SCHOOL OF COMPUTING

DIPLOMA IN INFORMATION TECHNOLOGY

ST0509 JAVA PROGRAMMING

2022/2023 SEMESTER 2
PRACTICAL ASSIGNMENT

Objective of Assignment

To allow students to practice what they have learnt in the module by developing a Java program to simulate a student management system.

Instructions and Guidelines:

1. The assignment should be done individually and will account for **30%** of your final grade.

2. The assignment should be submitted by **Monday, 2 Jan, 2023 11:59 pm**.

3. The development platform will be in Java using NetBeans IDE.

4. The interview will be conducted during the practical lessons on week 12. You are expected to explain the program logic and modify the program during the interview. If you are absent from the interview, **you will be awarded zero mark for the assignment.**

5. **No marks will be awarded**, if the work is copied or you have allowed others to copy your work.

6. **50%** of the marks will be deducted for assignments that are received within **ONE (1) calendar** day after the submission deadline. No marks will be given thereafter.

Warning: Plagiarism means passing off as one's own the ideas, works, writings, etc., which belong to another person. In accordance with this definition, you are committing plagiarism if you copy the work of another person and turning it in as your own, even if you would have the permission of that person.

Plagiarism is a serious offence, and if you are found to have **committed, aided, and/or abetted** the offence of plagiarism, disciplinary action will be taken against you. If you are guilty of plagiarism, you may **fail all modules** in the semester, or even be **liable for expulsion**.

Overview of the system

You are tasked to develop a Java program to simulate a **Comic Rental System**.

The example below shows an example of the System Menu that allows users to display, search and print statistic.

