MINI PROJECT CODE

```
#include<iostream>
#include<fstream>
#include<iomanip>
using namespace std;

// the class that stores data
class student
{
   int rollno;
   char name[50];
   int eng_marks, math_marks, sci_marks, lang2_marks, cs_marks;
   double average;
   char grade;

   public:
   void getdata();
   void showdata() const;
   void calculate();
   int retrollno() const;
};

void student::calculate()
{
   average=(eng_marks+math_marks+sci_marks+lang2_marks+cs_marks)/5.0;
   if(average>=90)
   grade='A';
```

```
    else if(average>=75)
    grade='B';
    else if(average>=50)
    grade='C';
    else
    grade='F';
}

void student::getdata()
{
    cout<<"Enter student's roll number: ";
    cin>>rollno;

    cout<<"Enter student's name: ";
    cin.ignore();
    cin.getline(name,50);

    cout<<"\n\nEnter Student's Marks"<<endl;
    cout<<"All marks should be out of 100";

    cout<<"\n\nEnter marks in English: ";
    cin>>eng_marks;

    cout<<"Enter marks in Math: ";
    cin>>math_marks;

    cout<<"Enter marks in Science: ";
    cin>>sci_marks;
```

```
    cout<<"Enter marks in Second language: ";
    cin>>lang2_marks;

    cout<<"Enter marks in Computer science: ";
    cin>>cs_marks;
    calculate();
}

void student::showdata() const
{
    cout<<"\nRoll number of student : "<<rollno;
    cout<<"\nName of student : "<<name;
    cout<<"\nEnglish : "<<eng_marks;
    cout<<"\nMaths : "<<math_marks;
    cout<<"\nScience : "<<sci_marks;
    cout<<"\nSecond Language : "<<lang2_marks;
    cout<<"\nComputer Science :"<<cs_marks;
    cout<<"\nAverage Marks :"<<average;
    cout<<"\nGrade of student is :"<<grade;
}

int student::retrollno() const
{
return rollno;
}

//function declaration

void create_student();
```

```
void display_sp(int);//display particular record
void display_all(); // display all records
void delete_student(int);//delete particular record
void change_student(int);//edit particular record

//Main Function

int main()
{
   char ch;
   cout<<setprecision(2);

   do
   {
      char ch;
      int num;
      system("cls");
      cout<<"Welcome to Student Registration Management System"<<endl;
      cout<<"\nMENU"<<endl;
      cout<<"\n1. Create a newstudent record"<<endl;
      cout<<"2. Search student record"<<endl;
      cout<<"3. Display all students records"<<endl;
      cout<<"4. Delete student record"<<endl;
      cout<<"5. Modify student record"<<endl;
      cout<<"6. Exit"<<endl;
      cout<<"\nEnter your choice"<<endl;
      cin>>ch;

      system("cls");
```

```
   switch(ch)
   {
      case '1': create_student(); break;
      case '2': cout<<"\n\n\tEnter The roll number: ";
      cin>>num;
      display_sp(num);
               break;
      case '3': display_all();
               break;
      case '4':    cout<<"\n\n\tEnter The roll number: ";
      cin>>num;
      delete_student(num);
                     break;
      case '5':    cout<<"\n\n\tEnter The roll number: "; cin>>num;
      change_student(num);
                     break;
      case '6':    cout<<"Exiting, Thank you!";exit(0);
   }

   }while(ch!='6');
   return 0;
}

//Write Student Details to File

void create_student()
{
   student stud;
```

```
    ofstream oFile;
    oFile.open("student.dat",ios::binary|ios::app);
    stud.getdata();
    oFile.write(reinterpret_cast<char *> (&stud), sizeof(student));
    oFile.close();
    cout<<"\nStudent's record has been created ";
    cin.ignore();
    cin.get();
}

// Read File Records

void display_all()
{
    student stud;
    ifstream inFile;
    inFile.open("student.dat",ios::binary);
    if(!inFile)
{
    cout<<"File could not be opened !! Press any Key to exit";
    cin.ignore();
    cin.get();
    return;
}

cout<<"\n\n\n\t\tDISPLAYING ALL RECORDS\n\n";
while(inFile.read(reinterpret_cast<char *> (&stud), sizeof(student)))
{
    stud.showdata();
```

```
    cout<<"\n\n====================================\n";
}

inFile.close();
cin.ignore();
cin.get();
}

//Read Specific Record Based on Roll Number

void display_sp(int n)
{
    student stud;
    ifstream iFile;
    iFile.open("student.dat",ios::binary);
    if(!iFile)

    {
        cout<<"File could not be opened... Press any Key to exit";
        cin.ignore();
        cin.get();
        return;
    }

    bool flag=false;
    while(iFile.read(reinterpret_cast<char *> (&stud), sizeof(student)))
    {
        if(stud.retrollno()==n)
        {
```

```
            stud.showdata();
            flag=true;
        }
    }

    iFile.close();
    if(flag==false)
    cout<<"\n\nrecord does not exist";
    cin.ignore();
    cin.get();
}

// Modify Record for Specified Roll Number

void change_student(int n)
{
    bool found=false;
    student stud;
    fstream fl;
    fl.open("student.dat",ios::binary|ios::in|ios::out);
    if(!fl)

    {
        cout<<"File could not be opened. Press any Key to exit...";
        cin.ignore();
        cin.get();
        return;
    }
```

```
   while(!fl.eof() && found==false)
   {
      fl.read(reinterpret_cast<char *> (&stud), sizeof(student));
      if(stud.retrollno()==n)
      {
         stud.showdata();
         cout<<"\nEnter new student details:"<<endl;
         stud.getdata();
         int pos=(-1)*static_cast<int>(sizeof(stud));
         fl.seekp(pos,ios::cur);
         fl.write(reinterpret_cast<char *> (&stud), sizeof(student));
         cout<<"\n\n\t Record Updated";
         found=true;
      }
   }

   fl.close();
   if(found==false)
   cout<<"\n\n Record Not Found ";
   cin.ignore();
   cin.get();
}

//Delete Record with Particular Roll Number

void delete_student(int n)
{
   student stud;
   ifstream iFile;
```

```
    iFile.open("student.dat",ios::binary);
    if(!iFile)
    {
       cout<<"File could not be opened... Press any Key to exit...";
       cin.ignore();
       cin.get();
       return;
    }
    ofstream oFile;
    oFile.open("Temp.dat",ios::out);
    iFile.seekg(0,ios::beg);
    while(iFile.read(reinterpret_cast<char *> (&stud), sizeof(student)))
    {
       if(stud.retrollno()!=n)
       {
          oFile.write(reinterpret_cast<char *> (&stud), sizeof(student));
       }
    }
    oFile.close();
    iFile.close();
    remove("student.dat");
    rename("Temp.dat","student.dat");
    cout<<"\n\n\tRecord Deleted ..";
    cin.ignore();
    cin.get();
}
```