Car Import Java Program

Charles Kent Labrador  
Ivan Bryll Joseco  
Genevieve Herera

Car Import (Main Method)

package JavaProject;

import java.util.Scanner;

public class CarImport {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        Menu menu = new Menu();

        Cars cars = new Cars();

        PaymentMethods paymentMethods = new PaymentMethods();

        System.out.println("Welcome to Luxury Cars Import \n");

        menu.Option();

       int userinput = input.nextInt();

        switch (userinput) {

            case 1:

                cars.CheckImports();

                menu.Purchase();

                break;

            case 2:

                menu.CheckCashBalance();

                break;

            case 3:

                menu.CheckCreditBalance();

                break;

            default:

            System.out.println("Invalid Input"+"\n");

            main(null);

                break;

        }

    }

}

Cars (Class)

package JavaProject;

import java.util.ArrayList;

public class Cars {

    private String brand;

    private String model;

    private double price;

    private String country;

    Cars[] cars = new Cars[9];

    public Cars(){

    cars[0] = new Cars("Toyota", "Camry", 35390.0, "U.S");

    cars[1] = new Cars("Honda", "Civic", 23950.0, "Japan");

    cars[2] = new Cars("Ford", "Mustang", 36445.0, "U.S");

    cars[3] = new Cars("Toyota", "Supra", 66370.0, "Japan");

    cars[4] = new Cars("Porsche", "911 Carrera GTS", 559400.0, "Germany");

    cars[5] = new Cars("NISSAN", "GTR R34", 116641., "Japan");

    cars[6] = new Cars("Toyota", "Trueno AE86", 20000.0, "Japan");

    cars[7] = new Cars("Mazda", "RX-7 FD", 43158.0, "Japan");

    cars[8] = new Cars("Honda", "NSX", 169500.0, "U.S");

}

    void CheckImports(){

        System.out.println("Here are the available imports:");

        for(int i = 0; i < cars.length; i++){

            System.out.println("Car " + (i+1) + ":");

            System.out.println("Make: " + cars[i].getBrand());

            System.out.println("Model: " + cars[i].getModel());

            System.out.println("Price: $" + cars[i].getPrice());

            System.out.println("Country: " + cars[i].getCountry());

            System.out.println();

        }

    }

    // Constructor

    public Cars(String brand, String model, double price, String country) {

        this.brand = brand;

        this.model = model;

        this.price = price;

        this.country = country;

    }

    // Getters and setters

    //set and get for (brand)

    public String getBrand() {

        return brand;

    }

    public String getBrand(int i){

        return cars[i].getBrand();

    }

    public void setBrand(String brand) {

        this.brand = brand;

    }

    //set and get for (model)

    public String getModel() {

        return model;

    }

    public String getModel(int i){

        return cars[i].getModel();

    }

    public void setModel(String model) {

        this.model = model;

    }

    //set and get for (price)

    public double getPrice() {

        return price;

    }

    public double getPrice(int i){

        return cars[i].getPrice();

    }

    public void setPrice(double price) {

        this.price = price;

    }

    //set and get for (country)

    public String getCountry() {

        return country;

    }

    public String getCountry(int i){

        return cars[i].getCountry();

    }

    public void setCountry(String country) {

        this.country = country;

}}

Menu (Class)

package JavaProject;

import java.util.Scanner;

public class Menu {

    Scanner input = new Scanner(System.in);

    Cars cars = new Cars();

    PaymentMethods paymentMethods = new PaymentMethods();

    void returnMainMenu(){

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

        System.out.println("Would you like to return to the main menu?");

        System.out.println("Yes / No");

        char choice = input.next().toLowerCase().charAt(0);

        if(choice == 'y'){

            CarImport.main(null);

        }

       if (choice =='n') {

        System.exit(0);

       }

       else{

            System.out.println("Invalid Input");

            returnMainMenu();

       }

    }

    void Option(){

       System.out.println("[1] Check Imports");

       System.out.println("[2] Check Balance");

       System.out.println("[3] Check Credit");

    }

    void cashBalance(String username, String password){

        if(paymentMethods.checkAccount(username, password)){

             paymentMethods.getBalanceCash(username);

        }

        else{

            System.out.println("Username and Password Error \n");

        }

    }

    void creditBalance(String username, String password){

        if(paymentMethods.checkAccount(username, password)){

             paymentMethods.getBalanceCredit(username);

        }

        else{

            System.out.println("Username and Password Error \n");

        }

    }

    void CheckCashBalance(){

        System.out.print("Enter Username: ");

        String username = input.nextLine();

        input.nextLine();

        System.out.print("Enter Password: ");

        String password = input.nextLine();

        if(paymentMethods.checkAccount(username, password) == false){

            System.out.println("Incorrect Username and Password");

        }

        else{

            System.out.println(username +"'s current cash balance is " + paymentMethods.getBalanceCash(username));

            returnMainMenu();

        }

            returnMainMenu();

    }

    void CheckCreditBalance(){

        System.out.print("Enter Username: ");

            String username = input.nextLine();

            input.nextLine();

            System.out.print("Enter Password: ");

            String password = input.nextLine();

            if(paymentMethods.checkAccount(username, password) == false){

                System.out.println("Incorrect Username and Password");

            }

            else{

                System.out.println(username +"'s current credit balance is " + paymentMethods.getBalanceCash(username));

                returnMainMenu();

            }

            returnMainMenu();

    }

    void Purchase(){

        System.out.println("Select car to purchase: ");

        int purchasenum = input.nextInt();

        System.out.println("Are you sure to purchase \n" + cars.getBrand(purchasenum - 1)

