## horizontal line



# Student Database Management System

Programming for problem-solving

Jawin Edsereown J

RA2111050010024

Ambarish S A

RA2111050010055

# Overview

**S**imple student database management system means the process of storing the information of students on the computer. Normally this system is used in schools and colleges to get the information of students and store it in a file. Information means student name, class, roll no or register no, performance, attendance, etc. To maintain this data or information, all universities use their own database management software to maintain student records neatly without any problem. This is very useful and helpful to the schools and colleges because these database management systems help maintain or store the records safely in the cloud or in their system.

So on this page, you will get the simple source code of the student database management system written using the C language

Below is the complete source code of the Student Database Management

#include"stdio.h"

#include"conio.h"

void addstudent();

void studentrecord();

void searchstudent();

void delete();

struct student {

char first\_name[20];

char last\_name[20];

int roll\_no;

char Class[10];

char vill[20];

float per;

};

void main()

{

int choice;

while(choice!=5){

printf("\t\t\t=====STUDENT DATABASE MANAGEMENT SYSTEM=====");

printf("\n\n\n\t\t\t\t 1. Add Student\n");

printf("\t\t\t\t 2. Students Records\n");

printf("\t\t\t\t 3. Search Student\n");

printf("\t\t\t\t 4. Delete Student\n");

printf("\t\t\t\t 5. Exit\n");

printf("\t\t\t\t \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

printf("\t\t\t\t ");

scanf("%d",&choice);

switch(choice){

case 1:

clrscr();

addstudent();

clrscr();

break;

case 2:

clrscr();

studentrecord();

printf("\t\t\t\t press any key to exit..... \n");

getch();

clrscr();

break;

case 3:

clrscr();

searchstudent();

printf("\n\t\t\t\t Press any key to exit.......\n");

getch();

clrscr();

break;

case 4:

clrscr();

delete();

printf("\n\t\t\t\tPress any key to exit.......\n");

getch();

clrscr();

break;

case 5:

clrscr();

printf("\n\t\t\t\tThank you, for used this software.\n\n");

exit(0);

break;

default :

clrscr();

getch();

printf("\n\t\t\t\t\tEnter a valid number\n\n");

printf("\t\t\t\tPress any key to continue.......");

getch();

clrscr();

break;

}

}

getch();

}

void addstudent(){

char another;

FILE \*fp;

int n,i;

struct student info;

do{

clrscr();

printf("\t\t\t\t=======Add Students Info=======\n\n\n");

fp=fopen("information.txt","a"); //use can give any file name. Give the name with extention or without extention.

printf("\n\t\t\tEnter First Name : ");

scanf("%s",&info.first\_name);

printf("\n\t\t\tEnter Last Name : ");

scanf("%s",&info.last\_name);

printf("\n\t\t\tEnter Roll-No : ");

scanf("%d",&info.roll\_no);

printf("\n\t\t\tEnter Class(course) : ");

scanf("%s",&info.Class);

printf("\n\t\t\tEnter Address : ");

scanf("%s",&info.vill);

printf("\n\t\t\tEnter Percentage : ");

scanf("%f",&info.per);

printf("\n\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

if(fp==NULL){

fprintf(stderr,"can't open file");

}else{

printf("\t\t\tRecord stored successfuly\n");

}

fwrite(&info, sizeof(struct student), 1, fp);

fclose(fp);

printf("\t\t\tYou want to add another record?(y/n) : ");

scanf("%s",&another);

}while(another=='y'||another=='Y');

}

void studentrecord(){

FILE \*fp;

struct student info;

fp=fopen("information.txt","r");

printf("\t\t\t\t=======STUDENTS RECORD=======\n\n\n");

if(fp==NULL){

fprintf(stderr,"can't open file\n");

exit(0);

}else{

printf("\t\t\t\tRECORDS :\n");

printf("\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\n\n");

}

while(fread(&info,sizeof(struct student),1,fp)){

printf("\n\t\t\t\t Student Name : %s %s",info.first\_name,info.last\_name);

printf("\n\t\t\t\t Roll NO : %d",info.roll\_no);

printf("\n\t\t\t\t Class : %s",info.Class);

printf("\n\t\t\t\t Village/City : %s",info.vill);

printf("\n\t\t\t\t Percentage : %f%",info.per);

printf("\n\t\t\t\t \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

}

fclose(fp);

getch();

}

void searchstudent(){

struct student info;

