3. Implementation/Code:

1. **Video Class:-**

```
class Video {
private String title;
private boolean checkedOut;
private double averageRating;
    private int ratingCount;

    public Video(String title) {
this.title = title;
this.checkedOut = false;
this.averageRating = 0.0;
        this.ratingCount = 0;
    }

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

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

```
    public void receiveRating(int rating) {
if (rating < 1 || rating > 5) {
            System.out.println("Invalid rating. Please rate between 1 and 5.");
return;
        }
        averageRating = (averageRating * ratingCount + rating) /
(++ratingCount);
        System.out.println("Received rating of " + rating + " for video \"" + title +
"\".");
    }

     public String getTitle() {
        return title;
    }

    public boolean isCheckedOut() {
        return checkedOut;
    }

    public double getAverageRating() {
        return averageRating;
    }
}
```

2. VideoStore Class:- class VideoStore { private Video[] videos;
    private int count;

    public VideoStore(int capacity) {
videos = new Video[capacity];

```
        count = 0;
    }

    public void addVideo(String title) {
if (count < videos.length) {
            videos[count++] = new Video(title);
System.out.println("Added video: " + title);
        } else {
            System.out.println("Inventory is full. Cannot add more videos.");
        }
    }

    public void checkOut(String title) {
Video video = findVideo(title);
        if (video != null) {
video.checkOut();
        } else {
            System.out.println("Video \"" + title + "\" not found.");
        }
    }

    public void returnVideo(String title) {
Video video = findVideo(title);
        if (video != null) {
video.returnVideo();
        } else {
            System.out.println("Video \"" + title + "\" not found.");
        }
    }

    public void receiveRating(String title, int rating) {
        Video video = findVideo(title);
if (video != null) {
            video.receiveRating(rating);
        } else {
```

```
            System.out.println("Video \"" + title + "\" not found.");
        }
    }

    public void listInventory() {
System.out.println("\nInventory:");        for
(int i = 0; i < count; i++) {
            Video video = videos[i];
            System.out.println("Title: " + video.getTitle() + ", Checked Out: " +
video.isCheckedOut() +
                    ", Average Rating: " + video.getAverageRating());
        }
    }

    private Video findVideo(String title) {
for (int i = 0; i < count; i++) {
            if (videos[i].getTitle().equalsIgnoreCase(title)) {
return videos[i];
            }
}
        return null;
    }
}
```

3. **VideoStoreLauncher Class:-** public class VideoStoreLauncher { public static void main(String[] args) { VideoStore store = new VideoStore(10);

```
        store.addVideo("The Matrix");
store.addVideo("Godfather II");
        store.addVideo("Star Wars Episode IV: A New Hope");

        store.receiveRating("The Matrix", 5);
store.receiveRating("Godfather II", 4);
        store.receiveRating("Star Wars Episode IV: A New Hope", 5);

        store.checkOut("Godfather II");
        store.returnVideo("Godfather II");

        store.listInventory();
    }
}
```