# Classroom

## Preparation

Download the skeleton provided in Judge. **Do not** change the **packages**.

**Pay attention to name the package (classroom), all the classes, their fields and methods exactly the same way they are presented in the following document. It is also important to keep the project structure as described above.**

**Problem Description**

Your task is to create a repository which stores departments by creating the classes described below.

### Student

First, write a Java class Student with the following fields:

* **firstName: String**
* **lastName: String**
* **bestSubject: String**

The class **constructor** should receive (**firstName, lastName** and **bestSubject**).

The class also should have the methods:

* getFirstName()
* getLastName()
* getBestSubject()
* Override the **toString()** method in the following format:

**"Student: First Name= {firstName}, Last Name= {lastName}, Best Subject= {bestSubject}"**

### Classroom

**Next**, write a **Java** class Classroom that has **students** (a collection which stores the entity **Student**). All entities inside the repository have the **same fields**. Also, the Students class should have those fields:

* **capacity:** int
* **data:** List<Student> **-** holds all added students in the classroom

The class **constructor** should receive (**capacity**), also it should initialize the **students** with a new instance of the collection**.**

Implement the following features:

* getCapacity()
* getStudents()
* getStudentCount() method– **returns** the **number** of **students in the classroom**
* registerStudent(Student student) method – **adds** an **entity** to the data **if** **there** **is** **room** for it
  + Returns **"Added student {firstName} {lastName}"** if the student is **successfully added**
  + Returns **"Student is already in the classroom"** if the student is already in the classroom
  + Returns **"No seats in the classroom"** if the classroom is full
* dismissStudent(Student student) method – **removes the student**
  + Returns **"Removed student {firstName} {lastName}"** if the student is **successfully removed**
  + Returns **"Student not found"** if the student is not in the classroom
* **getSubjectInfo(String subject)** method – returns all the students with the **given subject in the following format:**

**"Subject: {subjectName}  
Students:  
{firstName} {lastName}  
{firstName} {lastName}  
…"**

* + Returns **"No students enrolled for the subject"** if the student is not in the classroom
* **getStudent(String firstName, String lastName)** method **-** returns the student with the **given names**.
* **getStatistics()** – **returns** a **String** in the following **format**:
  + **"Classroom size: {added\_students\_count}  
     ==Student: First Name= {firstName} , Last Name= {lastName} , Best Subject= {bestSubject}   
     ==Student: First Name= {firstName} , Last Name= {lastName} , Best Subject= {bestSubject}**

**(…)**"

## Constraints

* The **combinations** of **names** will **always be unique**.
* The **capacity** will always be **a positive** **number**.

## Examples

This is an example how the **Classroom** class is **intended to be used**.

| Sample code usage |
| --- |
| *// Initialize the repository* Classroom classroom = new Classroom(10); *// Initialize entities* Student student = new Student("Peter", "Parker", "Geometry");  Student studentTwo = new Student("Sarah", "Smith", "Algebra");  Student studentThree = new Student("Sam", "Winchester", "Algebra");  Student studentFour = new Student("Dean", "Winchester", "Music");  Student studentFive = new Student("Ellie", "Goulding", "Music"); *// Print Student* System.*out*.println(student);  *// Student: First Name = Peter, Last Name = Parker, Subject = Geometry // Register Student* String register = classroom.registerStudent(student);  System.*out*.println(register); *// Added student Peter Parker* String registerTwo = classroom.registerStudent(studentTwo);  String registerThree = classroom.registerStudent(studentThree);  String registerFour = classroom.registerStudent(studentFour); *// Dismiss Student* String dismissed = classroom.dismissStudent(student);  System.*out*.println(dismissed); *// Removed student Peter Parker* String dismissedTwo = classroom.dismissStudent(studentFive);  System.*out*.println(dismissedTwo); *// Student not found // Subject info* String subjectInfo = classroom.getSubjectInfo("Algebra");  System.*out*.println(subjectInfo); *// Subject: Algebra // Students: // Sarah Smith // Sam Winchester* String anotherInfo = classroom.getSubjectInfo("Art");  System.*out*.println(anotherInfo); *// No students enrolled for the subject // Get Student* System.*out*.println(classroom.getStudent("Dean", "Winchester")); *// Student: First Name = Dean, Last Name = Winchester, Subject = Music* |

## Submission

Submit **single .zip file**, containing **christmas package, with the classes inside (Student, Classroom and the Main class**, there is no specific content required inside the Main class e. g. you can do any kind of local testing of you program there. However there should be **main(String[] args)** method inside.