Phase end proj1

Code:

**package** cameraRentalApplicationDefaultPackage;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** showWelcomeAndDeveloperDetils() {

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

System.***out***.println("Welcome to camera rental application");

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

}

**public** **static** **void** main(String[] args) {

UserLoginService loginService = **new** UserLoginService();

Scanner sc = **new** Scanner(System.***in***);

*showWelcomeAndDeveloperDetils*();

**boolean** isAuthenticated = **false**;

User user = **null**;

**while** (!isAuthenticated) {

System.***out***.println("Enter username:");

String usernameInput = sc.nextLine();

System.***out***.println("Enter password:");

String passwordInput = sc.nextLine();

isAuthenticated = loginService.authenticateUser(usernameInput, passwordInput);

**if** (isAuthenticated) {

user = loginService.getUserDetails(usernameInput);

**if** (user != **null**) {

System.***out***.println("\nLogin successful!");

**break**;

}

} **else** {

System.***out***.println("\nInvalid username or password. Please try again.");

System.***out***.println("\n\*-------------------------------------------------\*");

}

}

**if** (user != **null**) {

MenuHandler menuHandler = **new** MenuHandler(sc, loginService);

menuHandler.displayMenu(user);

}

}

}

package cameraRentalApplicationDefaultPackage;

import java.util.ArrayList;

import java.util.List;

public class UserRepository {

private static List<User> userDetails = new ArrayList<>();

static {

// Adding some sample user details to the ArrayList

userDetails.add(new User("user1", "password1", 100.0));

userDetails.add(new User("user2", "password2", 200.0));

userDetails.add(new User("user3", "password3", 300.0));

}

public static List<User> getUserDetails() {

return userDetails;

}

}

**package** cameraRentalApplicationDefaultPackage;

**import** java.util.List;

**public** **class** UserLoginService {

**public** **boolean** authenticateUser(String username, String password) {

List<User> userDetails = UserRepository.*getUserDetails*();

**for** (User user : userDetails) {

**if** (user.getUsername().equals(username) && user.getPassword().equals(password)) {

**return** **true**;

}

}

**return** **false**;

}

**public** User getUserDetails(String username) {

List<User> userDetails = UserRepository.*getUserDetails*();

**for** (User user : userDetails) {

**if** (user.getUsername().equals(username)) {

**return** user;

}

}

**return** **null**;

}

}

**package** cameraRentalApplicationDefaultPackage;

**import** java.util.Scanner;

**public** **class** MenuHandler {

**private** Scanner scanner;

**private** UserLoginService loginService;

**public** MenuHandler(Scanner scanner, UserLoginService loginService) {

**this**.scanner = scanner;

**this**.loginService = loginService;

}

**public** **void** displayMenu(User user) {

**while** (**true**) {

System.***out***.println("\nMenu:\n");

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");

**int** choice = scanner.nextInt();

scanner.nextLine();

**switch** (choice) {

**case** 1:

CameraAction.*showCameraMenu*();

**break**;

**case** 2:

CameraAction.*rentCamera*();

**break**;

**case** 3:

CameraAction.*ViewAllCamers*();

**break**;

**case** 4:

WalletHandler.*addAmount*(user, scanner);

**break**;

**case** 5:

System.***out***.println("EXITING !");

**return**;

**default**:

System.***out***.println("INVALID CHOICE PLEASE TRY AGAIN");

}

}

}

}

**package** cameraRentalApplicationDefaultPackage;

**public** **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** **void** setUsername(String username) {

**this**.username = username;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** **void** setWalletBalance(**double** walletBalance) {

**this**.walletBalance = walletBalance;

}

**public** String getUsername() {

**return** username;

}

**public** String getPassword() {

**return** password;

}

**public** **double** getWalletBalance() {

**return** walletBalance;

}

}

package cameraRentalApplicationDefaultPackage;

import java.util.ArrayList;

import java.util.List;

public class CameraRepository {

private static List<Camera> cameras = new ArrayList<>();

static {

cameras.add(new Camera(1, "Canon", "EOS R", 2500.0, true));

cameras.add(new Camera(2, "Nikon", "D850", 30.0, true));

cameras.add(new Camera(3, "Sony", "A7III", 35.0, false));

