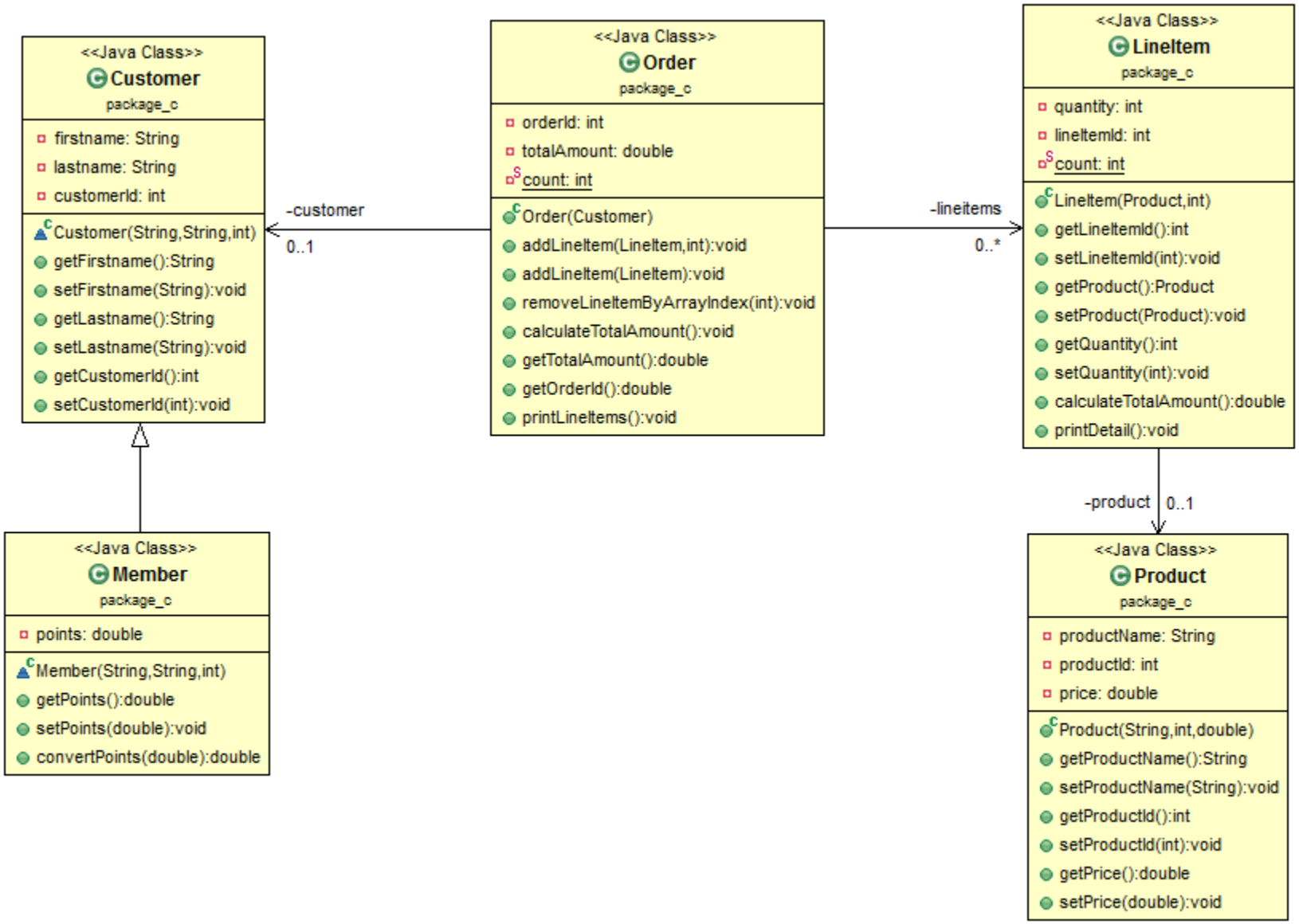
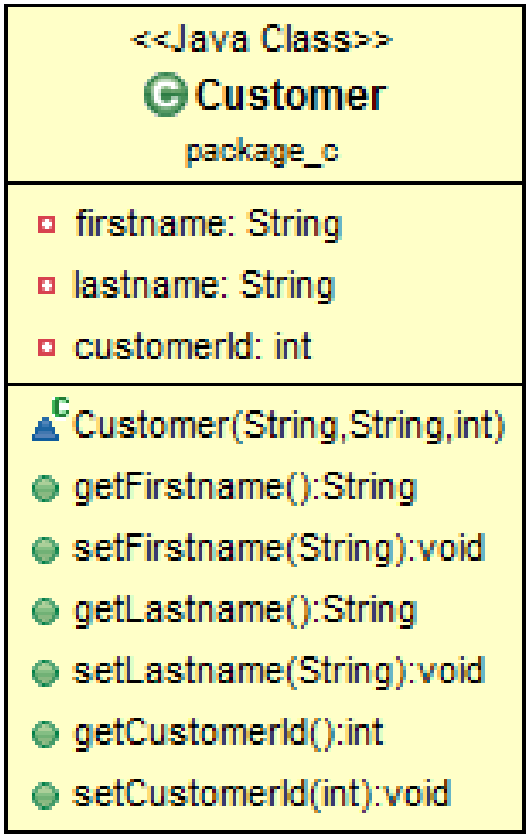
LAB 9

