Project

Class Product

```java
public class Product {
    3 usages
    private int productId;
    3 usages
    private String name;
    3 usages
    private float price;

    4 usages
    public Product(int id, String name, float price){
        this.productId = id;
        this.name = name;
        this.price = Math.abs(price);
    }

    no usages
    public int getProductId() { return productId; }

    no usages
    public void setProductId(int productId) { this.productId = productId; }

    4 usages
    public String getName() { return name; }

    no usages
    public void setName(String name) { this.name = name; }

    3 usages
    public float getPrice() { return price; }

    no usages
    public void setPrice(float price) { this.price = Math.abs(price); }
}
```

Class Electronic Product

```java
public class ElectronicProduct extends Product {
    3 usages
    private String brand;
    3 usages
    private int warrantyPeriod;

    1 usage
    public ElectronicProduct(int id, String name, float price, String brand, int warrantyPeriod){
        super(id, name, price);
        this.brand = brand;
        this.warrantyPeriod = Math.abs(warrantyPeriod);
    }

    no usages
    public String getBrand() { return brand; }

    no usages
    public void setBrand(String brand) { this.brand = brand; }

    no usages
    public int getWarrantyPeriod() { return warrantyPeriod; }

    no usages
    public void setWarrantyPeriod(int warrantyPeriod) { this.warrantyPeriod = Math.abs(warrantyPeriod); }
}
```

Class Clothing Product

```java
public class ClothingProduct extends Product {
    3 usages
    private String size;
    3 usages
    private String fabric;

    1 usage
    public ClothingProduct(int id, String name, float price, String size, String fabric){
        super(id, name, price);
        this.size = size;
        this.fabric = fabric;
    }

    no usages
    public String getSize() { return size; }

    no usages
    public void setSize(String size) { this.size = size; }

    no usages
    public String getFabric() { return fabric; }

    no usages
    public void setFabric(String fabric) { this.fabric = fabric; }
}
```

Class Book Product

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

    1 usage
    public BookProduct(int id, String name, float price, String author, String publisher){
        super(id, name, price);
        this.author = author;
        this.publisher = publisher;
    }

    no usages
    public String getAuthor() { return author; }

    no usages
    public void setAuthor(String author) {
        this.author = author;
    }

    no usages
    public String getPublisher() { return publisher; }

    no usages
    public void setPublisher(String publisher) { this.publisher = publisher; }
}
```

Class Customer

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

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

    1 usage
    public int getCustomerId() { return customerId; }

    no usages
    public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }

    no usages
    public String getName() { return name; }

    no usages
    public void setName(String name) { this.name = name; }

    no usages
    public String getAddress() { return address; }

    no usages
    public void setAddress(String address) { this.address = address; }
}
```

Class Cart

```java
import java.util.ArrayList;

2 usages
public class Cart {
    4 usages
    private int customerId;
    4 usages
    private int nProducts;
    8 usages
    private ArrayList<Product> products;

    1 usage
    public Cart(int customerId, int nProducts) {
        this.customerId = Math.abs(customerId);
        this.nProducts = Math.abs(nProducts);
        this.products = new ArrayList<>();
    }

    1 usage
    public ArrayList<Product> getProducts() { return products; }

    no usages
    public void setProducts(ArrayList<Product> products) { this.products = products; }

    no usages
    public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }

    no usages
    public int getCustomerId() { return customerId; }

    no usages
    public void setNProducts(int nProducts) { this.nProducts = Math.abs(nProducts); }
```

```java
    public int getNProducts() { return nProducts; }

    4 usages
    public void addProduct(Product product) {
        if (products.size() < nProducts) {
            products.add(product);
            System.out.println(product.getName() + " added to cart.");
        } else {
            System.out.println("Cart is full. Cannot add more products.");
        }
    }

    no usages
    public void removeProduct(String productName) {
        for (Product p : products) {
            if (p.getName().equals(productName)) {
                products.remove(p);
                System.out.println(productName + " removed from cart.");
                break;
            }
        }
        System.out.println(productName + " not found in cart.");
    }

    3 usages
    public float calculatePrice() {
        float total_price = 0;
        for (Product p : products) {
            total_price += p.getPrice();
        }
        return total_price;
    }
```

```java
    public void placeOrder() {
        float total_price = calculatePrice();
        if (total_price > 0) {
            System.out.println("Order placed for customer " + customerId + " with total price: " + total_price);
        } else {
            System.out.println("Cart is empty. Cannot place order.");
        }
    }
}
```

Class Order

```java
import java.util.ArrayList;

2 usages
public class Order {
    4 usages
    private int customerId;
    4 usages
    private int orderId;
    4 usages
    private ArrayList<Product> products;
    4 usages
    private float totalPrice;

    1 usage
    public Order(int customerId, int orderId, float totalPrice) {
        this.customerId = Math.abs(customerId);
        this.orderId = Math.abs(orderId);
        this.products = new ArrayList<>();
        this.totalPrice = Math.abs(totalPrice);
    }

    1 usage
    public int getCustomerId() { return customerId; }

    no usages
    public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }

    1 usage
    public int getOrderId() { return orderId; }

    no usages
    public void setOrderId(int orderId) { this.orderId = Math.abs(orderId); }
```

```java
    public ArrayList<Product> getProducts() { return products; }

    no usages
    public void setProducts(ArrayList<Product> products) { this.products = products; }

    no usages
    public float getTotalPrice() { return totalPrice; }

    no usages
    public void setTotalPrice(float totalPrice) { this.totalPrice = Math.abs(totalPrice); }

    no usages
    public void printOrderInfo(){
        System.out.println("Here's your order's summary: ");
        System.out.println("Order ID: " + orderId);
        System.out.println("Customer ID: " + customerId);
        System.out.println("Products: ");
        for(Product p : products){
            System.out.println(p.getName() + " - $" + p.getPrice());
        }
        System.out.println("Total Price: " + totalPrice);
    }
}
```