DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

Experiment 1.2

Student Name: Yuvraj Singh **UID: 21BCS10849**
Branch: CSE **Section/Group: FL-604_A**
Semester: 5th **Date of Performance:20/01/24**
Subject Name: Java with Lab **Subject Code: 21CSH-319**

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

2. **Objective:**
   - Develop a user-friendly and intuitive inventory control system for a small video rental store that efficiently manages and tracks the store's video rental collection.
   - Implement a secure and reliable database to store essential information about each video, including title, genre, release date, and availability status, ensuring accurate and up-to-date inventory records.

3. **Input/Software Used:**
   - Hardware Requirements: - Minimum 384MB RAM, 100 GB hard Disk, processor with 2.1 MHz
   - Software Requirements: - Eclipse, NetBeans, IntelliJ, etc.

4. **Procedure:**

Designing and implementing a simple inventory control system for a small video rental store involves several steps. Below is a procedural guide to help you through the process:

- Define the Video Class:
  Create a Video class to represent each video in the inventory.
- Create Inventory Class:

Develop an Inventory class to manage the collection of videos.

- Implement Rental System:
  Design a RentalSystem class to handle the rental transactions.

5. Code:

```
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class VideoRentalStore {
    private Map<String, Integer> inventory;

    public VideoRentalStore() {
        this.inventory = new HashMap<>();
    }

    public void addMovie(String title, int quantity) {
        inventory.put(title, quantity);
        System.out.println("Added " + quantity + " copies of " + title + " to the inventory.");
    }

    public void rentMovie(String title, int quantity) {
        if (inventory.containsKey(title)) {
            int availableCopies = inventory.get(title);
            if (availableCopies >= quantity) {
                inventory.put(title, availableCopies - quantity);
                System.out.println("Rented " + quantity + " copies of " + title + ".");
            } else {
                System.out.println("Sorry, not enough copies of " + title + " available.");
            }
        } else {
            System.out.println("Movie not found in the inventory.");
        }
    }
```

```
    public void displayInventory() {
        System.out.println("Current Inventory:");
        for (Map.Entry<String, Integer> entry : inventory.entrySet()) {
            System.out.println(entry.getKey() + ": " + entry.getValue() + " copies");
        }
    }

    public static void main(String[] args) {
        VideoRentalStore rentalStore = new VideoRentalStore();
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("\nMenu:\n1. Add Movie\n2. Rent Movie\n3. Display Inventory\n4.
Exit");
            System.out.print("Enter your choice: ");

            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume the newline character

            switch (choice) {
                case 1:
                    System.out.print("Enter movie title: ");
                    String title = scanner.nextLine();
                    System.out.print("Enter quantity: ");
                    int quantity = scanner.nextInt();
                    rentalStore.addMovie(title, quantity);
                    break;
                case 2:
                    System.out.print("Enter movie title: ");
                    title = scanner.nextLine();
                    System.out.print("Enter quantity to rent: ");
                    quantity = scanner.nextInt();
                    rentalStore.rentMovie(title, quantity);
                    break;
                case 3:
                    rentalStore.displayInventory();
                    break;
                case 4:
```

```
                System.out.println("Exiting program. Goodbye!");
                System.exit(0);
            default:
                System.out.println("Invalid choice. Please enter a valid option.");
        }
    }
  }
}
```

6. Output:

```
java -cp /tmp/KxXOa0B8uZ VideoRentalStore
Menu:
1. Add Movie
2. Rent Movie
3. Display Inventory
4. Exit
Enter your choice: 1
Enter movie title: La La Laa
Enter quantity: 40
Added 40 copies of La La Laa to the inventory.

Menu:
1. Add Movie
2. Rent Movie
3. Display Inventory
4. Exit
Enter your choice:
```

7. Learning Outcomes:

- Understand the Basics of Inventory Management.
- Master Java Programming Fundamentals.
- Database Design and Management.