
Tutorial 2.2

Object-Oriented Programming

Overview

- You will be doing online collaborative coding to solve the problem given in this activity.
- You will be coding in Dart programming language.
- The goal of this exercise is to learn some key features in Dart programming language, i.e. as stated in the title of this exercise

Breakout Sessions

- I will split the main meeting room on Webex into several rooms. Each room will have 3 or 4 members.
- You will be assigned to a room. See the attached file to find which room you belong to.
- Do introduce yourself to your room members before getting started.
- Turn on your webcam only during the introduction session. After that, turn off the webcam to reduce network traffic.

Collaborative Coding

- In each room, appoint one member to be the host. The host member will initiate a session for collaborative coding using Live Share.
- Other members will be invited as collaborators.
- All members (host and collaborators) will need to open VS Code.
- The host member will also need to share his/her screen via Webex.

Problem

Given a class diagram that models students along with their results as shown in Figure 1. There are three classes in the design, i.e., **Person**, **Result** and **Student**, and they form inheritance relationship as depicted in the diagram. Note that, class `Result` is a mixin.

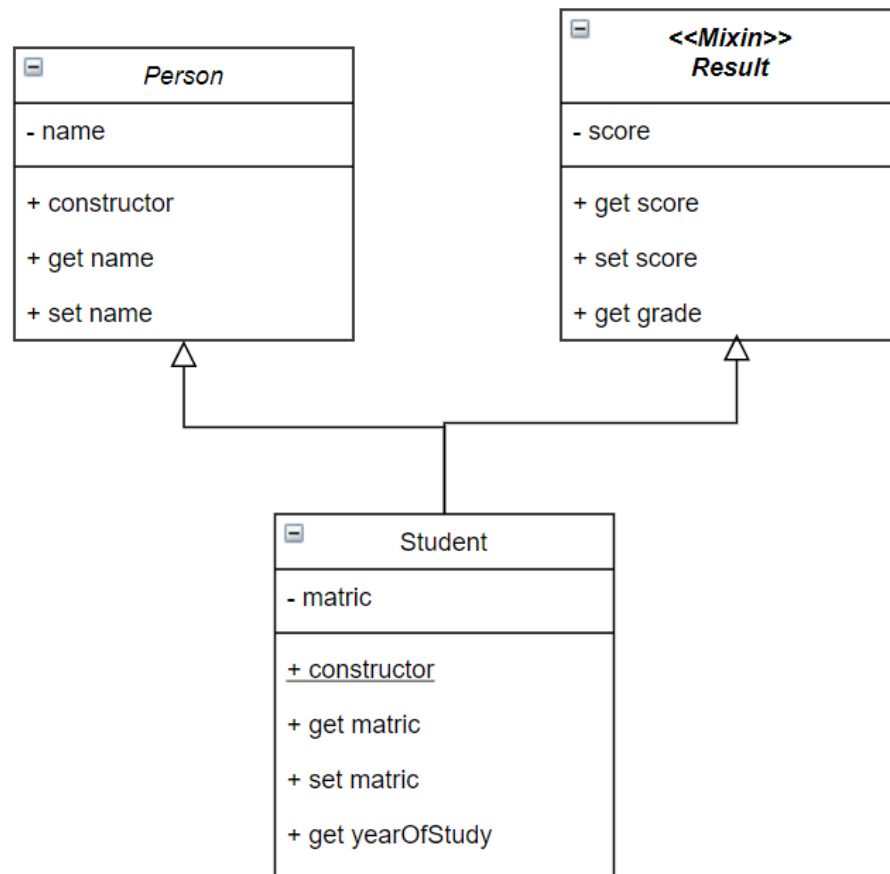


Figure 1: Class diagram

In the **Result** class, the getter method **grade ()** is meant to determine the letter grade of the score using the conversion in Table 1.

Table 1: Score to letter grade

Score range	Grade
85 – 100	A
75 – 84	B
65 – 74	C
50 – 64	D
0 – 49	E

In the **Student** class, the getter method **yearOfStudy ()** is meant to determine the year of study of the student from his/her matric number. Here is an example:

Student's matric number: A17EC4042
Year of study: 2020 – 2017 = 3, i.e., Third Year

Figure 2: Determining year of study from matric number

where, the current year is 2020, the student's intake year is 2017. The intake year is taken from the second and third characters of the matric number.

Later, to test your program out, use the input data given in Table 2. They will be hard-coded in your program.

