

## Student Performance Management

Create a console application to maintain the data about -

Create a database for the above requirements

Console application should have a menu system -

- Add Student
- Add course
- Add subject
- Enter marks of student
- Edit Student data
- Edit course
- Edit subject
- Delete Student
- Delete Subject
- Delete Course
- Delete marks
- List all students with details of course, subject, and total marks
- Ask roll number to display subject wise marks for a student
- List course wise average marks
- List course wise max marks
- List course wise topper

## Database Requirements

- Student

- Roll No, Name, email, address
- Each Student will belong to a course
- Sample Data

RollNo	Name	Email	Address
1001	Ak shay Shroff	<a href="mailto:akskayshroff@gmail.com">akskayshroff@gmail.com</a>	Ahmednagar

- Course

- Code, Title, description
- Sample Data

Code	Title	Description
SCI001	Science 1	Pure Science Course
SCI002	Science with Economics	Pure Science Course with Eco
ART001	Fine Arts	Elementary Drawing

- Subjects

- Code, Title, description
- Each Subject will belong to a course.
  - So, create the required foreign key field
- Sample Data

Code	Title	Description	???
PHY001	Physics	Elementary Physics	
PHY002	Modern Physics	Modern Physics	
SKT001	Sketching	Basic Art Sketching	

- Subject wise student marks

- Create the required fields
- Sample Data

StudentId	SubjectId	Marks

## Note

1. All the above operations should happen through Stored Procedures
  - a. So, Create one SP per menu Item
2. Create schema student, course and marks
3. Make sure to have proper PK and FK relationships
4. This project should not use cascading delete rule in database.
  - a. Instead handle the delete with code inside the stored procedure
5. In the Console application create the following class structure with methods
  - a. Models
    - i. class Student
    - ii. class Course
    - iii. class Subject
    - iv. class Marks
  - b. Services
    - i. class StudentService
      1. public void Add(Student s)
      2. public void Edit(Student s)
      3. public void Delete(int studentId)
      4. public void DisplayStudentReport()
    - ii. class CourseService
      1. public void Add(Course c)
      2. public void Edit(Course c)
      3. public void Delete(int courseId)
      4. public void DisplayCourseWiseAverageMarks()
      5. public void DisplayCourseWiseHighestMarks()
      6. public void DisplayCourseWiseTopper ()
    - iii. class SubjectService
      1. public void Add(Subject s)
      2. public void Edit(Subject s)
      3. public void Delete(int subjectId)
    - iv. class MarksService
      1. public void Add(string rollNo, string subject, int marks)
      2. public void Delete(string rollNo, string
      3. public void DisplayMarksByRollNo (string rollNumber)