FILE \*fp;

int roll\_no,found=0;

fp=fopen("information.txt","r");

printf("\t\t\t\t=======SEARCH STUDENTS RECORD=======\n\n\n");

printf("\t\t\tEnter the roll no : ");

scanf("%d",&roll\_no);

while(fread(&info,sizeof(struct student),1,fp)>0){

if(info.roll\_no==roll\_no){

found=1;

printf("\n\n\t\t\tStudent Name : %s %s",info.first\_name,info.last\_name);

printf("\n\t\t\tRoll NO : %d",info.roll\_no);

printf("\n\t\t\tClass : %s",info.Class);

printf("\n\t\t\tAddress : %s",info.vill);

printf("\n\t\t\tPercentage : %f%",info.per);

printf("\n\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

}

}

if(!found){

printf("\n\t\t\tRecord not found\n");

}

fclose(fp);

getch();

}

void delete(){

struct student info;

FILE \*fp, \*fp1;

int roll\_no,found=0;

printf("\t\t\t\t=======DELETE STUDENTS RECORD=======\n\n\n");

fp=fopen("information.txt","r");

fp1=fopen("temp.txt","w");

printf("\t\t\t\tEnter the roll no : ");

scanf("%d",&roll\_no);

if(fp==NULL){

fprintf(stderr,"can't open file\n");

exit(0);

}

while(fread(&info,sizeof(struct student),1,fp)){

if(info.roll\_no == roll\_no){

found=1;

}else{

fwrite(&info,sizeof(struct student),1,fp1);

}

}

fclose(fp);

fclose(fp1);

if(!found){

printf("\n\t\t\t\tRecord not found\n");

}

if(found){

remove("information.txt");

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

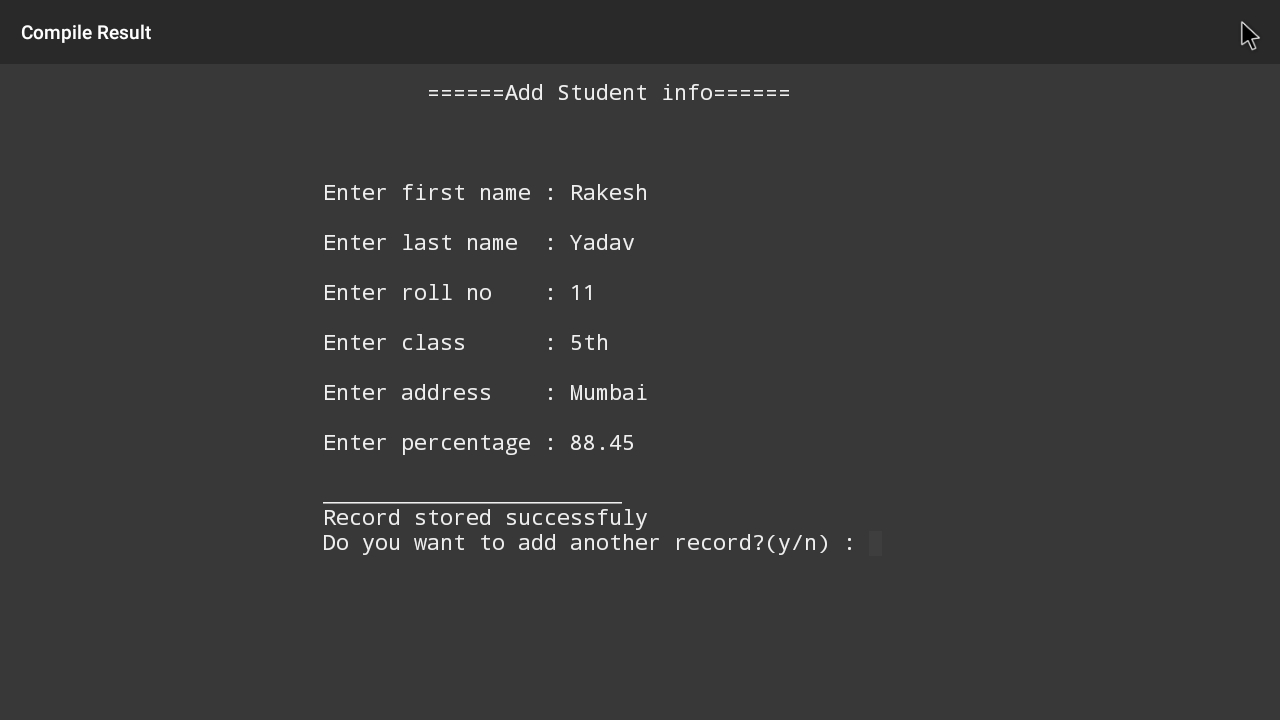
printf("\n\t\t\t\tRecord deleted succesfully\n");

