```java
package String;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import java.util.Scanner;


class Camera {

    private int id;

    private String brand;

    private String model;

    private double rentalAmount;


    private boolean isRented;


    public Camera(int id, String brand, String model, double rentalAmount) {

        this.id = id;

        this.brand = brand;

        this.model = model;

        this.rentalAmount = rentalAmount;

        this.isRented=false;

    }


    public int getId() {

        return id;

    }


    public String getBrand() {

        return brand;
```

```
    }

    public String getModel() {
        return model;
    }

    public double getRentalAmount() {
        return rentalAmount;
    }
    public boolean isRented() {
        return isRented;
    }
    public void setRented(boolean rented) {
        isRented = rented;
    }


    @Override
    public String toString() {
        String status = isRented ? "Rented" : "Available";
        return "Camera ID: "+"\n"+ + id +
                ", Brand: " + brand +
                ", Model: " + model +
                ", Rental Amount: $" + rentalAmount +
                ", Status: " + status;
    }
}
```

```
class User {
    private String username;
    private String password;
    private List<Camera> myCameras;
    private double walletBalance;

    public User(String username, String password) {
        this.username = username;
        this.password = password;
        this.myCameras = new ArrayList<>();
        this.walletBalance = 0.0;
    }

    public String getUsername() {
        return username;
    }

    public String getPassword() {
        return password;
    }

    public List<Camera> getMyCameras() {
        return myCameras;
    }

    public double getWalletBalance() {
        return walletBalance;
    }
```

```
    public void addCamera(Camera camera) {
        myCameras.add(camera);
    }

    public void removeCamera(Camera camera) {
        myCameras.remove(camera);
    }

    public void addToWallet(double amount) {
        walletBalance += amount;
    }

    public boolean deductFromWallet(double amount) {
        if (amount <= walletBalance) {
            walletBalance -= amount;
            return true;
        }
        return false;
    }
}

public class CameraRentalApplication {
    private List<Camera> cameraList;
    private User user;
    private Scanner scanner;

    public CameraRentalApplication(User user) {
```

```
        this.cameraList = new ArrayList<>();
        this.user = user;
        this.scanner = new Scanner(System.in);
    }

    public void addCamera(Camera camera) {
        cameraList.add(camera);
    }

    public Camera removeCamera(int id) {
        for (Camera camera : cameraList) {
            if (camera.getId() == id) {
                cameraList.remove(camera);
                return camera;
            }
        }
        return null;
    }

    public void addCameraToInventory() {
        System.out.println("Add a Camera to Inventory");
        System.out.print("Enter the camera ID: ");
        int id = scanner.nextInt();
        scanner.nextLine(); // Consume the newline character
        System.out.print("Enter the camera brand: ");
        String brand = scanner.nextLine();
        System.out.print("Enter the camera model: ");
        String model = scanner.nextLine();
```

```
        System.out.print("Enter the rental amount: ");
        double rentalAmount = scanner.nextDouble();

        Camera newCamera = new Camera(id, brand, model, rentalAmount);
        addCamera(newCamera);
    }

    public void removeCameraFromInventory() {
        System.out.println("Remove a Camera from Inventory");
        System.out.print("Enter the camera ID to remove: ");
        int id = scanner.nextInt();

        Camera removedCamera = removeCamera(id);
        if (removedCamera != null) {
            System.out.println("Camera removed successfully.");
        } else {
            System.out.println("Camera not found in the inventory.");
        }
    }

    public void viewMyCameras() {
        List<Camera> myCameras = user.getMyCameras();
        if (myCameras.isEmpty()) {
            System.out.println("You don't have any cameras.");
        } else {
            System.out.println("===== My Cameras =====");
            for (Camera camera : myCameras) {
                System.out.println(camera);
```

```
        }
    }
}

public void rentCamera() {
    System.out.println("Rent a Camera");
    System.out.println("Available Cameras:");
    viewAllCameras();
    System.out.print("Enter the camera ID to rent: ");
    int id = scanner.nextInt();

    Camera rentedCamera = null;
    for (Camera camera : cameraList) {
        if (camera.getId() == id) {
            rentedCamera = camera;
            break;
        }
    }

    if (rentedCamera != null) {
        if (user.deductFromWallet(rentedCamera.getRentalAmount())) {
            rentedCamera.setRented(true); // Set the status to rented
            user.addCamera(rentedCamera);
            cameraList.remove(rentedCamera);
            System.out.println("Camera rented successfully.");
        } else {
            System.out.println("Insufficient balance in your wallet.");
        }
```

```
    } else {
        System.out.println("Camera not found in the inventory.");
    }
}

public void viewAllCameras() {
    if (cameraList.isEmpty() && user.getMyCameras().isEmpty()) {
        System.out.println("No cameras available in the inventory.");
    } else {
        System.out.println("===== All Cameras =====");

        System.out.println("Available Cameras:");
        if (cameraList.isEmpty()) {
            System.out.println("No available cameras.");
        } else {
            for (Camera camera : cameraList) {
                System.out.println(camera);
            }
        }

        //System.out.println("\nRented Cameras:");
        List<Camera> rentedCameras = user.getMyCameras();
