

BALLARI INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(BITM)

STUDENT DATABASE MANAGEMENT SYSTEM

PREPARED BY VINAY B

Data is one of the most crucial and important entity in this modern IT world, hence it becomes an important responsibility for a software engineer to come up with the best solution to solve this problem. The main initiative of this project is to record the data of students in schools or colleges very easily. We might have seen such cases where it becomes very tedious task to add students details to the register books, it is more tedious to search any students details from the heavy bunch of records.

In order to solve such problems regarding to the management of student's data, this project will definitely help in performing such operations within no time. We can perform operations on the data like adding any students details, printing all the students details, searching for a particular students details, deleting a student etc.. For privacy concerns, the entire software is password protected. The user will only be allowed to perform such operations when he enters correct credentials.

This entire project is purely written in C-Language, due to which the software is very fast and easy to understand, all the operations are written in terms of functions, which will be called accordingly, all the functions will be explained clearly below. Each function is executed only when the login function is bypassed.

FUNCTIONALITY OF CODE:

- LANDING PAGE :-

When the program is executed, a landing page will be displayed on the terminal. Here you will be asked to choose any operation to perform as shown below. In case of an invalid choice you will again be asked to enter a valid input, this continues until you enter correct option. Once you enter a valid choice, you will be landed to the page according to the operation.

```
=====STUDENT DATABASE MANAGER=====
```

- 1) Add student
- 2) Display students
- 3) Search any student details
- 4) Delete any student
- 5) Exit from software

```
_____  
Enter any option:
```

- ADD STUDENT :-

This function allows the user to enter the details of students, the details of the students will be temporarily stored in a structure variable and further the data will be permanently copied or appended to the records.txt file. The user can enter the details of the multiple students continuously. And this operation can be performed when the login is successful, Otherwise the user will be asked to login with correct credentials again and again.

```
=====ADD ANY STUDENT'S DETAILS=====
```

```
Enter USN          : 3BR21EC180
```

```
Enter roll number  : 43
```

```
Enter name         : VINAY
```

```
Enter branch       : ECE
```

```
Enter gender (M/F): MALE
```

```
Enter CGPA         : 9.2
```

```
YOUR DATA HAS BEEN SAVED SUCCESSFULLY !
```

```
Do you want to add another student(y/n)
```

- DISPLAY STUDENT'S RECORDS :-

This function allows the user to display all the student's records saved in the file, it makes easier to analyze all the students information very efficiently. The details of the students are displayed in well formatted manner, which makes the user to access the data easily. Like other functions this function is also password protected, you will only be able to access the data only when you enter correct credentials. The details of the students are extracted from the .txt file and printed on to the terminal.

=====STUDENT'S RECORDS=====

USN : 3BR21AI054
Roll num: 54
Name : SAMETHA
Branch : AIML
Gender : FEMALE
CGPA : 8.00

USN : 3BR21EC180
Roll num: 43
Name : VINAY
Branch : ECE
Gender : MALE
CGPA : 9.20

Enter any key to exit...█

- SEARCH ANY STUDENTS DETAILS :-

Like the name itself suggests the functionality of this function, this function allows the user to get details of a particular function. The user is allowed to enter the roll number of the student and the details of that particular student will be deleted permanently from the records.

=====SEARCH ANY STUDENT'S DETAILS=====

Enter roll Number: 43

USN : 3BR21EC180
Roll num: 43
Name : VINAY
Branch : ECE
Gender : MALE
CGPA : 9.20

Enter any key to exit...█

- DELETE ANY STUDENT :-

In some circumstances we need to delete any students details permanently from the records, in such cases this functionality will help the user to perform the action very efficiently by using the function. In this method all the student's details will be copied to another file except the details of the student to be deleted. And the file will be replaced by original file.

```
=====DELETE STUDENTS RECORD=====
```

```
Enter the roll no : 43
```

```
Record deleted successfully !
```

```
Enter any key to exit...█
```

- EXIT FROM SOFTWARE :-

After completion of all the data modifications, the user can exit from software by choosing option 5.

```
PS D:\code> █
```

```
Thank you for using this software !
```

- LOGIN FUNCTION :-

This function creates an encryption layer to access other functions of the software, the user will only be able to access the functions of the software only when he enters username and password correctly, if incorrect credentials are entered then user is allowed to enter the details once again

```
=====LOGIN TO CONTINUE=====
```

```
USERNAME: vinay_b
```

```
PASSWORD: *****█
```

