

FIT9131 Programming Foundations

Week 2 Exercises

Introduction

Each week you will be given a set of exercises to work on. These are very important for your learning in this unit. They will reinforce the work covered in the lecture each week and will build your understanding and skills in programming.

Generally, you will work on these exercises in your lab class¹ (via online Zoom sessions, or on-campus labs) each week under the direction of your tutor and will complete any unfinished exercises/homework at home. From week 3 onwards, extra help will also be available from HelpDesk sessions scheduled throughout the week. There is also an online forum on Moodle for you to discuss any issues relating to the lab work.

You will be expected to complete all the exercises for the current week by the start of your lab class for the following week. So, for example, the exercises in this document should be completed by your class in week 3.

Homework will be set each week. Typically, this will be readings from the textbook and exercises not completed in the lab. These are important to help you prepare for the concepts and topics introduced in the following week. You must complete them.

Pre-lab tasks will be set and assessed each week. **This assessment will start in week 3.** The mark awarded (plus your lab participation) will contribute to your final mark for this unit. The deadline for submitting tasks is **48 hours before** the next lab.

For each week, there are **four sections** to this document:

- A. **Homework checklist** – a list of what we expect you to have completed from the *previous* week
- B. **Exercises** – exercises for the current week
- C. **Homework** – exercises/readings to be completed by the following week
- D. **Pre-lab tasks** – assessable tasks for the following week.

If anything is not clear please ask your tutor.

It is really important that you keep up with the work in this unit and seek help if you begin to fall behind.

¹ Reminder: “tutorial”, “workshop” & “lab” all mean the same thing in FIT9131

Week 2

A. Homework checklist

By this week you should have completed the following:

1. Installed Java and BlueJ on your computer.
2. The BlueJ tutorial, at least up to the end of Chapter 7.
3. Read chapter 1 and completed exercises 1.1-1.8 of the textbook *Objects First with Java*.
4. Read sections 2.1-2.14 of chapter 2 of the textbook.

Please see your tutor if you have had any problems with these tasks.

B. Exercises for Week 2

1. Exploring Class Diagrams

When working with object-oriented programming, all programs developed by programmers make use of a **Class Diagram**. A class diagram represents the various classes within a program along with the attributes (fields)² and the behaviours (methods). Below are examples of simplified class diagrams.

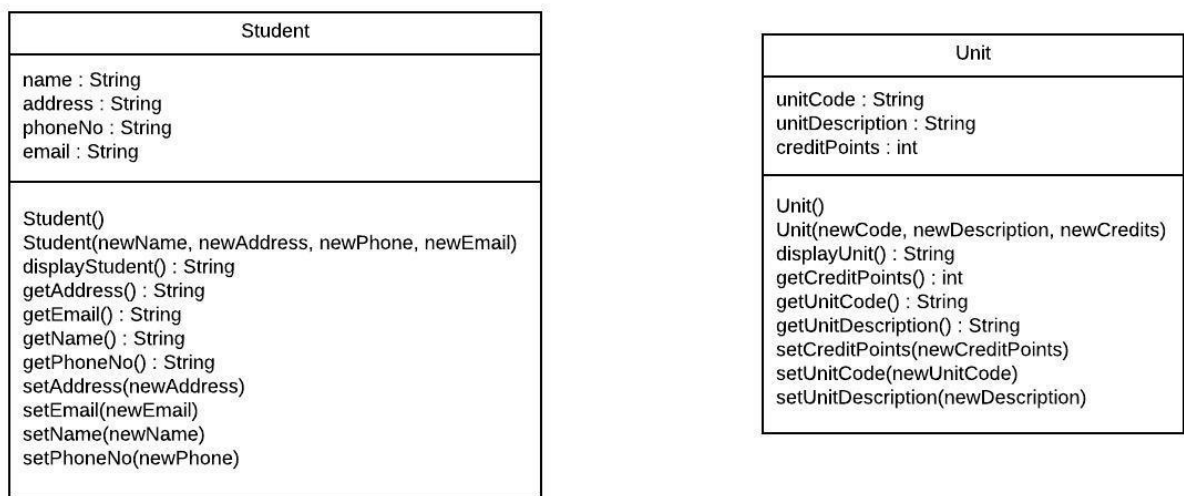


Figure 1: Basic Class Diagrams

A **Class Diagram** is made up of the following **3 components**:

- **Class Name**
- **Fields** (attributes) – each field *must* be of a specific type, this is indicated after the “:”
- **Methods** – each method *may* return data of a specific type, this is indicated after the “:”

² **Field** is the term used in *Objects First with Java* for ‘**attribute**’. Also known as ‘**instance variable**’.