                           +" \nModel: " + cars.getModel(purchasenum - 1 )

                            + "\nPrice: $" +  cars.getPrice(purchasenum - 1) +"\n");

        System.out.println("Enter [ 1 ] to purchase / [ 2 ] to go back to car list ");

        int ans = input.nextInt();

        if(ans == 1){

            System.out.print("Enter Username: ");

            input.nextLine();

            String username = input.nextLine();

            System.out.print("Enter Password: ");

            String password = input.nextLine();

            if(paymentMethods.checkAccount(username, password) == false){

                System.out.println("Incorrect Username and Password");

                Purchase();

            }

            System.out.println("-----------\n Enter payment method: [1] Cash [2] Credit");

            int cash\_credit = input.nextInt();

                if(cash\_credit == 1){

                System.out.println("Welcome " + username + " Your balance is " + paymentMethods.getBalanceCash(username));

                double tax = cars.getPrice(purchasenum - 1) \* 0.30;

                double total = (cars.getPrice(purchasenum - 1) + tax);

                System.out.println("\nInvoice \n --------- \n Price: " + cars.getPrice(purchasenum - 1)+ "\n Tax: " + tax + "\n");

                System.out.println("-------------\n Total: " + total );

                System.out.println(paymentMethods.getBalanceCash(username) + " will be deducted " + total);

                if(paymentMethods.canAfford(paymentMethods.getBalanceCash(username), total) == false){

                    System.out.println(username + "'s account has insuficient cash for this purchase. Please try again\n-------------------------------");

                    Purchase();

                }

                System.out.print("\n Type Yes to confirm:");

                char confirm = input.next().toLowerCase().charAt(0);

                if(confirm == 'y'){

                    System.out.println("Order confirmed: \n ---------------------");

                    System.out.println( cars.getBrand(purchasenum - 1) +" " + cars.getModel(purchasenum - 1 ) + "is now being delivered");

                    double newBalance = paymentMethods.getBalanceCash(username) - total;

                    System.out.println(username + "'s new cash balance is $" + newBalance);

                    returnMainMenu();

                }

            }

                 if (cash\_credit == 2) {

                System.out.println("Welcome " + username + " Your balance is " + paymentMethods.getBalanceCredit(username));

                double tax = cars.getPrice(purchasenum - 1) \* 0.30;

                double total = (cars.getPrice(purchasenum - 1) + tax);

                System.out.println("Invoice \n --------- \n Price: " + cars.getPrice(purchasenum - 1)+ "\n Tax: " + tax + "\n");

                System.out.println("-------------\n Total: " + total );

                System.out.println(paymentMethods.getBalanceCredit(username) + " will be deducted " + total);

                System.out.print("\n Type Yes to confirm:");

                char confirm = input.next().toLowerCase().charAt(0);

                if(confirm == 'y'){

                    System.out.println("Order confirmed: \n ---------------------");

                    System.out.println( cars.getBrand(purchasenum - 1) +" " + cars.getModel(purchasenum - 1 ) + " is now being delivered\n--------------");

                    double newBalance = paymentMethods.getBalanceCredit(username) - total;

                    System.out.println(username + "'s new credit balance is $" + newBalance);

                    returnMainMenu();

                }

            }

                    else{

                        System.out.println("Input Invalid Try again");

                        Purchase();

            }

            }

                else{

                Purchase();

                }

        }

    }

PaymentMethods (Class)

package JavaProject;

import java.util.ArrayList;

import java.util.Scanner;

public class PaymentMethods {

   private double balance;

    private ArrayList<String> accounts;

    private ArrayList<String> accountpass;

    private ArrayList<Double> cash;

    private ArrayList<Double> credit;

    public PaymentMethods(){

        accounts = new ArrayList<>();

        accounts.add("Ivan Bryll Joseco");

        accounts.add("Charles Kent");

        accounts.add("Ian Barrientos ");

        accounts.add("Gryxdane Maldo");

        accounts.add("Jericho Diezma");

        accounts.add("Kurt Iax Limos");

        accounts.add("Ismael Baguilar");

        accounts.add("Joven Rafael");

        accounts.add("Adie Saludares");

        accountpass = new ArrayList<>();

        accountpass.add("ivan");

        accountpass.add("ken");

        accountpass.add("ian");

        accountpass.add("gryx");

        accountpass.add("jeri");

        accountpass.add("kurt");

        accountpass.add("mael");

        accountpass.add("joven");

        accountpass.add("adie");

        cash= new ArrayList<>();

        cash.add(237420.0); //1

        cash.add(700000.0);

        cash.add(803920.0);

        cash.add(8923000.0);

        cash.add(1683200.0);

        cash.add(9820.0);

        cash.add(1000000.0);

        cash.add(7892300.0);

        cash.add(5000.0);

        cash.add(700.0);//10

        credit = new ArrayList<>();

        credit.add(1000000.0);//1

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);

        credit.add(1000000.0);//10

    }

    public boolean checkAccount(String username, String password){

        for (int i = 0; i < accounts.size(); i++) {

            if (username.equals(accounts.get(i)) && password.equals(accountpass.get(i))) {

                return true;

            }

        }

        return false;

    }

    public double getBalanceCash(String username){

        if(accounts.contains(username) == true){

            int index = accounts.indexOf(username);

            balance = cash.get(index);

        }

        return balance;

    }

    public double getBalanceCredit(String username){

        if(accounts.contains(username) == true){

            int index = accounts.indexOf(username);

            balance = credit.get(index);

        }

        return balance;

    }

    public boolean canAfford(double cash, double payment){

        if(cash > payment){

            return false;

        }

        else{

            return true;

        }

    }

}