SOURCE-CODE

```
//##### STUDENT DATABASE MANAGEMENT SYSTEM #####
#include <stdio.h>
#include <conio.h>
#include <windows.h>
void addstud();
void display();
void search();
void deleteRec();
void login();
struct student
{
    char usn[10];
    int roll;
    char name[100];
    char branch[40];
    char gender[10];
    float cgpa;
};
int main()
{
    int opt;
    while (opt != 5)
    {
        printf("\n\n\t\t\t\t\t=====STUDENT DATABASE MANAGER=====");
        printf("\n\n\n\t\t\t\t\t1) Add student");
        printf("\n\t\t\t\t\t2) Display students");
        printf("\n\t\t\t\t\t3) Search any student details");
        printf("\n\t\t\t\t\t4) Delete any student");
        printf("\n\t\t\t\t\t5) Exit from software\n\t\t\t\t\t_____");
        printf("\n\t\t\t\t\tEnter any option: ");
        scanf("%d", &opt);
        switch (opt)
        {
            case 1:
                system("cls");
                addstud();
                system("cls");
                break;
            case 2:
                system("cls");
                display();
                system("cls");
                break;
            case 3:
                system("cls");
                search();
```

```

        system("cls");
        break;
    case 4:
        system("cls");
        deleteRec();
        system("cls");
        break;
    case 5:
        system("cls");
        printf("\n\n\n\t\t\t\tThank you for using this software !");
        exit(0);
        break;
    default:
        system("cls");
        printf("\n\t\t\t\tIt was an invalid choice");
        printf("\n\t\t\t\tPlease try with other options");
    }
}
return 0;
}
void addstud()
{
    login();
    FILE *fp;
    int droll;
    char dusrn[10];
    char again;
    struct student s, t;
    fp = fopen("records.txt", "a+");
    if (fp == NULL)
    {
        printf("\t\t\t\tAn Error Occured");
        exit(0);
    }
x:
    printf("\n\n\t\t\t\t=====ADD ANY STUDENT'S DETAILS=====");
    printf("\n\n\t\t\t\t\tEnter USN          : ");
    scanf("%s", &t.usn);
    strcpy(dusrn, t.usn);
    printf("\n\t\t\t\t\tEnter roll number : ");
    scanf("%d", &t.roll);
    droll = t.roll;
    printf("\n\t\t\t\t\tEnter name          : ");
    scanf("%s", &t.name);
    printf("\n\t\t\t\t\tEnter branch        : ");
    scanf("%s", &t.branch);
    printf("\n\t\t\t\t\tEnter gender (M/F): ");
    scanf("%s", &t.gender);
    printf("\n\t\t\t\t\tEnter CGPA          : ");
    scanf("%f", &t.cgpa);

```

```

while (fread(&s, sizeof(struct student), 1, fp))
{
    if (droll == s.roll)
    {
        system("cls");
        printf("\n\t\t\t\t\tEntered data already exists !");
        printf("\n\t\t\t\t\tPlease enter a valid data...");
        goto x;
    }
    if (!strcmp(dusn, s.usn))
    {
        system("cls");
        printf("\n\t\t\t\t\tEntered data already exists !");
        printf("\n\t\t\t\t\tPlease enter a valid data...");
        goto x;
    }
}
fwrite(&t, sizeof(struct student), 1, fp);
printf("\n\t\t\t\t\tYOUR DATA HAS BEEN SAVED SUCCESFULLY !");
printf("\n\t\t\t\t\tEnter any key to exit...");
getch();
fclose(fp);
}
void display()
{
    login();
    FILE *fp;
    struct student s;
    fp = fopen("records.txt", "r");
    if (fp == NULL)
    {
        printf("\t\t\t\t\tAn Error Occured");
        exit(0);
    }
    printf("\n\n\t\t\t\t\t=====STUDENT'S RECORDS=====");
    while (fread(&s, sizeof(struct student), 1, fp))
    {
        printf("\n\n\t\t\t\t\tUSN      : %s", s.usn);
        printf("\n\t\t\t\t\tRoll num: %d", s.roll);
        printf("\n\t\t\t\t\tName      : %s", s.name);
        printf("\n\t\t\t\t\tBranch    : %s", s.branch);
        printf("\n\t\t\t\t\tGender    : %s", s.gender);
        printf("\n\t\t\t\t\tCGPA      : %.2f", s.cgpa);
        printf("\n\t\t\t\t\t_____");
    }
    fclose(fp);
    printf("\n\t\t\t\t\tEnter any key to exit...");
    getch();
}
void search()

```



```

while (fread(&s, sizeof(struct student), 1, fp))
{
    if (rollNo == s.roll)
        flag = 1;
    else
        fwrite(&s, sizeof(struct student), 1, fpt);
}
fclose(fp);
fclose(fpt);
if (!flag)
    printf("\n\n\t\t\tRecord not found !");
else
{
    remove("records.txt");
    rename("recover.txt", "records.txt");
    printf("\n\n\t\t\tRecord deleted successfully !");
}
printf("\n\n\t\t\tEnter any key to exit...");
getch();
}
void login()
{
    char c;
    int i, flag = 1;
    char usrName[30] = "vinay_b";
    char realPwd[30] = "vinay@123";
    char enteredPwd[30];
    system("cls");
y:
    printf("\n\t\t\t\t\t=====LOGIN TO CONTINUE=====");
    i = 0;
    printf("\n\n\t\t\t\t\tUSERNAME: ");
    scanf("%s", usrName);
    printf("\n\t\t\t\t\tPASSWORD: ");
    while ((c = getch()) != 13)
    {
        enteredPwd[i] = c;
        printf("*");
        i++;
    }
    enteredPwd[i] = '\0';
    if (!strcmp(enteredPwd, realPwd))
    {
        system("cls");
        printf("\n\n\n\t\t\t\t\tLogin Successful....");
    }
    else
    {
        system("cls");
        printf("\n\t\t\t\t\tIncorrect Credentials....");
    }
}

```

```
        printf("\n\t\t\t\tPlease Try again\n");  
        goto y;  
    }  
}
```

- VINAY_B

