الاسم : نور الدين محمود عبد البصير

ID :23011598

1-Product

```java
public class Product {
    protected  int productId;
    protected String nameProduct;
    protected double productPrice;


    public Product() {
    }

    public Product(int productId, String nameProduct, double productPrice) {
        this.productId = Math.abs(a: productId);
        this.nameProduct = nameProduct;
        this.productPrice = Math.abs(a: productPrice);
    }

    public int getProductId() {
        return productId;
    }

    public void setProductId(int productId) {
        this.productId = Math.abs(a: productId);
    }

    public String getNameProduct() {
        return nameProduct;
    }

```

```java
    public void setNameProduct(String nameProduct) {
        this.nameProduct = nameProduct;
    }

    public double getProductPrice() {
        return productPrice;
    }

    public void setProductPrice(double productPrice) {
        this.productPrice = Math.abs(a: productPrice);
    }
}
```

2- ElectronicProduct

```java
public class ElectronicProduct extends Product {
    private String brand;
    private int warrantyPeriod;


    public ElectronicProduct(int productId, String nameProduct, double productPrice, String brand, int warrantyPeriod) {
        super(productId, nameProduct, productPrice);
        this.brand = brand;
        this.warrantyPeriod = warrantyPeriod;
    }

    public String getBrand() {
        return brand;
    }

    public void setBrand(String brand) {
        this.brand = brand;
    }

    public int getWarrantyPeriod() {
        return warrantyPeriod;
    }

    public void setWarrantyPeriod(int warrantyPeriod) {
        this.warrantyPeriod = Math.abs(a: warrantyPeriod);
    }

```

```java
    @Override
    public String toString() {
        return "ElectronicProduct{" +
                "Brand: '" + brand + '\'' +
                ", WarrantyPeriod: " + warrantyPeriod +
                ", ProductId: " + productId +
                ", NameProduct: '" + nameProduct + '\'' +
                ", ProductPrice: " + productPrice +
                '}';
    }
}
```

3- ClothingProduct

```
public class ClothingProduct extends Product {
    private String size;
    private String fabric;
    public ClothingProduct(int productId, String nameProduct, double productPrice, String size, String fabric) {
        super(productId, nameProduct, productPrice);
        this.size = size;
        this.fabric = fabric;
    }

    public String getSize() {
        return size;
    }

    public void setSize(String size) {
        this.size = size;
    }

    public String getFabric() {
        return fabric;
    }

    public void setFabric(String fabric) {
        this.fabric = fabric;
    }

```

```
    @Override
    public String toString() {
        return "ClothingProduct{" +
                "Size: '" + size + '\'' +
                ", Fabric: '" + fabric + '\'' +
                ", ProductId: " + productId +
                ", NameProduct: '" + nameProduct + '\'' +
                ", ProductPrice: " + productPrice +
                '}';
    }
}
```

4- BookProduct

```java

public class BookProduct extends Product {
    private String author;
    private String publisher;

    public BookProduct(int productId, String nameProduct, double productPrice, String author, String publisher) {
        super(productId, nameProduct, productPrice);
        this.author = author;
        this.publisher = publisher;
    }
    public String getAuthor() {
        return author;
    }

    public void setAuthor(String author) {
        this.author = author;
    }

    public String getPublisher() {
        return publisher;
    }

    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }

```

```java

    @Override
    public String toString() {
        return "BookProduct{" +
                "Author: '" + author + '\'' +
                ", Publisher: '" + publisher + '\'' +
                ", ProductId: " + productId +
                ", NameProduct: '" + nameProduct + '\'' +
                ", ProductPrice: " + productPrice +
                '}';
    }
}

```

5- Customer

```java

public class Customer {
    protected int customerId;
    private String customerName;
    private String customerAddress;

    public Customer() {
    }


    public Customer(int customerId, String customerName, String customerAddress) {
        this.customerId = Math.abs(a: customerId);
        this.customerName = customerName;
        this.customerAddress = customerAddress;
    }

    public int getCustomerId() {
        return customerId;
    }

    public void setCustomerId(int customerId) {
        this.customerId = Math.abs(a: customerId);
    }

    public String getCustomerName() {
        return customerName;
    }

```

```
30                  Back (Alt+NumPad-Left)
31      public double getTotalPrice() {
32          Cart cart = new Cart();
33          this.totalPrice = cart.calculatePrice(products);
34          return totalPrice;
35      }
36
37      public int getCustomerId() {
38          return customerId;
39      }
40
41      public void setCustomerId(int customerId) {
42          this.customerId = customerId;
43      }
44
45      public Product[] getProducts() {
46          return products;
47      }
48
49      public void setProducts(Product[] products) {
50          this.products = products;
51      }
52
```

