

```
import java.util.Scanner;

public class EcommerceSystem {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        Scanner input=new Scanner(System.in);

        System.out.println("Welcome to E-Commerce System!");

        System.out.println("Please enter your ID:");
        int customerId = input.nextInt();
        input.nextLine();

        System.out.println("Please enter your name:");
        String name = input.nextLine();

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

        Customer c1 = new Customer(customerId, name, address);

        System.out.println("How many products would you like to add to your cart?");
        int nProducts = input.nextInt();
```

```
Cart cart = new Cart(customerId, nProducts);
```

```
ElectronicProduct E1= new ElectronicProduct(1, "smartPhone", 599.9f, "Apple", 1);
```

```
ClothingProduct Cc= new ClothingProduct(2, "T-shirt", 19.9f, "medium", "Satan");
```

```
BookProduct B1=new BookProduct(3, "OOP", 39.99f, "O'Reilly", "X Publications");
```

```
for (int i = 0; i < nProducts; i++) {
```

```
    System.out.println("Which product would you like to add? 1-SmartPhone 2-T-shirt 3-  
    OOP");
```

```
    int choice = input.nextInt();
```

```
    input.nextLine();
```

```
    switch (choice) {
```

```
        case 1:
```

```
            cart.addProduct(E1,i);
```

```
            break;
```

```
        case 2:
```

```
            cart.addProduct(Cc,i);
```

```
            break;
```

```
        case 3:
```

```
            cart.addProduct(B1,i);
```

```
            break;
```

```
        default:
```

```
            System.out.println("Invalid choice!");
```

```
}  
}
```

```
System.out.println("Your total is $" + cart.CalculateProduct()+ ". Would you like to  
place the order? 1-Yes 2-No");
```

```
int decision = input.nextInt();
```

```
if (decision == 1) {
```

```
    Order order = new Order(customerId,1, cart.getProducts()); //array of products
```

```
    order.printOrderInfo();
```

```
} else {
```

```
    System.out.println("Order not placed. Thank you for shopping with us!");
```

```
    System.out.println("Hope to see you next time");
```

```
}
```

```
}
```

```
}
```



```
public class Product {  
    private int productId;  
    private String name;  
    private float price;  
    public Product () {  
  
    }  
    public Product (int productId,String name,float price){  
        this.productId=Math.abs(productId);  
        this.name=name;  
        this.price=Math.abs(price);  
  
    }  
  
    public int getProductId() {  
        return productId;  
    }  
  
    public void setProductId(int productId) {  
  
        this.productId = productId;  
    }  
  
    public String getName() {  
        return name;  
    }  
}
```

```
public void setName(String name) {  
    this.name = name;  
}
```

```
public float getPrice() {  
    return price;  
}
```

```
public void setPrice(float price) {  
  
    this.price = price;  
}
```

```
}
```

```
public class ElectronicProduct extends Product {  
    private String brand;  
    private int WarrantyPeriod;  
  
    public ElectronicProduct(int productId,String name ,float price,String brand,int  
WarrantyPeriod){  
        super(productId,name,price);  
        this.brand=brand;  
        this.WarrantyPeriod=Math.abs(WarrantyPeriod);  
    }  
  
    public String getBrand() {  
        return brand;  
    }  
  
    public void setBrand(String brand) {  
        this.brand = brand;  
    }  
  
    public int getWarrantyPeriod() {  
        return WarrantyPeriod;  
    }  
  
    public void setWarrantyPeriod(int WP) {
```

```
        this.WarrantyPeriod = WP;
    }

    public class ClothingProduct extends Product {

        private String Size;

        private String Fabric;


        public ClothingProduct(int productId,String name,float price,String Size,String Fabric){

            super(productId,name,price);

            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;  
}
```

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

```
    public BookProduct(int productId, String name, float price, String author, String publisher)  
    {  
        super(productId, name, price);  
        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;  
}
```

```
}
```

```
public class Customer {  
    private int customerId;  
    private String name;  
    private String address;
```

```
public Customer(int customerId, String name, String address) {  
    this.customerId = Math.abs(customerId);  
    this.name = name;  
    this.address = address;  
}
```

```
public int getCustomerId() {  
    return customerId;  
}
```

