

Assignment

Create a Java project. Add 3 classes named **Grading**, **Student** and **Assignment** respectively to the project. You must comply with the following variable names.

In this assignment, you will have to operate the **Grading class** and **Student class** in the **Assignment class** to record the score of Students' assignments.

Each Student has a **name**, **grades (containing five grades)** and **gradesIndex (recording max index of grades)**. Each Grading has a **passMark** attribute which determines the **passing threshold** of the grades. In the **Assignment class**, you will have to create a Student object and Grading object. The Student object records the basic information of the student, and the Grading object is used to analyze the grades of the Student. Finally, you have to show the basic information of the student and the analyzed result generated by the Grading object in the console like the sample output shows.

Requirements:

- Read instructions and create classes needed. You are supposed to add 3 classes (2 required + 1 Assignment) to the project.
- Your code must be properly formatted with sensible variable names! Refer to the text for code format examples.
- The instruction for Assignment and outputs is for your reference.
- Make sure your classes are correctly implemented with the public interfaces.
- Note that all instance variables are private. Please use public interfaces to access private variables.
- The following diagram describes two classes you need to implement.

Class Name	Grading	Student
Attributes	int passMark	String name int[5] grades int gradesIndex
Methods	String toLetterGrade(int) double calculateAvg(int[]) String summarizeGrade(int[])	int getGrade(int) void addGrade(int) String info()

Create **Student** class

Student	
Modifier and type	Method (or Variable) and description
Instance variable	
String	name The student's name
int[]	grades An array that can store 5 grades.

int	gradesIndex The initial value is 0. This variable is used as a counter for grades.
Constructor	
Student(String name) Constructs a student object with a given student name and an empty array of grades.	
Instance methods	
-	getter(s) & setter(s)
int	getGrade(int idx) Gets the value in grades by specific index.
void	addGrade(int grade) If gradesIndex is in valid range, add a new grade to grades at gradesIndex and gradesIndex + 1. If the index is out of bound, that is, gradesIndex is 5 or greater, print an error message "Array index out of bounds."
String	info() Returns a formatted String that describes the information about the student. (See sample output.) Please follow the right alignment as sample output.

Create **Grading** class

Grading															
Modifier and type	Method (or Variable) and description														
Instance variable															
int	passMark The pass marks. (For example, the pass mark for undergraduate school is 60.)														
Constructor															
Grading(int passMark) Constructs a grading object with a given passMark.															
Instance methods															
-	getter(s) & setter(s)														
String	toLetterGrade(int score) Converts the grade to the corresponding letter grade and returns it (see table 1 for grade reference). <div style="text-align: center; margin: 10px 0;"> Table 1 Grade reference </div> <table border="1" style="margin: auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 50%;">Score Range</th> <th style="width: 50%;">Letter Grade</th> </tr> </thead> <tbody> <tr> <td>100 ~ 80</td> <td>A</td> </tr> <tr> <td>70 ~ 79</td> <td>B</td> </tr> <tr> <td>60 ~ 69</td> <td>C</td> </tr> <tr> <td>50 ~ 59</td> <td>D</td> </tr> <tr> <td>1 ~ 49</td> <td>E</td> </tr> <tr> <td>0</td> <td>X</td> </tr> </tbody> </table>	Score Range	Letter Grade	100 ~ 80	A	70 ~ 79	B	60 ~ 69	C	50 ~ 59	D	1 ~ 49	E	0	X
Score Range	Letter Grade														
100 ~ 80	A														
70 ~ 79	B														
60 ~ 69	C														
50 ~ 59	D														
1 ~ 49	E														
0	X														
double	calculateAvg(int[] grades) Calculates the average of the input array and returns the avg. score. (To make it simple, don't count 0)														
String	summarizeGrade(int[] grades) Returns a string that describes the average score, and pass/failed count (Don't count 0) of the input parameter. (See sample output.) Must call calculateAvg(...) and toLetterGrade(...) .														

Assignment (you have to fill the blank _____)

```
public class Assignment {  
  
    public static void main(String[] args) {  
        Student stu1 = new Student(_____);  
        Grading grading1 = new Grading(60);  
        Grading grading2 = new Grading(80);  
        stu1.addGrade(100);  
        stu1.addGrade(78);  
        stu1.addGrade(55);  
        stu1.addGrade(67);  
        stu1.addGrade(98);  
        stu1.addGrade(90);  
  
        System.out.println("information:");  
        System.out.println(_____);  
        System.out.println("grading1 summarizeGrade(...)");  
        System.out.println(_____);  
        System.out.println("grading2 summarizeGrade(...)");  
        System.out.println(_____);  
    }  
}
```

Output

your output must exactly the same as below except student name

```
Array index out of bounds.  
information:  
Name: Jack  
Grades: 100 78 55 67 98  
  
grading1 summarizeGrade(...)  
Avg. Score: 79.60  
Pass: 4, Failed: 1  
All Grades Letter: A B D C A  
  
grading2 summarizeGrade(...)  
Avg. Score: 79.60  
Pass: 2, Failed: 3  
All Grades Letter: A B D C A
```