Lab 9

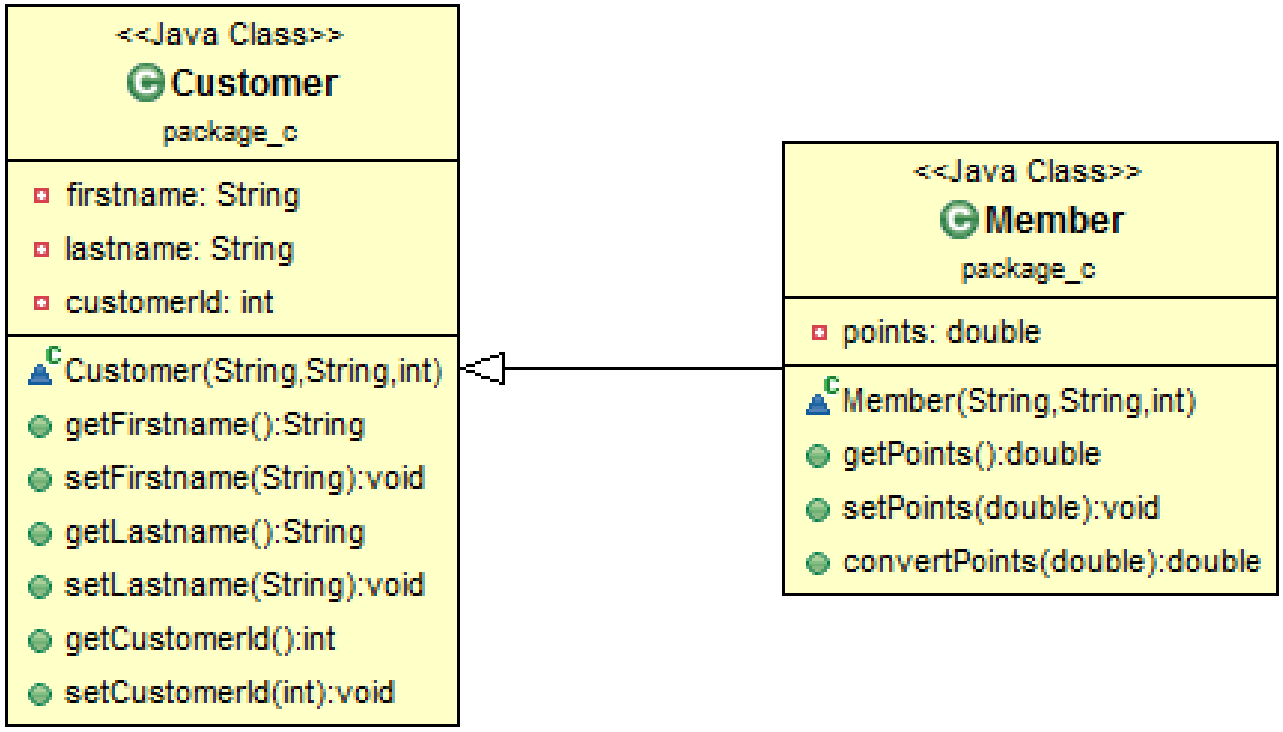

<<Java Class>>
Customer
package_c
firstname: String
lastname: String
customerId: int
Customer(String,String,int)
getFirstname():String
setFirstname(String):void
getLastname():String
setLastname(String):void
getCustomerId():int
setCustomerId(int):void
-customer
0..1
<<Java Class>>
Order
package_c
orderId: int
totalAmount: double
count: int
Order(Customer)
addLineItem(LineItem,int):void
addLineItem(LineItem):void
removeLineItemByArrayIndex(int):void
calculateTotalAmount():void
getTotalAmount():double
getOrderId():double
printLineItems():void
-lineitems
0..*
<<Java Class>>
LineItem
package_c
quantity: int
lineItemId: int
count: int
LineItem(Product,int)
getLineItemId():int
setLineItemId(int):void
getProduct():Product
setProduct(Product):void
getQuantity():int
setQuantity(int):void
calculateTotalAmount():double
printDetail():void
-product
0..1
<<Java Class>>
Member
package_c
points: double
Member(String,String,int)
getPoints():double
setPoints(double):void
convertPoints(double):double