}

public static List<Camera> getAllCameras() {

return cameras;

}

public static void removeCamera(Camera camera) {

cameras.remove(camera);

}

public static void addCamera(int id, String brand, String model, double perDayRent, boolean available) {

cameras.add(new Camera(id, brand, model, perDayRent, available));

}

}

**package** cameraRentalApplicationDefaultPackage;

**public** **class** Camera {

**private** **int** id;

**private** String brand;

**private** String model;

**private** **double** perDayRent;

**private** **boolean** available;

**public** Camera(**int** id, String brand, String model, **double** perDayRent, **boolean** available) {

**this**.id = id;

**this**.brand = brand;

**this**.model = model;

**this**.perDayRent = perDayRent;

**this**.available = available;

}

**public** **int** getId() {

**return** id;

}

**public** String getBrand() {

**return** brand;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** **void** setBrand(String brand) {

**this**.brand = brand;

}

**public** **void** setModel(String model) {

**this**.model = model;

}

**public** **void** setPerDayRent(**double** perDayRent) {

**this**.perDayRent = perDayRent;

}

**public** String getModel() {

**return** model;

}

**public** **double** getPerDayRent() {

**return** perDayRent;

}

**public** **boolean** getAvailable() {

**return** available;

}

**public** **void** setAvailable(**boolean** available) {

**this**.available = available;

}

}

**package** cameraRentalApplicationDefaultPackage;

**import** java.util.\*;

**public** **class** CameraAction {

**static** Scanner *scanner* = **new** Scanner(System.***in***);

**static** CameraRepository *cameraRepository* = **new** CameraRepository();

**static** UserRepository *userRepository* = **new** UserRepository();

**public** **static** **void** showCameraMenu() {

**int** choice;

**do** {

System.***out***.println("1. ADD");

System.***out***.println("2. REMOVE");

System.***out***.println("3. VIEW MY CAMERAS");

System.***out***.println("4. GO BACK TO PREVIOUS MENU");

choice = *scanner*.nextInt();

*scanner*.nextLine();

**switch** (choice) {

**case** 1:

*addNewCamera*();

**return**;

**case** 2:

*removeCamera*();

**return**;

**case** 3:

*ViewAllCamers*();

**return**;

**case** 4:

System.***out***.println("Going back to Main Menu...");

**break**;

**default**:

System.***out***.println("Invalid choice. Please enter a valid option.");

**break**;

}

} **while** (choice != 4);

}

**public** **static** **void** addNewCamera() {

System.***out***.println("ENTER THE CAMERA BRAND -");

String brand = *scanner*.next();

System.***out***.println("ENTER THE MODEL -");

String model = *scanner*.next();

System.***out***.println("ENTER THE PER DAYPRICE (INR) -");

**double** perDayRent = *scanner*.nextDouble();

*cameraRepository*.*addCamera*(*cameraRepository*.*getAllCameras*().size() + 1, brand, model, perDayRent, **true**);

System.***out***.println("YOUR CAERA HAS BEEN SUCCESSFULL ADDED TO THE LIST");

}

**public** **static** **void** removeCamera() {

List<Camera> cameras = *cameraRepository*.*getAllCameras*();

**if** (cameras.isEmpty()) {

System.***out***.println("No cameras available to remove.");

**return**;

}

System.***out***.println(

"\n====================================================================================================");

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", "CAMERA Id", "BRAND", "MODEL", "PRICE( PER DAY)",

"STATUSUSER");

System.***out***.println(

"\n====================================================================================================");

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

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", camera.getId(), camera.getBrand(), camera.getModel(),

camera.getPerDayRent(), camera.getAvailable() ? "Available" : "Rented");

}

System.***out***.print("Enter THE CAMERA ID TO REMOVE");

**int** selectedId = *scanner*.nextInt();

**boolean** found = **false**;

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

**if** (camera.getId() == selectedId) {

*cameraRepository*.*removeCamera*(camera);

found = **true**;

**break**;

}

}

**if** (found) {

System.***out***.println("CAMERA SUCCESSFULLY REMOVED FROM THE LIST ");

} **else** {

System.***out***.println("CAMERA WITH THE ID " + selectedId + " NOT FOUND");

