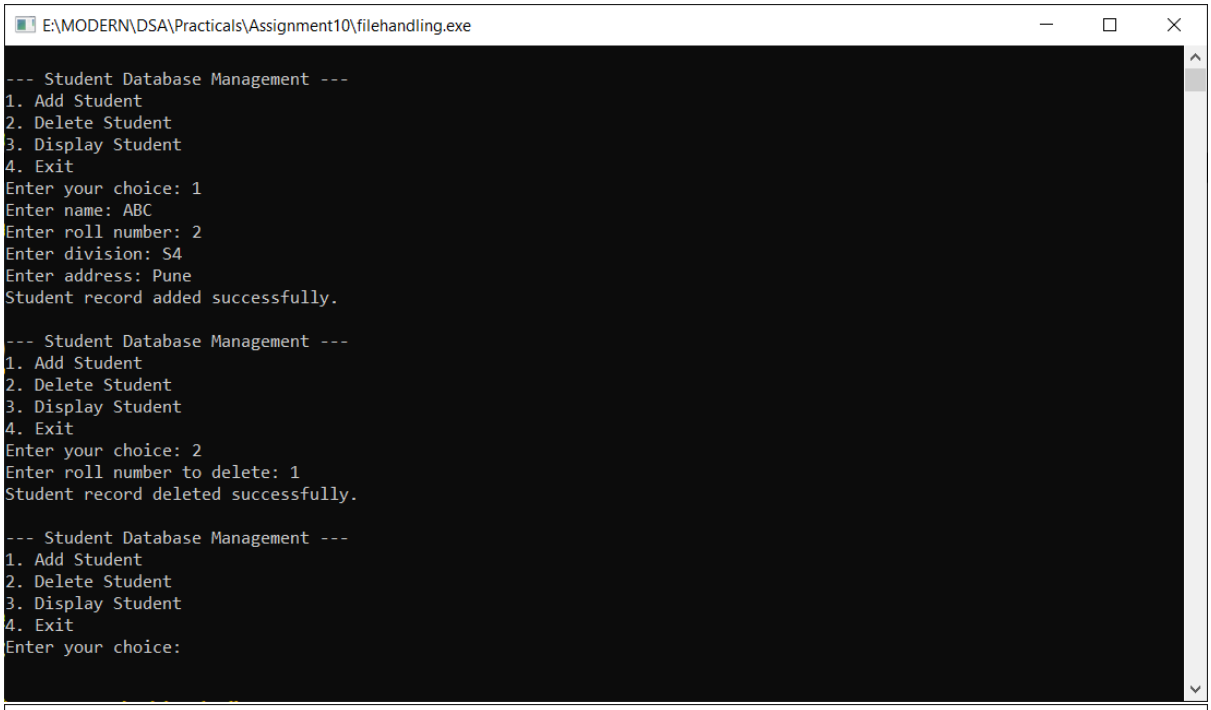
```
 /*
                            ASSIGNMENT NO.10
NAME- ABRAR SHAIKH                                  ROLL NO. - 23570
                       TOPIC- File Handling
*/


#include <iostream>
#include <fstream>
#include <cstring>

using namespace std;

struct Student {
    char name[50];
    int rollNo;
    char division[10];
    char address[100];
};

// Function to add a student record
void addStudent() {
    Student student;
    ofstream file("student_database.txt", ios::app); // Open file in
append mode


    cout << "Enter name: ";
    cin.ignore(); // Clear input buffer
    cin.getline(student.name, 50);
    cout << "Enter roll number: ";
    cin >> student.rollNo;
    cout << "Enter division: ";
    cin >> student.division;
```

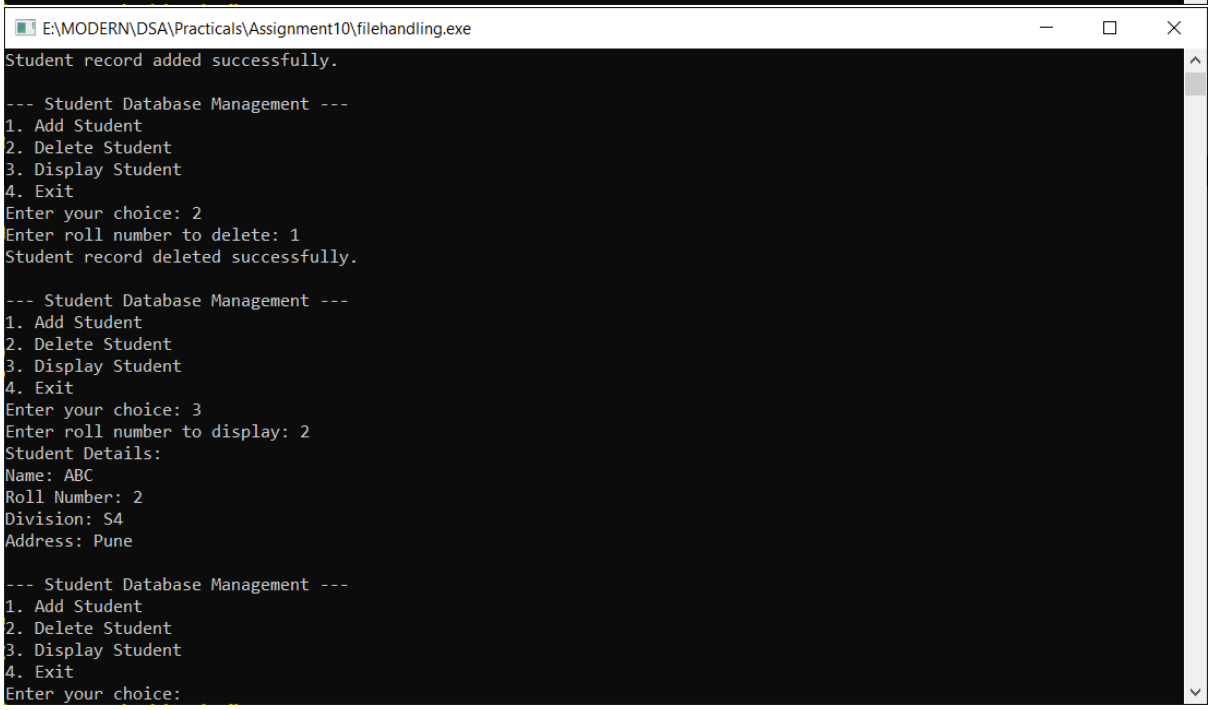
```
    cout << "Enter address: ";
    cin.ignore(); // Clear input buffer
    cin.getline(student.address, 100);

    // Write student data to file in a readable format
    file << student.name << "," << student.rollNo << "," <<
student.division << "," << student.address << endl;
    file.close();
    cout << "Student record added successfully.\n";
}

// Function to delete a student record
void deleteStudent(int rollNo) {
    Student student;
    ifstream file("student_database.txt", ios::in);
    ofstream tempFile("temp.txt", ios::out); // Temporary file to
store records
    bool found = false;

    while (file.getline(student.name, 50, ',') && // Read until
comma
           (file >> student.rollNo).ignore() && // Read rollNo and
ignore the next comma
           file.getline(student.division, 10, ',') && // Read
division until comma
           file.getline(student.address, 100)) { // Read address
until newline
        if (student.rollNo != rollNo) {
            // Write to temp file if not matching
            tempFile << student.name << "," << student.rollNo << ","
<< student.division << "," << student.address << endl;
        } else {
```

```
            found = true; // Mark as found if matching record
        }
    }

    file.close();
    tempFile.close();
    remove("student_database.txt"); // Remove original file
    rename("temp.txt", "student_database.txt"); // Rename temp file
to original

    if (found) {
