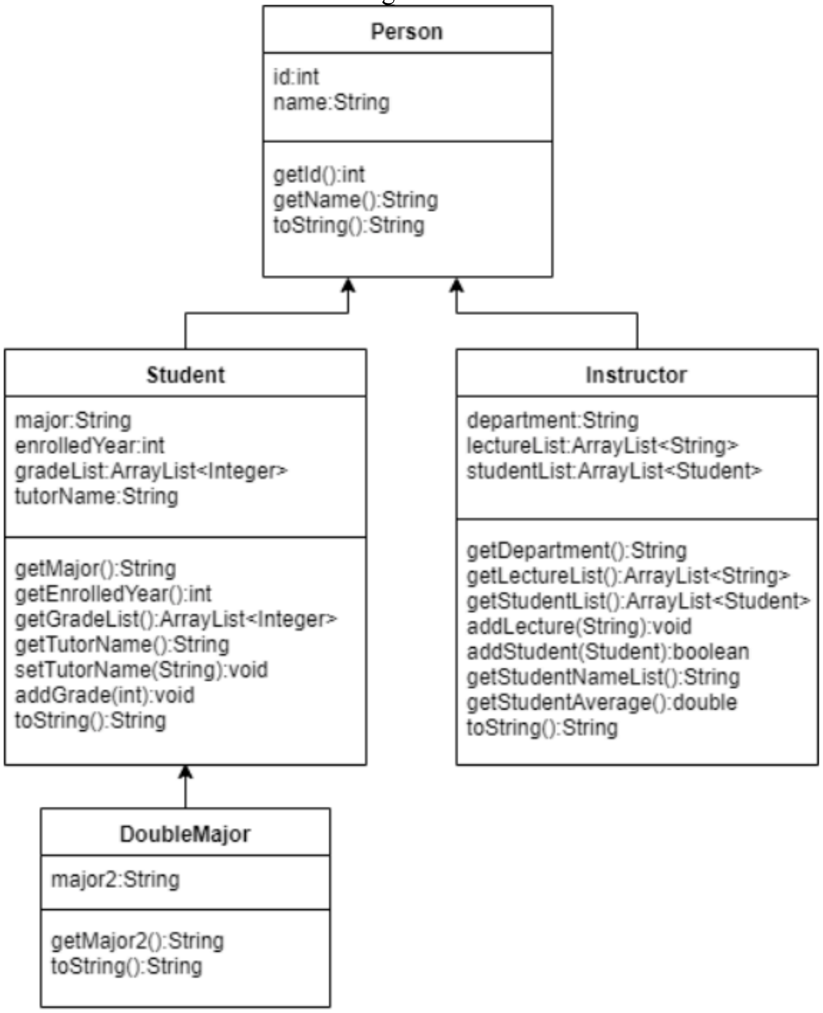
Assignment 5

Requirements:

- Create a Java project named yourStudentId_HW5
- Read instructions and create classes needed. You are supposed to add 5 classes (4 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 four classes you need to implement.

Figure 1



1. Create **Person** class

Person	
Modifier and type	**Method (or Variable) and description**
Instance variable	
int	id The person’s ID.
String	name The person’s name.
Constructor	
Person(int id, String name) **Constructs a Person object with given id and name.**	
Instance methods	
-	For setter and getter please refer to Figure 1.
String	toString() Returns a String description of the person’s information. (See example output below)

2. Create **Student** class

Student	
Modifier and type	**Method (or Variable) and description**
Instance variable	
String	major The major of this student.
int	enrolledYear The year that this student enrolled.
ArrayList<Integer>	gradeList An ArrayList that contains the grades of this student.
String	tutorName The name of this student’s tutor.
Constructor	
Student(int id, String name, String major, int enYear) **Constructs a Student object with given id, name, major, enrolledYear and initialize gradeList.**	
Instance methods	
-	For setter and getter please refer to Figure 1.
void	addGrade(int grade) Puts given grade into the gradeList.
String	toString() Returns a String description of the student. (See example output below)

3. Create **Instructor** class

Instructor	
Modifier and type	**Method (or Variable) and description**
Instance variable	
String	department The department of this instructor.
ArrayList<String>	lectureList An ArrayList that contains the lectures of this Instructor.
ArrayList<Student>	studentList An ArrayList that contains the students of this Instructor.
Constructor	
Instructor(int id, String name, String dep) **Constructs an Instructor object with given id, name, department and initialize lectureList and studentList.**	
Instance methods	
-	For setter and getter please refer to Figure 1.
void	addLecture(String lecture) Puts given lecture into the lectureList.
boolean	addStudent(Student stu) Checks if this student has a tutor or not. If does, return false; Otherwise, add this student into the studentList, and set the tutor name of this student. Also don't forget to return a true boolean result.
String	getStudentNameList() Returns all students' name in studentList. (See example output below)
double	getStudentAverage() Calculates and returns the average score of students that taught by this instructor. Calculates average to the second decimal place.
String	toString() Returns a String description of the instructor. (See example output below)

4. Create **DoubleMajor** class

DoubleMajor	
Modifier and type	**Method (or Variable) and description**
Instance variable	
String	major2 The second major of this student..
Constructor	
DoubleMajor(int id, String name, String major, int enYear, String major2) **Constructs a DoubleMajor object with given id, name, major enrolledYear and double major.**	
Instance methods	
-	For setter and getter please refer to Figure 1.
String	toString() Returns a String description of the student. (See example output below)

5. Hint: String.substring(int beginIndex, int endIndex)

Tester

```
public class Tester {
      public static void main(String[] args) {

            Person person = new Person(123, "Leon");
```

```
        Instructor ins1 = new Instructor(111, "Jennifer", "MIS");
        ins1.addLecture("OOPI");
        ins1.addLecture("OOPII");

        Student stu1 = new Student(789, "Simon", "MIS", 107);
        stu1.addGrade(85);
        stu1.addGrade(95);

        DoubleMajor stu2 = new DoubleMajor(456, "David", "MIS", 106, "STAT");
        stu2.addGrade(93);
        stu2.addGrade(98);

        System.out.println(ins1.addStudent(stu1));
        System.out.println(ins1.addStudent(stu2));
        System.out.println(ins1.addStudent(stu1));

        System.out.println(person.toString());
        System.out.println(ins1.toString());
        System.out.println(stu1.toString());
        System.out.println(stu2.toString());
        System.out.println("-----------------------");
        System.out.println(ins1.getStudentNameList());
        System.out.println(ins1.getStudentAverage());
    }
}
```

Output

```
true
true
false
Person[id=123, name=Leon]
Instructor[id=111, name=Jennifer, department=MIS, lectureList=OOPI, OOPII,
studentList=Simon, David]
Student[id=789, name=Simon, major=MIS, tutorName=Jennifer, enrolledYear=107,
gradeList=85, 95]
DoubleMajor[id=456, name=David, major=MIS, major2=STAT, tutorName=Jennifer,
enrolledYear=106, gradeList=93, 98]
-----------------------
Simon, David
92.75
```

Submission: ***IMPORTANT**

1. Submit **"class" file** via https://140.119.19.74:8443/oop/

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

Deadline: 2019/12/29 23:59 (for both Mon56 and Tue23)