Backend Technologies - Experiment: 2

**StudentName**: Archana Srivastava **UID**:22MCA20244

**Branch**: UIC **Section/Group**: MCA - 2(A)

**Semester**:3rdsemester **SubjectCode**:22CAH-706

# 1) Task to be done :Store the data obtained in the experiment -1 in file and use that file in other program.

# HTML CODE:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Employee Details</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-image: url('path-to-your-image-file.jpg');

background-color: #f5f5f5;

background-size: cover;

background-repeat: no-repeat;

background-attachment: fixed;

}

.header {

text-align: center;

position: relative;

padding: 20px;

background-color: #35495e;

color: #0dc0f7;

}

.headerimg {

position: absolute;

top: 20px;

left: 20px;

width: 100px;

}

.frame {

width: 80%;

margin: auto;

padding: 20px;

border: 1px solid #ccc;

background-color: rgba(255, 255, 255, 0.9); /\* Slightly transparent background \*/

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

border-radius: 10px;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

font-size: 18px;

background-color: #fff;

border: 1px solid #ddd;

margin-bottom: 20px;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);

border-radius: 10px;

}

th, td {

padding: 12px;

text-align: left;

border: 1px solid #ddd;

}

th {

background-color: #f8c009;

color: #fff;

border: none;

}

.employee-row:nth-child(even) {

background-color: #f2f2f2;

}

.employee-row:nth-child(odd) {

background-color: #e0e0e0;

}

#result {

text-align: center;

margin-top: 20px;

font-size: 18px;

}

#calculateButton {

background-color: #e74c3c; /\* Red color \*/

color: #fff;

padding: 12px 24px;

border: none;

border-radius: 30px; /\* Rounder button \*/

cursor: pointer;

transition: background-color 0.3s;

}

#calculateButton:hover {

background-color: #c0392b; /\* Darker red on hover \*/

}

</style>

</head>

<body>

<div class="header">

<imgsrc="pgnlogo.jpg" alt="Company Logo" width="100">

<h1>Employee Management System</h1>

</div>

<div class="frame">

<h2>Employee Details</h2>

<table>

<tr>

<th>Emp ID</th>

<th>Emp Name</th>

<th>Emp Role</th>

<th>Emp Salary</th>

</tr>

<tr>

<td>1</td>

<td>Abhishek Bhardwaj</td>

<td>Software Developer</td>

<td id="salary1"></td>

</tr>

<tr>

<td>2</td>

<td>Avinash </td>

<td>UX Designer</td>

<td id="salary2"></td>

</tr>

<tr>

<td>3</td>

<td>Mohit</td>

<td>Project Manager</td>

<td id="salary3"></td>

</tr>

<tr>

<td>4</td>

<td>Rohit</td>

<td>Graphic Designer</td>

<td id="salary4"></td>

</tr>

<tr>

<td>5</td>

<td>gagankumar</td>

<td>QA Engineer</td>

<td id="salary5"></td>

</tr>

<tr>

<td>6</td>

<td>Rakesh</td>

<td>Marketing Specialist</td>

<td id="salary6"></td>

</tr>

<tr>

<td>7</td>

<td>Md. Ali </td>

<td>Data Analyst</td>

<td id="salary7"></td>

</tr>

<tr>

<td>8</td>

<td>Rajesh ali</td>

<td>HR Manager</td>

<td id="salary8"></td>

</tr>

<tr>

<td>9</td>

<td>Munish</td>

<td>Sales Representative</td>

<td id="salary9"></td>

</tr>

<tr>

<td>10</td>

<td>Godwill</td>

<td>Customer Support</td>

<td id="salary10"></td>

</tr>

<!-- Add more rows for other employees -->

</table>

<button id="calculateButton">Show Salary</button>

<div id="result"></div>

</div>

<script>

const calculateButton = document.getElementById('calculateButton');

const resultDiv = document.getElementById('result');

calculateButton.addEventListener('click', () => {

const salaries = [80000, 70000, 90000, 60000, 75000, 65000, 70000, 85000, 55000, 50000];

for (let i = 0; i<salaries.length; i++) {

const salaryElement = document.getElementById(`salary${i + 1}`);

salaryElement.textContent = `₹${salaries[i]}`;

