STUDENT MANAGEMENT SYSTEM

Submitted by

Name of the Student: **Abhiroop Sarkar**
Enrolment Number: **12022002016041**
Section: **D**
Class Roll Number: **03**
Stream: **CSE (AIML)**
Subject: **Programming for problem solving using C**
Subject Code: **ESC103(Pr.)**
Semester:**2ⁿᵈ(1ˢᵗ year)**
Department: **Basic Science and Humanities**

Academic Year: 2022-23

INSTITUTE OF ENGINEERING & MANAGEMENT
Good Education, Good Jobs

Introduction

The Student Records Management System is a software project designed to facilitate efficient management of student information. It provides functionalities for adding student records, displaying records, deleting records, and updating student grades. The system utilizes file handling techniques to store and retrieve student data in a structured manner.

The Student Records Management System offers a streamlined approach to managing student information, ensuring accuracy, and facilitating easy access. By leveraging file handling techniques and intuitive functions, it provides a user-friendly experience for administrators or educators responsible for maintaining student records.

Functions

- ## void addStudent()

Variables:

file	fp
char	first_name[20], last_name[20], Course[10], city[20]
int	roll_no
float	grade

Use:

This function allows us to add a student's information to the student database. It prompts the user to enter the student's first name, last name, roll number, course, and city. By default, the grade is set to 0.0. The function then stores this information in a file called "information.txt" for later retrieval.

- ## void displayStudentRecords()

Variables:

file	fp
char	first_name[20], last_name[20], Course[10], city[20]
int	roll_no
float	grade

This function allows us to view all the existing student records stored in the database. It opens the "information.txt" file and reads each record line by line. For each record, it retrieves the student's first name, last name, roll number, course, city, and grade. Then, it prints out the information in a formatted manner.

• void deleteStudent()

Variables:

file	fp, temp_fp
char	first_name[20], last_name[20], Course[10], city[20]
int	roll_no, delete_roll_no, found
float	grade

Use:

This function allows us to delete a student record from the database based on the provided roll number. It opens the "information.txt" file for reading and creates a temporary file "temp.txt" for writing the updated records without the deleted one. It prompts us to enter the roll number of the student to be deleted. Then it scans each record from the file, compares the roll number with the provided input, and writes the non-matching records to the temporary file. Finally, it replaces the original file with the temporary file, effectively deleting the specified student record.

• void examination()

Variables:

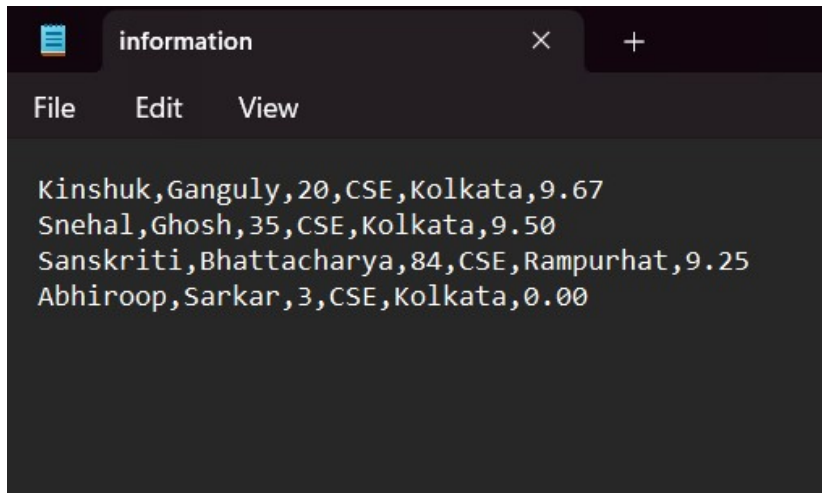| file	fp,temp_fp
char	first_name[20], last_name[20], Course[10], city[20]
int	roll_no, roll_no_to_change, found, n, marks[n], tot
float	grade

Use:

The examination function allows you to change the grades of a student's record based on their roll number. It opens the "information.txt" file for reading and creates a temporary file "temp.txt" for writing the updated records. The function prompts you to enter the roll number of the student whose grades you want to change. If the roll number is found, it then asks us to enter the number of subjects and their respective marks. It calculates the

grade based on the marks and updates the record in the temporary file. Finally, it replaces the original file with the temporary file, effectively updating the grades of the specified student.
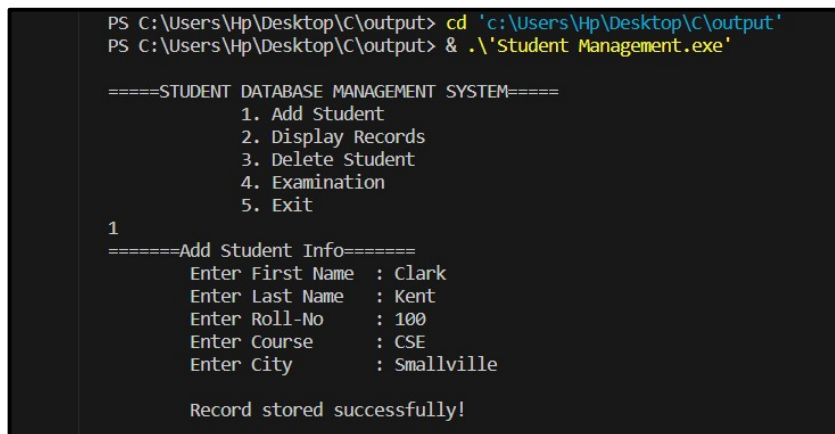
File

- information.txt

```
information                                    ×    +

File    Edit    View

Kinshuk,Ganguly,20,CSE,Kolkata,9.67
Snehal,Ghosh,35,CSE,Kolkata,9.50
Sanskriti,Bhattacharya,84,CSE,Rampurhat,9.25
Abhiroop,Sarkar,3,CSE,Kolkata,0.00
```

Outputs

1. Adding new Student:

```
PS C:\Users\Hp\Desktop\C\output> cd 'c:\Users\Hp\Desktop\C\output'
PS C:\Users\Hp\Desktop\C\output> & .\'Student Management.exe'

=====STUDENT DATABASE MANAGEMENT SYSTEM=====
            1. Add Student
            2. Display Records
            3. Delete Student
            4. Examination
            5. Exit
1
=======Add Student Info=======
        Enter First Name  : Clark
        Enter Last Name   : Kent
        Enter Roll-No     : 100
        Enter Course      : CSE
        Enter City        : Smallville

        Record stored successfully!
```

2. Displaying Student Records:

```
=====STUDENT DATABASE MANAGEMENT SYSTEM=====
          1. Add Student
          2. Display Records
          3. Delete Student
          4. Examination
          5. Exit
2
=======Student Records=======
Name: Kinshuk Ganguly
Roll no: 20
Course: CSE
City: Kolkata
Grade: 9.67

Name: Snehal Ghosh
Roll no: 35
Course: CSE
City: Kolkata
Grade: 9.50

Name: Sanskriti Bhattacharya
Roll no: 84
Course: CSE
City: Rampurhat
Grade: 0.00

Name: Abhiroop Sarkar
Roll no: 3
Course: CSE
City: Kolkata
Grade: 0.00

Name: Clark Kent
Roll no: 100
Course: CSE
City: Smallville
Grade: 0.00
```

3. Deleting a Record:

```
=====STUDENT DATABASE MANAGEMENT SYSTEM=====
          1. Add Student
          2. Display Records
          3. Delete Student
          4. Examination
          5. Exit
3
=======Delete Student Record=======
Enter the roll number of the student to delete: 100
Record deleted successfully for roll number: 100
```

4. Examination Portal:

```
=====STUDENT DATABASE MANAGEMENT SYSTEM=====
              1. Add Student
              2. Display Records
              3. Delete Student
              4. Examination
              5. Exit
4
=======Examination Portal=======
Enter the roll number of the student to change grades: 84
Enter no of subjects:4
Enter marks for Subject 1:82
Enter marks for Subject 2:80
Enter marks for Subject 3:85
Enter marks for Subject 4:95
Grades updated successfully for roll number: 84
```

Future Prospects

Future prospects for the Student Management System may include incorporating multiple files for different courses and examinations, accommodating students from multiple academic years, implementing advanced search and sorting functionalities, and integrating a user-friendly graphical interface for enhanced usability and data visualization.