

NAME : TANISHQ GUPTA  
CLASS : SE  
BATCH : A4  
ROLL NO.: SEAD21168  
F24

Problem Statement:

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 <stdio.h>

using namespace std;

// Employee class Declaration
class Employee
{
private:
    int code;
    char name[20];
    float salary;

public:
    void read();
    void display();
    // will return employee code
    int getEmpCode() { return code; }
    // will return employee salary
    int getSalary() { return salary; }
    // will update employee salary
    void updateSalary(float s) { salary = s; }
};

// Read employee record
void Employee::read()
{
    cout << "Enter employee code: ";
    cin >> code;
    cout << "Enter name: ";
    cin.ignore(1);
    cin.getline(name, 20);
    cout << "Enter salary: ";
    cin >> salary;
}

// Display employee record
void Employee::display()
{
    cout << code << " " << name << "\t" << salary << endl;
}

// global declaration
fstream file;
```

```

// Will delete file when program is being executed
// because we are create file in append mode
void deleteExistingFile()
{
    remove("EMPLOYEE.DAT");
}

// function to append record into file
void appendToFille()
{
    Employee x;

    // Read employee record from user
    x.read();

    file.open("EMPLOYEE.DAT", ios::binary | ios::app);
    if (!file)
    {
        cout << "ERROR IN CREATING FILE\n";
        return;
    }
    // write into file
    file.write((char *)&x, sizeof(x));
    file.close();
    cout << "Record added sucessfully.\n";
}

void displayAll()
{
    Employee x;

    file.open("EMPLOYEE.DAT", ios::binary | ios::in);
    if (!file)
    {
        cout << "ERROR IN OPENING FILE \n";
        return;
    }
    while (file)
    {
        if (file.read((char *)&x, sizeof(x)))
            if (x.getSalary() >= 10000 && x.getSalary() <= 20000)
                x.display();
    }
    file.close();
}

void searchForRecord()
{
    // read employee id
    Employee x;
    int c;
    int isFound = 0;

    cout << "Enter employee code: ";
    cin >> c;

    file.open("EMPLOYEE.DAT", ios::binary | ios::in);

```

```

    if (!file)
    {
        cout << "ERROR IN OPENING FILE \n";
        return;
    }
    while (file)
    {
        if (file.read((char *)&x, sizeof(x)))
        {
            if (x.getEmpCode() == c)
            {
                cout << "RECORD FOUND\n";
                x.display();
                isFound = 1;
                break;
            }
        }
    }
    if (isFound == 0)
    {
        cout << "Record not found!!!\n";
    }
    file.close();
}

// Function to increase salary
void increaseSalary()
{
    // read employee id
    Employee x;
    int c;
    int isFound = 0;
    float sal;

    cout << "enter employee code \n";
    cin >> c;

    file.open("EMPLOYEE.DAT", ios::binary | ios::in);
    if (!file)
    {
        cout << "ERROR IN OPENING FILE \n";
        return;
    }
    while (file)
    {
        if (file.read((char *)&x, sizeof(x)))
        {
            if (x.getEmpCode() == c)
            {
                cout << "Salary hike? ";
                cin >> sal;
                x.updateSalary(x.getSalary() + sal);
                isFound = 1;
                break;
            }
        }
    }
    if (isFound == 0)

```

```

    {
        cout << "Record not found!!!\n";
    }
    file.close();
    cout << "Salary updated successfully." << endl;
}

// Insert record by assuming that records are in
// ascending order
void insertRecord()
{
    // read employee record
    Employee x;
    Employee newEmp;

    // Read record to insert
    newEmp.read();

    fstream fin;
    // read file in input mode
    file.open("EMPLOYEE.DAT", ios::binary | ios::in);
    // open file in write mode
    fin.open("TEMP.DAT", ios::binary | ios::out);

    if (!file)
    {
        cout << "Error in opening EMPLOYEE.DAT file!!!\n";
        return;
    }
    if (!fin)
    {
        cout << "Error in opening TEMP.DAT file!!!\n";
        return;
    }
    while (file)
    {
        if (file.read((char *)&x, sizeof(x)))
        {
            if (x.getEmpCode() > newEmp.getEmpCode())
            {
                fin.write((char *)&newEmp, sizeof(newEmp));
            }
            // no need to use else
            fin.write((char *)&x, sizeof(x));
        }
    }

    fin.close();
    file.close();

    rename("TEMP.DAT", "EMPLOYEE.DAT");
    remove("TEMP.DAT");
    cout << "Record inserted successfully." << endl;
}

int main()
{
    char ch;

```

```

// if required then only remove the file
deleteExistingFile();

do
{
    int n;

    cout << "ENTER CHOICE\n"
         << "1.ADD AN EMPLOYEE\n"
         << "2.DISPLAY\n"
         << "3.SEARCH\n"
         << "4.INCREASE SALARY\n"
         << "5.INSERT RECORD\n";
    cout << "Make a choice: ";
    cin >> n;

    switch (n)
    {
    case 1:
        appendToFile();
        break;
    case 2:
        displayAll();
        break;
    case 3:
        searchForRecord();
        break;
    case 4:
        increaseSalary();
        break;
    case 5:
        insertRecord();
        break;

    default:
        cout << "Invalid Choice\n";
    }

    cout << "Do you want to continue ? : ";
    cin >> ch;

} while (ch == 'Y' || ch == 'y');

return 0;
}

```

OUTPUT:

```
PS C:\Users\tanis\OneDrive\Desktop\tanishq_gupta_21168> g++ .\f24.cpp
PS C:\Users\tanis\OneDrive\Desktop\tanishq_gupta_21168> .\a.exe
ENTER CHOICE
1.ADD AN EMPLOYEE
2.DISPLAY
3.SEARCH
4.INCREASE SALARY
5.INSERT RECORD
Make a choice: 1
Enter employee code: 2109
Enter name: shubham
Enter salary: 25000
Record added successfully.
Do you want to continue ? : y
ENTER CHOICE
1.ADD AN EMPLOYEE
2.DISPLAY
3.SEARCH
4.INCREASE SALARY
5.INSERT RECORD
Make a choice: 2
Do you want to continue ? : y
ENTER CHOICE
1.ADD AN EMPLOYEE
2.DISPLAY
3.SEARCH
4.INCREASE SALARY
5.INSERT RECORD
Make a choice: 3
Enter employee code: 2109
RECORD FOUND
2109 shubham      25000
Do you want to continue ? : 2
PS C:\Users\tanis\OneDrive\Desktop\tanishq_gupta_21168> |
```