}

});

</script>

</body>

</html>

**Server.js**

// Server.js

const http = require('http');

const fs = require('fs');

http.createServer((req, resp) => {

fs.readFile('employee.html', function (err, html) {

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

resp.write(html);

resp.end();

});

}).listen(8000);

**FileHandling.js**

// FileHandling.js

const fs = require("fs").promises;

const http = require("http");

try {

const read = async () => {

const data = await fs.readFile("employee.html");

fs.writeFile("NewFile.html", data);

};

read();

} catch (e) {

console.log("YOU HAVE AN ERROR", e);

}

try {

setTimeout(() => {

const ReadandDisplay = async () => {

const data = await fs.readFile("NewFile.html");

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

res.write(data);

res.end();

}).listen(8000, () => {

console.log("Listening to Port 8000");

});

};

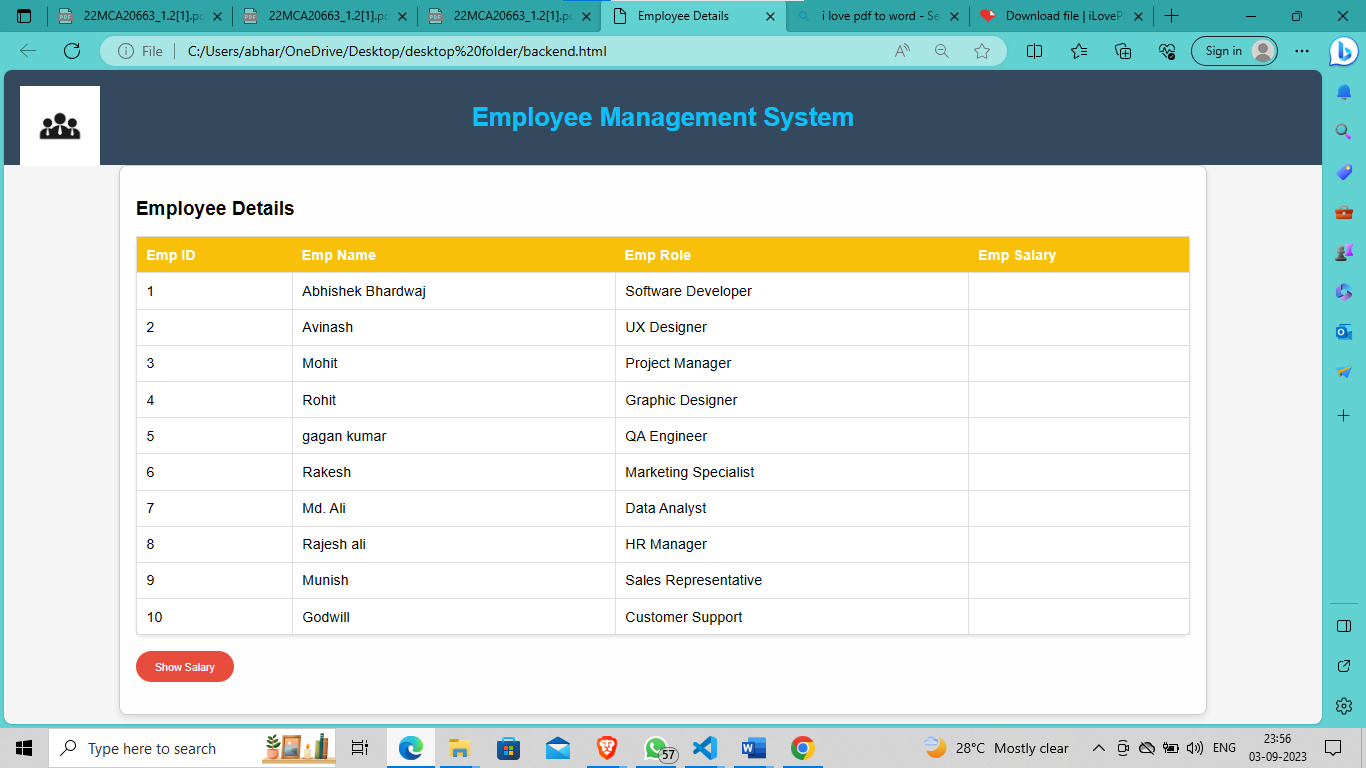
ReadandDisplay();

