// Student Record Management System in C++

// Uses file handling + OOP + structured design

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

class Student {

public:

int rollNo;

string name;

string course;

float marks;

void input() {

cout << "Enter Roll No: ";

cin >> rollNo;

cin.ignore();

cout << "Enter Name: ";

getline(cin, name);

cout << "Enter Course: ";

getline(cin, course);

cout << "Enter Marks: ";

cin >> marks;

}

void display() {

cout << "\nRoll No: " << rollNo;

cout << "\nName: " << name;

cout << "\nCourse: " << course;

cout << "\nMarks: " << marks << endl;

}

};

void addRecord() {

ofstream file("students.txt", ios::app);

Student s;

s.input();

file << s.rollNo << " " << s.name << " " << s.course << " " << s.marks << endl;

file.close();

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

}

void displayRecords() {

ifstream file("students.txt");

Student s;

cout << "\n--- All Student Records ---\n";

while (file >> s.rollNo >> s.name >> s.course >> s.marks) {

s.display();

cout << "-----------------------" << endl;

}

file.close();

}

void searchRecord() {

ifstream file("students.txt");

int r;

bool found = false;

cout << "Enter Roll No to search: ";

cin >> r;

Student s;

while (file >> s.rollNo >> s.name >> s.course >> s.marks) {

if (s.rollNo == r) {

cout << "\nRecord Found:" << endl;

s.display();

found = true;

break;

}

}

if (!found) {

cout << "\nRecord not found." << endl;

}

file.close();

}

void deleteRecord() {

ifstream file("students.txt");

ofstream temp("temp.txt");

int r;

bool found = false;

cout << "Enter Roll No to delete: ";

cin >> r;

Student s;

while (file >> s.rollNo >> s.name >> s.course >> s.marks) {

if (s.rollNo != r) {

temp << s.rollNo << " " << s.name << " " << s.course << " " << s.marks << endl;

} else {

found = true;

}

}

file.close();

temp.close();

remove("students.txt");

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

if (found)

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

else

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

}

void updateRecord() {

ifstream file("students.txt");

ofstream temp("temp.txt");

int r;

bool found = false;

cout << "Enter Roll No to update: ";

cin >> r;

Student s;

while (file >> s.rollNo >> s.name >> s.course >> s.marks) {

if (s.rollNo == r) {

cout << "Enter new details:\n";

s.input();

found = true;

}

temp << s.rollNo << " " << s.name << " " << s.course << " " << s.marks << endl;

}

file.close();

temp.close();

remove("students.txt");

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

if (found)

cout << "\nRecord updated successfully!\n";

else

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

}

int main() {

int choice;

do {

cout << "\n\*\*\* Student Record Management System \*\*\*\n";

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

cout << "2. Display All Records\n";

cout << "3. Search Record\n";

cout << "4. Delete Record\n";

cout << "5. Update Record\n";

cout << "0. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

addRecord();

break;

case 2:

displayRecords();

break;

case 3:

searchRecord();

break;

case 4:

deleteRecord();

break;

case 5:

updateRecord();

break;

case 0:

cout << "Exiting program.\n";

break;

default:

cout << "Invalid choice. Try again.\n";

}

} while (choice != 0);

return 0;

}