}

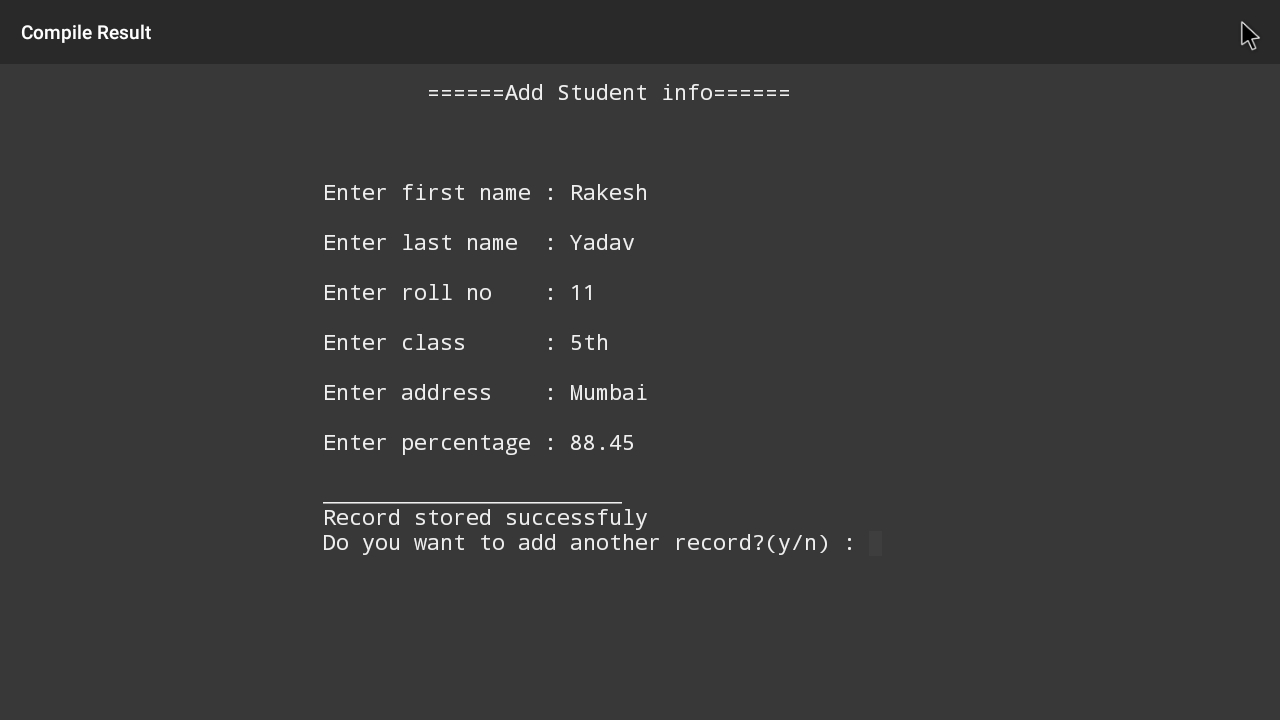
getch();

