

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help E-Commerce - Apache NetBeans IDE 19 Search (Ctrl+I)

Start Page x ECommerce.java x Product.java x ElectronicProduct.java x ClothingBrand.java x BookProduct.java x Customer.java x Cart.java x Order.java x

Source History

```
5 package e.commerce;
6 import java.util.Scanner;
7 public class ECommerce {
8
9     /**
10      * @param args the command line arguments
11      */
12     public static void main(String[] args) {
13         ElectronicProduct smartphone = new ElectronicProduct(productId: 1, name: "Smartphone", price: 599.99f, brand: "Samsung", warrantyPeriod: 1);
14         ClothingBrand tShirt = new ClothingBrand(productId: 2, name: "T-Shirt", price: 19.99f, size: "Medium", fabric: "Cotton");
15         BookProduct oop = new BookProduct(productId: 3, name: "OOP", price: 39.99f, author: "O'Reilly", publisher: "X Publications");
16
17         Scanner scanner = new Scanner(System.in);
18         System.out.println("Welcome to the E-Commerce System!");
19         System.out.print("Please enter your id: ");
20         int customerId = scanner.nextInt();
21         scanner.nextLine();
22         System.out.print("Please enter your name: ");
23         String name = scanner.nextLine();
24         System.out.print("Please enter your address: ");
25         String address = scanner.nextLine();
26         Customer customer = new Customer(customerId, name, address);
27
28
29         System.out.print("How many products do you want to add to your cart? ");
30         int nProducts = scanner.nextInt();
31         scanner.nextLine();
32         Cart cart = new Cart(customerId: customer.getCustomerId(), nProducts);
33
34
35         for (int i = 0; i < nProducts; i++) {
36             System.out.println("Which product would you like to add? 1- Smartphone 2- T-Shirt 3- OOP");
```

Notifications 27:1 INS Windows (CRLF)

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help E-Commerce - Apache NetBeans IDE 19 Search (Ctrl+I)

<default config> 255.9/3810MB

Start Page x ECommerce.java x Product.java x ElectronicProduct.java x ClothingBrand.java x BookProduct.java x Customer.java x Cart.java x Order.java x

Source History

```
34
35 for (int i = 0; i < nProducts; i++) {
36     System.out.println("Which product would you like to add? 1- Smartphone 2- T-Shirt 3- OOP");
37     int choice = scanner.nextInt();
38     scanner.nextLine();
39     switch (choice) {
40     case 1:
41         cart.addProduct(product:smartphone);
42         break;
43     case 2:
44         cart.addProduct(product:tShirt);
45         break;
46     case 3:
47         cart.addProduct(product:oop);
48         break;
49     default:
50         System.out.println("Invalid choice. Skipping this product.");
51         break;
52     }
53 }
54 System.out.printf(format: "Your total is $%.5f. Would you like to place the order? 1- Yes 2- No\n", args: cart.calculatePrice());
55 int orderChoice = scanner.nextInt();
56 scanner.nextLine();
57 if (orderChoice == 1) {
58     cart.placeOrder(customer);
59 } else {
60     System.out.println("Order canceled.");
61 }
62 }
63 }
64
65
```

Notifications 27:1 INS Windows (CRLF)

Start Page × ECommerce.java × Product.java × ElectronicProduct.java × ClothingBrand.java × BookProduct.java ×

Source History

Navigator

Services

Files

Projects

0
1
2
3
4
5
6
7
8
9 * @author user
10 */
11
12 public class Product {
13 private int productId;
14 private String name;
15 private float price;
16
17 public Product() {
18 }
19
20 public Product(int productId, String name, float price) {
21 this.productId = Math.abs(productId);
22 this.name = name;
23 this.price = Math.abs(price);
24 }
25
26 public void setProductId(int productId) {
27 this.productId = productId;
28 }
29
30 public void setName(String name) {
31 this.name = name;
32 }
33
34 public void setPrice(float price) {
35 this.price = price;
36 }
37
38 public int getProductId() {
39 return productId;
40 }
41
42 }