}

}

**public** **static** **void** ViewAllCamers() {

List<Camera> cameras = *cameraRepository*.*getAllCameras*();

List<User> userDetails = *userRepository*.*getUserDetails*();

**if** (cameras.isEmpty()) {

System.***out***.println("NO CAMERAS AVAILABLE FOR RENT.");

**return**;

}

System.***out***.println("FOLLOWING IS THE LIST OF AVAILABEL CAMERA(S)");

System.***out***.println(

"\n====================================================================================================");

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", "CAMERA Id", "BRAND", "MODEL", "PRICE( PER DAY)",

"STATUSUSER");

System.***out***.println(

"\n====================================================================================================");

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

**if** (camera.getAvailable()) {

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", camera.getId(), camera.getBrand(),

camera.getModel(), camera.getPerDayRent(), camera.getAvailable() ? "Available" : "Rented");

}

}

}

**public** **static** **void** rentCamera() {

List<Camera> cameras = *cameraRepository*.*getAllCameras*();

List<User> userDetails = *userRepository*.*getUserDetails*();

**if** (cameras.isEmpty()) {

System.***out***.println("NO CAMERAS AVAILABLE FOR RENT.");

**return**;

}

System.***out***.println("FOLLOWING IS THE LIST OF AVAILABEL CAMERA(S)");

System.***out***.println(

"\n====================================================================================================");

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", "CAMERA Id", "BRAND", "MODEL", "PRICE( PER DAY)",

"STATUSUSER");

System.***out***.println(

"\n====================================================================================================");

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

**if** (camera.getAvailable()) {

System.***out***.printf("\n%-15s %-15s %-20s %-20s %s%n", camera.getId(), camera.getBrand(),

camera.getModel(), camera.getPerDayRent(), camera.getAvailable() ? "Available" : "Rented");

}

}

System.***out***.print("Enter THE CAMERA ID TO RENT");

**int** selectedId = *scanner*.nextInt();

Camera selectedCamera = **null**;

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

**if** (camera.getId() == selectedId && camera.getAvailable()) {

selectedCamera = camera;

**break**;

}

}

**if** (selectedCamera != **null**) {

User currentUser = userDetails.get(0);

**if** (currentUser.getWalletBalance() >= selectedCamera.getPerDayRent()) {

System.***out***.println(

"YOUR TRANSACTION FOR CAMERA - " + selectedCamera.getBrand() + selectedCamera.getModel()

+ " WITH RENT " + selectedCamera.getPerDayRent() + " HAS SUCCESSFULLY COMPLETED");

**double** newBalance = currentUser.getWalletBalance() - selectedCamera.getPerDayRent();

currentUser.setWalletBalance(newBalance);

selectedCamera.setAvailable(**false**);

} **else** {

System.***out***.println("Insufficient balance to rent this camera.");

}

} **else** {

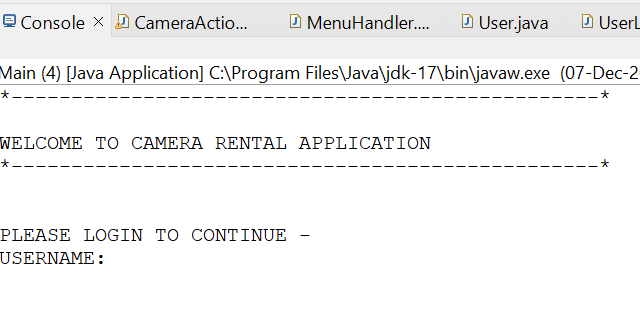
System.***out***.println("Invalid camera selection or camera not available for rent.");