Fig 1.1

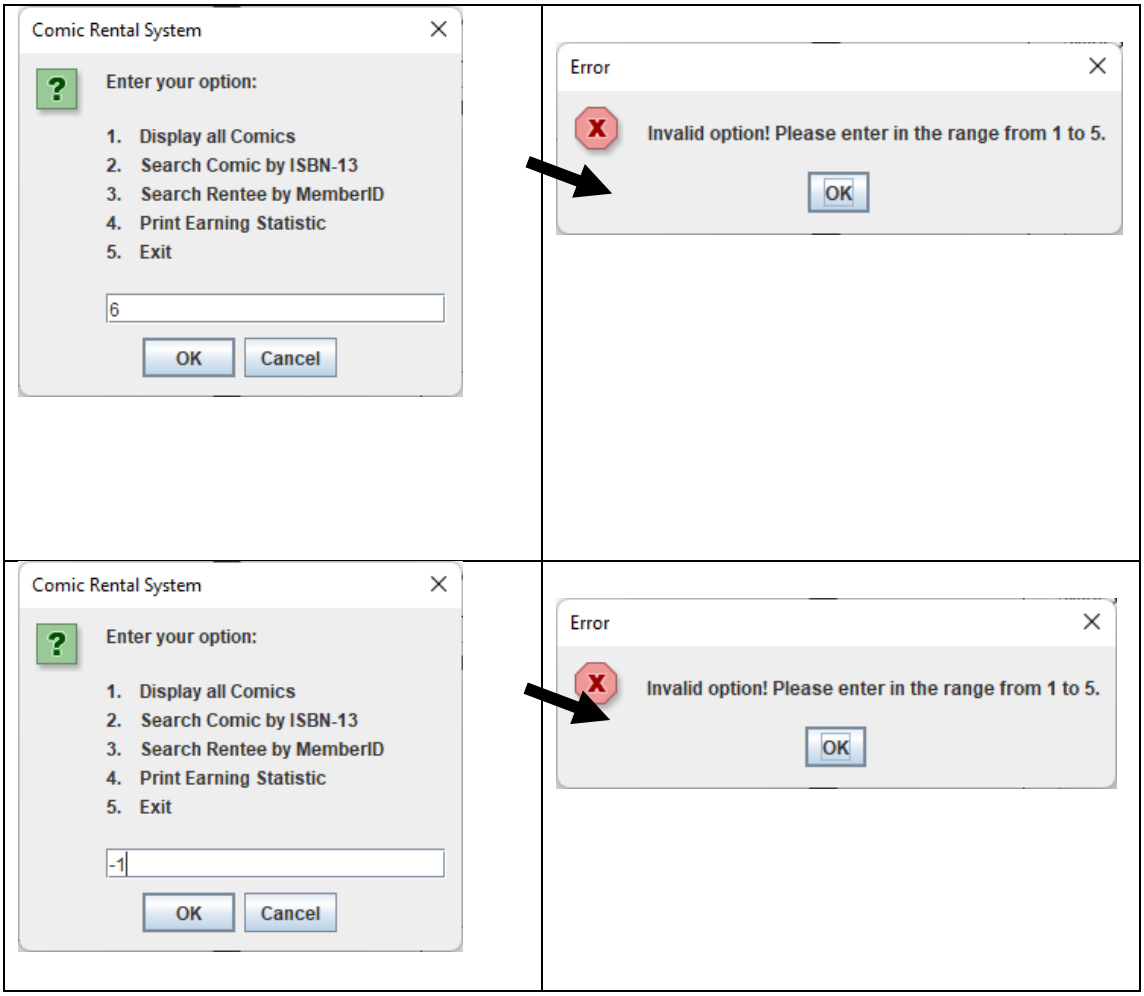


Figure1.1: The program displays the menu and allows the user to enter his option. You should validate the user input. If the