        if (rentedCameras.isEmpty()) {
            System.out.println("No rented cameras.");
        } else {
            for (Camera camera : rentedCameras) {
                System.out.println(camera);
            }
```

```
            }
        }
    }
```

```
    public void viewMyWallet() {
        double walletBalance = user.getWalletBalance();
        System.out.println("Wallet Balance: $" + walletBalance);
    }

    public void addFundsToWallet() {
        System.out.println("Add Funds to Wallet");
        System.out.print("Enter the amount to add: ");
        double amount = scanner.nextDouble();

        user.addToWallet(amount);
        System.out.println("Funds added to your wallet successfully.");
    }

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


        System.out.println("+--------------------------------------+");
        System.out.println("|Welcome to Camera Rental Application|");
        System.out.println("+--------------------------------------+");
        System.out.println("login to continue");
```

```
        System.out.print("Enter your username: ");
        String username = scanner.nextLine();
        System.out.print("Enter your password: ");
        String password = scanner.nextLine();

        if (validateLogin(username, password)) {
            User user = new User(username, password);
            CameraRentalApplication application = new
CameraRentalApplication(user);

            application.addCamera(new Camera(1, "Canon", "EOS R5", 50.0));
            application.addCamera(new Camera(2, "Nikon", "Z7 II", 45.0));
            application.addCamera(new Camera(3, "Sony", "A7R IV", 55.0));
            application.addCamera(new Camera(4, "Nikon", "Auu I", 60.0));
            application.addCamera(new Camera(5, "Canon", "EOS 5D Mark IV",
40.0));
            application.addCamera(new Camera(6, "Sony", "A9 II", 65.0));
            application.addCamera(new Camera(7, "Fujifilm", "X-T4", 35.0));
            application.addCamera(new Camera(8, "Nikon", "D850", 55.0));
            application.addCamera(new Camera(9, "Canon", "EOS 6D Mark II",
30.0));
            application.addCamera(new Camera(10, "Sony", "A7 III", 50.0));
            application.addCamera(new Camera(11, "Panasonic", "Lumix GH5",
45.0));
            application.addCamera(new Camera(12, "Nikon", "Z6 II", 55.0));
            application.addCamera(new Camera(13, "Canon", "EOS R6", 40.0));
            application.addCamera(new Camera(14, "Sony", "A7S III", 60.0));
            application.addCamera(new Camera(15, "Fujifilm", "X-T3", 35.0));
```

```
            application.addCamera(new Camera(16, "Nikon", "D750", 50.0));
            application.addCamera(new Camera(17, "Canon", "EOS 90D", 30.0));
            application.addCamera(new Camera(18, "Sony", "A6600", 45.0));
            application.addCamera(new Camera(19, "Olympus", "OM-D E-M1 Mark
III", 55.0));
            application.addCamera(new Camera(20, "Nikon", "D780", 40.0));



            int choice;
           do {
              System.out.println("\n===== RentMyCam.io =====");

                System.out.println("Welcome, " + user.getUsername() + "!");
                System.out.println("1. My Cameras");
                System.out.println("2. Rent a Camera");
                System.out.println("3. View All Cameras");
                System.out.println("4. My Wallet");
                System.out.println("5. Add Funds to Wallet");
                System.out.println("6. Exit");
                System.out.print("Enter your choice: ");
                choice = scanner.nextInt();

                switch (choice) {
                    case 1:
                        int cameraChoice;
                        do {
                            System.out.println("\n===== My Cameras =====");
                            System.out.println("1. Add");
```

```
                    System.out.println("2. Remove");
                    System.out.println("3. View Customer ordered
Cameras");
                    System.out.println("4. Go to previous Menu");
                    System.out.print("Enter your choice: ");
                    cameraChoice = scanner.nextInt();

                    switch (cameraChoice) {
                        case 1:
                            application.addCameraToInventory();
                            break;

                        case 2:
                            application.removeCameraFromInventory();
                            break;
                        case 3:
                            application.viewMyCameras();
                             break;
                        case 4:

                            break;

                        default:
                            System.out.println("Invalid choice.
Please try again.");
                            break;
                    }
                } while (cameraChoice != 4);
                break;
```

```
                    case 2:
                        application.rentCamera();
                        break;
                    case 3:
                        application.viewAllCameras();
                        break;
                    case 4:
                        application.viewMyWallet();
                        break;
                    case 5:
                        application.addFundsToWallet();
                        break;
                    case 6:
                        System.out.println("Thank you for using RentMyCam.io.
Goodbye!");
                        break;
                    default:
                        System.out.println("Invalid choice. Please try
again.");
                        break;
                }
            } while (choice != 6);

        } else {
            System.out.println("Invalid username or password. Please try
again.");
        }
    }
```

```
    private static boolean validateLogin(String username, String password)
{

      return  username.equals("admin") && password.equals("password");

}

}
```