<<Java Class>>
Product
package_c
productName: String
productId: int
price: double
Product(String,int,double)
getProductName():String
setProductName(String):void
getProductId():int
setProductId(int):void
getPrice():double
setPrice(double):void

Exercise 1: Create a Customer Class



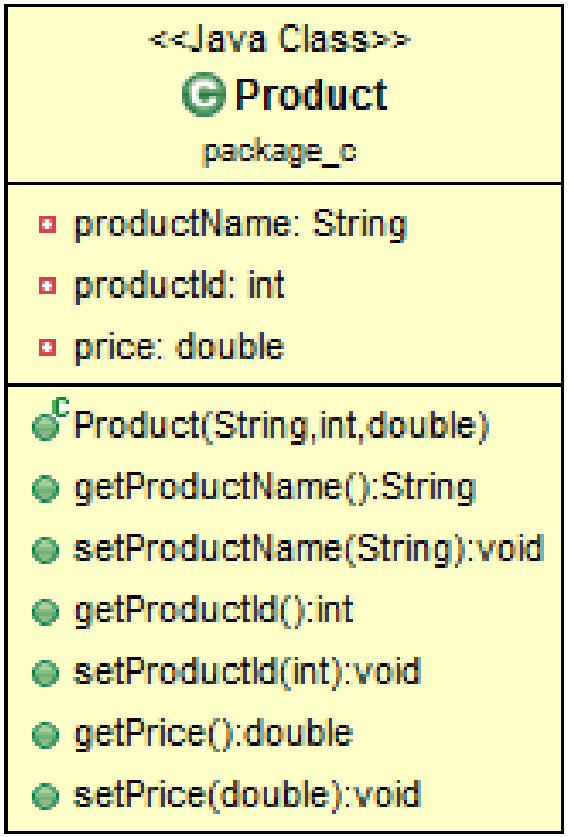
```java
public class Customer {
    private String firstname;
    private String lastname;
    private int customerId;

    Customer(String fn, String ln, int id){
        firstname = fn;
        lastname = ln;
        customerId = id;
    }

    public String getFirstname() {
        return firstname;
    }
    public void setFirstname(String firstname) {
        this.firstname = firstname;
    }
    public String getLastname() {
        return lastname;
    }
    public void setLastname(String lastname) {
        this.lastname = lastname;
    }
    public int getCustomerId() {
        return customerId;
    }
    public void setCustomerId(int customerId) {
        this.customerId = customerId;
    }
}
```