input is invalid, it displays an error message and prompts the user to re-enter.

Fig 1.2

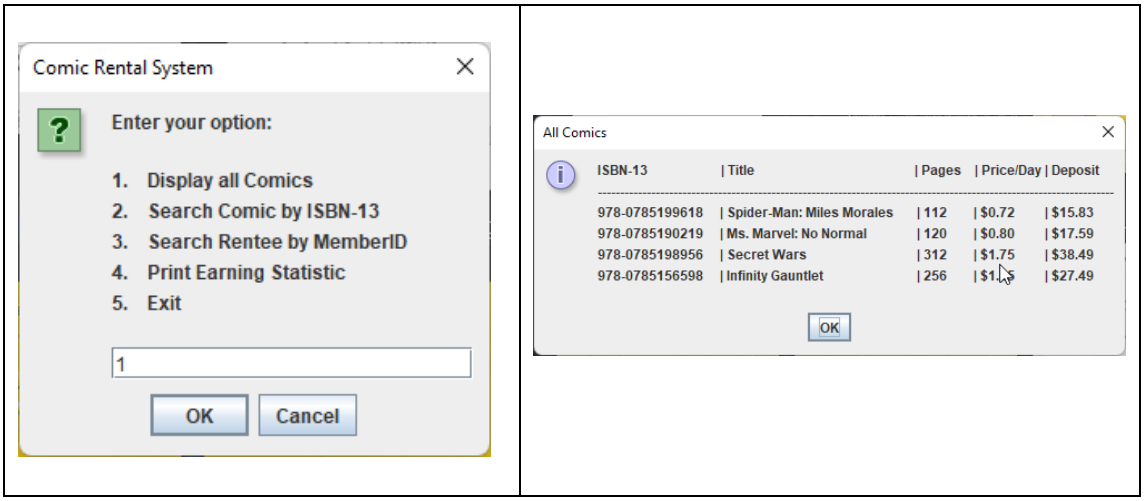
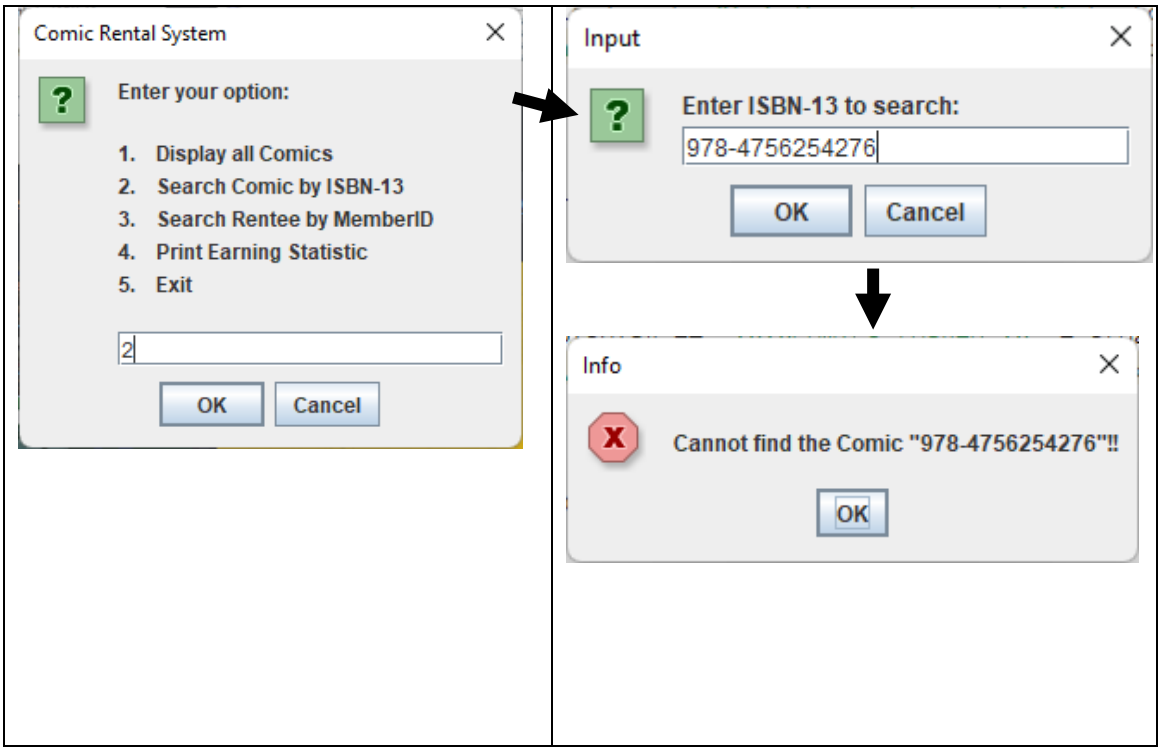