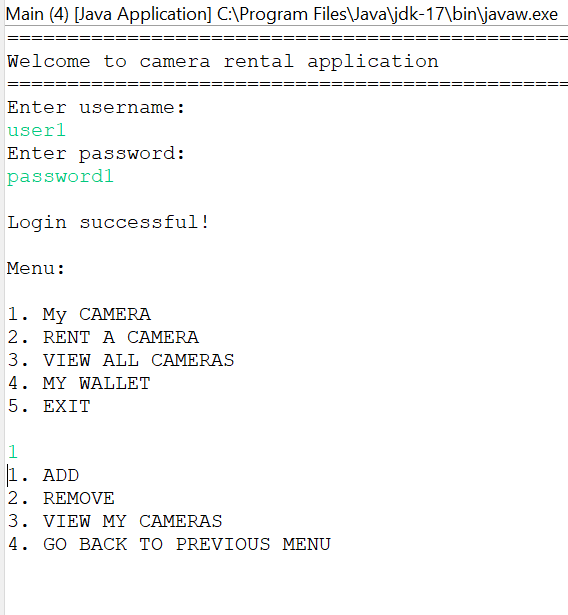
}

}

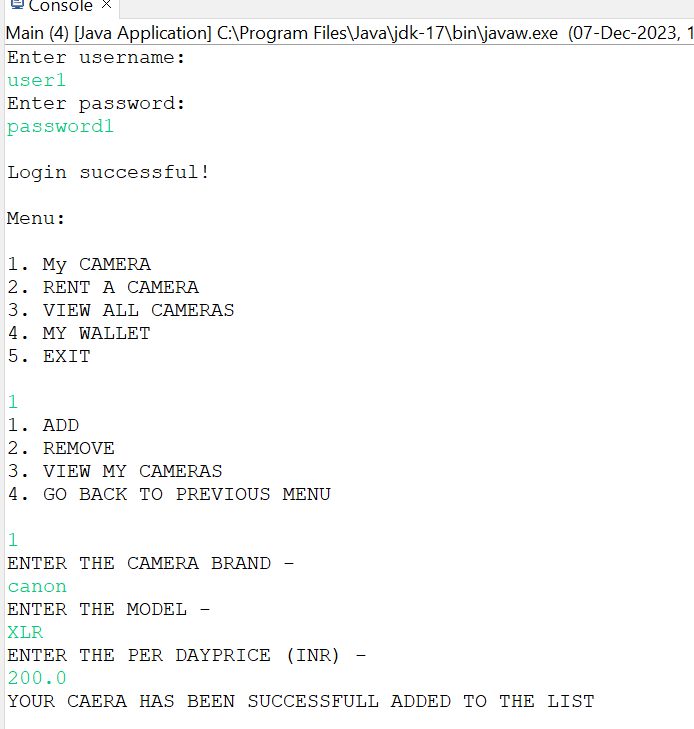
}

Output:

