SOFT 7004 – OOP - Project

## Project: Completion Date: 14th December 2018

## On completion please zip up your files including any documents used for drawing the class diagram etc. Upload to BlackBoard.

You are to create an shopping application as outlined below. You are to create a UML class diagram which details all the java classes used in your application.

**PART 1**

You are to write the following java classes:

1.

A Product class. This will be a generic class for Products that a company may sell. It will have attributes for the following: name, description, price and productID. Use appropriate data types. Write getters and setters for all attributes. Write a method called print() that prints the Product information to the screen.

A unique productID is assigned to the product when it is created. You can use a static int to achieve this.  
  
2.

A Phone class. It will Inherit from the Product class. It will have attributes for make (Apple, Motorola, Samsung, etc), model (iPhone 6, Moto X, Galaxy S5, etc), and storage space (in gigabytes). Write getters and setters and Overload the print() method that's inherited from the Product class. Make use of the super.print() call.  
  
3.

A TV class. It will Inherit from the Product class. It will have attributes for make, screen size, type (LCD, LED, Plasma) and whether or not it is 3D capable. Write getters and setters and overload the print() method.

4.  
Write a ProductDB class which manages an ArrayList of Products. The class provides methods to add, remove and find a product. The find method returns a product with a matching productID.

5.

Write an Order class which manages an ArrayList of Products.

6.

Write a Customer class that has attributes name, address. A Customer object also has an ArrayList of Orders.

7.

Write a Test class that instantiates a number of phones and TVs and sets their attributes. Create some Customer objects. Create a ProductDB object. Add all your products to the object. Create some Order objects for a given Customer and add some orders to that Customer object. See example code below (Note not all parameters are shown).

*Phone p = new Phone(“Apple”, “iphone6”, ..); // all parameters not shown*

*Phone p1 = new Phone(“Samsung”,”Galaxy 8”, ...);*

*ProductDB database = new ProductDB();*

*database.add(p1);*

*database.add(p);*

*Customer Mary = new Customer(“Mary” ..);*

*Order o = new Order();*

*o.add(p,12); // ordered 12 iphone 6 products*

*o.add(p1,1); //ordered 1 Galaxy 8*

*Mary.addOrder(o);*

**PART 2**

You are to create a UML class diagram which details all the java classes used in your application.

**PART 3**

Provide a menu in your Test class which has the following options:

1. Create a new phone.

2. Create a new Customer.

3. Search for a product by supplying the productid.

4. Display all products in the database.

5. Allow a customer to order some products.

6. Display all orders that a customer has made and all the products that are in a given order.

7. Display all orders for a given product by supplying the productid.

8. Quit

**Example run of the menu system user input shown in bold:**

1. Create a new phone.

2. Create a new Customer.

3. Search for a product by supplying the productid.

4. Display all products in the database.

5. Allow a customer to order some products.

6. Display all orders that a customer has made and all the products that are in a given order.

7. Display all orders for a given product by supplying the productid.

8. Quit

**5**

Enter the customers name

**Joe**

Enter a product id and a quantity. Enter -1 to finish

**2 3**

You ordered 3 iphones6

Enter a product id and a quantity. Enter -1 to finish

**3 2**

You ordered 2 galaxy s6

Enter a product id and a quantity. Enter -1 to finish

**-1**

Please enter a menu option

1. Create a new phone.

2. Create a new Customer.

3. Search for a product by supplying the productid.

4. Display all products in the database.

5. Allow a customer to order some products.

6. Display all orders that a customer has made and all the products that are in a given order.

7. Display all orders for a given product by supplying the productid.

8. Quit

**5**

Enter the customers name

**Joe**

Enter a product id and a quantity. Enter -1 to finish

**2 1**

You ordered 1 iphones6

Enter a product id and a quantity. Enter -1 to finish

**3 1**

You ordered 1 galaxy s6

Enter a product id and a quantity. Enter -1 to finish

**-1**

Please enter a menu option

1. Create a new phone.

2. Create a new Customer.

3. Search for a product by supplying the productid.

4. Display all products in the database.

5. Allow a customer to order some products.

6. Display all orders that a customer has made and all the products that are in a given order.

7. Display all orders for a given product by supplying the productid.

8. Quit

**6**

Enter the customers name

**Joe**

**The Orders for Joe are as follows:**

**Order 1:**

3 iphones6

2 galaxy s6

**Order 2:**

1 iphones6

1 galaxy s6

Please enter a menu option

1. Create a new phone.

2. Create a new Customer.

3. Search for a product by supplying the productid.

4. Display all products in the database.

5. Allow a customer to order some products.

6. Display all orders that a customer has made and all the products that are in a given order.

7. Display all orders for a given product by supplying the productid.

8. Quit

**8**

Goodbye