**Practical GroupF\_23**

**Aim:**

Department maintains a student information. The file contains roll number, name, division and address. Allow user to add, delete information of student. Display information of particular employee. If record of student does not exist an appropriate message is displayed. If it is, then the system displays the student details. Use sequential file to main the data.

**Code:**

#include <iostream>

#include <fstream>

#include <cstring>

using namespace std;

// Structure to store student information

struct Student {

int rollNo;

char name[50];

char division;

char address[100];

};

// Function to display a student's information

void displayStudent(int rollNo) {

ifstream fin("students.txt", ios::binary);

if (!fin) {

cout << "Error opening file!";

return;

}

Student s;

bool found = false;

while (fin.read((char\*)&s, sizeof(Student))) {

if (s.rollNo == rollNo) {

cout << "Roll No: " << s.rollNo << endl;

cout << "Name: " << s.name << endl;

cout << "Division: " << s.division << endl;

cout << "Address: " << s.address << endl;

found = true;

break;

}

}

if (!found) {

cout << "Record not found!" << endl;

}

fin.close();

}

// Function to add a new student's information to the file

void addStudent() {

ofstream fout("students.txt", ios::binary | ios::app);

if (!fout) {

cout << "Error opening file!";

return;

}

Student s;

cout << "Enter Roll No: ";

cin >> s.rollNo;

cin.ignore();

cout << "Enter Name: ";

cin.getline(s.name, 50);

cout << "Enter Division: ";

cin >> s.division;

cin.ignore();

cout << "Enter Address: ";

cin.getline(s.address, 100);

fout.write((char\*)&s, sizeof(Student));

fout.close();

cout << "Record added successfully!" << endl;

}

// Function to delete a student's information from the file

void deleteStudent(int rollNo) {

ifstream fin("students.txt", ios::binary);

ofstream fout("temp.txt", ios::binary);

if (!fin || !fout) {

cout << "Error opening file!";

return;

}

Student s;

bool found = false;

while (fin.read((char\*)&s, sizeof(Student))) {

if (s.rollNo != rollNo) {

fout.write((char\*)&s, sizeof(Student));

} else {

found = true;

}

}

fin.close();

fout.close();

if (!found) {

cout << "Record not found!" << endl;

remove("temp.txt");

} else {

remove("students.txt");

rename("temp.txt", "students.txt");

cout << "Record deleted successfully!" << endl;

}

}

int main() {

int choice, rollNo;

while (true) {

cout << "1. Display student information" << endl;

cout << "2. Add student information" << endl;

cout << "3. Delete student information" << endl;

cout << "4. Exit" << endl;

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Enter Roll No: ";

cin >> rollNo;

displayStudent(rollNo);

break;

case 2:

addStudent();

break;

case 3:

cout << "Enter Roll No: ";

cin >> rollNo;

deleteStudent(rollNo);

break;

case 4:

return 0;

default:

cout << "Invalid choice!" <<