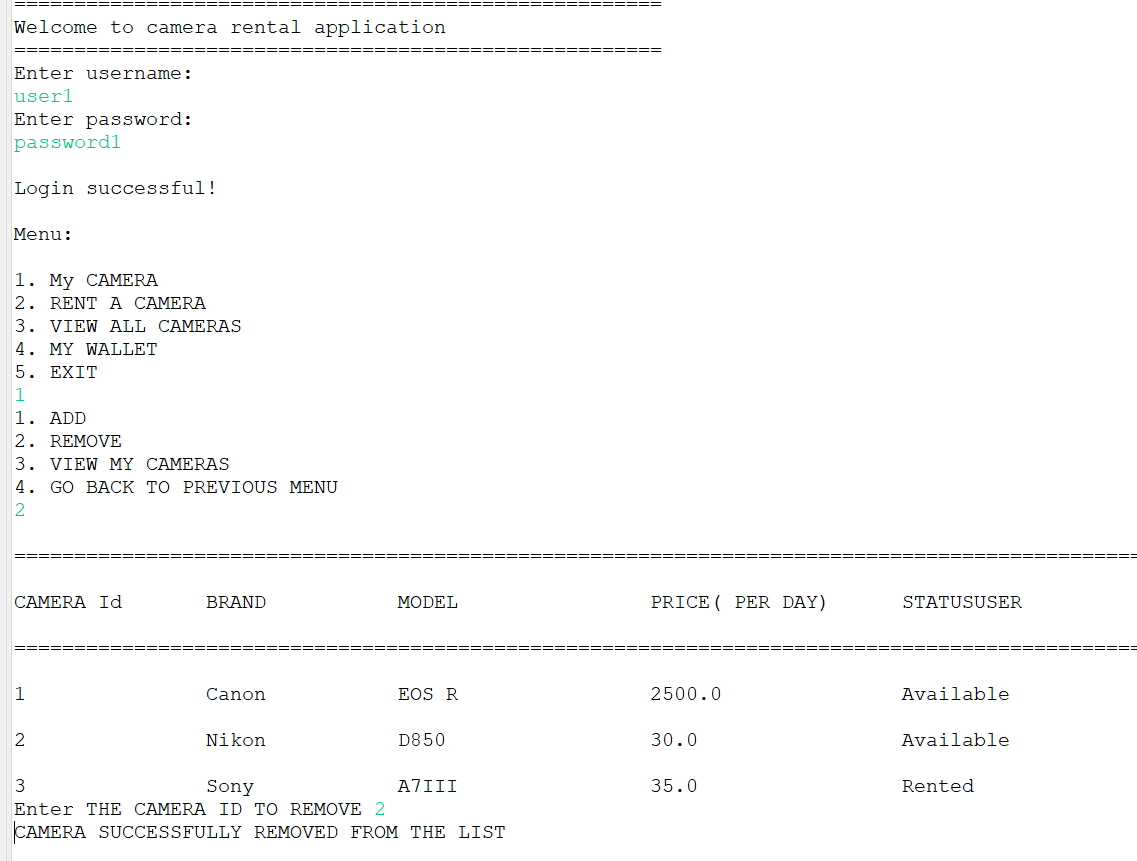




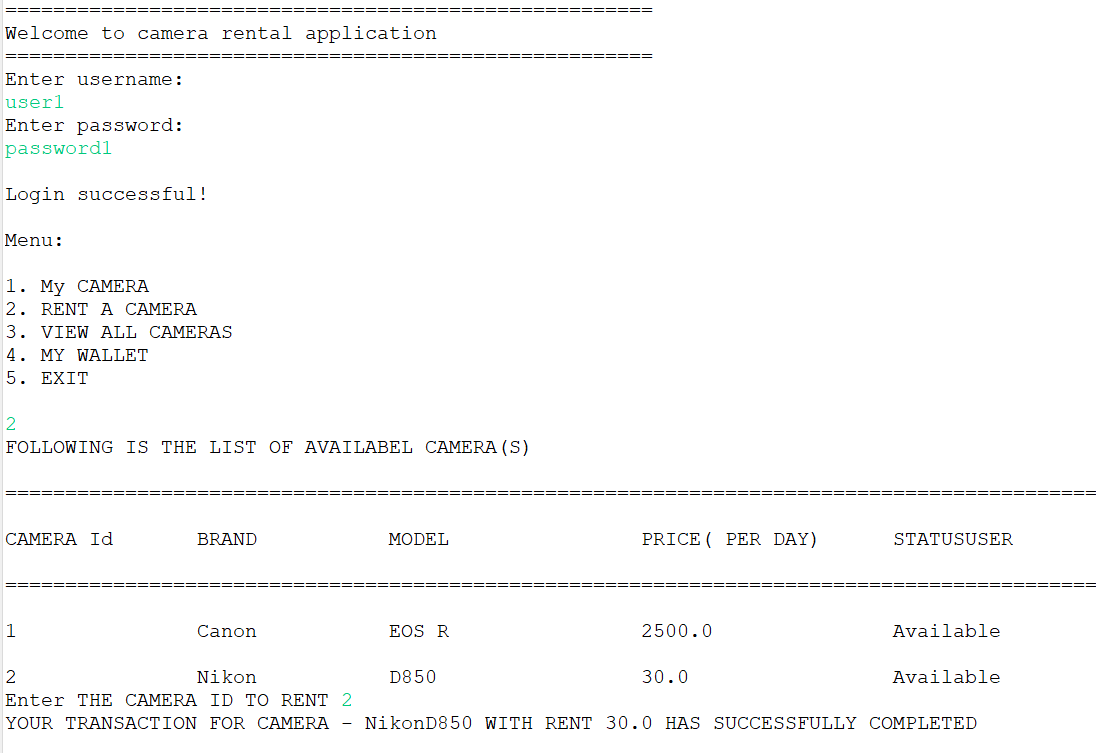
1.Adding Camera



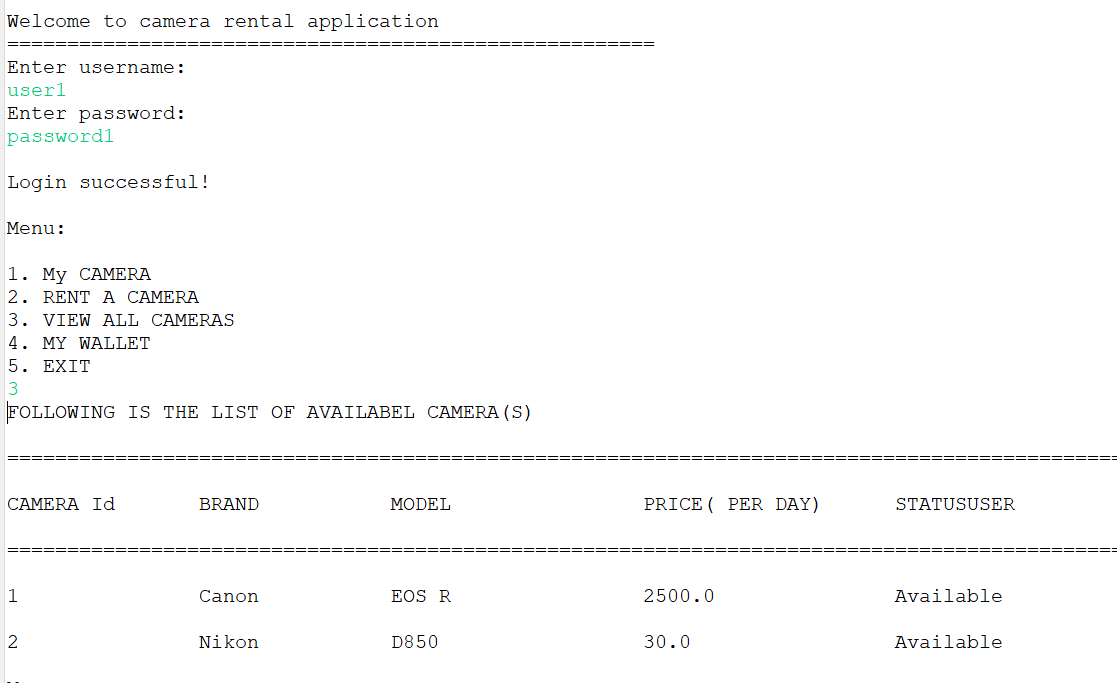
