

Introduction To Programming: Independent Investigative Effort 8

Due: See [Canvas→Assignments](#) for due dates, marks and submission link.

Obtaining help method 1: Create a post in the corresponding Canvas→[Discussions→IIE forum](#), post a screenshot of your preliminary/faulty [along with a brief explanation](#) and your tutor will help in a general way. Debugging will still need to be done by the student as it is an important aspect of programming.

Obtaining help method 2: Weekly tutor chats (not to be confused with ‘live lecture’) of this week will give directions (not solutions) on how to approach this IIE. [Check recordings](#) if you cannot attend.

Getting feedback: For specific questions asked, you will receive feedback via the forum within 1-2 working days by your group tutor. If you ask during a tutor chat, you would receive feedback immediately.

Late submissions: Accepted for up to 1 week with an automatic 10% penalty for each day late (unless special consideration received).

Solution: Solution, recommended approach, common mistakes to avoid, etc. for main tasks will be shown during the [weekly live lecture](#) that starts at the time of submission.

Marks: IIEs are not tests. Marks are given based on effort and not on correctness. Marks turnaround time is approx. 10 working days after submissions close.

Access issues: For non-programming technical issues (relating to infrastructure, passwords, etc.) please call the [RMIT IT Service and Support Centre](#) for quick help on 03-9925 8888 and remember to ask for a reference number and pass it on to your instructor.

Extensions: For all new extensions, [apply for special consideration online](#). Contacting your tutors, instructors first will lead to delays.

Please follow/complete all steps below in the given sequence:

1. Check your [official @student.rmit.edu.au email account](#) for announcements and other communication from the university. If getting in touch with your instructors, please only use this account (not Canvas inbox, messages, personal email, phone, Microsoft Teams, etc.)

2. [Watch any unwatched recordings](#) of the **Weekly Live Lecture** and complete all missed tutorials **before going further**. For your convenience, the time stamps of recordings are sent via student email/Canvas→Announcements.

3. Is there something that you have not fully grasped from what has been covered so far? Please have your doubts clarified via one of the relevant forums under [Canvas→Discussions](#). Leaving gaps has shown to be severely detrimental to learning.

4. Did you want to make any additions to the previous IIE? Please do by replying to your original post. i.e. do not edit, change the images of existing posts as it affects submission timing.

5. This week’s programming task will cover concepts required by Assignment 3. You should aim to get the help of your tutors and make further revisions.

Coding exercise steps (Hint: Need help? Ask your tutor via Canvas→Discussions→"IIE08"):

Complete the 18/Jan/2021 solution first as this extends upon that work. Next follow Canvas→[Modules→Week 8](#) where the topics of creating multiple classes is explained. **Your tutors will provide further explanations on these general topics and how they relate to this IIE during their weekly “tutor chats”.** Please also follow the announcement ‘[How to debug large programs and get help on debugging...](#)’

a) In this week’s IIE, modify [your own adaptation](#) of the IIE07 student manager (which has to be different to the student manager) so that it has the same functionality but across several class files as follows.

Start by creating the Application.java (do not modify its contents or even rename the class to anything else) which is the new entry point to your program. When run, it will allow a graphical interface and a console interface to the same program and records that you add from one interface should be visible across all interfaces (via manually invoking a refresh method).

Application.java

```
public class Application {
    private BackEnd backEnd;
    private FrontEndGTerm uiGT;
    private FrontEndConsole uiConsole;

    public Application() {
        this.backEnd = new BackEnd();
        this.uiGT = new FrontEndGTerm(this.backEnd);
        this.uiConsole = new FrontEndConsole(this.backEnd);
    }

    public static void main(String[] args) {
        Application app = new Application();
    }
}
```

```
}
```

BackEnd.java: This file should contain the primary data structures (arrays relating to records), methods to add/remove records, etc. It should not have any mentions of GTerm, System.in/out, JOptionPane, etc. It must not make any assumptions about there being a GTerm interface and/or a console interface. All values returned by its methods should use standard Java API classes (e.g. String or Strings[]) or primitive data types. There is no need to prefix the method names with “backEnd” as done in IIE07.

FrontEndGTerm.java: This file should contain only the methods and object member variables related to the GTerm interface. It must not have any awareness of any other interface. It should use tables, buttons, text fields, etc. to present the information. There is no need to prefix the method names with “gui” as done in IIE07.

FrontEndConsole.java: This file should contain only the methods and object member variables related to the Console interface. It must not have any awareness of any other interface. A user should be able to perform the same operations (e.g. adding, querying, removing, etc.) as they can do with the graphical one above. There is no need to prefix the method names with “console” as done in IIE07.

An important note about Assignment 2: In Assignment 2, there must be only one interface (GTerm). This IIE08 is relevant to Assignment 3 requirements.

Submission Checklist for Step 5:

- a. Ensure steps above have been followed in sequence.
- b. Ensure that there are no red dots (compilation errors) in your code. Code with red dots are not valid Java and cannot be marked.
- c. If you have not made a final submission for your Assignment 2, make a dummy submission for Assignment 2 by submitting **your .java file to Canvas→Assignments→Assignment 2**. Do the same for Assignments 3 as well. Remember, you can overwrite this submission any time when you have a proper submission for your assignment.
- d. Take screenshots of the code and the running program (as you did for IIE01) and embed the **screenshots in a post under Canvas→Discussions→Independent Investigative Exercise 8**. If you are unable to embed screenshots, please follow the announcement [‘Having issues embedding images? Here's the alternative...’](#) The mark for this week’s work will be given based on this submission.
- e. Download your own file(s) from the discussion forum and ensure that it is correct. If it is not, you can edit/delete your post and retry.