

  fs.readFile(__dirname + "/imarks.json", 'utf8', function(err, data) {

    res.end(data);

  });

});

app.get('/emarks', function(req, res) {

  fs.readFile(__dirname + "/emarks.json", 'utf8', function(err, data) {

    res.end(data);

  });

});

app.listen(5000, () => {

  console.log("REST Service running on port 8080");

});
```

JSON -----

emarks.json

```
[  
  { "id": 2001, "marks": 85 },  
  { "id": 2002, "marks": 78 },  
  { "id": 2003, "marks": 90 },  
  { "id": 2004, "marks": 88 },  
  { "id": 2005, "marks": 92 }  
]
```

imarks.json

```
[  
  { "id": 2001, "marks": 28 },  
  { "id": 2002, "marks": 26 },  
  { "id": 2003, "marks": 30 },  
  { "id": 2004, "marks": 27 },  
  { "id": 2005, "marks": 29 }  
]
```

students. Json

```
[  
  { "id": 2001, "name": "Ayesha" },  
  { "id": 2002, "name": "Sara" },  
  { "id": 2003, "name": "Rahul" },  
  { "id": 2004, "name": "Neha" },  
  { "id": 2005, "name": "Kiran" }  
]
```


Spa/App.js

```
import React, { Component } from 'react';
import './App.css';

class App extends Component {
  constructor(props) {
    super(props);
    this.state = {
      studentData: [],
      internalMarks: [],
      externalMarks: []
    };
  }
  componentDidMount() {
    this.loadStudents();
  }
  loadStudents() {
    fetch("http://localhost:8080/students")
      .then(res => res.json())
      .then(result => {
        this.setState({ studentData: result });
      });
  }
  showInternalMarks() {
    fetch("http://localhost:8080/imarks")
      .then(res => res.json())
      .then(result => {
        this.setState({ internalMarks: result });
      });
  }
}
```

```

    });
  }
  showExternalMarks() {
    fetch("http://localhost:8080/emarks")
      .then(res => res.json())
      .then(result => {
        this.setState({ externalMarks: result });
      });
  }
  render() {
    const { studentData, internalMarks, externalMarks } = this.state;
    return (
      <div className="App">
        <h1>Student Marks Information</h1>
        <button onClick={() => this.showInternalMarks()}>
          Show Internal Marks
        </button>
        <button onClick={() => this.showExternalMarks()}>
          Show External Marks
        </button>
        <table border="1" align="center">
          <tr>
            <th>Student Name</th>
            <th>Internal</th>
            <th>External</th>
          </tr>
          <tr>
            <td>

```

```
      {studentData.map(s => (  
        <div key={s.id}>{s.name}</div>  
      ))}  
    </td>  
  
    <td>  
      {internalMarks.map(i => (  
        <div key={i.id}>{i.marks}</div>  
      ))}  
    </td>  
  
    <td>  
      {externalMarks.map(e => (  
        <div key={e.id}>{e.marks}</div>  
      ))}  
    </td>  
  </tr>  
</table>  
</div>  
  
);  
  
}  
  
}
```

```
export default App;
```

Output:

Student Marks Information

Show Internal Marks		Show External Marks	
Student Name	Internal	External	
Ayesha			
Sara			
Rahul			
Neha			
Kiran			

Student Marks Information

Show Internal Marks		Show External Marks	
Student Name	Internal	External	
Ayesha	28		
Sara	26		
Rahul	30		
Neha	27		
Kiran	29		

Student Marks Information

Show Internal Marks	Show External Marks	
Student Name	Internal	External
Ayesha	28	85
Sara	26	78
Rahul	30	90
Neha	27	88
Kiran	29	92

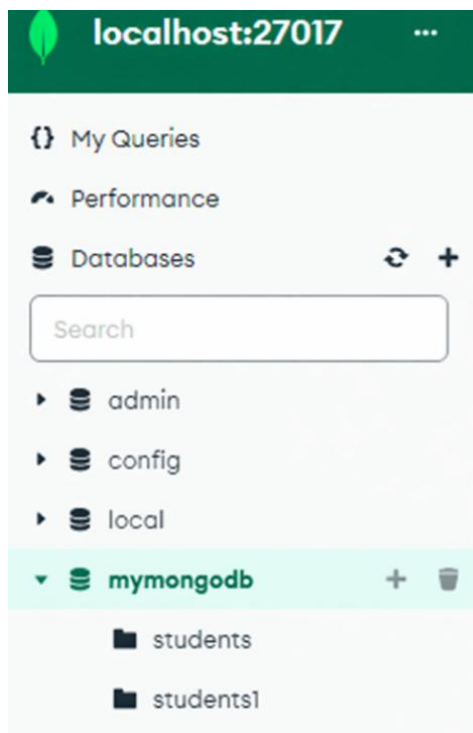
PROGRAM-13

Write a Node.js program to create DB and Collections in MongoDB

index.js