Exercise 2: Create a Member Class



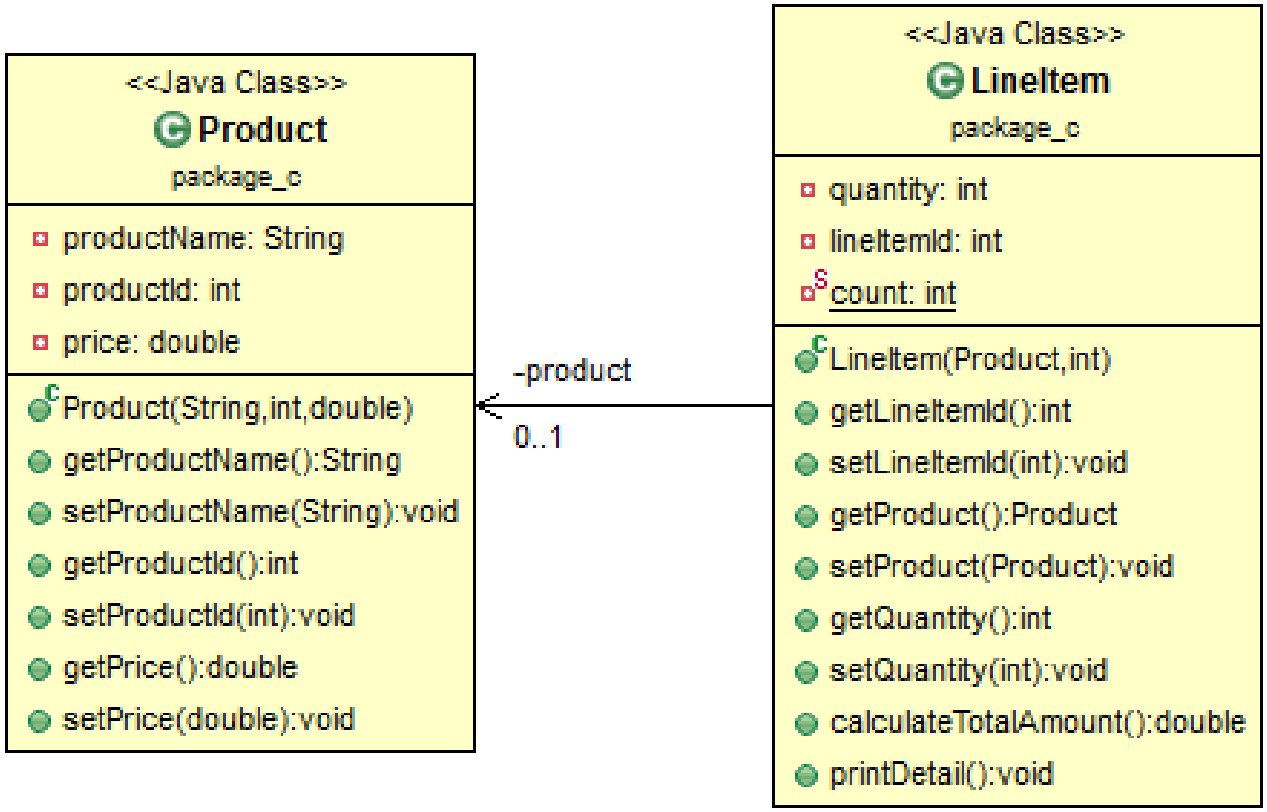
```java
public class Member extends Customer {
    private double points;

    Member(String fn, String ln, int id) {
        super(fn, ln, id);
        points = 0;
    }

    public double getPoints() {
        return points;
    }

    public void setPoints(double shoppingAmount) {
        points = points + (shoppingAmount/1000);
    }

    public double convertPoints(double p){
        double converted = p*20;
        points = points - p;
        return converted;
    }

}
```