2.Remove camera:



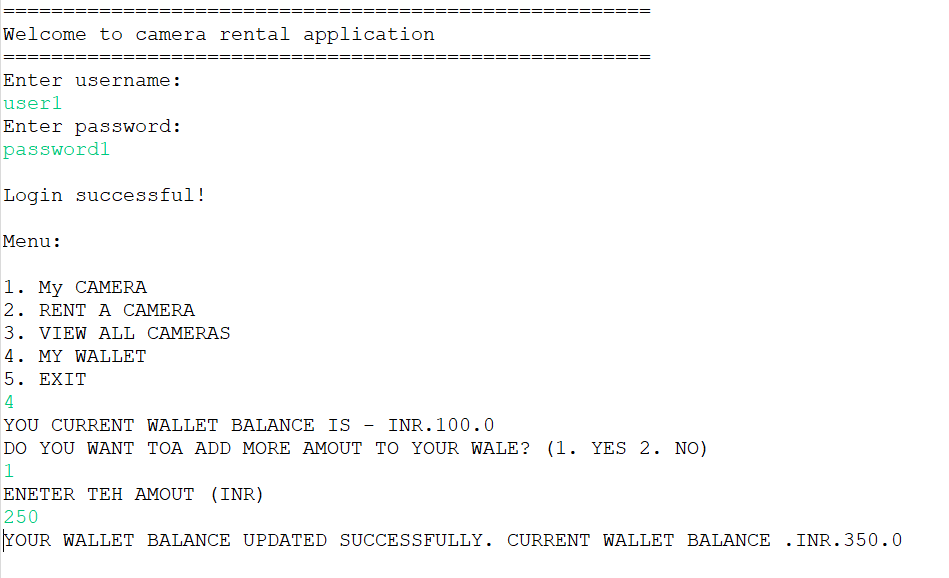
1. Rent a Camera



1. View all cameras



1. My Wallet or Add Money



1. Failed Transaction