6- Cart

```java
public class Cart {
    private int numberOfProducts;
    private Product[] products;
    private int customerId;

    public Cart() {
    }

    public Cart(int customerId, int numberOfProducts) {
        this.customerId = Math.abs(a: customerId);
        this.numberOfProducts = Math.abs(a: numberOfProducts);
        this.products = new Product[numberOfProducts];
    }

    public Product[] getProducts() {
        return products;
    }

    public void setProducts(Product[] products) {
        this.products = products;
    }

    public int getNumberOfProducts() {
        return numberOfProducts;
    }

    public void setNumberOfProducts(int numberOfProducts) {
        this.numberOfProducts = numberOfProducts;
        products = new Product[numberOfProducts];
```

```
    }

    public int getCustomerId() {
        return customerId;
    }

    public void setCustomerId(int customerId) {
        this.customerId = customerId;
    }

    public Product[] addProduct(Product product, int index) {
        if (products[index] == null)
            this.products[index] = product;
        return products;
    }

    public Product[] removeProduct(int indexProductToDelete, Product[] product) {
        Product[] newProducts = new Product[product.length - 1];

        if (indexProductToDelete >= 0 && indexProductToDelete <= product.length) {
            System.arraycopy(src: product, srcPos: 0, dest: newProducts, destPos: 0, length: indexProductToDelete);

            if (numberOfProducts - (indexProductToDelete + 1) >= 0)
                System.arraycopy(src: product, indexProductToDelete + 1, dest: newProducts, indexProductToDelete + 1 - 1, product.length - (indexProductToDelete + 1));
        } else {
            System.out.println(x: "We can't found this in the cart!");
        }
        this.products = newProducts;
        return products;
    }

    public double calculatePrice(Product[] products) {
        double total = 0;
        for (Product product1 : products) {
            total += product1.getProductPrice();
        }
        return total;
    }

    public boolean placeOrder(String check) {
        if (check.equals(anObject: "Yes") || check.equals(anObject: "yes") || check.equals(anObject: "Y") || check.equals(anObject: "y"))
            return true;
        else if (check.equals(anObject: "No") || check.equals(anObject: "no") || check.equals(anObject: "N") || check.equals(anObject: "n"))
            return false;
        else
            System.out.println(x: "Invalid Input!");
        return false;
    }

    public void printProducts() {
        for (int i = 0; i < numberOfProducts; i++) {
            System.out.println((i + 1) + "- " + products[i].getNameProduct() + " and the price is: " + products[i].getProductPrice() + " $");

        }
    }
}
```

7- Order

```java

public class Order {
    private int orderId;  // there is more work on it
    private double totalPrice;
    private int customerId;
    private Product[] products;

    public Order() {

    }

    public Order(int customerId, Product[] products, int orderId) {
        this.products = products;
        this.customerId = customerId;
        this.orderId = orderId;
    }

    public void setOrderId(int orderId) {
        this.orderId = orderId;
    }

    public int getOrderId() {
        return orderId;
    }

    public void setTotalPrice(double totalPrice) {
        this.totalPrice = totalPrice;
    }

```

```
                Back (Alt+NumPad-Left)
    public double getTotalPrice() {
        Cart cart = new Cart();
        this.totalPrice = cart.calculatePrice(products);
        return totalPrice;
    }

    public int getCustomerId() {
        return customerId;
    }

    public void setCustomerId(int customerId) {
        this.customerId = customerId;
    }

    public Product[] getProducts() {
        return products;
    }

    public void setProducts(Product[] products) {
        this.products = products;
    }

public void printOrderInfo() {
    System.out.println(x: "====Here's your order's summary==== ");
    System.out.println("Customer ID: " + customerId);
    System.out.println("Order ID: " + getOrderId());

    System.out.println(x: "========================================");

    System.out.println(x: "##### Products #####");
    for (Product product : products) {
        if (product != null)
            System.out.println(product.getNameProduct() + " -> " + product.getProductPrice() + " $.");
    }

    System.out.println(x: "========================================");

    System.out.println("Total Price: " + getTotalPrice() + " $.");

    System.out.println(x: "========================================");

}

}
```