Exercise 3: Create a Product Class



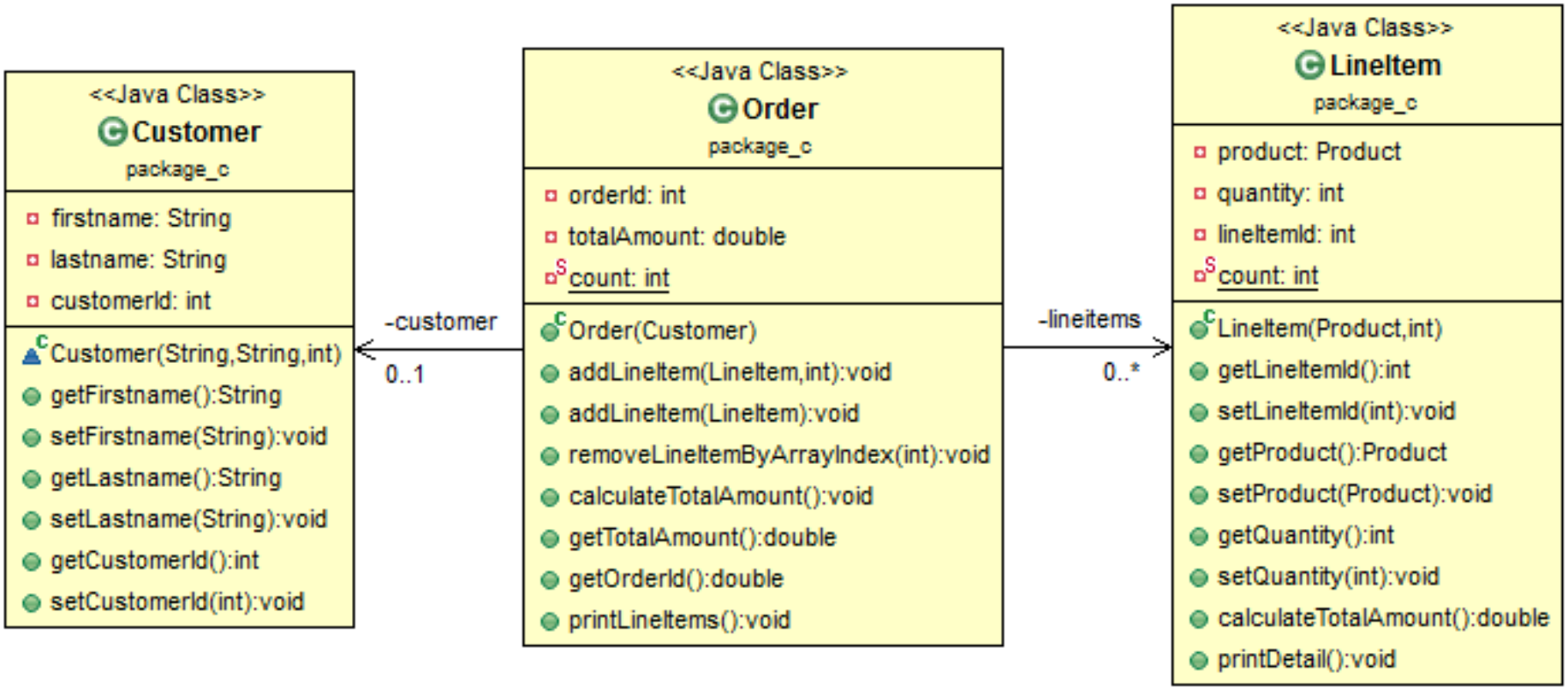
```java
public class Product {
    private String productName;
    private int productId;
    private double price;

    public Product(String productName, int productId, double price){
        this.productName = productName;
        this.productId = productId;
        this.price = price;
    }

    public String getProductName() {
        return productName;
    }
    public void setProductName(String productName) {
        this.productName = productName;
    }
    public int getProductId() {
        return productId;
    }
    public void setProductId(int productId) {
        this.productId = productId;
    }
    public double getPrice() {
        return price;
    }
    public void setPrice(double price) {
        this.price = price;
    }
}
```