}

## **Options in the program :**

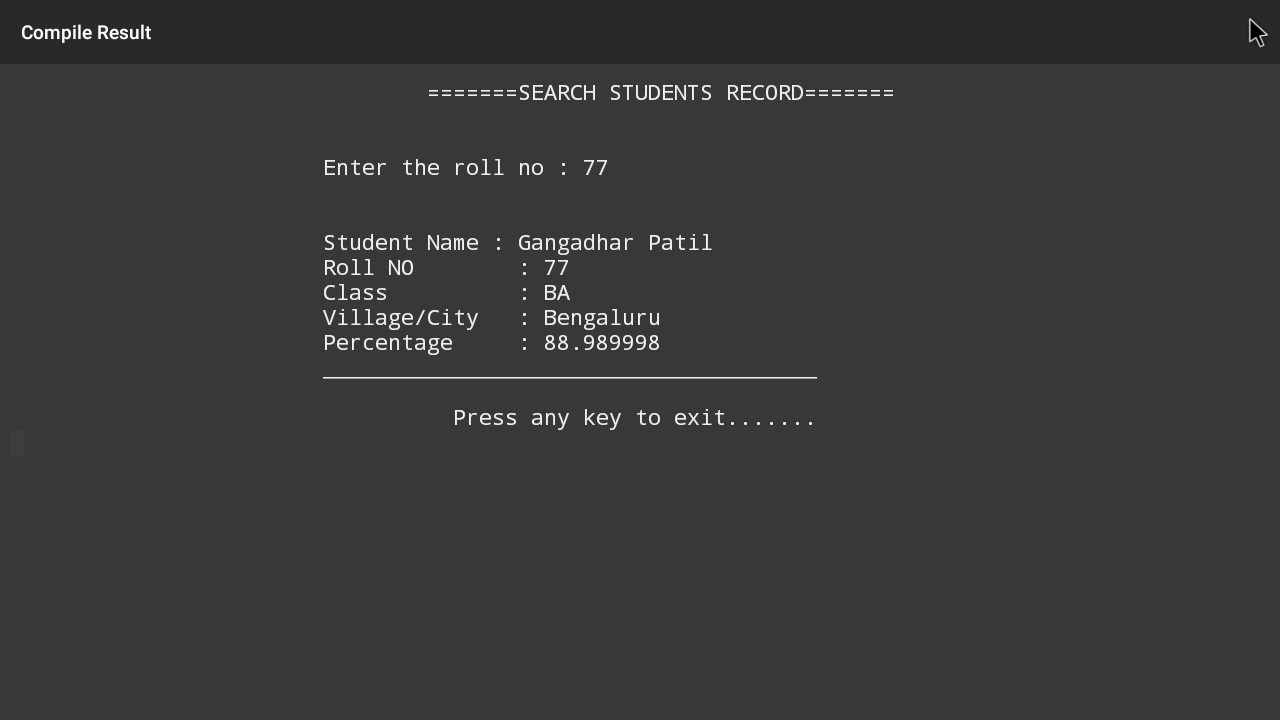


1. Add student record :



(Add student record) is used to insert or add the records of students in the list or in the file. You can add repeatedly and unlimited records of students

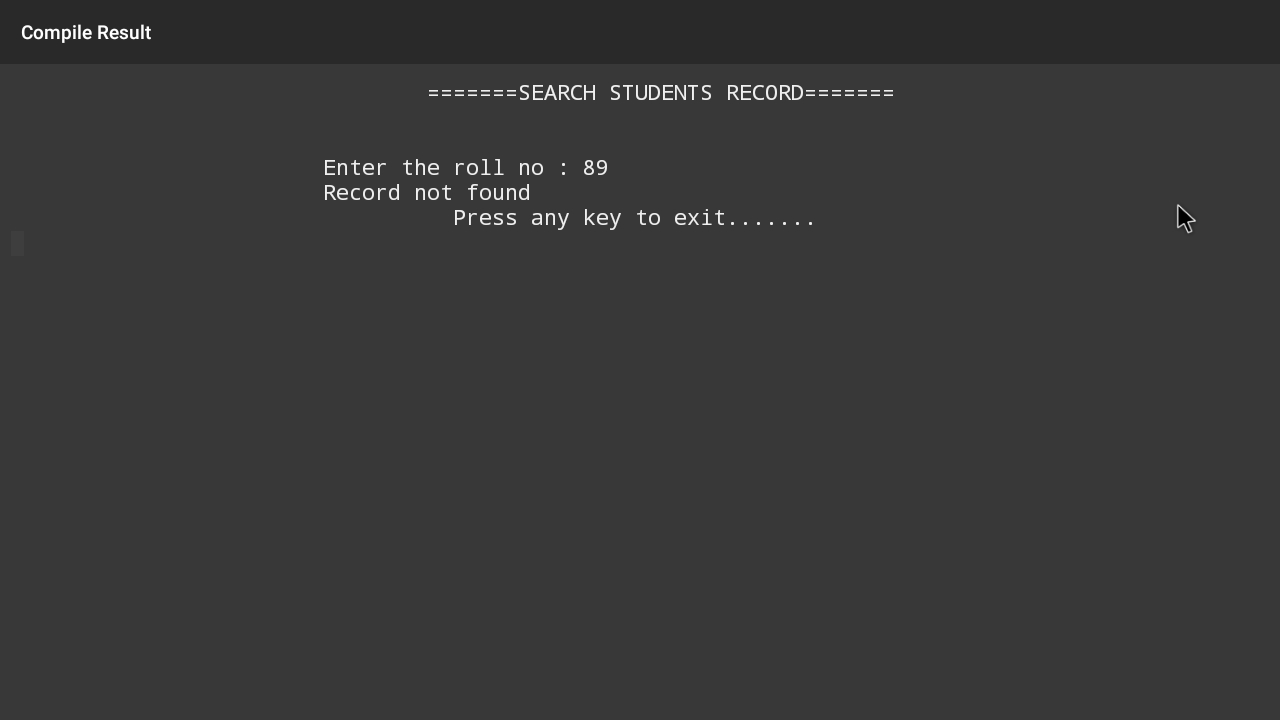
2. Student records :



