

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

class Camera {
    private int id;
    private String brand;
    private String model;
    private double rentalAmountPerDay;
    private boolean rented;

    public Camera(int id, String brand, String model, double rentalAmountPerDay) {
        this.id = id;
        this.brand = brand;
        this.model = model;
        this.rentalAmountPerDay = rentalAmountPerDay;
        this.rented = false; // Initialize status as not rented
    }

    public int getId() {
        return id;
    }

    public String getBrand() {
        return brand;
    }

    public String getModel() {
        return model;
    }

    public double getRentalAmountPerDay() {
        return rentalAmountPerDay;
    }

    public boolean isRented() {
        return rented;
    }

    public void setRented(boolean rented) {
        this.rented = rented;
    }
}

class Wallet {
    private double balance;

    public Wallet(double balance) {
        this.balance = balance;
    }

    public double getBalance() {
        return balance;
    }
}
```

```

    public void addMoney(double amount) {
        balance += amount;
    }

    public void deductMoney(double amount) {
        balance -= amount;
    }
}

public class CameraRentalApplication {
    private static List<Camera> cameraList = new ArrayList<>();
    private static Wallet wallet;
    private static boolean loggedIn = false;

    public static void main(String[] args) {
        displayWelcomeScreen();

        Scanner scanner = new Scanner(System.in);

        // Create some initial cameras
        cameraList.add(new Camera(1, "Nikon", "D5600", 25.0));
        cameraList.add(new Camera(2, "Canon", "EOS 80D", 30.0));
        cameraList.add(new Camera(3, "Sony", "A7 III", 35.0));

        int choice;
        do {
            displayMenu();
            choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline character

            switch (choice) {
                case 1:
                    if (login(scanner)) {
                        loggedIn = true;
                        System.out.println("Login successful.");
                    } else {
                        loggedIn = false;
                        System.out.println("Login failed. Please try again.");
                    }
                    break;
                case 2:
                    if (loggedIn) {
                        addCamera(scanner);
                    } else {
                        System.out.println("Please log in to access this feature.");
                    }
                    break;
                case 3:
                    if (loggedIn) {
                        removeCamera(scanner);
                    } else {
                        System.out.println("Please log in to access this feature.");
                    }
                    break;
            }
        } while (choice != 0);
    }
}

```

```

        case 4:
            if (loggedIn) {
                rentCamera(scanner);
            } else {
                System.out.println("Please log in to access this feature.");
            }
            break;
        case 5:
            if (loggedIn) {
                viewAllCameras(scanner, cameraList);
            } else {
                System.out.println("Please log in to access this feature.");
            }
            break;
        case 6:
            if (loggedIn) {
                manageWallet(scanner);
            } else {
                System.out.println("Please log in to access this feature.");
            }
            break;
        case 7:
            System.out.println("Thank you for using the Camera Rental
Application. Goodbye!");
            break;
        default:
            System.out.println("Invalid choice. Please try again.");
            break;
    }
} while (choice != 7);
}

private static boolean login(Scanner scanner) {
    System.out.print("Enter your username: ");
    String username = scanner.nextLine();
    System.out.print("Enter your password: ");
    String password = scanner.nextLine();
    return !username.isEmpty() && !password.isEmpty();
}

private static void removeCamera(Scanner scanner) {
    System.out.println("\nRemove Camera");

    if (cameraList.isEmpty()) {
        System.out.println("No cameras available to remove.");
        goBackToPreviousMenu(scanner);
    } else {
        System.out.println("Available Cameras:");
        for (int i = 0; i < cameraList.size(); i++) {
            Camera camera = cameraList.get(i);
            System.out.println((i + 1) + ". " + camera.getBrand() + " " +
camera.getModel());
        }

        System.out.print("Enter the number of the camera to remove: ");
    }
}

```

```

        int cameraNumber = scanner.nextInt();
        scanner.nextLine(); // Consume newline character

        if (cameraNumber >= 1 && cameraNumber <= cameraList.size()) {
            cameraList.remove(cameraNumber - 1);
            System.out.println("Camera removed successfully.");
        } else {
            System.out.println("Invalid camera number. Please try again.");
        }
        goBackToPreviousMenu(scanner);
    }
}

private static void rentCamera(Scanner scanner) {
    System.out.println("Available Cameras for Rent:\n");
    if (cameraList.isEmpty()) {
        System.out.println("No cameras available for rent at the moment.");
    } else {
        System.out.println("+-----+-----+-----+-----+");
        System.out.println("| ID | Brand | Model | Rental Amount |");
        System.out.println("+-----+-----+-----+-----+");

        for (Camera camera : cameraList) {
            System.out.printf("| %-3d | %-19s | %-20s | $%-12.2f |\n",
                camera.getId(), camera.getBrand(), camera.getModel(),
                camera.getRentalAmountPerDay());
        }

        System.out.println("+-----+-----+-----+-----+");