        cout << "Student record deleted successfully.\n";
    } else {
        cout << "Student record not found.\n";
    }
}

// Function to display a student record
void displayStudent(int rollNo) {
    Student student;
    ifstream file("student_database.txt", ios::in);
    bool found = false;

    while (file.getline(student.name, 50, ',') && // Read until
comma
           (file >> student.rollNo).ignore() && // Read rollNo and
ignore the next comma
           file.getline(student.division, 10, ',') && // Read
division until comma
           file.getline(student.address, 100)) { // Read address
until newline
        if (student.rollNo == rollNo) {
```

```
            cout << "Student Details:\n";
            cout << "Name: " << student.name << endl;
            cout << "Roll Number: " << student.rollNo << endl;
            cout << "Division: " << student.division << endl;
            cout << "Address: " << student.address << endl;
            found = true; // Mark as found if matching record
            break; // Exit the loop after finding the record
        }
    }

    file.close();

    if (!found) {
        cout << "Student record not found.\n";
    }
}

// Main function to display menu and handle user input
int main() {
    int choice, rollNo;

    do {
        cout << "\n--- Student Database Management ---\n";
        cout << "1. Add Student\n";
        cout << "2. Delete Student\n";
        cout << "3. Display Student\n";
        cout << "4. Exit\n";
        cout << "Enter your choice: ";
        cin >> choice;
```

```
        switch (choice) {
            case 1:
                addStudent();
                break;
            case 2:
                cout << "Enter roll number to delete: ";
                cin >> rollNo;
                deleteStudent(rollNo);
                break;
            case 3:
                cout << "Enter roll number to display: ";
                cin >> rollNo;
                displayStudent(rollNo);
                break;
            case 4:
                cout << "Exiting program.\n";
                break;
            default:
                cout << "Invalid choice! Please try again.\n";
        }
    } while (choice != 4);

    return 0;
}
```





GitHub Repository- https://github.com/abssha/DSA.git