

```

class product{
    3 usages
    private Integer product_id ;
    3 usages
    private String name ;
    3 usages
    private Float Price ;

    3 usages
    public product(Integer product_id, String name, Float price) {
        this.product_id = Math.abs(product_id);
        this.name = name;
        this.Price = Math.abs(price);
    }

    no usages
    public void setPrice(Float price) {
        Price = Math.abs(price);
    }

    no usages
    public void setName(String name) {
        this.name = name;
    }

    no usages
    public void setProduct_id(Integer product_id) {
        this.product_id = Math.abs(product_id);
    }

    4 usages
    public String getName() {
        return name;
    }

    no usages
    public Integer getProduct_id() {
        return product_id;
    }
}

```

```

class ElectronicProduct extends product{
    3 usages
    private String brand ;
    3 usages
    private Integer warrantyPeriod ;

    2 usages
    public ElectronicProduct(Integer product_id, String name, Float price, String brand, Integer warrantyPeriod) {
        super(product_id, name, price);
        this.brand = brand;
        this.warrantyPeriod = Math.abs(warrantyPeriod);
    }

    no usages
    public void setBrand(String brand) {
        this.brand = brand;
    }

    no usages
    public void setWarrantyPeriod(Integer warrantyPeriod) {
        this.warrantyPeriod = Math.abs(warrantyPeriod);
    }

    no usages
    public Integer getWarrantyPeriod() {
        return warrantyPeriod;
    }

    no usages
    public String getBrand() {
        return brand;
    }
}

```

```

class ClothingProduct extends product {
    3 usages
    private String size ;
    3 usages
    private String fabric;

    2 usages
    public ClothingProduct(Integer product_id, String name, Float price, String size, String fabric) {
        super(product_id, name, price);
        this.size = size;
        this.fabric = fabric;
    }

    no usages
    public void setFabric(String fabric) {
        this.fabric = fabric;
    }

    no usages
    public void setSize(String size) {
        this.size = size;
    }

    no usages
    public String getFabric() {
        return fabric;
    }

    no usages
    public String getSize() {
        return size;
    }
}
4 usages

```

```

88 class BookProduct extends product{
89     3 usages
90     private String author ;
91     3 usages
92     private String publisher ;
93
94     2 usages
95     public BookProduct(Integer product_id, String name, Float price, String author, String publisher) {
96         super(product_id, name, price);
97         this.author = author;
98         this.publisher = publisher;
99     }
100
101     no usages
102     public String getAuthor() {
103         return author;
104     }
105
106     no usages
107     public String getPublisher() {
108         return publisher;
109     }
110
111     no usages
112     public void setAuthor(String author) {
113         this.author = author;
114     }
115
116     no usages
117     public void setPublisher(String publisher) {
118         this.publisher = publisher;
119     }
120 }

```

```

public class cart {
    3 usages
    private Integer customerId ;
    3 usages
    private Integer nproducts ;
    12 usages
    product [] products ;
    1 usage
    public cart(Integer customerId, Integer nproducts, product[] products) {
        this.customerId = Math.abs(customerId);
        this.nproducts = Math.abs(nproducts);
        this.products = products;
    }
    no usages
    public Integer getCustomerId() {
        return customerId;
    }
    no usages
    public Integer getNproducts() {
        return nproducts;
    }
    no usages
    public product[] getProducts() {
        return products;
    }
    no usages
    public void setCustomerId(Integer customerId) {
        this.customerId = Math.abs(customerId);
    }
    no usages
    public void setNproducts(Integer nproducts) {
        this.nproducts = Math.abs(nproducts);
    }
    no usages
    public void setProducts(product[] products) {

```

Problems Terminal Services Build

```

25
no usages
26 public void setNproducts(Integer nproducts) {
27     this.nproducts = Math.abs(nproducts);
28 }
29
no usages
30 public void setProducts(product[] products) {
31     this.products = products;
32 }
33
no usages
34 public void addProduct(product product) {
35     for (int i = 0; i < products.length; i++) {
36         if (products[i] == null) {
37             products[i] = product;
38             return;
39         }
40     }
41
no usages
42 public void removeProduct(product xproduct) {
43     for (int i = 0; i < products.length; i++) {
44         if (products[i] != null && products[i].equals(xproduct)) {
45             products[i] = null;
46             System.out.println("removed product from cart: " + xproduct.getName());
47             return; }
48     System.out.println("product is not there : " + xproduct.getName());
49 }
50
no usages
51 public double getPrice(){
52     double total=0;
53     for (int i = 0; i < products.length; i++) {
54         total = total + products[i].getPrice();
55     }
56     return total;
57 }
58

