

SOURCE CODE : - CAMERA RENTAL APPLICATION

```
package camera;
```

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

```
import java.util.HashMap;
```

```
import java.util.List;
```

```
import java.util.Map;
```

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

```
class Camera {
```

```
    private String cameraId;
```

```
    private String brand;
```

```
    private String model;
```

```
    private double perDayPrice;
```

```
    private boolean isAvailable;
```

```
    public Camera(String cameraId, String brand, String model, double  
perDayPrice, boolean isAvailable) {
```

```
        this.cameraId = cameraId;
```

```
        this.brand = brand;
```

```
        this.model = model;
```

```
        this.perDayPrice = perDayPrice;
```

```
        this.isAvailable = isAvailable;
```

```
    }
```

```
public String getCameraId() {  
    return cameraId;  
}
```

```
public String getBrand() {  
    return brand;  
}
```

```
public String getModel() {  
    return model;  
}
```

```
public double getPerDayPrice() {  
    return perDayPrice;  
}
```

```
public boolean isAvailable() {  
    return isAvailable;  
}
```

```
public void setAvailable(boolean available) {  
    isAvailable = available;  
}
```

```
@Override
```

```
public String toString() {  
    return "Camera ID: \t" + cameraId +  
        ", Brand: \t" + brand +  
        ", Model: \t " + model +  
        ", Per Day Price: $\t" + perDayPrice +  
        ", Status: " + (isAvailable ? "Available" : "Rented");  
}  
}
```

```
class User {  
    private String username;  
    private String password;  
    private double walletBalance;  
  
    public User(String username, String password, double walletBalance) {  
        this.username = username;  
        this.password = password;  
        this.walletBalance = walletBalance;  
    }  
  
    public String getUsername() {  
        return username;  
    }  
  
    public String getPassword() {  
        return password;  
    }  
}
```

```
}
```

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

```
public void setWalletBalance(double walletBalance) {  
    this.walletBalance = walletBalance;  
}
```

```
@Override  
public String toString() {  
    return "Username: " + username +  
        ", Wallet Balance: $" + walletBalance;  
}  
}
```

```
public class CameraRentalApplication {  
    private static Map<String, User> users = new HashMap<>();  
    private static List<Camera> cameras = new ArrayList<>();  
    private static User currentUser;  
    private static Scanner scanner = new Scanner(System.in);  
  
    public static void main(String[] args) {  
        initializeUsers();  
        initializeCameras();  
    }  
}
```

```
boolean quit = false;
while (!quit) {
    displayLoginPage();
    int choice = scanner.nextInt();
    scanner.nextLine(); // Consume newline character

    switch (choice) {
        case 1:
            login();
            break;
        case 2:
            quit = true;
            System.out.println("Thank you for using the Camera Rental
Application. Goodbye!");
            break;
        default:
            System.out.println("Invalid choice. Please try again.");
            break;
    }
}

private static void initializeUsers() {
    users.put("raj", new User("raj", "raj123", 100.0));
    users.put("user2", new User("user2", "password2", 200.0));
}
```

```
private static void initializeCameras() {  
    cameras.add(new Camera("C1", "Canon", "Model X", 10.0, true));  
    cameras.add(new Camera("C2", "Nikon", "Model Y", 15.0, true));  
    cameras.add(new Camera("C3", "Sony", "Model Z", 12.0, false));  
}
```

```
private static void displayLoginPage() {  
    System.out.println("===== Camera Rental Application  
=====");  
    System.out.println("1. Login");  
    System.out.println("2. Quit");  
    System.out.print("Enter your choice: ");  
}
```

```
private static void login() {  
    System.out.print("Enter username: ");  
    String username = scanner.nextLine();  
    System.out.print("Enter password: ");  
    String password = scanner.nextLine();  
  
    User user = users.get(username);  
    if (user != null && user.getPassword().equals(password)) {  
        currentUser = user;  
        System.out.println("Login successful. Welcome, " +  
currentUser.getUsername() + "!");  
        showMainMenu();  
    }
```

```
    } else {  
        System.out.println("Invalid username or password. Login failed.");  
    }  
}
```

```
private static void showMainMenu() {  
    boolean backToLogin = false;  
    while (!backToLogin) {  
        System.out.println("\n===== Main Menu  
=====");  
        System.out.println("1. My Camera");  
        System.out.println("2. Rent a Camera");  
        System.out.println("3. View All Cameras");  
        System.out.println("4. Wallet");  
        System.out.println("5. Logout");  
        System.out.print("Enter your choice: ");  
        int choice = scanner.nextInt();  
        scanner.nextLine(); // Consume newline character  
  
        switch (choice) {  
            case 1:  
                handleMyCameraMenu();  
                break;  
            case 2:  
                handleRentCameraMenu();  
                break;  
            case 3:
```

```

        displayAllCameras();
        break;
    case 4:
        handleWalletMenu();
        break;
    case 5:
        backToLogin = true;
        currentUser = null;
        break;
    default:
        System.out.println("Invalid choice. Please try again.");
        break;
    }
}
}
}

```

```

private static void handleMyCameraMenu() {
    boolean backToMenu = false;
    while (!backToMenu) {
        System.out.println("\n===== My Camera =====");
        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: ");
        int choice = scanner.nextInt();
    }
}