        System.out.print("Enter the Camera id of the camera to rent: ");
        int index = scanner.nextInt();

        if (index >= 1 && index <= cameraList.size()) {
            Camera selectedCamera = cameraList.get(index - 1);

            if (selectedCamera.isRented()) {
                System.out.println("Camera is already rented. Please select another camera.");
            } else {
                double rentalAmount = selectedCamera.getRentalAmountPerDay();

                if (wallet == null) {
                    System.out.println("Failed due to insufficient balance in wallet.");
                    System.out.println("Please deposit an amount to the wallet.");
                } else if (wallet.getBalance() < rentalAmount) {
                    System.out.println("Not enough money in your wallet to rent a camera.");
                }
            }
        }
    }
}

```

```

        System.out.println("Please deposit more money to your
wallet.");
    } else {
        wallet.deductMoney(rentalAmount);
        selectedCamera.setRented(true);
        System.out.println("Camera " + selectedCamera.getBrand() + "
" + selectedCamera.getModel() + " rented successfully.");
        System.out.println("Your transaction with rent of Rs " +
rentalAmount + " has successfully completed");
    }
}
} else {
    System.out.println("Invalid index. No camera rented.");
}
goBackToPreviousMenu(scanner);
}
}

private static void goBackToPreviousMenu(Scanner scanner) {
    System.out.println("Press enter to go back to the previous menu...");
    scanner.nextLine(); // Wait for the user to press enter
    scanner.nextLine(); // Consume the newline character
    displayMenu(); // Display the main menu again
}

private static void viewAllCameras(Scanner scanner, List<Camera> cameras) {
    System.out.println("----- View All Cameras -----");

    if (cameras.isEmpty()) {
        System.out.println("No cameras available at the moment.");
    } else {
        System.out.println("ID\tBrand\tModel\tRental Amount\tStatus");
        System.out.println("-----");
        for (Camera camera : cameras) {
            String status = camera.isRented() ? "Rented" : "Available";
            System.out.printf("%d\t%s\t%s\t$%.2f\t\t%s\n", camera.getId(),
camera.getBrand(), camera.getModel(), camera.getRentalAmountPerDay(), status);
        }

        System.out.println();
        goBackToPreviousMenu(scanner);
    }
}

private static void manageWallet(Scanner scanner) {
    boolean exit = false;
    while (!exit) {
        System.out.println("----- Manage Wallet -----");
        System.out.println("1. Add money to wallet");
        System.out.println("2. View wallet amount");
        System.out.println("3. Go back to main menu");
        System.out.print("Enter your choice: ");
        int choice = scanner.nextInt();

        switch (choice) {
            case 1:
                addMoneyToWallet(scanner);

```

```

        break;
    case 2:
        viewWalletAmount();
        break;
    case 3:
        exit = true;
        break;
    default:
        System.out.println("Invalid choice. Please try again.");
    }

    System.out.println();
}
displayMenu();
}

private static void addMoneyToWallet(Scanner scanner) {
    System.out.print("Enter the amount to deposit: ");
    double depositAmount = scanner.nextDouble();

    if (wallet == null) {
        wallet = new Wallet(depositAmount);
    } else {
        wallet.addMoney(depositAmount);
    }

    System.out.println("Amount deposited successfully.");
}

private static void viewWalletAmount() {
    if (wallet != null) {
        System.out.println("Current wallet balance: $" + wallet.getBalance());
    } else {
        System.out.println("Wallet not initialized.");
    }
}

private static void displayWelcomeScreen() {
    System.out.println("-----");
    System.out.println("                Camera Rental Application");
    System.out.println("                Developed by Syed Ariz Haider");
    System.out.println("-----");
    -\n");
    System.out.println("                Welcome to the Camera Rental Application!\n");
    System.out.println("-----");
    -\n");
}

private static void displayMenu() {
    System.out.println("1. Log in");
    System.out.println("2. Add camera");
    System.out.println("3. Remove camera");
    System.out.println("4. Rent a camera");
    System.out.println("5. View all cameras");
}

```

```

        System.out.println("6. My wallet or Add money");
        System.out.println("7. Close the application");
        System.out.print("\nEnter your choice: ");
    }

    private static void addCamera(Scanner scanner) {
        System.out.println("\nAdd Camera");

        boolean addAnotherCamera = true;
        do {
            System.out.print("Enter the brand: ");
            String brand = scanner.nextLine();
            System.out.print("Enter the model: ");
            String model = scanner.nextLine();
            System.out.print("Enter the rental amount per day: $");
            double rentalAmountPerDay = scanner.nextDouble();
            scanner.nextLine(); // Consume newline character

            int id = cameraList.size() + 1; // Generate a unique id for the camera

            Camera camera = new Camera(id, brand, model, rentalAmountPerDay);
            cameraList.add(camera);
            System.out.println("Camera added successfully.");

            System.out.print("Do you want to add another camera? (yes/no): ");
            String choice = scanner.nextLine();

            if (choice.equalsIgnoreCase("no")) {
                addAnotherCamera = false;
            }
        } while (addAnotherCamera);

        goBackToPreviousMenu(scanner);
    }
}

```