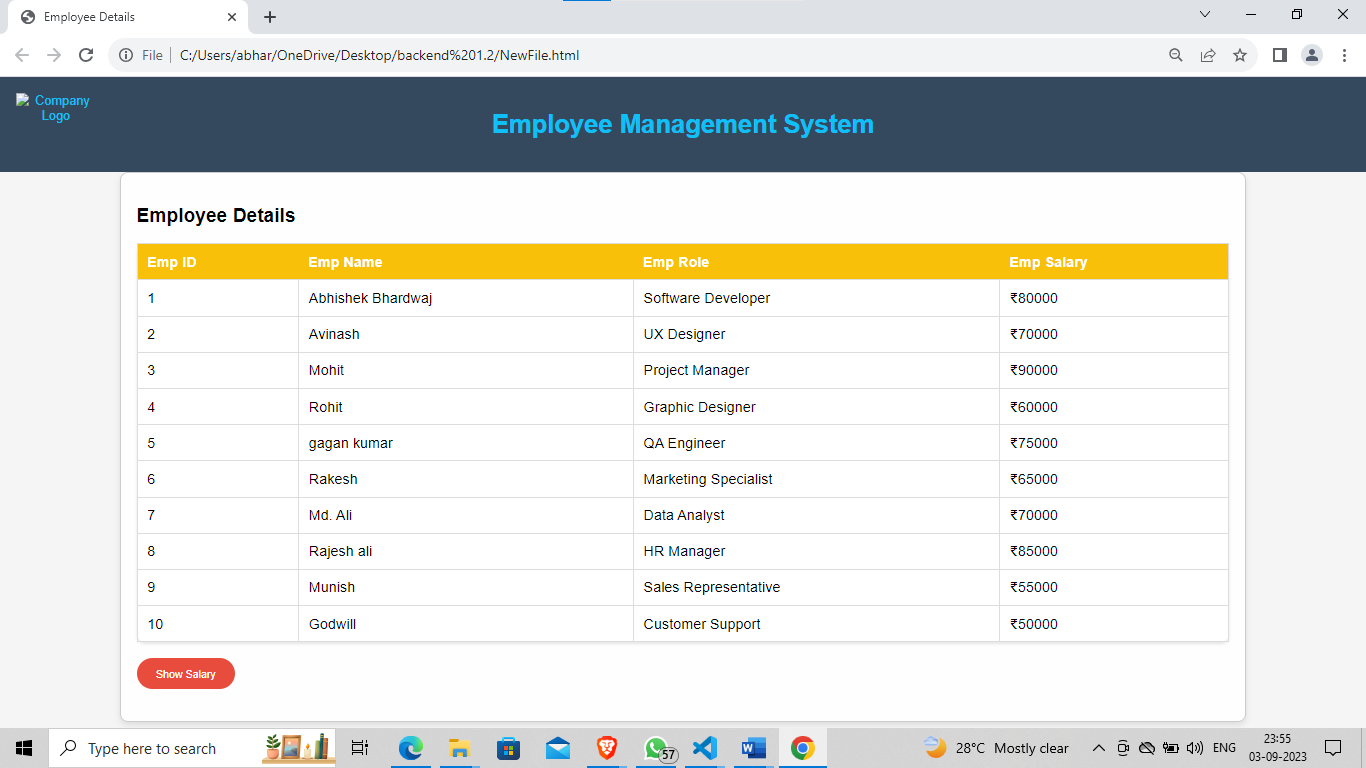
}, 8000);

} catch (e) {

console.log("YOU HAVE AN ERROR", e);

**Output:**

****



**Learning outcomes (What I have learnt):**

1. Understand application of node.js
2. Learn about file system
3. Learn how to create text file in node.js