```
class Camera {
    private String brand;
    private String model;
    private double rentAmount;

    public Camera(String brand, String model, double rentAmount) {
        this.brand = brand;
        this.model = model;
        this.rentAmount = rentAmount;
    }

    public String getBrand() {
        return brand;
    }

    public String getModel() {
        return model;
    }

    public double getRentAmount() {
        return rentAmount;
    }
}
```

```
class User1 {
    private double walletBalance;

    public User1(double walletBalance) {
        this.walletBalance = walletBalance;
    }

    public double getWalletBalance() {
        return walletBalance;
    }

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

    public void addMoneyToWallet(double amount) {
        walletBalance += amount;
        System.out.println("Amount added successfully. Current wallet
balance: " + walletBalance);
    }
}
```

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

public class CameraRentalApp {
    private List<Camera> cameraList;
    private User1 user;
    private boolean loggedIn;

    public CameraRentalApp() {
        cameraList = new ArrayList<>();
        user = new User1(0.0);
        loggedIn = false;
    }

    public void login() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = scanner.nextLine();
        System.out.print("Enter password: ");
        String password = scanner.nextLine();

        if (authenticate(username, password)) {
            loggedIn = true;
            System.out.println("Logged in successfully!");
        } else {
            System.out.println("Invalid username or password. Please
try again.");
        }
    }

    public void logout() {
        loggedIn = false;
        System.out.println("Logged out successfully!");
    }

    private boolean authenticate(String username, String password) {

        if (username.equals("admin") && password.equals("password")) {
            return true;
        }
        return false;
    }

    public void checkWalletBalance() {
        System.out.println("Your current wallet balance is: " +
user.getWalletBalance());
    }

    public void addMoneyToWallet(double amount) {
        user.addMoneyToWallet(amount);
```

```
    }

    public void addCamera(String brand, String model, double
rentAmount) {
        Camera camera = new Camera(brand, model, rentAmount);
        cameraList.add(camera);
        System.out.println("Camera added successfully!");
    }

    public void removeCamera(int cameraIndex) {
        if (cameraIndex >= 0 && cameraIndex < cameraList.size()) {
            cameraList.remove(cameraIndex);
            System.out.println("Camera removed successfully!");
        } else {
            System.out.println("Invalid camera selection.");
        }
    }

    public void displayCameraList() {
        if (cameraList.isEmpty()) {
            System.out.println("No cameras available for rent.");
        } else {
            System.out.println("Cameras available for rent:");
            for (int i = 0; i < cameraList.size(); i++) {
                Camera camera = cameraList.get(i);
                System.out.println("Camera " + (i + 1) + ":");
                System.out.println("Brand: " + camera.getBrand());
                System.out.println("Model: " + camera.getModel());
                System.out.println("Rent Amount: " +
camera.getRentAmount() + " per day");
                System.out.println("-----------------------");
            }
        }
    }

    public void rentCamera(int cameraIndex) {
        if (cameraIndex >= 0 && cameraIndex < cameraList.size()) {
            Camera camera = cameraList.get(cameraIndex);
            if (user.getWalletBalance() >= camera.getRentAmount()) {
                user.setWalletBalance(user.getWalletBalance() -
camera.getRentAmount());
                System.out.println("Camera rented successfully!");
            } else {
                System.out.println("Insufficient balance. Please
recharge your wallet.");
            }
        } else {
            System.out.println("Invalid camera selection.");
        }
    }

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

```
        CameraRentalApp app = new CameraRentalApp();
        Scanner scanner = new Scanner(System.in);
        int choice;

        do {
            System.out.println("----- WELCOME TO CAMERA RENTAL APP ---
--");
            System.out.println("1. Log in");
            System.out.println("2. Exit");
            System.out.print("Enter your choice: ");
            choice = scanner.nextInt();

            switch (choice) {
                case 1:
                    app.login();
                    if (app.loggedIn) {
                        int userChoice;
                        do {
                            System.out.println("----- CAMERA RENTAL
APP MENU -----");
                            System.out.println("1. Add a camera");
                            System.out.println("2. Remove a camera");
                            System.out.println("3. Rent a camera");
                            System.out.println("4. View all cameras");
                            System.out.println("5. My wallet / Add
money");
                            System.out.println("6. Failed
transaction");
                            System.out.println("7. Log out");
                            System.out.print("Enter your choice: ");
                            userChoice = scanner.nextInt();

                            switch (userChoice) {
                                case 1:
                                    System.out.print("Enter camera
brand: ");
                                    String brand = scanner.next();
                                    System.out.print("Enter camera
model: ");
                                    String model = scanner.next();
                                    System.out.print("Enter rent
amount per day: ");
                                    double rentAmount =
scanner.nextDouble();
                                    app.addCamera(brand, model,
rentAmount);
                                    break;
                                case 2:
                                    System.out.print("Enter the camera
number to remove: ");
                                    int removeIndex =
scanner.nextInt() - 1;
```

```
                                    app.removeCamera(removeIndex);
                                    break;
                                case 3:
                                    System.out.print("Enter the camera
number to rent: ");
                                    int cameraIndex =
scanner.nextInt() - 1;
                                    app.rentCamera(cameraIndex);
                                    break;
                                case 4:
                                    app.displayCameraList();
                                    break;
                                case 5:
                                    System.out.println("1. View wallet
balance");
                                    System.out.println("2. Add money
to wallet");
                                    System.out.print("Enter your
choice: ");
                                    int walletChoice =
scanner.nextInt();
                                    if (walletChoice == 1) {
                                        app.checkWalletBalance();
                                    } else if (walletChoice == 2) {
                                        System.out.print("Enter the
amount to add to your wallet: ");
                                        double amount =
scanner.nextDouble();
                                        app.addMoneyToWallet(amount);
                                    } else {
                                        System.out.println("Invalid
choice.");
                                    }
                                    break;
                                case 6:
                                    System.out.println("Failed
transaction. Please try again.");
                                    break;
                                case 7:
                                    app.logout();
                                    break;
                                default:
                                    System.out.println("Invalid
choice. Please try again.");
                            }
                            System.out.println();
                        } while (userChoice != 7);
                    }
                    break;
                case 2:
                    System.out.println("Exiting the application...");
                    break;
```

```
                default:
                    System.out.println("Invalid choice. Please try
again.");
            }
            System.out.println();
        } while (choice != 2);

        scanner.close();
    }
}
```