Figure1.2: If the user selects option 1, the program displays all comics details. A comic details should include the ISBN-13 number, Title, number of Pages, Price to Rent Per Day (Price/Day) and the Deposit Fee (Deposit). The Price/Day and Deposit Fee are calculated from the Cost Price of the comic.

Fig 1.3



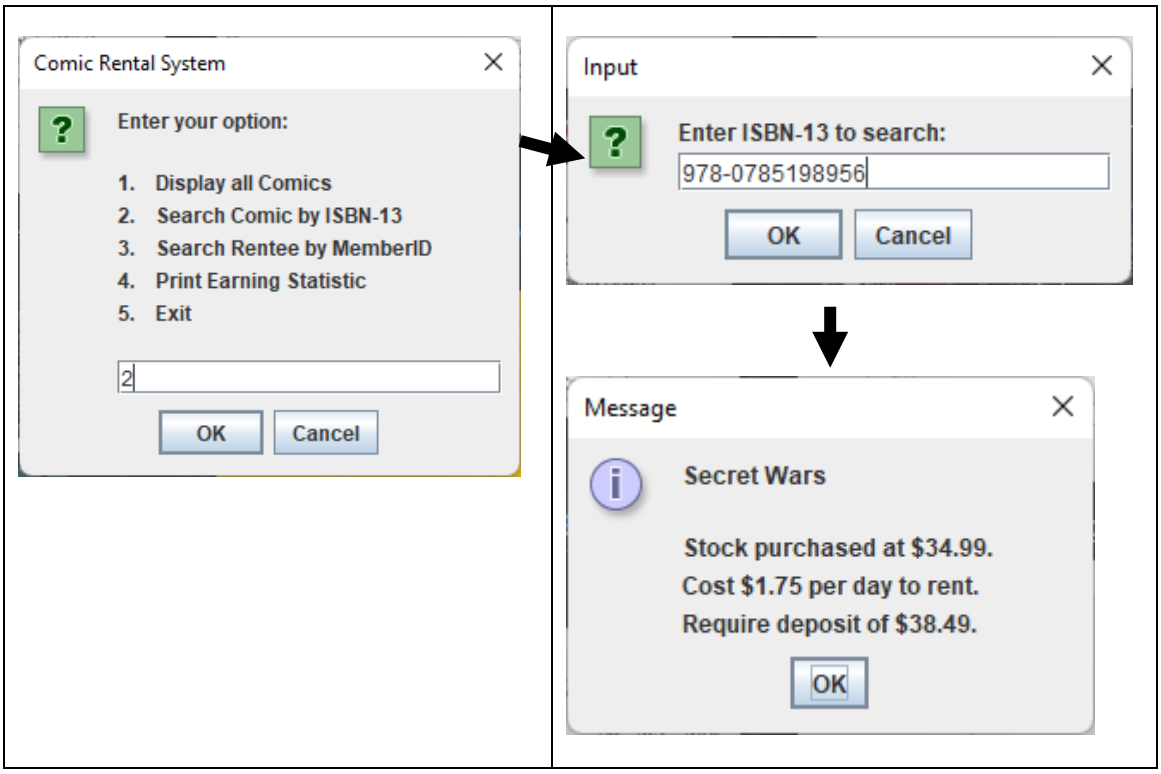
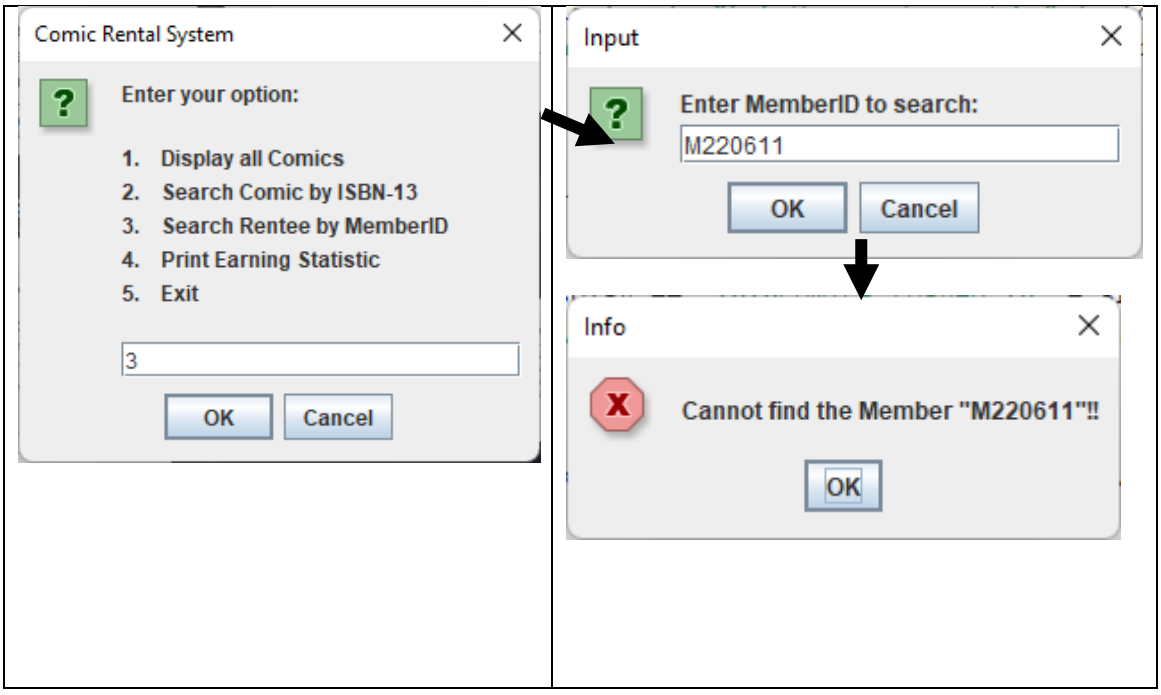


Figure 1.3: If the user chooses to search a Comic, the program prompts the user to enter the ISBN-13 of comic to search and displays the result accordingly. It should display the title, cost price of the stock, daily rental fee and deposit fee required.

