

e-commerce-system-challenge

Examples

Normal Case:

Code:

```
Customer customer = new Customer("Amr", 4500);
Product Cheese = new Cheese("Cheese", 50, 10, 2, LocalDate.of(2025, 9, 25));
Product TV = new TV("TV", 1500, 5, 10);
Product Mobile = new Mobile("Mobile", 1500, 5, 10);
Product ScratchCard = new ScratchCard("ScratchCard", 100, 30);
Product Biscuits = new Biscuits("Biscuits", 5, 60, 0.2, LocalDate.of(2025, 9, 25));

Cart cart = new Cart();
try {
    cart.addItem(Cheese, 5);
    cart.addItem(TV, 1);
    cart.addItem(ScratchCard, 5);
    cart.addItem(Mobile, 1);
    cart.addItem(Biscuits, 1);

    customer.checkout(cart);
} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
```

Console Output:

```
** Shipment notice **
1x Cheese 2.0Kg
1x TV 10.0Kg
1x Mobile 10.0Kg
1x Biscuits 0.2Kg

Total package weight 22.2kg

** Checkout receipt **
5x Cheese 250.0
1x TV 1500.0
5x ScratchCard 500.0
1x Mobile 1500.0
1x Biscuits 5.0
```

```
-----  
Subtotal      3755.0  
Shipping      563.3  
Amount        4318.3  
Your current balance 181.8
```

No Enough balance

Code:

```
Customer customer = new Customer("Amr", 2500);  
Product Cheese = new Cheese("Cheese", 50, 10, 2, LocalDate.of(2025, 9, 25));  
Product TV = new TV("TV", 1500, 5, 10);  
Product Mobile = new Mobile("Mobile", 1500, 5, 10);  
Product ScratchCard = new ScratchCard("ScratchCard", 100, 30);  
Product Biscuits = new Biscuits("Biscuits", 5, 60, 0.2, LocalDate.of(2025, 9,  
25));  
  
Cart cart = new Cart();  
try {  
    cart.addItem(Cheese, 5);  
    cart.addItem(TV, 1);  
    cart.addItem(ScratchCard, 5);  
    cart.addItem(Mobile, 1);  
    cart.addItem(Biscuits, 1);  
  
    customer.checkout(cart);  
} catch (Exception ex) {  
    System.out.println(ex.getMessage());  
}
```

Console Output:

```
** Shipment notice **  
1x Cheese 2.0Kg  
1x TV 10.0Kg  
1x Mobile 10.0Kg  
1x Biscuits 0.2Kg  
  
Total package weight 22.2kg  
  
** Checkout receipt **  
No Enough balance
```

Product is out of stock

Code:

```
Customer customer = new Customer("Amr", 2500);
Product Cheese = new Cheese("Cheese", 50, 10, 2, LocalDate.of(2025, 9, 25));
Product TV = new TV("TV", 1500, 5, 10);
Product Mobile = new Mobile("Mobile", 1500, 5, 10);
Product ScratchCard = new ScratchCard("ScratchCard", 100, 30);
Product Biscuits = new Biscuits("Biscuits", 5, 60, 0.2, LocalDate.of(2025, 9, 25));

Cart cart = new Cart();
try {
    cart.addItem(Cheese, 60);
    cart.addItem(TV, 1);
    cart.addItem(ScratchCard, 5);
    cart.addItem(Mobile, 1);
    cart.addItem(Biscuits, 1);

    customer.checkout(cart);
} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
```

Console Output:

Product is out of stock

Cart is empty

Code:

```
Customer customer = new Customer("Amr", 2500);
Product Cheese = new Cheese("Cheese", 50, 10, 2, LocalDate.of(2025, 1, 25));
Product TV = new TV("TV", 1500, 5, 10);
Product Mobile = new Mobile("Mobile", 1500, 5, 10);
Product ScratchCard = new ScratchCard("ScratchCard", 100, 30);
Product Biscuits = new Biscuits("Biscuits", 5, 60, 0.2, LocalDate.of(2025, 9, 25));

Cart cart = new Cart();
try {
    customer.checkout(cart);
} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
```

Console Output:

Cart is Empty

Cart is Expired

Code:

```
Customer customer = new Customer("Amr", 2500);
Product Cheese = new Cheese("Cheese", 50, 10, 2, LocalDate.of(2025, 1, 25));
Product TV = new TV("TV", 1500, 5, 10);
Product Mobile = new Mobile("Mobile", 1500, 5, 10);
Product ScratchCard = new ScratchCard("ScratchCard", 100, 30);
Product Biscuits = new Biscuits("Biscuits", 5, 60, 0.2, LocalDate.of(2025, 9, 25));

Cart cart = new Cart();
try {
    customer.checkout(cart);
} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
```

Console Output:

Product is Expired