/\*\*

\* The Order class represents an order placed by a customer in the e-commerce system. It includes

\* attributes such as order ID, customer details, and a list of purchased products with their quantities.

\* The class provides methods to create and manage orders, and calculates the total cost of the order.

\* It also updates the stock quantity of the purchased products.

\*/

package com.ecommerce.orders;

import com.ecommerce.Customer;

import com.ecommerce.Product;

import java.util.HashMap;

import java.util.Map;

public class Order {

private int orderID;

private Customer customer;

private Map<Product, Integer> products;

private double orderTotal;

public Order(int orderID, Customer customer) {

if (orderID <= 0) {

throw new IllegalArgumentException("Order ID must be positive.");

}

if (customer == null) {

throw new IllegalArgumentException("Customer cannot be null.");

}

this.orderID = orderID;

this.customer = customer;

this.products = new HashMap<>(customer.getShoppingCart());

this.orderTotal = calculateOrderTotal();

updateProductStock();

}

private double calculateOrderTotal() {

double total = 0;

for (Map.Entry<Product, Integer> entry : products.entrySet()) {

total += entry.getKey().getPrice() \* entry.getValue();

}

return total;

}

private void updateProductStock() {

for (Map.Entry<Product, Integer> entry : products.entrySet()) {

Product product = entry.getKey();

int quantity = entry.getValue();

if (quantity > product.getStockQuantity()) {

throw new IllegalArgumentException("Insufficient stock for product: " + product.getName());

}

product.setStockQuantity(product.getStockQuantity() - quantity);

}

}

public void generateOrderSummary() {

System.out.println("Order ID: " + orderID);

System.out.println("Customer: " + customer.getName());

System.out.println("Products:");

for (Map.Entry<Product, Integer> entry : products.entrySet()) {

System.out.println(entry.getKey() + " Quantity: " + entry.getValue());

}

System.out.println("Order Total: " + orderTotal);

}

@Override

public String toString() {

return "Order [ID=" + orderID + ", Customer=" + customer.getName() + ", Total=" + orderTotal + "]";

}

}