```
var MongoClient=require('mongodb').MongoClient;
var url="mongodb://localhost:27017/mymongodb";
MongoClient.connect(url,function(err,db){
  if(err) throw WriteError;
  console.log("Database created!");
  db.close();
});
MongoClient.connect(url,function(err,db){
  if(err) throw err;
  var dbo=db.db("mymongodb");
  dbo.createCollection("students1",function(err,res) {
    if(err) throw err;
    console.group("Collection created!");
    db.close();
  });
});
```

Output:



```
PS C:\Users\Ameena\OneDrive\Desktop\7th SEM\WT LAB  
13> node index.js  
Database created!  
Collection created!
```

PROGRAM-14

Write a react.js program to retrieve data from MongoDB

Restservice index.js

```
var express = require('express');

var MongoClient = require('mongodb').MongoClient;

var cors = require('cors');

var app = express();

app.use(cors());

var port = 8080;

var url = 'mongodb://127.0.0.1:27017/mymongodb';

var databasename = 'mymongodb';

var database;

app.get('/retrieve', function (req, res) {

  database.collection("students121").find({}).toArray((err, result) => {

    if (err) throw err;

    console.log(result);

    res.send(result);

  });

})

app.listen(port, ()=>{

  console.log('Mongo Connection Initiated...');

  MongoClient.connect(url, function (err, client) {

    database = client.db(databasename);

    console.log('Mongo Connected');

  });

});
```

Retrievedata App.js

```
import logo from './logo.svg';
```

```
import './App.css';

import { Component } from 'react';

class App extends Component {

  constructor(props) {

    super(props);

    this.state={

      students1:[]

    }

  }

  componentDidMount(){

    this.refreshStudents();

  }

  async refreshStudents(){

    fetch("http://localhost:8080/retrieve")

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

    .then(data=> {

      this.setState({students1:data});

    })

  }

  render() {

    const{students1} = this.state;

    return (

      <div className="App">

        <h1>Students Data</h1>

        <table border='1' align='center'>

          <tr>

            <th>Id</th>

            <th>Names</th>
```








```
<th>Internal Marks</th>
<th>External Marks</th>
</tr>
<tr>
<td>
{students1.map(st=><div>{st.id}</div>)}
</td>
<td>
{students1.map(st=><div>{st.name}</div>)}
</td>
<td>
{students1.map(st=><div>{st.imarks}</div>)}
</td>
<td>
{students1.map(st=><div>{st.emarks}</div>)}
</td>
</tr>
</table>
</div>
);
}
}
export default App;
```

Output:



mymongodb.student121

Documents Aggregations Schema Indexes Validation

Filter  Type a query: { field: 'value' }

 ADD DATA  EXPORT DATA  UPDATE  DELETE

```
_id: ObjectId('6776171e31c4ac75891db61f')
emarks: 234
id: 12
imarks: 47
name: "am"
```

←   localhost:8080/retrieve

```
1 [
2   {
3     "_id": "6776171e31c4ac75891db61f",
4     "emarks": 234,
5     "id": 12,
6     "imarks": 22,
7     "name": "am"
8   }
9 ]
```

Students Data

Id	Names	Internal Marks	External Marks
12	am	22	234