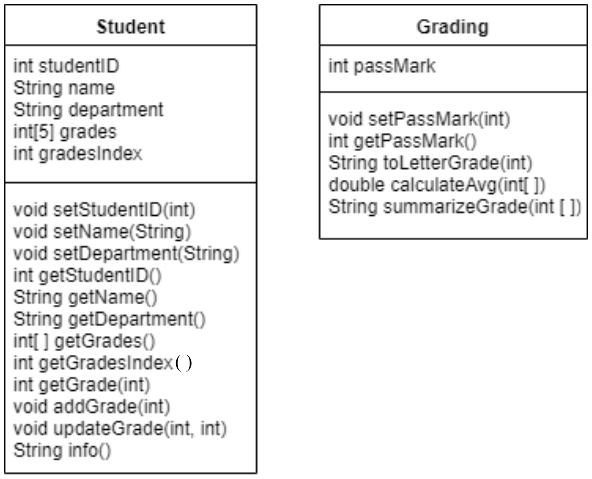
Assignment 3

Requirements:

- Create a Java project named yourStudentId_HW3
- Read instructions and create classes needed. You are supposed to add 3 classes (2 required + 1 Tester) to the project.
- Your code must be properly formatted with sensible variable names! Refer to the text for code format examples.
- The instruction for Tester and output are for your reference.
- Make sure your classes correctly implement the public interfaces.

The following diagram describes two class you need to implement.



```
Student
int studentID
String name
String department
int[5] grades
int gradesIndex

void setStudentID(int)
void setName(String)
void setDepartment(String)
int getStudentID()
String getName()
String getDepartment()
int[ ] getGrades()
int getGradesIndex( )
int getGrade(int)
void addGrade(int)
void updateGrade(int, int)
String info()
```

```
Grading
int passMark

void setPassMark(int)
int getPassMark()
String toLetterGrade(int)
double calculateAvg(int[ ])
String summarizeGrade(int [ ])
```

1. Create **Student** class

Student	
Modifier and type	**Method (or Variable) and description**
Instance variable	
int	studentID The student ID.
String	name The student's name
String	department The department the student belongs to.
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(int studentID, String name, String department) **Constructs a student object with given student id, name, and department, and an empty array of grades.**	
Instance methods	
-	3 setter for 3 attributes. 5 getter for 5 attributes.
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**".
void	updateGrade(int idx, int grade) Updates the value in grades at given index. (Suppose idx is in valid range)
String	info() Returns a **formatted String** that describe the information about the student. (See sample output.)

2. 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 given passMark.**	
Instance methods	
-	one getter for an attribute. one setter for an attribute.
String	toLetterGrade(int score) Converts the grade to the corresponding letter grade and returns it (see table 1 for grade reference).
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 describe the score, the average score, and pass/failed count **(Don't count 0)** of the input parameter. (See sample output.) Must call calculateAvg(…).

3. Write **Javadoc** and generate Javadoc by using "Project -> Generate Javadoc…" in Eclipse.

4. Note that all instance variables are private. Please use public interfaces to access private variables.

5. Hint: the input parameter in calculateAvg(…) and summarizeGrade(…) are from the student's object

Table 1 Grade reference

Score Range	Letter Grade
100 ~ 80	A
70 ~ 79	B
60 ~ 69	C
50 ~ 59	D
1 ~ 49	E
0	X

Tester

```
public class Tester {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Student stu1 = new Student(…);
        Grading grading1 = new Grading(60);
        stu1.addGrade(100);
        stu1.addGrade(70);
        stu1.addGrade(50);
        stu1.addGrade(90);
        stu1.addGrade(67);
        stu1.addGrade(98);
        stu1.updateGrade(4, 98);
        System.out.println("------info()");
        System.out.println(…);
        System.out.println("summarizeGrade(…)");
        System.out.println(…);
    }

}
```

Output

```
Grade added! 100
Grade added! 70
Grade added! 50
Grade added! 90
Grade added! 67
Array index out of bounds
Updated! 98
------info()
Student ID: 107356015
      Name:     Simon
Department:       MIS
    Grades: 100  70  50  90  98
summarizeGrade(…)
Avg. Score: 81.6
Pass: 4, failed: 1
```

Submission: Submit your project as "zip (or rar) file" via WM5. No other submissions will be graded.

Reminder: Please zip the whole project

Deadline: 2019/11/17 23:59 (for both Mon56 and Tue23)