Exercise 4: Create a LineItem Class



```java
public class LineItem {
    private Product product;
    private int quantity;
    private int lineItemId;
    private static int count;

    public LineItem(Product product, int quantity){
        count++;
        this.quantity = quantity;
        this.product = product;
        this.lineItemId = count;
    }

    public int getLineItemId() {
        return lineItemId;
    }
    public void setLineItemId(int lineItemId) {
        this.lineItemId = lineItemId;
    }
    public Product getProduct() {
        return product;
    }
    public void setProduct(Product product) {
        this.product = product;
    }
    public int getQuantity() {
        return quantity;
    }
    public void setQuantity(int quantity) {
        this.quantity = quantity;
    }

    public double calculateTotalAmount() {
        return quantity*(product.getPrice());
    }
    public void printDetail(){
        System.out.println("LineItem ID: " + lineItemId + " " +
                           "Product: " + product.getProductName() + " " +
                           "Quantity: " + quantity + " "+
                           "TotalAmount: " + calculateTotalAmount());
    }
}
```

Exercise 5: Create an Order Class

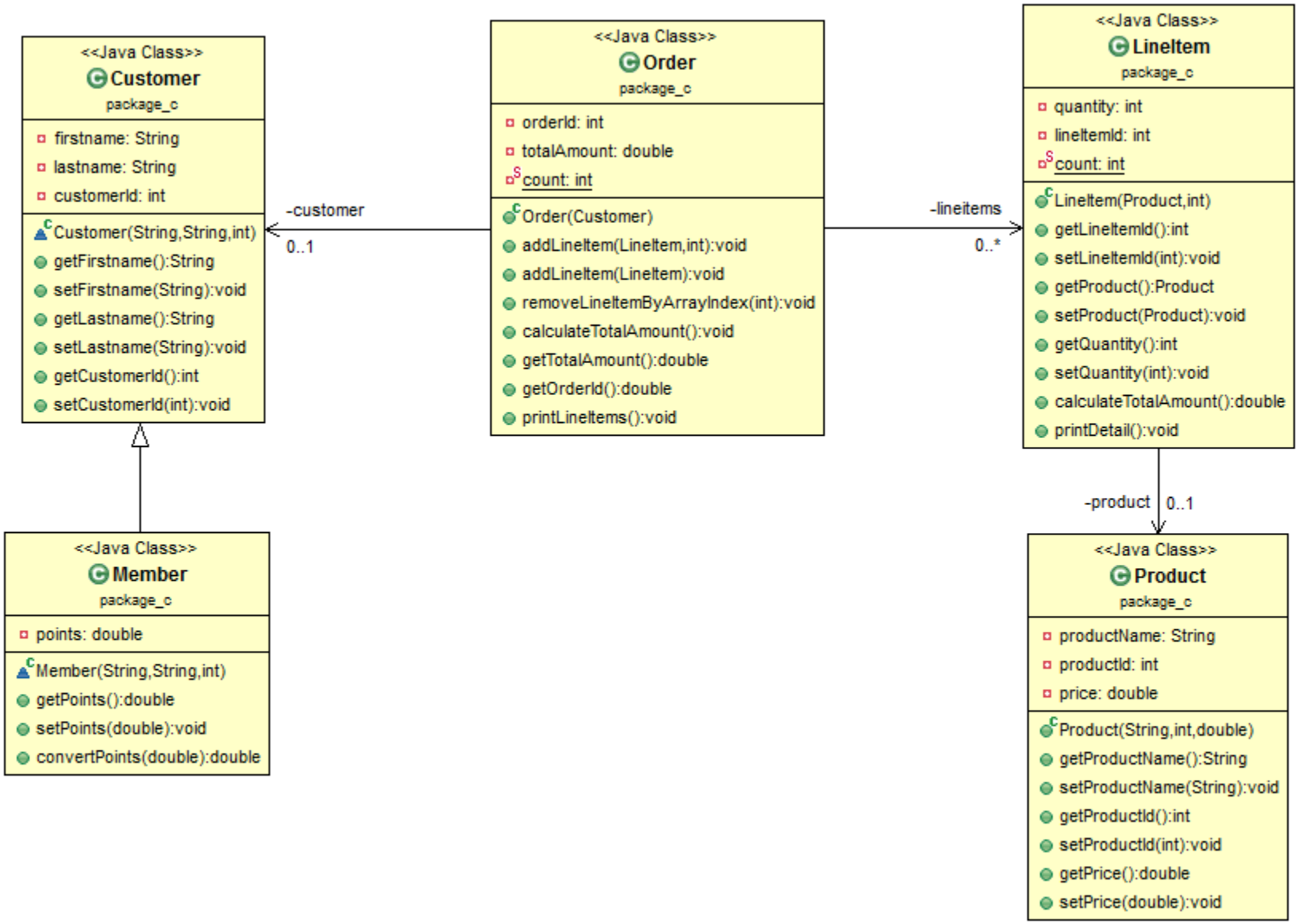


```java
public class Order {
    private int orderId;
    private LineItem lineitems[];
    private double totalAmount;
    private static int count;
    private Customer customer;

    public Order(Customer c){
        this.orderId = count++;
        this.totalAmount = 0;
        this.lineitems = new LineItem[20];
        this.customer = c;
    }

    public void addLineItem(LineItem lineitem, int index){
        if(lineitems[index]==null){
            lineitems[index] = lineitem;
            System.out.println("Add line item successfully.");
        }
        else{
            System.out.println("The array is not empty. Line item cannot be added.");
        }
    }
    public void addLineItem(LineItem lineitem){
        boolean addSuccess = false;
        for(int i = 0; i<lineitems.length; i++){
            if(lineitems[i]==null){
                lineitems[i] = lineitem;
                addSuccess = true;
                break;
            }
        }
        if(addSuccess){
            System.out.println("Add line item successfully.");
        }
        else{
            System.out.println("Your basket is full. Cannot add a line item.");
        }
    }
```