Fig 1.4



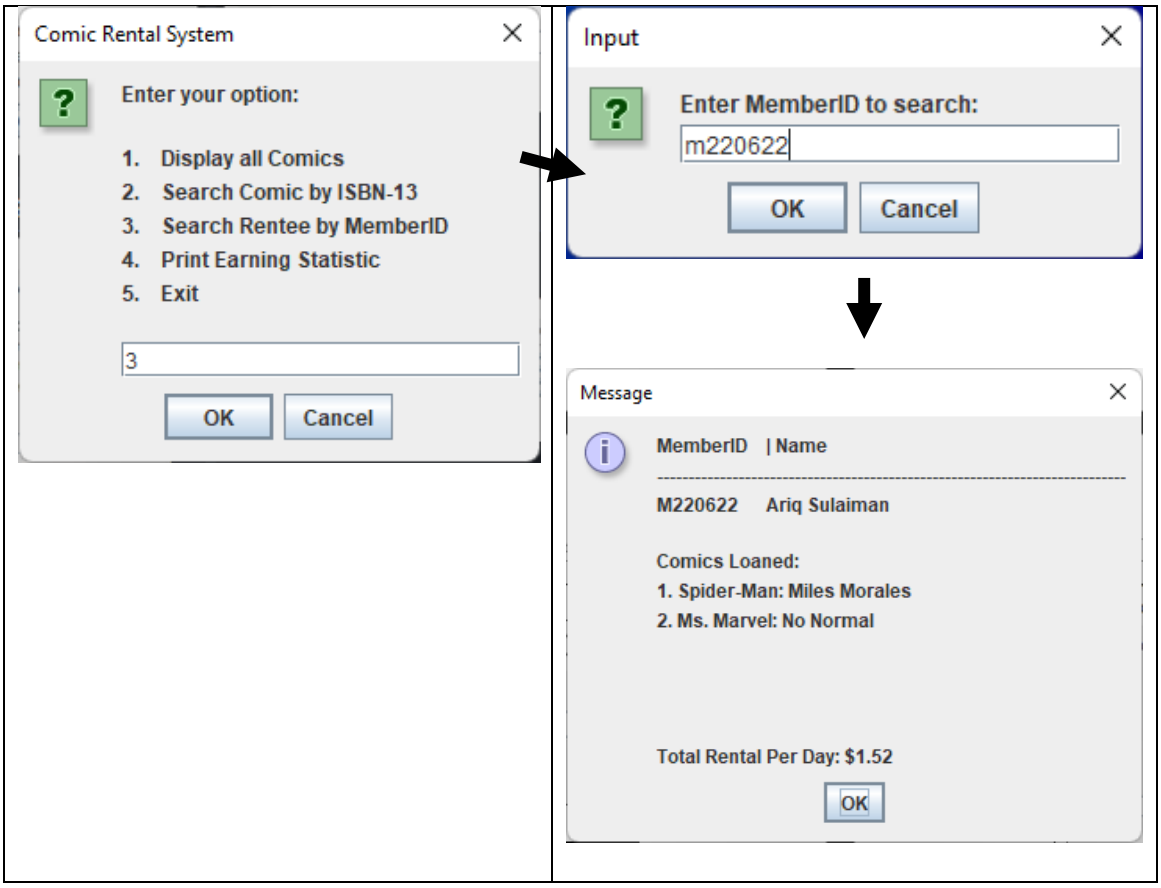


Figure 1.4: If the user chooses to search a Rentee, the program prompts the user to enter the MemberID to search, and displays the result accordingly. If the Rentee rented comics, it displays list of comics loaned by the Rentee and the Rental Fee Per Day.

