Introduction To Programming: Independent Investigative Effort 5

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 2. 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→"IIE05"):
Follow Canvas→Modules→Week 5 first. It covers topics on creating and manipulating arrays.

a. Make a copy of your IIE04 Eclipse project (GTerm version that takes an entire record using one input) and rename the project to IIE05 or similar, as shown during the week 1 'weekly live lecture'. Ensure that your .java file is not PleaseRenameMe.java and it is something relevant to your application (avoid names such as IIE05.java; give it a personality!) but remember to follow class naming conventions (refer to IIE01). You must be able to **show how your work is different to the "student manager" examples** shown in the live lectures, if it can be thought of as being similar. Also note that, whenever the instructions use the term "array", it **does not refer to the array obtained by the String class' .split method**, unless otherwise stated.

Ensure that your program first asks the user how many records they would like to process. Then declare an array reference of an appropriate data type for each field of your records. E.g.

Instead of the variable:

```
        String name;
```

Create an array to store many such values as:

```
        String[] names;
```

If your code took inputs for 5 different pieces of information for each record, you must have 5 separate arrays. In other words, you will have an array reference for each column in the table but the arrays will be of different data types (whereas data in a table exist as Strings).

Now create the actual arrays based on the num of records specified by the user for each of the arrays. For now, keep these two steps separate.

```
names=new String[numRecords];
```

Are you stuck? Please ask your friendly tutor by creating a post in the relevant IIE forum.

b. When input String is split and the values are extracted/parsed and before they are added to the table, assign them to the arrays. i.e. if you have a names array and an ages array, names[0] and ages[0] must refer to the same student, names[1] and ages[1] should refer to the next student, etc. To achieve this, you will need to use your outer loop's loop variable to refer to the same index across all arrays. Of course, your code must not be about students. Are you stuck? Please ask your friendly tutor by creating a post in the relevant IIE forum.

c. After 5b, your values would be stored in the arrays. Modify your addRowToTable to add these same values to the table as you did in IIE04. Are you stuck? Please ask your friendly tutor by creating a post in the relevant IIE forum.

d. Investigative exercise: After taking all inputs and storing them in the arrays, perform some statistical calculations on one or more of the numerical fields/columns. Are you stuck? Please ask your friendly tutor by creating a post in the relevant IIE forum.

e. Optional to do but must follow when solution shown: How can we make the program not ask for numRecords at the start and make it still work? Are you stuck? Please ask your friendly tutor by creating a post in the relevant IIE forum.

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 5**. 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.