Experiment 2

Student Name: Arjun **UID: 22BCS10214**
Branch: CSE **Section/Group: 631/A**
Semester: 06 **Date of Performance:24/01/25**
Subject Name: Project Based Learning in Java with Lab **Subject Code: 22CSH-359**

1. **Aim: Design a inventory system for video rental store.**

2. **Objective:** To develop a system for managing a video rental store, allowing users to add videos, rent and return them, rate videos, and view inventory details efficiently.

3. **Code:**

```
import java.util.ArrayList;
import java.util.Scanner;

// Class 1: Video
class Video {
    private String title;
    private boolean checkedOut;
    private ArrayList<Integer> ratings;

    public Video(String title) {
        this.title = title;
        this.checkedOut = false;
        this.ratings = new ArrayList<>();
    }

    public String getTitle() {
        return title;
    }

    public boolean isCheckedOut() {
        return checkedOut;
```

```
}

public void checkOut() {
    if (!checkedOut) {
        checkedOut = true;
        System.out.println(title + " has been checked out.");
    } else {
        System.out.println(title + " is already checked out.");
    }
}

public void returnVideo() {
    if (checkedOut) {
        checkedOut = false;
        System.out.println(title + " has been returned.");
    } else {
        System.out.println(title + " was not checked out.");
    }
}

public void receiveRating(int rating) {
    if (rating >= 1 && rating <= 5) {
        ratings.add(rating);
        System.out.println("Rating " + rating + " added for " + title);
    } else {
        System.out.println("Invalid rating! Please provide a rating between 1 and 5.");
    }
}

public double getAverageRating() {
    if (ratings.isEmpty()) {
        return 0;
    }
    int sum = 0;
    for (int rating : ratings) {
        sum += rating;
    }
    return (double) sum / ratings.size();
}

@Override
public String toString() {
```

```
        return "Title: " + title + ", Checked Out: " + checkedOut + ", Average Rating: " +
String.format("%.2f", getAverageRating());
    }
}

// Class 2: VideoStore
class VideoStore {
    private ArrayList<Video> inventory;

    public VideoStore() {
        inventory = new ArrayList<>();
    }

    public void addVideo(String title) {
        inventory.add(new Video(title));
        System.out.println("Video " + title + " added to the store.");
    }

    public void checkOutVideo(String title) {
        for (Video video : inventory) {
            if (video.getTitle().equalsIgnoreCase(title)) {
                video.checkOut();
                return;
            }
        }
        System.out.println("Video not found in inventory.");
    }

    public void returnVideo(String title) {
        for (Video video : inventory) {
            if (video.getTitle().equalsIgnoreCase(title)) {
                video.returnVideo();
                return;
            }
        }
        System.out.println("Video not found in inventory.");
    }
```

```
    public void receiveRating(String title, int rating) {
        for (Video video : inventory) {
            if (video.getTitle().equalsIgnoreCase(title)) {
                video.receiveRating(rating);
                return;
            }
        }
        System.out.println("Video not found in inventory.");
    }

    public void listInventory() {
        System.out.println("Video Inventory:");
        for (Video video : inventory) {
            System.out.println(video);
        }
    }
}

// Class 3: Main
public class Main {
    public static void main(String[] args) {
        VideoStore store = new VideoStore();
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("\nVideo Rental System Menu:");
            System.out.println("1. Add Video");
            System.out.println("2. Check Out Video");
            System.out.println("3. Return Video");
            System.out.println("4. Receive Rating");
            System.out.println("5. List Inventory");
            System.out.println("6. Exit");

            System.out.print("Enter your choice: ");
            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline
```

```
                switch (choice) {
                    case 1:
                        System.out.print("Enter video title to add: ");
                        String titleToAdd = scanner.nextLine();
                        store.addVideo(titleToAdd);
                        break;
                    case 2:
                        System.out.print("Enter video title to check out: ");
                        String titleToCheckOut = scanner.nextLine();
                        store.checkOutVideo(titleToCheckOut);
                        break;
                    case 3:
                        System.out.print("Enter video title to return: ");
                        String titleToReturn = scanner.nextLine();
                        store.returnVideo(titleToReturn);
                        break;
                    case 4:
                        System.out.print("Enter video title to rate: ");
                        String titleToRate = scanner.nextLine();
                        System.out.print("Enter rating (1-5): ");
                        int rating = scanner.nextInt();
                        store.receiveRating(titleToRate, rating);
                        break;
                    case 5:
                        store.listInventory();
                        break;
                    case 6:
                        System.out.println("Exiting system. Goodbye!");
                        scanner.close();
                        return;
                    default:
                        System.out.println("Invalid choice. Please try again.");
                }
            }
        }
    }
```

4. Output

```
Title: Venom, Checked Out: false, Average Rating: 0.00
Title: Intesteller, Checked Out: false, Average Rating: 0.00
Title: inception, Checked Out: false, Average Rating: 0.00
Title: catch me if you can, Checked Out: false, Average Rating: 0.00
Title: wolf of warstreet, Checked Out: false, Average Rating: 0.00
Title: the dark world, Checked Out: false, Average Rating: 0.00
Title: batman, Checked Out: false, Average Rating: 0.00
Title: superman, Checked Out: false, Average Rating: 0.00

Video Rental System Menu:
1. Add Video
2. Check Out Video
3. Return Video
4. Receive Rating
5. List Inventory
6. Exit
Enter your choice:
```

5. Learning Outcome

- Understand Object-Oriented Programming (OOP) concepts.
- Learn to use Java collections like ArrayList.
- Develop skills to design modular and reusable code.
- Gain experience in handling user input and validation.
- Improve debugging and problem-solving abilities.