**GROUP F**

**Company maintains employee information as employee ID, name, designation and salary.**

**Allow user to add, delete information of employee. Display information of particular**

**employee. If employee does not exist an appropriate message is displayed. If it is, then**

**the system displays the employee details. Use index sequential file to maintain the data.**

#include <iostream>

#include <fstream>

#include <sstream>

using namespace std;

const string DATA\_FILE = "employee\_data.txt";

struct Employee {

int employee\_id;

string name;

string designation;

double salary;

};

void addEmployee() {

ofstream file(DATA\_FILE, ios::app);

Employee employee;

cout << "Enter Employee ID: ";

cin >> employee.employee\_id;

cout << "Enter Name: ";

cin.ignore();

getline(cin, employee.name);

cout << "Enter Designation: ";

getline(cin, employee.designation);

cout << "Enter Salary: ";

cin >> employee.salary;

file << employee.employee\_id << ',' << employee.name << ',' << employee.designation <<

',' << employee.salary << '\n';

file.close();

cout << "Employee added successfully!\n";

}

void deleteEmployee() {

int employee\_id;

cout << "Enter Employee ID to delete: ";

cin >> employee\_id;

ifstream inputFile(DATA\_FILE);

ofstream tempFile("temp.txt");

bool found = false;

string line;

while (getline(inputFile, line)) {

stringstream ss(line);

string field;

getline(ss, field, ',');

int id = stoi(field);

if (id == employee\_id) {

found = true;

} else {

tempFile << line << '\n';

}

}

inputFile.close();

tempFile.close();

if (found) {

remove(DATA\_FILE.c\_str());

rename("temp.txt", DATA\_FILE.c\_str());

cout << "Employee deleted successfully!\n";

} else {

remove("temp.txt");

cout << "Employee not found.\n";

}

}

void displayEmployee() {

int employee\_id;

cout << "Enter Employee ID: ";

cin >> employee\_id;

ifstream file(DATA\_FILE);

string line;

bool found = false;

while (getline(file, line)) {

stringstream ss(line);

string field;

getline(ss, field, ',');

int id = stoi(field);

if (id == employee\_id) {

found = true;

cout << "Employee ID: " << field << '\n';

getline(ss, field, ',');

cout << "Name: " << field << '\n';

getline(ss, field, ',');

cout << "Designation: " << field << '\n';

getline(ss, field, ',');

cout << "Salary: " << field << '\n';

break;

}

}

file.close();

if (!found) {

cout << "Employee not found.\n";

}

}

int main() {

int choice;

do {

cout << "\nEmployee Management System\n";

cout << "---------------------------\n";

cout << "1. Add Employee\n";

cout << "2. Delete Employee\n";

cout << "3. Display Employee\n";

cout << "4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

addEmployee();

break;

case 2:

deleteEmployee();

break;

case 3:

displayEmployee();

break;

case 4:

cout << "Exiting...\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

break;

}

} while (choice != 4);

return 0;

}

**OUTPUT:**

**Employee Management System**

**1. Add Employee**

**2. Delete Employee**

**3. Display Employee**

**4. Exit**

**Enter your choice: 1**

**Enter Employee ID: 1**

**Enter Name: Vedant Gogawale**

**Enter Designation: HR**

**Enter Salary: 90000**

**Employee added successfully!**

**Employee Management System**

**1. Add Employee**

**2. Delete Employee**

**3. Display Employee**

**4. Exit**

**Enter your choice:**