Notifications

```
36
37 [- public int getProductId() {
38     return productId;
39 }
40
41 [- public String getName() {
42     return name;
43 }
44
45 [- public float getPrice() {
46     return price;
47 }
48 }
49
50
```

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help E-Commerce - Apache NetBeans IDE 19 Search (Ctrl+I)

<default config> 258.4/506.0MB

Start Page x ECommerce.java x Product.java x ElectronicProduct.java x ClothingBrand.java x BookProduct.java x Customer.java x Cart.java x Order.java x

Source History

```
9  * @author user
10 */
11 public class ElectronicProduct extends Product{
12     private String brand;
13     private int warrantyPeriod;
14
15     public ElectronicProduct(int productId, String name, float price, String brand, int warrantyPeriod) {
16         super(productId, name, price);
17         this.brand = brand;
18         this.warrantyPeriod = Math.abs(warrantyPeriod);
19     }
20
21     public void setBrand(String brand) {
22         this.brand = brand;
23     }
24
25     public void setWarrantyPeriod(int warrantyPeriod) {
26         this.warrantyPeriod = warrantyPeriod;
27     }
28
29     public String getBrand() {
30         return brand;
31     }
32
33     public int getWarrantyPeriod() {
34         return warrantyPeriod;
35     }
36 }
37
38
```

Notifications 26:41 INS Windows (CRLF)

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help E-Commerce - Apache NetBeans IDE 19 Search (Ctrl+I)

Start Page x ECommerce.java x Product.java x ElectronicProduct.java x ClothingBrand.java x BookProduct.java x Customer.java x Cart.java x Order.java x

Source History

```
7  /**
8   *
9   * @author user
10  */
11  public class ClothingBrand extends Product {
12      private String size;
13      private String fabric;
14
15      public ClothingBrand(int productId, String name, float price, String size, String fabric) {
16          super(productId, name, price);
17          this.size = size;
18          this.fabric = fabric;
19      }
20
21      public void setSize(String size) {
22          this.size = size;
23      }
24
25      public void setFabric(String fabric) {
26          this.fabric = fabric;
27      }
28
29      public String getSize() {
30          return size;
31      }
32
33      public String getFabric() {
34          return fabric;
35      }
36  }
37
38
```

Projects Files Services Navigator

Notifications 29:24 INS Windows (CRLF)

Start Page × ECommerce.java × Product.java × ElectronicProduct.java × ClothingBrand.java × BookProduct.java × Customer.java

Source History

Source

History

Find

Copy

Paste

Undo

Redo

Run

Stop

Breakpoint

Watch

Log

Console

Output

Errors

Warnings

Messages

Stack

Debugger

Help

Tools

Windows

View

Layout

Theme

Settings

Help

Exit

Navigator

Services

Files

Projects

```
8      *
9      * @author user
10     */
11     public class BookProduct extends Product {
12         private String author;
13         private String publisher;
14
15         public BookProduct(int productId, String name, float price, String author, String publisher) {
16             super(productId, name, price);
17             this.author = author;
18             this.publisher = publisher;
19         }
20
21         public void setAuthor(String author) {
22             this.author = author;
23         }
24
25         public void setPublisher(String publisher) {
26             this.publisher = publisher;
27         }
28
29         public String getAuthor() {
30             return author;
31         }
32
33         public String getPublisher() {
34             return publisher;
35         }
36     }
37 }
```

Start Page × ECommerce.java × Product.java × ElectronicProduct.java × ClothingBrand.java × BookProduct.java × Customer.java × Cart.java × Order.java

Source History

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

```
/**
 *
 * @author user
 */
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 void setCustomerId(int customerId) {
        this.customerId = customerId;
    }

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

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

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

Notifications


```
32  
33  
34  
35 public int getCustomerId() {  
36     return customerId;  
37 }  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47
```

```
38 public String getName() {  
39     return name;  
40 }  
41  
42  
43  
44  
45  
46  
47
```

```
42 public String getAddress() {  
43     return address;  
44 }  
45 }  
46  
47
```