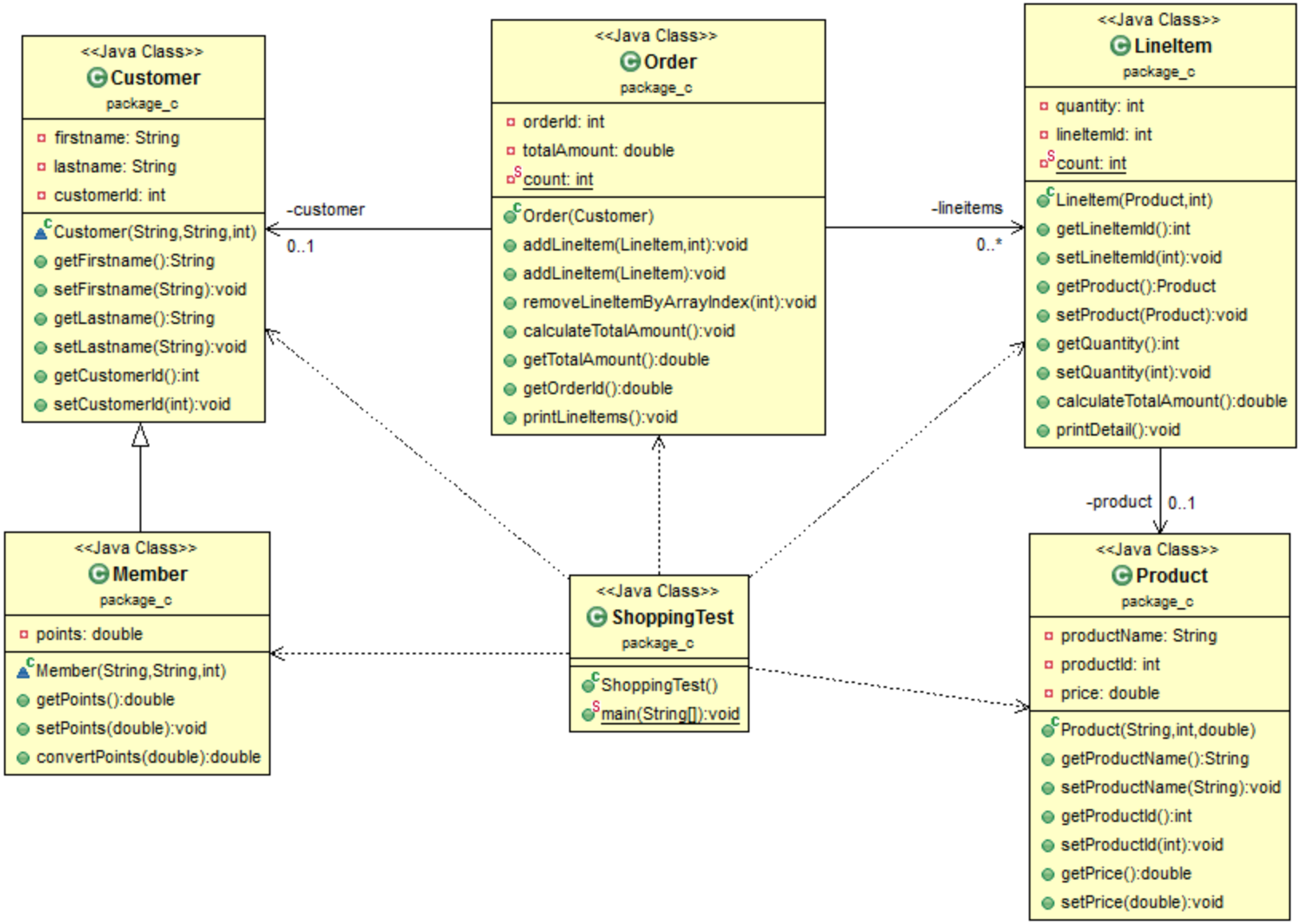
Cont.

```
43      public void removeLineItemByArrayIndex(int index){
44          if(lineitems[index]!=null){
45              lineitems[index] = null;
46              System.out.println("Remove line item successfully.");
47          }
48          else{
49              System.out.println("The array is empty. Nothing removed.");
50          }
51      }
52      public void calculateTotalAmount(){
53          double sum = 0;
54          for(LineItem l : lineitems){
55              if(l!=null){
56                  sum = sum + l.calculateTotalAmount();
57              }
58          }
59          totalAmount = sum;
60      }
61      public double getTotalAmount(){
62          calculateTotalAmount();
63          return totalAmount;
64      }
65      public double getOrderId(){
66          return orderId;
67      }
68      public void printLineItems(){
69          for(LineItem l : lineitems){
70              if(l!=null){
71                  l.printDetail();
72              }
73          }
74      }
75  }
```