```

```





1  public class order {
    4 usages
2      private Integer customerId;
    4 usages
3      private Integer orderId ;
    4 usages
4      product[] products ;
    4 usages
5      float totalprice ;
    1 usage
6      public order(Integer customerId, Integer orderId, product[] products, float totalprice) {
7          this.customerId = Math.abs(customerId);
8          this.orderId = Math.abs(orderId);
9          this.products = products;
10         this.totalprice = Math.abs(totalprice);
11     }
    no usages
12     public Integer getCustomerId() {
13         return customerId;
14     }
15
    no usages
16     public product[] getProducts() {
17         return products;
18     }
19
    no usages
20     public float getTotalprice() {
21         return totalprice;
22     }
23
    no usages
24     public Integer getOrderId() {
25         return orderId;
26     }
27
    no usages
28     public void setCustomerId(Integer customerId) {
29         this.customerId = Math.abs(customerId);
30     }

```

```

1  import javax.swing.*;
2  import java.awt.event.ActionEvent;
3  import java.awt.event.ActionListener;
4
5  ▶ public class EcommerceSystemUi extends JFrame implements ActionListener {
6      3 usages
7      private JTextField customerIdField;
8      3 usages
9      private JTextField nameField;
10     3 usages
11     private JTextField addressField;
12     3 usages
13     private JTextField numProductsField;
14     3 usages
15     private JButton submitButton;
16
17     1 usage
18     public EcommerceSystemUi() {
19         setTitle("Ecommerce System");
20         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
21         setSize( width: 400, height: 300);
22         setLocationRelativeTo(null);
23
24         JPanel panel = new JPanel();
25         panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));
26
27         panel.add(new JLabel( text: "Customer ID:"));
28         customerIdField = new JTextField();
29         panel.add(customerIdField);
30
31         panel.add(new JLabel( text: "Name:"));
32         nameField = new JTextField();
33         panel.add(nameField);
34
35         panel.add(new JLabel( text: "address:"));
36         addressField = new JTextField();
37         panel.add(addressField);
38
39         panel.add(new JLabel( text: "Number of Products:"));
40         numProductsField = new JTextField();
41         panel.add(numProductsField);

```

 Problems
  Terminal
  Services
  Build

```

1 public class customer {
2     private Integer customerId ;
3     private String name ;
4
5     private String address ;
6
7     public customer(Integer customerId, String name, String address) {
8         this.customerId = Math.abs(customerId);
9         this.name = name;
10        this.address = address;
11    }
12
13    public String getName() {
14        return name;
15    }
16
17    public String getAddress() {
18        return address;
19    }
20
21    public Integer getCustomerId() {
22        return customerId;
23    }
24
25    public void setName(String name) {
26        this.name = name;
27    }
28
29    public void setAddress(String address) {
30        this.address = address;
31    }

```

```

1 import java.util.Scanner;
2
3 public class EcommerceSystem {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         System.out.println("Welcome to the Ecommerce System!");
8
9         ElectronicProduct smartphone = new ElectronicProduct( product_id: 1, name: "smartphone", price: 599.9f, brand: "Samsung", warrantyPeriod: 1);
10
11        ClothingProduct tShirt = new ClothingProduct( product_id: 2, name: "T-shirt", price: 19.99f, size: "Medium", fabric: "Cotton");
12
13        BookProduct oopBook = new BookProduct( product_id: 3, name: "OOP", price: 39.99f, author: "O'Reilly", publisher: "X Publications");
14
15        System.out.print("Enter your ID: ");
16        int customerId = scanner.nextInt();
17        scanner.nextLine();
18        System.out.print("please enter your name: ");
19        String name = scanner.nextLine();
20        System.out.print("please Enter your address: ");
21        String address = scanner.nextLine();
22        customer customer = new customer(customerId, name, address);
23
24        System.out.print("How many products do you want to order? ");
25        int numProducts = scanner.nextInt();
26        scanner.nextLine();
27        product[] products = new product[numProducts];
28        for (int i = 0; i < numProducts; i++) {
29            System.out.println("which product do you want to add " + (i + 1) + ": " + " 1-smartphone 2-T-shirt 3-OOP");
30            int productId = scanner.nextInt();
31            scanner.nextLine();
32            switch (productId) {
33                case 1:
34                    products[i] = smartphone;
35                    break;
36                case 2:
37                    products[i] = tShirt;
38                    break;
39                case 3:
40                    products[i] = oopBook;
41                    break;