In the class diagrams above, find one example of each of the following:

a) *Class name*

b) *Field*

c) *Constructor*

d) *Method*

e) *Accessor method*

f) *Mutator Method*

Write down, and then discuss your answers with your classmates or you tutor if you are unsure.

2. Projects to be used for work during the semester

Before starting to work on the exercises below, download the entire sample *Projects* folder as a ZIP file from Moodle under Week 1: [Sample Projects for the textbook](#). Then unzip the folder and place it on your computer's desktop.

3. Working with code

Start BlueJ, and open the *naive-ticket-machine* project and study the code for the **TicketMachine** class. This project is located in the **chapter02** subfolder of the *Projects* folder you just unzip above.

In the *naive-ticket-machine* code, find examples of the following. Write down your answers ready to show to your tutor during your lab. If unsure, discuss your answers with your classmates or tutor.

a) *Class*

b) *Field*

c) *Constructor*

d) *Method*

e) *Accessor method*

f) *Mutator Method*

g) *Assignment operator*

h) *Arithmetic operator*

i) *Expression*

j) *Statement*

k) *Comment*

l) *Method header*

m) *Method Body*

n) *Method Signature*

o) *Parameter*

p) *Visibility Modifier*

q) *Return Type*

4. In the same code for the **TicketMachine** class, find examples of the following:
- a) A statement that *initialises* a field.

 - b) A statement that will cause the program to *display* something on the screen.

5. Answer the following questions about the code for the **TicketMachine** class:
- a) Where are the attributes defined?

 - b) Where are the behaviours defined?

 - c) What is the type of the field balance?

6. The following exercises are from the textbook, *Objects First with Java*, 6th Edition. These will test and reinforce your understanding of the concepts covered in the week 2 lecture. As usual, do these using *BlueJ*.
- a) Open the *lab-classes* project from the **chapter01** subfolder in the projects folder. Do exercises 1.21–1.29. These will provide a review of the concepts covered in week 1.
 - complete exercises 1.30, 1.31 (data types)
 - b) Open the *naive-ticket-machine* project from **chapter02** again. The following exercises will mostly refer to this project. Do the following exercises:
 - 2.1–2.5 (exploring the behaviour of a class)
 - 2.6–2.10 (components of a class)
 - 2.12–2.17 (fields)
 - 2.18, 2.19; 2.20, 2.22 (constructors)
 - 1.33–1.34; 2.26, 2.27; 2.32 (methods)
 - 2.37, 2.41 (displaying information)

C. Homework

The following tasks should be completed before your next class/online session.

1. Finish all the lab exercises set for this week (and any uncompleted tasks from week 1).
2. Read sections 2.16–2.23 of chapter 2.

D. Pre-lab tasks to be submitted/assessed in Week 3

The following exercises should be attempted for assessment by your tutor next week. ***Submit the exercise to Moodle – there will be a link in the Week 3 section for you to do this.*** All questions must be attempted and submitted **48 hours** prior to your scheduled class, otherwise zero marks will be awarded.

Your tutor will explain the submission requirements in more detail during the class.

The first four questions require a written response and the fifth exercise requires writing code. ***Write your answers in one document (e.g. in Microsoft Word or PDF format) for submission for assessment.*** Note that these exercises are from Chapter 2 of the textbook and relate to the *naive-ticket-machine* project from **chapter02**.

1. 2.11
2. 2.17
3. 2.28, 2.29
4. 2.31
5. 2.43–2.45

The link for submitting this set of tasks will be available later, under the Week 3 section.