Fig 1.5

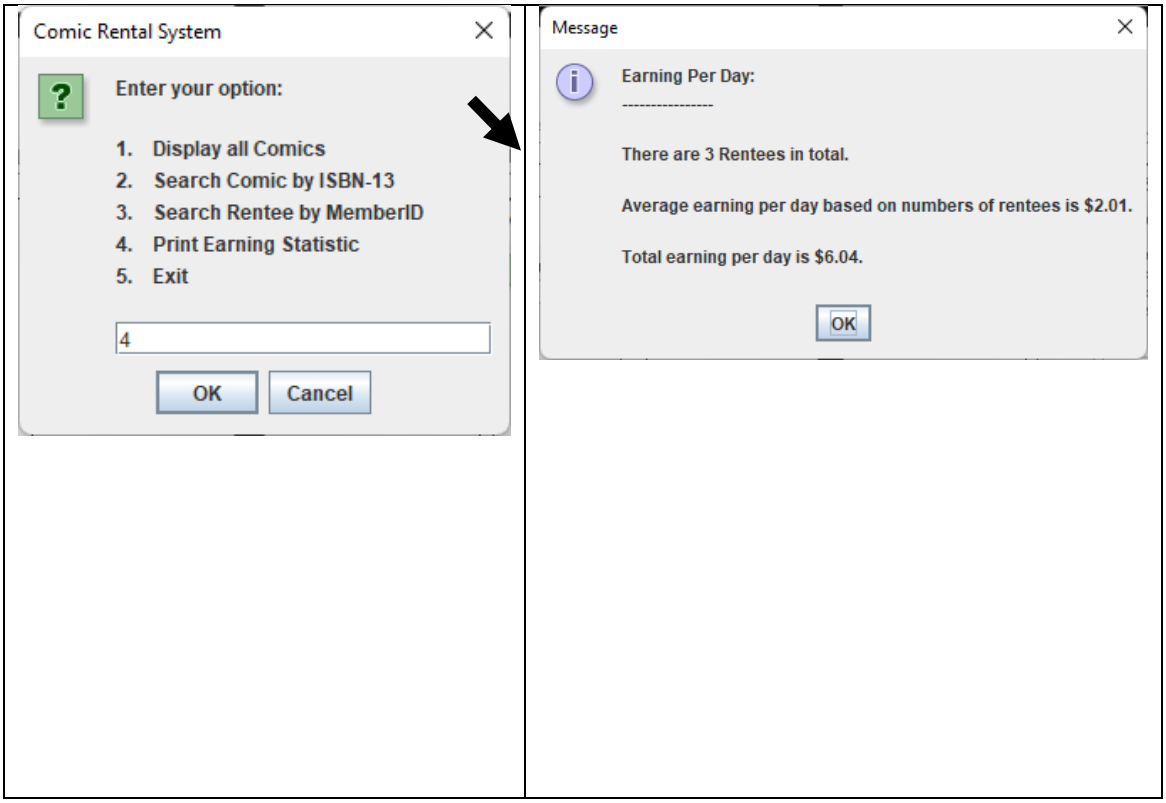


Figure 1.5: If the user wants to print statistic, the program should report the number of rentees in the system, estimating the earning per day based on average per rentee and total earning per day.

