```
package camerarentalapplication;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

class UserDetails {
    String USERNAME;
    String PASSWORD;
    public UserDetails(String USERNAME, String PASSWORD) {
        this.USERNAME = USERNAME;
        this.PASSWORD = PASSWORD;
    }
    public String getUSERNAME() {
        return USERNAME;
    }
    public String getPASSWORD() {
        return PASSWORD;
    }
}
class Camera {
    int CAMERA_ID;
    String BRAND;
    String MODEL;
    double PRICE;
    String STATUS;
    public Camera(int CAMERA_ID, String BRAND, String MODEL, double PRICE, String
STATUS) {
        this.CAMERA_ID = CAMERA_ID;
        this.BRAND = BRAND;
        this.MODEL = MODEL;
        this.PRICE = PRICE;
        this.STATUS = STATUS;
    }
    public int getCAMERA_ID() {
        return CAMERA_ID;
    }
    public String getBrand() {
        return BRAND;
    }
    public String getModel() {
        return MODEL;
    }
    public double getPrice() {
        return PRICE;
    }
    public String getStatus() {
        return STATUS;
    }
```

```
    public void setStatus(String status) {
        this.STATUS = status;
    }
}
class Wallet {
    double amount;
    public Wallet() {
        this.amount =1000;
    }
    public double getAmount() {
        return amount;
    }
    public void addAmount(double amount) {
        this.amount += amount;
    }
}
public class User {
            private static List<Camera> cameraList;
    public static void main(String[] args) {
        cameraList = new ArrayList<>();
        cameraList.add(new Camera(1, "Sony", "DSLR", 1000, "Rented"));
        cameraList.add(new Camera(2, "Canon", "Ds1234", 1500, "Available"));
        cameraList.add(new Camera(3, "LG", "L123", 500, "Available"));
        cameraList.add(new Camera(4, "Nikon", "2030", 500, "Available"));
        cameraList.add(new Camera(5, "Samsung", "DS123", 500, "Available"));
        displayLoginPage();
        optionsDisplay();
    }
    private static void displayLoginPage() {
        System.out.println("+------------------------------+");
        System.out.println("| WELCOME TO CAMERA RENTAL APP |");
        System.out.println("+------------------------------+");
        System.out.println("PLEASE LOGIN TO CONTINUE - ");
        Scanner scanner = new Scanner(System.in);
        System.out.println("USERNAME - ");
        String USERNAME = scanner.nextLine();
        System.out.println("PASSWORD - ");
        String PASSWORD = scanner.nextLine();
        if (isValidUser(USERNAME, PASSWORD)) {
            UserDetails user = new UserDetails(USERNAME, PASSWORD);
            System.out.println("Login successful\n");
        } else {
            System.out.println("Invalid username or password. Try again");
            displayLoginPage();
        }
    }
    private static boolean isValidUser(String username, String password) {
        return username.equals("admin") && password.equals("password");
```

```
}
private static void optionsDisplay() {
    Scanner scanner = new Scanner(System.in);
    int option;
    do {
        System.out.println("1. MY CAMERA");
        System.out.println("2. RENT A CAMERA");
        System.out.println("3. VIEW ALL CAMERAS");
        System.out.println("4. MY WALLET");
        System.out.println("5. EXIT");
        System.out.print("Enter your choice : ");
        option = scanner.nextInt();
        scanner.nextLine();
        switch (option) {
            case 1:
                myCameraOption();
                break;
            case 2:
                rentCameraOption();
                break;
            case 3:
                viewAllCameras();
                break;
            case 4:
                myWalletOption(scanner);
                break;
            case 5:
                System.out.println("Exiting..");
                break;
            default:
                System.out.println("Invalid option.");
        }
    } while (option != 5);
}
private static void myCameraOption() {
    Scanner S= new Scanner(System.in);
    int subOption;
    do {
        System.out.println("1. ADD");
        System.out.println("2. REMOVE");
        System.out.println("3. VIEW MY CAMERAS");
        System.out.println("4. GO TO PREVIOUS MENU");
        System.out.print("Enter your choice: ");
        subOption = S.nextInt();
        S.nextLine();
        switch (subOption) {
            case 1:
                addCamera();
```

```
                break;
            case 2:
                removeCamera();
                break;
            case 3:
                viewMyCameras();
                break;
            case 4:
                System.out.println("Returning to the previous menu...");
                break;
            default:
                System.out.println("Invalid option.");
        }
    } while (subOption != 4);
  }
  private static void addCamera() {
        Scanner sc=new Scanner(System.in);
        System.out.println("Camera id- ");
                int CAMERA_ID=sc.nextInt();
