INTRODUCTION TO PROGRAMMING I - ASSIGNMENT WEEK 5

Deliverables

You must deliver on Moodle:

- a single PDF file containing all your textual answers and Class Diagram using the indicated tool;
- a single compacted file (e.g. ZIP) containing ONLY your source code (.java files). Your code should not use any additional library but standard Java API.

Questions about this assignment should be submitted on Moodle, section 'Announcements and forum', topic 'QA - Assignment of week 5'.

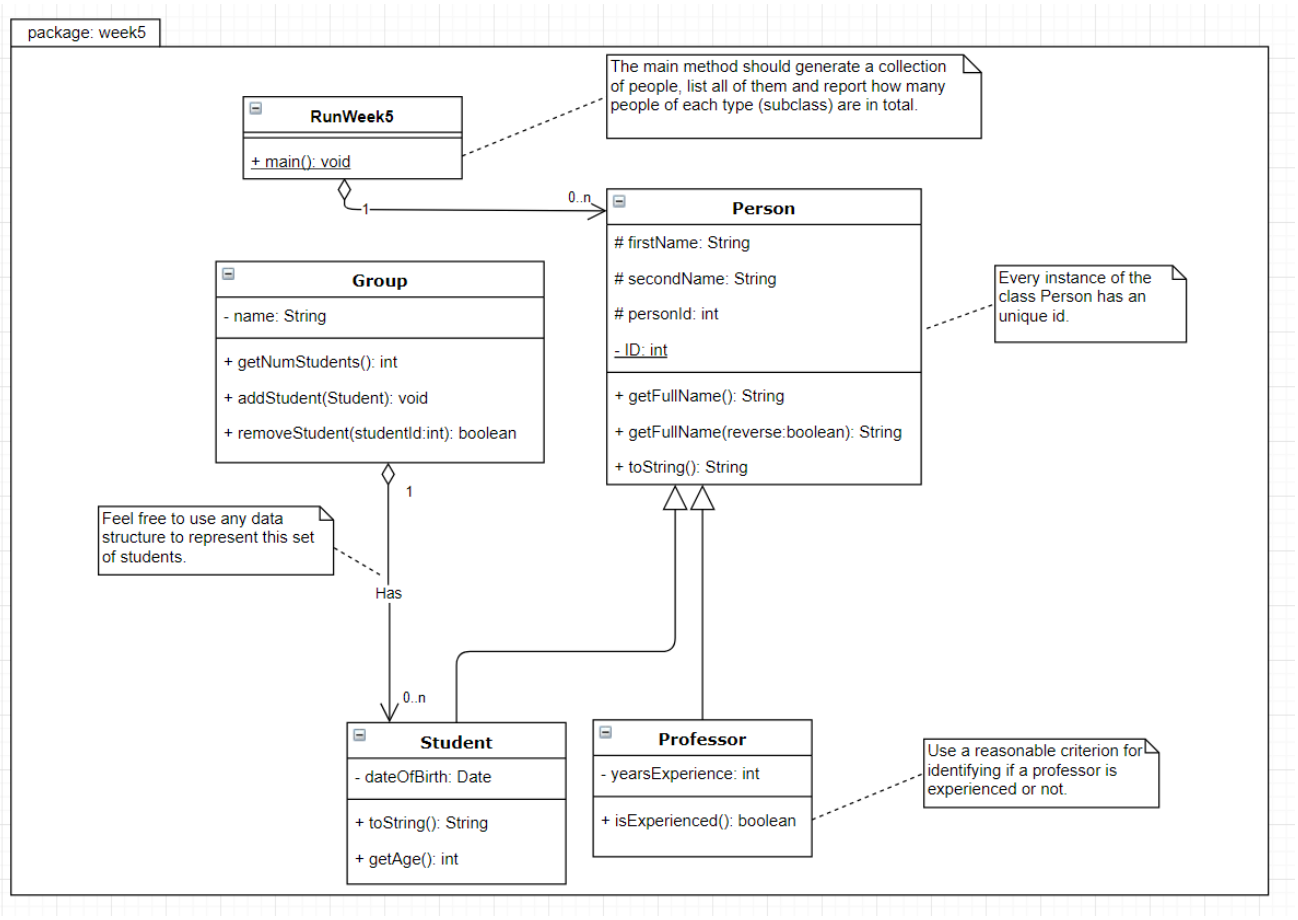
Question 1 (20 points) - Consider the following class declaration.

```
class QuestionOne {
	public final int A = 345;
	public int b;
	private float c;

	private void methodOne(int a) {
		b = a;
	}

	public float methodTwo() {
		return 23;
	}
}
```

Identify invalid statements in the class Q1Main. For each invalid statement, write one short phrase why it is invalid. Each phrase should consist of at most 90 characters and mention the line in which the error occurs (it is included in the 90 characters). Failing in attending these requirements for **any** statement will automatically and unquestionably imply in loss of all the **entire grade** for this question, i.e. 20 points.

```
public class Q1Main {
	public static void main(String[] args) {
		QuestionOne q1;
		q1 = new QuestionOne();
		q1.A = 12;
		q1.b = 12;
		q1.c = 12;
		q1.methodOne(12);
		q1.methodOne();
		System.out.println(q1.methodTwo(12));
		q1.c = q1.methodTwo();
	}
}
```

Question 2 (80 points) – Given the following partial UML Class Diagram, do as follows:



2.1. (30 points) Create three additional classes in the diagram. At most two of these classes should extend the class Person. The diagram should be generated using https://www.draw.io/

Link for the diagram above:
https://drive.google.com/file/d/1psuzmJe4XB_jwAYzjzxZVBV_4X0WFsLT/view?usp=sharing

2.2. (50 points) Implement the entire diagram. Follow all the instructions in the comments, as some of the features in the diagram require additional attention:

a) Method RunWeek5.main()

b) Use any data structure provided in the standard Java API for storing collections of people and students.

c) Each person has a unique ID.