```
6
7 public class Cart {
8     private int customerId;
9     private int nProducts;
10    private Product[] products;
11    private int orderId = 1; // Counter for order ID
12
13    public Cart(int customerId, int nProducts) {
14        this.customerId = Math.abs(customerId);
15        this.nProducts = Math.abs(nProducts);
16        this.products = new Product[nProducts];
17    }
18
19    public void setCustomerId(int customerId) {
20        this.customerId = customerId;
21    }
22
23    public void setnProducts(int nProducts) {
24        this.nProducts = nProducts;
25    }
26
27    public void setProducts(Product[] products) {
28        this.products = products;
29    }
30
31    public void setOrderId(int orderId) {
32        this.orderId = orderId;
33    }
34
35    public int getCustomerId() {
36        return customerId;
37    }
```

```

38
39     public int getnProducts() {
40         return nProducts;
41     }
42
43     public Product[] getProducts() {
44         return products;
45     }
46
47     public int getOrderId() {
48         return orderId;
49     }
50
51     public void addProduct(Product product) {
52         for (int i = 0; i < nProducts; i++) {
53             if (products[i] == null) {
54                 products[i] = product;
55                 return;
56             }
57         }
58         System.out.println("Cart is full. Cannot add more products.");
59     }
60
61     public float calculatePrice() {
62         float totalPrice = 0.0f;
63         for (Product product : products) {
64             if (product != null) {
65                 totalPrice += product.getPrice();
66             }
67         }
68         return totalPrice;
69     }
  
```

```
69  },
70
71  public void placeOrder(Customer customer) {
72      float totalPrice = calculatePrice();
73      Order order = new Order(orderId++, customerId: customer.getCustomerId(), nProducts: this.nProducts, products: this.products, totalPrice);
74      order.printOrderInfo(customer);
75  }
76  }
77
78
79
80
81
```

```

11 public class Order {
12     private int orderId;
13     private int customerId;
14     private Product[] products;
15     private float totalPrice;
16
17     public Order(int orderId, int customerId, int nProducts, Product[] products, float totalPrice) {
18         this.orderId = orderId;
19         this.customerId = customerId;
20         this.products = products;
21         this.totalPrice = totalPrice;
22     }
23
24     public void setOrderId(int orderId) {
25         this.orderId = orderId;
26     }
27
28     public void setCustomerId(int customerId) {
29         this.customerId = customerId;
30     }
31
32     public void setProducts(Product[] products) {
33         this.products = products;
34     }
35
36     public void setTotalPrice(float totalPrice) {
37         this.totalPrice = totalPrice;
38     }
39
40     public int getOrderId() {
41         return orderId;
42     }

```


Source History

```

37         this.totalPrice = totalPrice;
38     }
39
40     public int getOrderId() {
41         return orderId;
42     }
43
44     public int getCustomerId() {
45         return customerId;
46     }
47
48     public Product[] getProducts() {
49         return products;
50     }
51
52     public float getTotalPrice() {
53         return totalPrice;
54     }
55
56     public void printOrderInfo(Customer customer) {
57         System.out.println("Here's your order's summary:");
58         System.out.println("Order ID: " + this.orderId);
59         System.out.println("Customer ID: " + this.customerId);
60         System.out.println("Products:");
61         for (Product product : this.products) {
62             if (product != null) {
63                 System.out.println(product.getName() + " $" + product.getPrice());
64             }
65         }
66         System.out.printf("Total Price: $%.2f\n", args: this.totalPrice);
67     }
68 }

```

Start Page ×ECommerce.java ×Product.java ×ElectronicProduct.java ×ClothingBrand.java ×BookProduct.java ×Customer.java ×Cart.java ×Order.java ×

Output - E-Commerce (run)

run:
Welcome to the E-Commerce System!
Please enter your id: 23012096
Please enter your name: Roshan Mohamed Helmy
Please enter your address: Louran
How many products do 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
Your total is \$679.95996. Would you like to place the order? 1- Yes 2- No
1
Here's your order's summary:
Order ID: 1
Customer ID: 23012096
Products:
T-Shirt \$19.99
OOP \$39.99
T-Shirt \$19.99
Smartphone \$599.99
Total Price: \$679.96
BUILD SUCCESSFUL (total time: 1 minute 5 seconds)

NotificationsOutput

24:40INS