Experiment - 1.2

Student Name: Sanchit.	**UID: 21BCS1569**
Branch: BE-CSE	**Section/Group:21BCS_CC_606-B**
Semester: 6	**Date of Performance:20-01-2024**
Subject Name: Java Lab	**Subject Code:21CSH-319**

1. **Aim** Design and implement a simple inventory control system for a small video rental store.

2. **Objective:** The goal of this project is to design and implement a simple inventory control system for a small video rental store. Define least two classes: a class Video to model a video and a class VideoStore to model the actual store.

3. **Input/Apparatus used:**
   - VS Code
   - Mac M1 with 8GB RAM and 256GB storage.

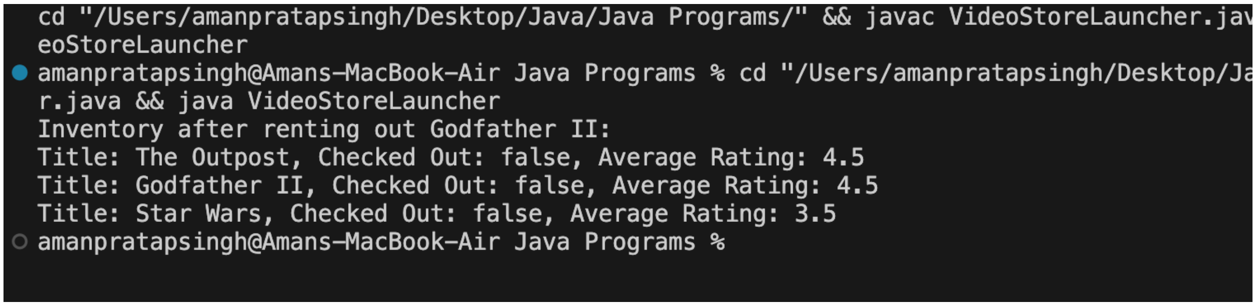
4. **Procedure/Algorithm:**
   - Create a Video class with private fields for title, checkedOut status, averageRating, and totalRatings.
   - Implement methods in the Video class for checking out, returning, and receiving ratings for a video.
   - Create a VideoStore class with a private array of Video objects as inventory.
   - Implement methods in the VideoStore class for adding videos to the inventory, checking out, returning, receiving ratings, and listing the inventory.
   - Define a functional interface VideoAction with a method perform that takes a Video as a parameter.
   - Implement the findAndPerformAction method in the VideoStore class, which takes a title and a VideoAction and performs the action on the corresponding video in the inventory.

- In the VideoStore class, create a method listInventory that prints the details of videos in the inventory.
- Create a VideoStoreLauncher class with a main method.
- In the main method, create an instance of VideoStore.
- Add videos to the store, receive ratings, check out and return videos, and list the inventory in the main method of VideoStoreLauncher.

5. Code:

```
class Video {
    private String title;
    private boolean checkedOut;
    private double averageRating;
    private int totalRatings;
    public Video(String title) {
        this.title = title;
    }
    public void checkOut() {
        checkedOut = true;
    }
    public void returnVideo() {
        checkedOut = false;
    }
    public void receiveRating(int rating) {
        averageRating = (averageRating * totalRatings + rating) / (++totalRatings);
    }
    public String getTitle() {
        return title;
    }
    public String toString() {
        return "Title: " + title + ", Checked Out: " + checkedOut + ", Average Rating: " + averageRating;
    }
}
class VideoStore {
    private Video[] inventory = new Video[10];
    public void addVideo(String title) {
        for (int i = 0; i < inventory.length; i++) {
            if (inventory[i] == null) {
                inventory[i] = new Video(title);
                break;
            }
        }
    }
    public void checkOut(String title) {
        findAndPerformAction(title, Video::checkOut);
    }
    public void returnVideo(String title) {
        findAndPerformAction(title, Video::returnVideo);
    }
    public void receiveRating(String title, int rating) {
        findAndPerformAction(title, video -> video.receiveRating(rating));
    }
    private void findAndPerformAction(String title, VideoAction action) {
        for (Video video : inventory) {
            if (video != null && video.getTitle().equals(title)) {
                action.perform(video);
                break;
            }
        }
    }
    public void listInventory() {
        for (Video video : inventory) {
            if (video != null) {
                System.out.println(video);
            }
        }
    }
}
interface VideoAction {
    void perform(Video video);
}
public class VideoStoreLauncher {
    public static void main(String[] args) {
        VideoStore videoStore = new VideoStore();
        videoStore.addVideo("The Outpost");
        videoStore.addVideo("Godfather II");
        videoStore.addVideo("Star Wars ");
        videoStore.receiveRating("The Outpost ", 4);
        videoStore.receiveRating("The Outpost ", 5);
        videoStore.receiveRating("Godfather II", 5);
        videoStore.receiveRating("Godfather II", 4);
        videoStore.receiveRating("Star Wars ", 3);
        videoStore.receiveRating("Star Wars ", 4);
        videoStore.checkOut("The Outpost ");
        videoStore.returnVideo("The Outpost ");
        videoStore.checkOut("Godfather II");
        videoStore.returnVideo("Godfather II");
        videoStore.checkOut("Star Wars ");
        videoStore.returnVideo("Star Wars ");
        System.out.println("Inventory after renting out Godfather II:");
        videoStore.listInventory();
    }
}
```

6. Output:



```
  cd "/Users/amanpratapsingh/Desktop/Java/Java Programs/" && javac VideoStoreLauncher.jav
  eoStoreLauncher
● amanpratapsingh@Amans-MacBook-Air Java Programs % cd "/Users/amanpratapsingh/Desktop/Ja
  r.java && java VideoStoreLauncher
  Inventory after renting out Godfather II:
  Title: The Outpost, Checked Out: false, Average Rating: 4.5
  Title: Godfather II, Checked Out: false, Average Rating: 4.5
  Title: Star Wars, Checked Out: false, Average Rating: 3.5
○ amanpratapsingh@Amans-MacBook-Air Java Programs %
```