

```
Run EcommerceSystem x
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Fil
Welcome to the E-commerce system
Please enter your ID
23011587
Please enter your Name
nada
Please enter your address
alex
How many products you want to add to your cart?
4
Which product would you like to add? 1-smartphone 2- T-shirt 3-OOP
2
Which product would you like to add? 1-smartphone 2- T-shirt 3-OOP
3
Which product would you like to add? 1-smartphone 2- T-shirt 3-OOP
2
Which product would you like to add? 1-smartphone 2- T-shirt 3-OOP
1
You're total is $679.8700256347656Do you want to place an order? 1-yes
1
Here is your order's summary :
Order ID:1
Customer ID:23011587
Products:
T-shirt - Price: $19.989999771118164
OOP - Price: $39.9900016784668
T-shirt - Price: $19.989999771118164
Smartphone - Price: $599.9000244140625
Totl price :$679.8700256347656

Process finished with exit code 0
|
```

```
EcommerceSystem.java  ElectronicProduct.java  product.java  ClothingProduct.java

1  public class Order { 2 usages
2      protected int customerId; 2 usages
3      protected int orderId; 2 usages
4      protected product[] products; 2 usages
5      protected double totalPrice; 2 usages
6
7  @ public Order(int customerId, int orderId, product[] products, Cart c) { 1 usage
8      this.customerId = Math.abs(customerId);
9      this.orderId = Math.abs(orderId);
10     this.products = products;
11     this.totalPrice = c.calculatePrice();
12 }
13 public void printOrderInfo(){ 1 usage
14     System.out.println("Here is your order's summary :");
15     System.out.println("Order ID:"+orderId);
16     System.out.println("Customer ID:"+customerId);
17     System.out.println("Products:");
18     for (product p : products) {
19         if (p != null) {
20             System.out.println( p.getName() + " - Price: $" + p.getPrice());
21         }
22     }
23     System.out.println("Totl price :$"+totalPrice);
24 }
25 }
26 }
```

EcommerceSystem.java ElectronicProduct.java product.java Clothing

```
1 public class Customer { 2 usages
2     protected int customerId; 4 usages
3     protected String name; 3 usages
4     protected String address; 3 usages
5
6     public Customer(int customerId, String name, String address) { 1 usa
7         this.customerId = customerId;
8         this.name = name;
9         this.address = address;
10    }
11    public int getCustomerId() { 1 usage
12        return customerId;
13    }
14    public void setCustomerId(int customerId) { no usages
15        if(customerId<0)
16            this.customerId = Math.abs(customerId);
17        else
18            this.customerId = customerId;
19    }
20    public String getName() { no usages
21        return name;
22    }
23    public void setName(String name) { no usages
24        this.name = name;
25    }
26    public String getAddress() { no usages
27        return address;
28    }
29    public void setAddress(String address) { no usages
30        this.address = address;
31    }
32 }
```

```
EcommerceSystem.java  ElectronicProduct.java  product.java  ClothingProduct.java  BookProduct.java

1  public class Cart { 3 usages
2      protected int customerId; 3 usages
3      protected int nProducts; 5 usages
4      protected product[] products; 6 usages
5      public Cart(int customerId, int nProducts) { 1 usage
6          this.customerId = Math.abs(customerId);
7          this.nProducts = Math.abs(nProducts);
8          this.products = new product[nProducts];
9      }
10     > public int getCustomerId() { return customerId; }
13     > public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }
16     > public int getnProducts() { return nProducts; }
19     public void setnProducts(int nProducts) { no usages
20         this.nProducts = Math.abs(nProducts);
21     }
22     public product[] getProducts() { 1 usage
23         return products;
24     }
25     public void setProducts(product[] products) { no usages
26         this.products = products;
27     }
28     public void addProduct(product p, int index) { 3 usages
29         if (index >= 0 && index < nProducts)
30             products[index] = p;
31     }
32     public void removeProduct(int index) { no usages
33         if (index >= 0 && index < nProducts)
34             products[index] = null;
35     }
36     public double calculatePrice() { 2 usages
37         double totalPrice = 0;
38         for (product p : products) {
39             if (p != null)
40                 totalPrice += p.getPrice();
41         }
42         return totalPrice;
43     }
44     public boolean placeOrder(int choice) { 1 usage
45         if (choice == 1)
46             return true;
47         else
48             return false;}}
```

```
EcommerceSystem.java  ElectronicProduct.java  product.java  ClothingProduct.java  BookProduct.java

1  public class BookProduct extends product { 2 usages
2      protected String author; 3 usages
3      protected String publisher; 3 usages
4
5      public BookProduct(int productId, String name, float price, String author, String publisher) { 1 us
6          super(productId, name, price);
7          this.author=author;
8          this.publisher=publisher;
9      }
10 > public String getAuthor() { return author; }
13 > public void setAuthor(String author) { this.author = author; }
16 > public String getPublisher() { return publisher; }
19 > public void setPublisher(String publisher) { this.publisher = publisher; }
22 }
23
```

```
1 public class ClothingProduct extends product { 2 usages
2     protected String size; 3 usages
3     protected String fabric; 3 usages
4
5     public ClothingProduct(int productId, String name, float price,String size,String fabric) {
6         super(productId, name, price);
7         this.size=size;
8         this.fabric=fabric;
9     }
10 > public String getSize() { return size; }
13 > public void setSize(String size) { this.size = size; }
16 > public String getFabric() { return fabric; }
19 > public void setFabric(String fabric) { this.fabric = fabric; }
22 }
23
```

