Introduction To Programming
Tutorial 7

(See Canvas→Assignments for due dates and marks)

Note: Do not include your name, student ID or any personally identifiable info in your submission as the submission may be used for peer reviews; your submission will not be lost as Canvas keeps track of these internally.

Please follow all of the steps below in the given sequence:

1. Read all unread announcements and unread replies to announcements under Canvas→Announcements.

2.1 Do any missed tutorials before going further.

2.2 Watch any unwatched recordings of the compulsory **Weekly Live Lecture** and any important videos in the Extra Videos Playlist.

2.3 If you need help in addition to what has been shown in the compulsory weekly live lecture, you are also expected to speak to your **group tutor via discussion forums** and attend/watch their live sessions. Please note that group tutors cannot debug your assessment code on your behalf as debugging is a part of every programming assessment.

2.4 **If you still have have any unresolved questions** or **if you need further feedback**, post the relevant parts of your submitted work in a new post under Canvas→Discussions→Tutorial discussions and ask from your group tutor. E.g. you can ask "*In the live lesson Gayan did ___ with ___. I didn't do ___ so should I be doing this as well?*", etc. Please note that the university requires teaching to be conducted in an equitable manner so your tutors will require you to post questions in the discussion forums.

3. Check any available feedback of your previous submissions and if you have any unresolved questions or if you need further feedback, post the relevant parts of your submitted work in a new post under Canvas→Discussions→Tutorial discussions and ask from your tutor. E.g. you can ask "*Gayan showed _____ but I did mine like ______, so which is the better approach and why?*", etc. Please note that the university requires teaching to be conducted in an equitable manner so please only use email for matters such as special consideration.

4. Follow the materials under Canvas→Modules→Week 7…

5.1 Complete this exercise with the help of your group tutor. However, your tutor will not be able to write the code on your behalf. Instead, they will give general guidance.

Get the `MusicLibraryW7.java` file from Canvas→Discussions→Tutorial Discussions→Class ? (where ? is your group) and get it in to your Eclipse (If you cannot find the file, ask your group tutor). Familiarise yourself with the program first (such as by running and tracing through the code) and then modify it as follows:

1. Create a constructor and move all of the code from the main method to the constructor and get the code to work as before. When written correctly, the main method should only contain the following two lines:

```
        MusicLibrary ml;
        ml=new MusicLibrary();
```

2. Move the declarations of maxNumSongs, songTitles, songLocations and currentNumSongs from the constructor to the class block and make them private. These will be known as object member variables. E.g.

```
public class MusicLibrary {
        private int maxNumSongs;
        private String[] songTitles;
        private String[] songLocations;
        private int currentNumSongs;
```

Note: You must not have any = (equal) signs in front of the above declarations; All object member initialisations must be performed in the constructor. Also, the word 'static' must only be used in the main method's definition.

3. In your constructor, add `this.` (this dot) to the start of any mention of an object member variable name.

E.g.

```
                maxNumSongs = 3;
```

Should be changed to:

```
                this.maxNumSongs = 3;
```

… and so forth for all object member variables (there are many such places).

4. Create a method each for the three menu items of the application.

5. Move at least one segment of repeated code to its own method. This method should take one or more parameters and return an appropriate value.

6. Make the necessary changes to make the constructor accept a parameter specifying the maximum number of songs.

7. Make the necessary changes so that, when adding a song, if a song by the same title already exists, the user is able to specify a new location for the song. When written correctly, this should not result in duplicated entries for the same song title.

8. Make the necessary changes so that, when adding a song and if out of room, the arrays are expanded. You must not use ArrayList or similar data structures to achieve this functionality. Ask your group tutors if you are not sure.

5.2. Add comments in the style required by Assignment 2. See rubric in section 9 of the Assignment 2 PDF.

6. If you have not submitted your final version of A2, add comments explaining your plan. Note that this will not be marked but it is to help you progress.

Submission Checklist:
1. Ensure that your code does not have any red dots (Java errors) as code with such errors cannot be tested/marked and will receive 0 for that submission. If your code has red-dots, refer back to similar code and fix the error or remove the code that is causing the problem. You must not leave any commenteded out code in your submissions. Yellow dots are warnings and these are different.
2. Ensure that you have added comments to your .java file explaining what you have done and any potential alternative approaches.
3. Format your code (e.g. Eclipse→Source→Format).
4. Go to Canvas→Assignments→**Independent Investigative Effort 7** and select ‘submit assignment’.
5. Select to attach files from your computer, navigate to your Eclipse workspace folder→Project folder→src folder and select the (one) final version of your **`MusicLibraryW7.java`** file. Please **do not submit more than 1 file** as it delays the marking process. You can also context select on the .java file name from package/project explorer inside Eclipse and find its exact location. Only the last submission is the official submission.
5. Verify your submitted files as shown during the week 1 session.

Having trouble with usernames, passwords, access, 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.

Need extensions or special consideration? Please follow details and process below:
https://www.rmit.edu.au/students/student-essentials/assessment-and-exams/assessment/special-consideration