```

```
private JTextField customerIdField;  
private JTextField nameField;  
private JTextField addressField;  
private JTextField numProductsField;  
private JButton submitButton;  
  
public EcommerceSystemUi() {  
    setTitle("Ecommerce System");  
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    setSize(400, 300);  
    setLocationRelativeTo(null);  
  
    JPanel panel = new JPanel();  
    panel.setLayout(new BorderLayout());  
  
    panel.add(new JLabel(text: "Customer ID:"));  
    customerIdField = new JTextField();  
    panel.add(customerIdField);  
  
    panel.add(new JLabel(text: "Name:"));  
    nameField = new JTextField();  
    panel.add(nameField);  
  
    panel.add(new JLabel(text: "Address:"));  
    addressField = new JTextField();  
    panel.add(addressField);  
  
    panel.add(new JLabel(text: "Number of Products:"));  
    numProductsField = new JTextField();  
    panel.add(numProductsField);  
  
    submitButton = new JButton("Submit");  
    panel.add(submitButton);  
}
```

Ecommerce System

Customer ID:	23012143
Name:	adham
address:	ibrahimya
Number of Products:	4
Submit	

agent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.24\lib\idea_rt.jar=53620:C:\Program Files\JetBrains\IntelliJ


```
Run: EcommerceSystem < EcommerceSystemUI <
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.24\lib\idea
Welcome to the Ecommerce System!
Enter your ID: 23012143
please enter your name: Adham mansour
please Enter your address: Ibrahimya
How many products do you want to order? 4
which product do you want to add 1: 1-smartphone 2-T-shist 3-00P
1
which product do you want to add 2: 1-smartphone 2-T-shist 3-00P
2
which product do you want to add 3: 1-smartphone 2-T-shist 3-00P
3
which product do you want to add 4: 1-smartphone 2-T-shist 3-00P
1
Do you want to place an order for the products in your cart? (Total Price: $1259.78) (Y/N) y
Here is your order summary
order id : 1
customer id: 23012143
products:
name: smartphone
price: $599.9
name: T-shirt
price: $19.99
name: 00P
price: $39.99
name: smartphone
price: $599.9
total Price: $1259.78

Process finished with exit code 0

Version Control Run Python Packages TODO Problems Terminal Services Build
// Build completed successfully in 1 sec, 262 ms (a minute ago)
```

```
textField addressField;

textField numProductsField;

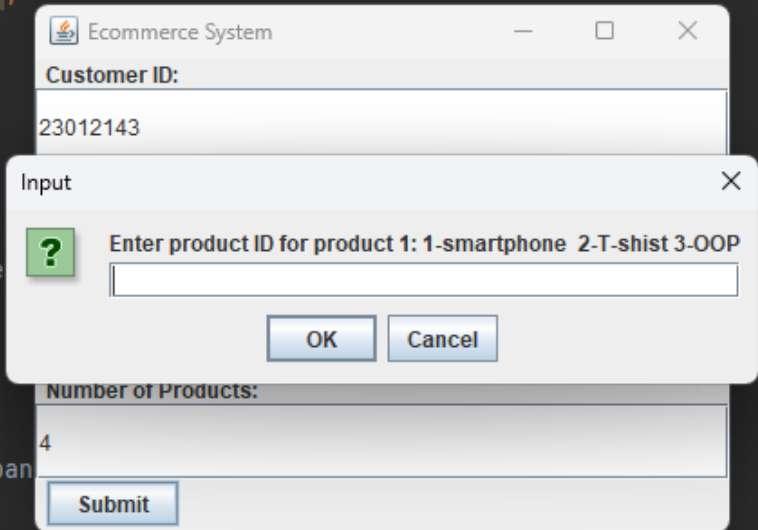
button submitButton;

EcommerceSystemUi() {
    Le("Ecommerce System");
   aultCloseOperation(JFrame
    e( width: 400, height: 300);
    ationRelativeTo(null);

    panel = new JPanel();
    setLayout(new BoxLayout(panel));

    add(new JLabel( text: "Customer ID:"));
    erIdField = new JTextField();
    add(customerIdField);

    add(new JLabel( text: "Name:"));
```



```
TextField customerIdField;  
  
TextField nameField;  
  
TextField addressField;  
  
TextField numProductsField;  
  
Button submitButton;
```

```
commerceSystemUi() {  
    title("Ecommerce System");  
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    setSize( width: 400, height: 300);  
    setLocationRelativeTo(null);  
  
    JPanel panel = new JPanel();  
    panel.setLayout(new BorderLayout());  
  
    panel.add(new JLabel( text: "Customer ID:"));  
    customerIdField = new JTextField();  
    panel.add(customerIdField);  
  
    panel.add(new JLabel( text: "Name:"));
```

Ecommerce System	
Customer ID:	23012143
Name:	adham
Address:	ibrahimya
Number of Products:	4
Submit	

Order Information

Order details:

Customer ID: 23012143

Name: adham

Address: ibrahimya

Number of Products: 4

Total Price: \$1259.78

Products:

- smartphone
- T-shirt
- OOP
- smartphone

OK