EcommerceSystem.java ElectronicProduct.java x product.java ClothingProduct.java BookProduct.java

```
1 public class ElectronicProduct extends product{ 2 usages
2     protected String brand; 3 usages
3     protected int warrantyPeriod; 4 usages
4     public ElectronicProduct(int productId, String name, float price,String brand,int warrantyPeriod ) { 1 u
5         super(productId, name, price);
6         this.brand=brand;
7         this.warrantyPeriod=warrantyPeriod;
8     }
9     > public String getBrand() { return brand; }
12    > public void setBrand(String brand) { this.brand = brand; }
15    > public int getWarrantyPeriod() { return warrantyPeriod; }
18    > public void setWarrantyPeriod(int warrantyPeriod) { no usages
19        if (warrantyPeriod<0)
20            this.warrantyPeriod = Math.abs(warrantyPeriod);
21        else
22            this.warrantyPeriod = warrantyPeriod;
23    }
24 }
25
```



```
EcommerceSystem.java    ElectronicProduct.java    product.java x
1  public class product { 12 usages 3 inheritors
2      protected int productId; 4 usages
3      protected String name; 3 usages
4      protected float price; 4 usages
5      public product(int productId, String name, float price) {
6          this.productId = productId;
7          this.name = name;
8          this.price = price;
9      }
10     public int getProductId() { no usages
11         return productId;
12     }
13     public void setProductId(int productId) { no usages
14         if (productId<0)
15             this.productId=Math.abs(productId);
16         else
17             this.productId = productId;
18     }
19     public String getName() { 1 usage
20         return name;
21     }
22     public void setName(String name) { no usages
23         this.name = name;
24     }
25     public double getPrice() { 2 usages
26         return price;
27     }
28     public void setPrice(float price) { no usages
29         if (price<0)
30             this.price=Math.abs(price);
31         else
32             this.price = price;
33     }
34 }
```



```

EcommereceSystem.java x ElectronicProduct.java product.java ClothingProduct.java BookProduct.java Customer.java C
1 import java.util.Scanner;
2 public class EcommereceSystem {
3     public static void main(String[] args) {
4         Scanner input = new Scanner(System.in);
5         ElectronicProduct e = new ElectronicProduct( productId: 1, name: "Smartphone", price: 599.9f, brand: "samsung", warrantyPeriod: 1);
6         ClothingProduct c = new ClothingProduct( productId: 2, name: "T-shirt", price: 19.99f, size: "medium", fabric: "cotton");
7         BookProduct b = new BookProduct( productId: 3, name: "OOP", price: 39.99f, author: "O'Reilly", publisher: "X publisher");
8         System.out.println("Welcome to the E-commerce system");
9         System.out.println("Please enter your ID");
10        int id = input.nextInt();
11        System.out.println("Please enter your Name");
12        String name = input.next();
13        System.out.println("Please enter your address");
14        String address = input.next();
15        Customer customer = new Customer(id, name, address);
16        System.out.println("How many products you want to add to your cart? ");
17        int num = input.nextInt();
18        Cart cart = new Cart(id, num);
19        for (int i = 0; i < num; i++) {
20            System.out.println("Which product would you like to add? 1-smartphone 2- T-shirt 3-OOP");
21            int productNumber = input.nextInt();
22            switch (productNumber) {
23                case 1:
24                    cart.addProduct(e, i);
25                    break;
26                case 2:
27                    cart.addProduct(c, i);
28                    break;
29                case 3:
30                    cart.addProduct(b, i);
31                    break;
32                default:
33                    System.out.println("Invalid product number.");
34            }
35        }
36        System.out.println("Your total is $" + cart.calculatePrice() + "Do you want to place an order? 1-yes 2-no");
37        int choice = input.nextInt();
38        if(cart.placeOrder(choice)){
39            Order order = new Order(customer.getCustomerId(), orderId: 1, cart.getProducts(), cart);
40            order.printOrderInfo();
41        } else
42            System.out.println("Order didn't place");
43    }
44 }

```