                System.out.println("ENTER THE CAMERA BRAND- ");
                String BRAND=sc.next();
                System.out.println("ENTE THE MODEL- ");
                String MODEL=sc.next();
                System.out.println("ENTER THE PRICE PER DAY(INR)- ");
                double PRICE=sc.nextDouble();
                System.out.println("ENTER THE STATUS- ");
                String STATUS=sc.next();
                Camera camera = new Camera(CAMERA_ID, BRAND, MODEL, PRICE,
STATUS);
    cameraList.add(camera);
                System.out.println("YOUR CAMERA HAS BEEN SUCCESSFULLY ADDED
TO THE LIST.");
  }
  private static void removeCamera() {
    // Implement your logic to remove a camera from the user's list
    System.out.println("Remove Camera option selected.");
    System.out.println("ENTER THE CAMERA ID TO REMOVE- ");
    Scanner sc=new Scanner(System.in);
    int cameraId = sc.nextInt();
    sc.nextLine();
    boolean removed = false;
    for (Camera camera : cameraList) {
        if (camera.getCAMERA_ID() == cameraId) {
            cameraList.remove(camera);
            removed = true;
            break;
        }
    }
```

```
        System.out.println("CAMERA SUCCESSFULLY REMOVED FROM THE LIST");
    }
    private static void viewMyCameras() {
            System.out.println("view camera list(1.yes 2.no)- ");
            Scanner sc=new Scanner(System.in);
            int m=sc.nextInt();
        if (m==1) {
            System.out.println("My Camera List:");

System.out.println("==========================================================
==================");
                System.out.println("CAMERA_ID\tBRAND\t\tMODEL\t\tPRICE\t\tSTATUS");

System.out.println("==========================================================
==================");
                for (int i=0;i<cameraList.size();i++) {
                    Camera camera=cameraList.get(i);

System.out.println("\n"+camera.getCAMERA_ID()+"\t\t"+camera.getBrand()+"\t\t"+camera.ge
tModel()+"\t\t"+camera.getPrice()+"\t\t"+camera.getStatus());
            }
        }
    }
    private static void rentCameraOption() {
        System.out.println("FOLLOWING IS THE LIST OF Available Cameras:");

System.out.println("==========================================================
==================");
        System.out.println("CAMERA_ID\tBRAND\t\tMODEL\t\tPRICE\t\tSTATUS");

System.out.println("==========================================================
==================");
        for (Camera camera : cameraList) {
            if (camera.getStatus().equals("Available")) {

System.out.println("\n"+camera.getCAMERA_ID()+"\t\t"+camera.getBrand()+"\t\t"+camera.ge
tModel()+"\t\t"+camera.getPrice()+"\t\t"+camera.getStatus());
            }
        }
        System.out.println("ENTER THE CAMERA ID YOU WANT TO RENT- ");
        Scanner sc=new Scanner(System.in);
        int cameraId = sc.nextInt();
        sc.nextLine();
        boolean rented = false;
        for (Camera camera : cameraList) {
            if(camera.getCAMERA_ID() == cameraId && camera.getStatus().equals("Available"))
{
                    double cameraPrice = camera.getPrice();
```

```
            Wallet wallet=new Wallet();
                                if (wallet.getAmount() >= cameraPrice) {
                camera.setStatus("Rented");
                rented = true;
                wallet.addAmount(-cameraPrice);
            System.out.println("YOUR TRANSACTION FOR CAMERA" + cameraId + " rented
successfully.");
                break;
          }else {
                    System.out.println("TRANSACTION FAILED DUE TO ISUFFICIENT WALLET
BALANCE. PLEASE DEPOSITE THE AMOUNT TO YOUR WALLET");
          }
          }
        }
   }
          private static void viewAllCameras() {
       System.out.println("All Available Cameras:");

System.out.println("==========================================================
==================");
       System.out.println("CAMERA_ID\tBRAND\t\tMODEL\t\tPRICE\t\tSTATUS");

System.out.println("==========================================================
==================");
       for (Camera camera : cameraList) {

System.out.println("\n"+camera.getCAMERA_ID()+"\t\t"+camera.getBrand()+"\t\t"+camera.ge
tModel()+"\t\t"+camera.getPrice()+"\t\t"+camera.getStatus());
       }
   }
   private static void myWalletOption(Scanner scanner) {
          Wallet wallet = new Wallet();
                    System.out.println("YOUR CURRENT WALLET BALANCE IS- " +
wallet.getAmount());
       System.out.print("Do you want to deposit more amount to your wallet? (1.yes 2.no): ");
       int depositChoice = scanner.nextInt();
       if (depositChoice==1) {
          System.out.print("Enter the amount : ");
          double depositAmount = scanner.nextDouble();
          scanner.nextLine(); // Consume newline character
          wallet.addAmount(depositAmount);
          System.out.println("YOUR WALLET BALANCE UPDATED SUCCESSFULLY.
CURRENT WALLET BALANCE- "+wallet.getAmount());
       }
   }
}
```