Table 2: Input data

Student's Name	Matric	Score
Mario Max	A16EC4041	57
Petey Cruiser	A17EC4042	85
Anna Sthesia	A18EC4056	66
Paul Molive	A16EC9043	80
Anna Mull	A19EC4002	57
Gail Forcewind	A16EC4032	75
Paige Turner	A16EC3002	85
Bob Frapples	A17EC3003	82
Robin Banks	A17EC4043	83
Jimmy Changa	A18EC4044	84
Barry Wine	A18EC3004	67
Wilma Mumduya	A16EC4045	70
Buster Hyman	A17EC4040	53

Tasks

Based on the specifications given above, write the code (in Dart programming language) to accomplish the following tasks:

***Note:** Some references are provided in the end of this document to give you tips in accomplishing some tasks.*

1. Based on the class diagram in Figure 1, define all the classes including their methods and relationship between classes. Then test out the classes by creating an object from each class (except the class Result) and printing the objects' attributes onto the screen.

Tips:

- the implementation involves inheritance and mixin
- refer to the lecture slide about OOP
- In the **Student** class, you may need to use constructor initializer. Refer to Resource 1.
- In the **yearOfStudy()** method of class Student, you will need to extract the intake year from the matric number. You can use the **substring()** method of String class. Refer to Resource 2.
- In the **yearOfStudy()** method, you will need to convert a string to an integer. Refer to Resource 3.

2. Store the input data (from Table 2), in a single string using a multi-line string. Separate the fields for each student with commas. For example:

Mario Max,A16EC4041,57

You are storing data in this way because later on you will be exploring on how to manipulate strings in Dart. Test the string out by printing its length.

Tips:

- Refer to Resource 4 about how to use multi-line strings

3. Split the multi-line string into a list of single-line strings. Test the code out by printing the length of the list.

Tips:

- Refer to Resource 5 about how to split strings
- Note that in a multi-line string, each line is separated by a **\n** character (i.e., new line)

4. Using the list from (3), create a list of Student objects. Use an high-order method (instead of traditional loop) to accomplish this task.

Tips:

- The logic for this task:
 - iterate each each string (from the list). You can use the **map ()** higher-order method to iterate the list. Refer to Resource 6.
 - split the string into their fields (i.e, name, matric and score). Refer to Resource 5 on how to split strings
 - pass the extracted student's fields to the Student constructor

5. Using the list of Student objects from (4), Print the list of students with their scores, grades and other info. Use an high-order method (instead of traditional loop) to accomplish this task.

Expected output:

Name	Year	Matric	Score	Grade
----	----	-----	-----	----
Mario Max	Fourth Year	A16EC4041	57	D
Petey Cruiser	Third Year	A17EC4042	85	A
Anna Sthesia	Second Year	A18EC4056	66	C
Paul Molive	Fourth Year	A16EC9043	80	A
Anna Mull	First Year	A19EC4002	57	D
Gail Forcewind	Fourth Year	A16EC4032	75	B
Paige Turner	Fourth Year	A16EC3002	85	A
Bob Frapples	Third Year	A17EC3003	82	A
Robin Banks	Third Year	A17EC4043	83	A
Jimmy Changa	Second Year	A18EC4044	84	A
Barry Wine	Second Year	A18EC3004	67	C
Wilma Mumduya	Fourth Year	A16EC4045	70	B
Buster Hyman	Third Year	A17EC4040	53	D

Tips:

- You can use the **forEach()** method to iterate the list. Refer to Resource 6.

- Using the list of `Student` objects from (4), calculate the average score and the grade. Use an high-order method (instead of traditional loop) to accomplish this task.

Expected output:

```
Average score: 73
Average grade: B
```

Tips:

- You can use the **fold()** method to calculate the sum of scores. Refer to Resource 6.
- Use the **round()** method to round the average score to an integer. Refer to Resource 7.
- To get the average grade, create a `Student` object and pass the average score to the object

Resources

- How do I call a super constructor in Dart?
<https://stackoverflow.com/questions/13272035/how-do-i-call-a-super-constructor-in-dart>
- substring method
<https://api.flutter.dev/flutter/dart-core/String/substring.html>
- How do I parse a string into a number with Dart?
<https://stackoverflow.com/questions/13167496/how-do-i-parse-a-string-into-a-number-with-dart>
- Multi-lines string in Dart
<https://stackoverflow.com/questions/13104729/multi-lines-string-in-darttext>
- split method
<https://api.flutter.dev/flutter/dart-core/String/split.htmltext>
- Top 10 Array utility methods you should know (Dart)
<https://codeburst.io/top-10-array-utility-methods-you-should-know-dart-feb2648ee3a2>
- round method
<https://api.dart.dev/stable/2.7.1/dart-core/num/round.html>