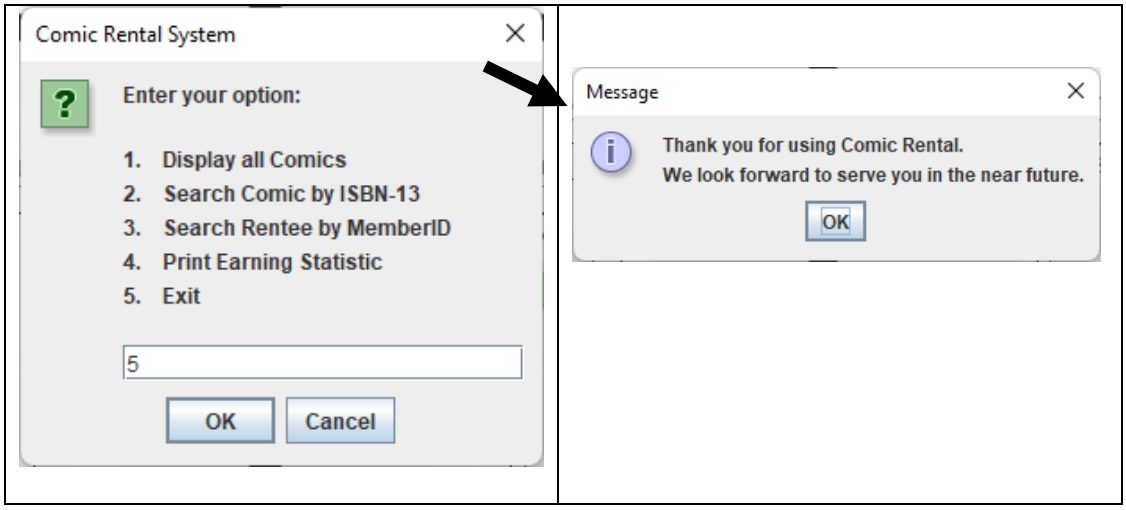


Figure 1.6: If the user chooses to exit the program, it displays a thank you message and terminates the program.

Basic Requirements

Write the Java programs using appropriate methods, objects and arrays to perform the tasks stated in **Overview of the system** section (shown in Fig1.1 to Fig 1.6).

1. Write a Java class ***Comic*** (or similar class) to represent a module object, you should have instance variables to store the ISBN-13 number, Title name, number of Pages and the Cost Price of the Comic (The amount of money the rental store spent to acquire the stock).

    - Add a constructor, appropriate **get** or **set** methods, or any other methods to the class if necessary.

2. Write a Java class ***Rentee*** (or similar class) to represent a rentee object, you should have instance variables to store the Member ID number, Name, and a list of **Comics** loaned by the Rentee. Kindly take note that a Rentee will rent more than one Comic.

    - Add a constructor, appropriate **get** or **set** methods, or any other methods to the class if necessary.

3. Write a Java class ***RentalSystem*** (or similar class) to manage the comic archives and members. The class should have methods that will be invoked by the ***main*** method such as to create Comics, display Comics, search Comics, search Rentee, print Earning Statistic, …., etc.

    In this class, please create at least
    - **4 Comic objects stored in an array and**
    - **3 Rentee objects stored in an array.**

4. Write a Java class ***RentalMenu*** (or similar class) that contains the ***main()*** method for the entire application. The ***main()*** method is where your program will start to run. It displays the menu and calls the methods in ***RentalSystem*** class accordingly.

5. Implement the advanced features such as, but not limited to the followings:
   - Validate for non-numeric input
   - Display all information in a tabulate format. (eg. Money in 2 decimal places)
   - Sound effects.
   - Add in a new user type for the system: ***Administrator***, so that the administrator can add/delete/modify comic and rentee details.
   - Generated more earning statistic for the report and allow user to print
   - Keep track of the time when a user uses the system.
   - Any other features that enhance the system.

Note: Do NOT implement a full Graphic User Interface (GUI) program for CA1 as this will be the requirement for CA2. For the purpose of CA1, you should be using the dialogs available from the ***JOptionPane*** class.

6. Zip the complete NetBeans project of your CA1 and submit to Brightspace. **Provide your Class, Admission Number and Name on each of the Java classes.**

7. You are **REQUIRED** to use the **Github Classroom repository** provided to commit all your progress and changes. It will be reviewed for marking.

Please keep in mind that advanced features are just bonus features. The main bulk of marks are allocated to the completion of a workable program that meets the minimum requirements. **You should try to fulfill the minimum requirements before you attempt to include any advance features.**

Assessment Guidelines

The assignment will be assessed based on the following criteria:

- Ability to demonstrate the minimum requirements of the system

- Program design such as:
  - Correct and efficient usage of classes and programming constructs
  - Appropriate method decomposition
  - Appropriate validations
  - Code efficiency

- Program readability:
  - Meaningful identifiers
  - Meaningful comments and indentation in source code

- Innovation and creativity, and/or any advanced features

- Independent work and the understanding of the concepts & methodology

- Question & Answer during interview

Note:

The **Rental Fee Per Day** for each Comic is computed using the formula below:

$$Rental\ Fee\ Per\ Day = \frac{Cost\ Price\ of\ Comic}{20\ working\ days}$$

The **Deposit Fee** for each Comic is computed using the formula below:

$$Rental\ Fee = \ Cost\ Price\ of\ Comic + 10\%$$

-- End --