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

```
public String getName() {  
    return name;  
}
```

```
public void setName(String name) {  
    this.name = name;  
}
```

```
public String getAddress() {  
    return address;  
}
```

```
public void setAddress(String address) {  
    this.address = address;  
}  
}
```

```
public class Cart {  
    private int customerId;  
    private int nProducts;  
    private Product[] Products;  
  
    public Cart(int customerId, int nProducts) {  
        this.customerId = Math.abs(customerId);  
        this.nProducts = nProducts;  
        this.Products = new Product[this.nProducts];  
    }  
  
    public int getCustomerId() {  
        return customerId;  
    }  
  
    public void setCustomerId(int customerId) {  
        this.customerId = customerId;  
    }  
  
    public int getnProducts() {  
        return nProducts;  
    }  
}
```

```
public void setnProducts(int nProducts) {
```

```
    this.nProducts = nProducts;
```

```
}
```

```
public Product[] getProducts() {
```

```
    return Products;
```

```
}
```

```
public void setProducts(Product[] Products) {
```

```
    this.Products = Products;
```

```
}
```

```
public void addProduct(Product product,int index){
```

```
    if(index>=0&&index<nProducts)
```

```
        Products[index]=product;
```

```
    else
```

```
        System.out.println("invalid");
```

```
}
```

```
public void removeProudct(Product product,int in){
```

```
    if(in>=0&&in<nProducts)
```

```
        Products[in]=null;
```

```
    else
```

```
        System.out.println("invalid");
```

```
}
```

```
public double CalculateProduct(){
```

```
    float totalPrice=0;
```

```
    for(int i=0;i<nProducts;i++){
```

```
        if(Products[i]!=null){
```

```
            totalPrice+=Products[i].getPrice();
```

```
        }
```

```
    }
```

```
    return totalPrice;
```

```
}
```

```
public void PlaceOrder(){
```

```
    System.out.println("customerId"+customerId);
```

```
    System.out.println("Total price:"+CalculateProduct()+"$");
```

```
}
```

```
}
```

```
public class Order {
```

```
    private int customerId;
```

```
    private int orderId;
```

```
    private Product[] Products;
```

```
    private float totalPrice;
```

```
public Order(int customerId,int orderId ,Product[] Products) {  
    this.customerId = Math.abs(customerId);  
    this.orderId = Math.abs(orderId);  
    this.Products =Products ;  
    this.totalPrice = Math.abs(totalPrice);  
}  
  
private double calculateTotalProduct(){  
    float total=0;  
    for(int i=0;i<Products.length;i++){  
        if(Products[i]!=null){  
            total+=Products[i].getPrice(); //إذا تحقق الشرط,ضيف السعر للمجموع  
        }  
    }  
  
    return total;  
}
```

```
public void printOrderInfo(){  
    System.out.println("Here is your Order summary:");  
    System.out.println("your orderId:"+orderId);  
    System.out.println("your customer ID:"+customerId);  
    System.out.println("the Products:");  
    for(int i=0;i<Products.length;i++){  
        if(Products[i]!=null)
```

```
        System.out.println(Products[i].getName()+"$"+Products[i].getPrice());

    }

    System.out.println("the totalPrice"+ calculateTotalProduct()+"$" );

}

public int getCustomerId() {
    return customerId;
}

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

public int getOrderId() {
    return orderId;
}

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

public Product[] getProducts() {
    return Products;
}
```



```
}
```

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

```
public float getTotalPrice() {  
    return totalPrice;  
}
```

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

```
}
```

The output:

Welcome to E-Commerce System!

Please enter your ID:

23011923

Please enter your name:

Madawi

Please enter your address:

Alex

How many products would you like to add to your cart?

3

Which product would you like to add? 1-SmartPhone 2-T-shirt 3-OOP

1

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

Your total is \$659.7900390625. Would you like to place the order? 1-Yes 2-No

1

Here is your Order summary:

your orderId:1

your customer ID:23011923

the Products:

smartPhone\$599.9

T-shirt\$19.9

OOP\$39.99

the totalPrice659.7900390625\$