Class Diagram



Exercise 6: Create a ShoppingTest



```
3  public class ShoppingTest {
4      public static void main(String args[]){
5          Customer julia = new Customer("Julia", "Smith", 1001);
6          Member bob = new Member("Bob", "Roger", 2005);
7
8          Product pepsi = new Product("Pepsi 1-liter", 100, 35);
9          Product juice = new Product("Splash", 200, 15);
10         Product milk = new Product("NongPho", 300, 12);
11
12         LineItem l1 = new LineItem(pepsi, 50);
13         LineItem l2 = new LineItem(juice, 100);
14         LineItem l3 = new LineItem(milk, 200);
15         LineItem l4 = new LineItem(pepsi, 100);
16
17         Order order1 = new Order(julia);
18         order1.addLineItem(l1);
19         order1.addLineItem(l2);
20         order1.removeLineItemByArrayIndex(0);
21         //---order1---
22         order1.printLineItems();
23         order1.addLineItem(l1);
24         order1.printLineItems();
25
26         Order order2 = new Order(bob);
27         order2.addLineItem(l3);
28         order2.addLineItem(l4);
29
30
31         order2.printLineItems();
32
33         System.out.println(order2.getTotalAmount());
34         bob.setPoints(order2.getTotalAmount());
35         System.out.println(bob.getPoints());
36
37     }
38 }
39
```