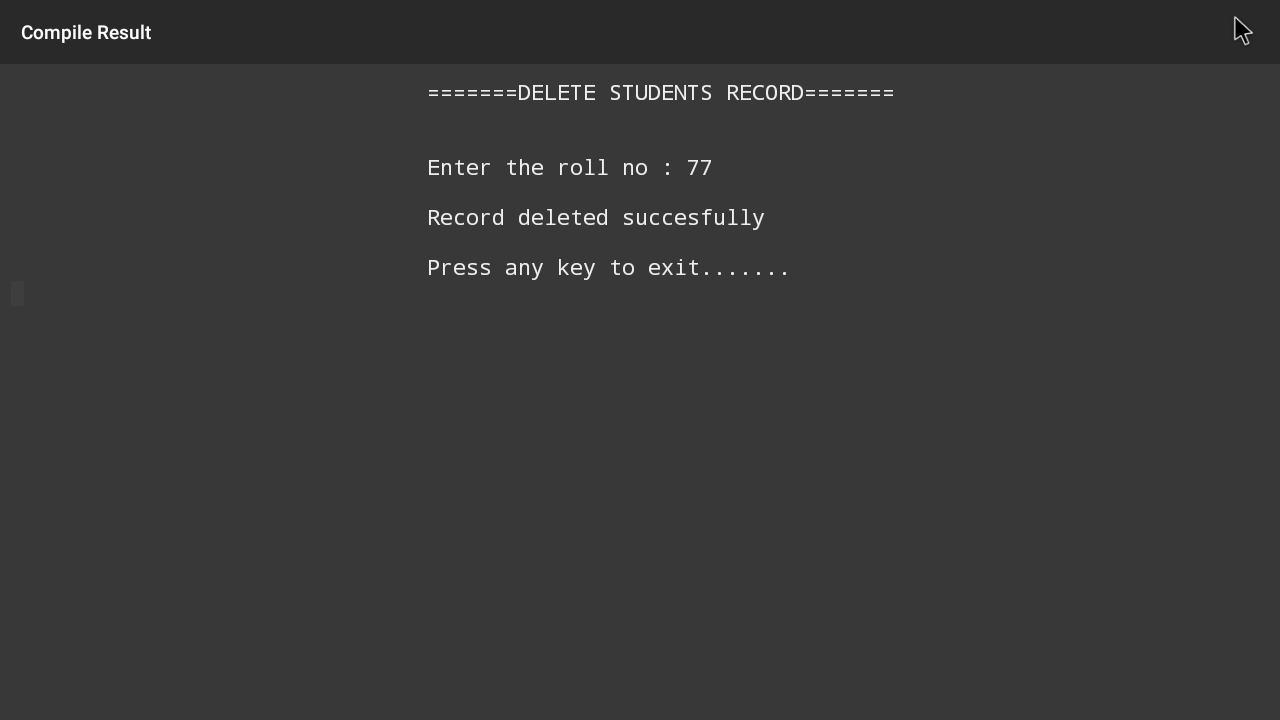
(Students Record) is used to watch the stored student records from the file or list in a systematic manner.

|  |
| --- |
|  |

3. Search student record:



(Search Student) is used to search or find the student record from the file or list. If the record is found then the program shows or prints the student information on the desktop.

4. Delete student record:

(delete a record) is used to delete the unwanted student record from the file or list. If the record is present and that record was deleted then the program prints Record deleted successfully.

|  |
| --- |

| 5. Exit:    (exit) is used to get out from the program or you can say exit from the program. |
| --- |
|  |