```



```
scanner.nextLine(); // Consume newline character
```

```
switch (choice) {  
    case 1:  
        addCamera();  
        break;  
    case 2:  
        removeCamera();  
        break;  
    case 3:  
        viewMyCameras();  
        break;  
    case 4:  
        backToMenu = true;  
        break;  
    default:  
        System.out.println("Invalid choice. Please try again.");  
        break;  
}  
}  
}
```

```
private static void addCamera() {  
    System.out.print("Enter Brand: ");  
    String brand = scanner.nextLine();  
    System.out.print("Enter Model: ");
```

```
String model = scanner.nextLine();  
System.out.print("Enter Per Day Price: ");  
double perDayPrice = scanner.nextDouble();  
scanner.nextLine(); // Consume newline character
```

```
String cameraId = "C" + (cameras.size() + 1);  
Camera camera = new Camera(cameraId, brand, model, perDayPrice,  
true);  
cameras.add(camera);  
  
System.out.println("Your camera has been successfully added to the list.");  
}
```

```
private static void removeCamera() {  
    System.out.println("\n===== Remove Camera =====");  
    displayCameraList();
```

```
    System.out.print("Enter Camera ID to remove: ");  
    String cameraId = scanner.nextLine();
```

```
    boolean cameraRemoved = false;  
    for (Camera camera : cameras) {  
        if (camera.getCameraId().equalsIgnoreCase(cameraId)) {  
            cameras.remove(camera);  
            cameraRemoved = true;  
            break;  
        }  
    }
```

```
}
```

```
if (cameraRemoved) {
```

```
    System.out.println("Camera successfully removed from the list.");
```

```
} else {
```

```
    System.out.println("Camera not found in the list.");
```

```
}
```

```
}
```

```
private static void viewMyCameras() {
```

```
    System.out.println("\n===== My Cameras =====");
```

```
    boolean cameraFound = false;
```

```
    for (Camera camera : cameras) {
```

```
        if (!camera.isAvailable() &&
```

```
camera.getId().equals(currentUser.getUsername())) {
```

```
            System.out.println(camera);
```

```
            cameraFound = true;
```

```
        }
```

```
    }
```

```
    if (!cameraFound) {
```

```
        System.out.println("You have no rented cameras.");
```

```
    }
```

```
}
```

```
private static void handleRentCameraMenu() {
```

```

        System.out.println("\n===== Rent a Camera
=====");

        displayCameraList();

        System.out.print("Enter Camera ID to rent: ");
        String cameraId = scanner.nextLine();

        Camera selectedCamera = null;
        for (Camera camera : cameras) {
            if (camera.getCameraId().equalsIgnoreCase(cameraId)) {
                selectedCamera = camera;
                break;
            }
        }

        if (selectedCamera != null) {
            if (selectedCamera.isAvailable()) {
                double rentAmount = selectedCamera.getPerDayPrice();
                if (currentUser.getWalletBalance() >= rentAmount) {
                    currentUser.setWalletBalance(currentUser.getWalletBalance() -
rentAmount);
                    selectedCamera.setAvailable(false);
                    System.out.println("Camera rented successfully. Rent amount: $" +
rentAmount);
                } else {
                    System.out.println("Transaction failed due to insufficient balance.");
                }
            }
        }
    }
}

```

```

        } else {
            System.out.println("Camera is already rented.");
        }
    } else {
        System.out.println("Camera not found in the list.");
    }
}

private static void displayAllCameras() {
    System.out.println("\n===== All Cameras
=====");
    displayCameraList();
}

// private static void displayCameraList() {
//     for (Camera camera : cameras) {
//         System.out.println(camera);
//     }
// }

private static void displayCameraList() {
    if (cameras.isEmpty()) {
        System.out.println("No cameras available.");
        return;
    }

    //System.out.println("\n===== All Cameras
=====");

```

```
System.out.println("\n=====
=====");
```

```
System.out.format("%-10s %-10s %-15s %-15s %s\n", "Camera ID",
"Brand", "Model", "Per Day Price", "Status");
```

```
System.out.println("=====
=====");
```

```
for (Camera camera : cameras) {
    System.out.format("%-10s %-10s %-15s $%-15.2f %s\n",
        camera.getCameraId(), camera.getBrand(), camera.getModel(),
        camera.getPerDayPrice(), (camera.isAvailable() ? "Available" :
"Rented"));
}
}
```

```
private static void handleWalletMenu() {
    System.out.println("\n===== Wallet
=====");

    System.out.println("Current Wallet Balance: $" +
currentUser.getWalletBalance());

    System.out.print("Do you want to add amount? (yes/no): ");

    String choice = scanner.nextLine();

    if (choice.equalsIgnoreCase("yes")) {
        System.out.print("Enter the amount to add: ");

        double amount = scanner.nextDouble();

        scanner.nextLine(); // Consume newline character
```

```
        currentUser.setWalletBalance(currentUser.getWalletBalance() +  
amount);
```

```
        System.out.println("Wallet balance updated. Current balance: $" +  
currentUser.getWalletBalance());
```

```
    }
```

```
}
```

```
}
```