8- EcommerceSystem

```java
import java.util.Scanner;

public class EcommerceSystem {
    public static void welcomeMessage() {
        System.out.println("Welcome to our E-Commerce!");
    }


    public static void main(String[] args) {
        Scanner input = new Scanner(source: System.in);

        Product product1 = new ElectronicProduct(productId: 1, nameProduct: "smart phone", productPrice: 599.9, brand: "Samsung", warrantyPeriod: 1);
        Product product2 = new ClothingProduct(productId: 2, nameProduct: "T-shirt", productPrice: 19.99, size: "Medium", fabric: "Cotton");
        Product product3 = new BookProduct(productId: 3, nameProduct: "OOP", productPrice: 39.99, author: "O'Reilly", publisher: "X Publications");

        Cart cart1 = new Cart();

        welcomeMessage();
```

```java
        System.out.print("Please enter your ID: ");
        int userId = input.nextInt();

        input.nextLine();

        System.out.print("Please enter your Name: ");
        String userName = input.nextLine();

        System.out.print("Please enter your address: ");
        String userAddress = input.nextLine();

        System.out.println("How many products you want to add to you cart?");
        int numberOfProducts = input.nextInt();

        Customer customer1 = new Customer(customerId: userId, customerName: userName, customerAddress:userAddress);

        cart1.setNumberOfProducts(numberOfProducts);

        System.out.println("Choose form this list:- ");
        System.out.println(
                x: """
                        1- smart phone.
                        2- T-shirt.
                        3- Book.
                        """
        );
```

```
49          for (int i = 0; i < numberOfProducts; i++) {
50              System.out.println(x: "Enter your key products to put in the cart: ");
51              int productKey = input.nextInt();
52              switch (productKey) {
53                  case 1:
54                      cart1.addProduct(product:product1, index: i);
55                      System.out.println(x: "The product added.\n");
56                      break;
57
58                  case 2:
59                      cart1.addProduct(product:product2, index: i);
60                      System.out.println(x: "The product added.\n");
61                      break;
62                  case 3:
63                      cart1.addProduct(product:product3, index: i);
64                      System.out.println(x: "The product added.\n");
65                      break;
66                  default:
67                      System.out.println(x: "Invalid input!");
68                      break;
69              }
70          }
71          input.nextLine();
72
73          System.out.println("Your total price is " + cart1.calculatePrice(products:cart1.getProducts()) + " $");
74
75          System.out.println(x: "Do you want to place order? (yes/no)");
76          String userCheck = input.nextLine();
```

```
77
78          if (cart1.placeOrder(check: userCheck)) {
79              Order order = new Order(customerId: userId, products:cart1.getProducts(), orderId:1);
80              order.printOrderInfo();
81          } else if (!cart1.placeOrder(check: userCheck) && userCheck.equals(anObject:"No") || userCheck.equals(anObject:"no") || userCheck.equals(anObject:"N")
82              System.out.println(x: "Do you want to remove a product from the cart or cancel the order? [yes (remove) / no (cancel)]");
83              userCheck = input.nextLine();
84              if (userCheck.equals(anObject:"Yes") || userCheck.equals(anObject:"yes") || userCheck.equals(anObject:"Y") || userCheck.equals(anObject:"y")) {
85                  System.out.println(x: "Here your cart: ");
86                  cart1.printProducts();
87                  do {
88                      System.out.println(x: "Which product do you want to remove? ");
89                      int userProductIndex = input.nextInt();
90
91                      cart1.removeProduct((userProductIndex - 1), product:cart1.getProducts());
92
93                      Order order = new Order(customerId: userId, products:cart1.getProducts(), orderId:1);
94                      order.printOrderInfo();
95
96                      System.out.println();
97
98                      input.nextLine();
99                      System.out.println(x: "Do want to remove any thing else from the cart? ");
100                     userCheck = input.nextLine();
101
```

```
101
102             } while (userCheck.equals(anObject:"Yes") || userCheck.equals(anObject:"yes") || userCheck.equals(anObject:"Y") || userCheck.equals(anObject:"y"));
103
104             Order order = new Order(customerId: userId, products:cart1.getProducts(), orderId:1);
105
106             System.out.println();
107
108             order.printOrderInfo();
109
110         } else if (userCheck.equals(anObject:"No") || userCheck.equals(anObject:"no") || userCheck.equals(anObject:"N") || userCheck.equals(anObject:"n")) {
111             System.out.println(x: "The order is canceled.");
112         }
113     }
114     System.out.println(x: "Thank you for use our E-commerce system.");
115 }
116
```

Output :

```
Welcome to our E-Commerce!
Please enter your ID: 23011598
Please enter your Name: Nour ElDeen Mahmoud
Please enter your address: Address
How many products you want to add to you cart?
3
Choose form this list:-
1- smart phone.
2- T-shirt.
3- Book.

Enter your key products to put in the cart:
2
The product added.

Enter your key products to put in the cart:
1
The product added.

Enter your key products to put in the cart:
3
The product added.

Your total price is 659.88 $
Do you want to place order? (yes/no)
yes
====Here's your order's summary====
```

```
The product added.

Your total price is 659.88 $
Do you want to place order? (yes/no)
yes
====Here's your order's summary====
Customer ID: 23011598
Order ID: 1
==========================================
##### Products #####
T-shirt -> 19.99 $.
smart phone -> 599.9 $.
OOP -> 39.99 $.
==========================================
Total Price: 659.88 $.
==========================================
Thank you for use our E-commerce system.
BUILD SUCCESSFUL (total time: 52 seconds)
```