**iTech Technologies**

**Date: 11 Jul 2020 Time: 2.30 Hrs**

Instructions:

 Must read the problem statement thoroughly before start working.

 Give project name as yourname\_ year\_001 (Ex. Rohinth\_2020\_001)

 Create all java classes in package “**com.itech**”.

 Use exactly same class names as mentioned.

 Use exactly same method signature (method name, return type, method parameter type)

 Define constructors and getter setters.

 Must follow java coding standards.

 Ensure attributes are private and other methods which will be called from main method, getter-setter methods and constructor are public.

 Use main method only for input and output and testing object creation and object methods.

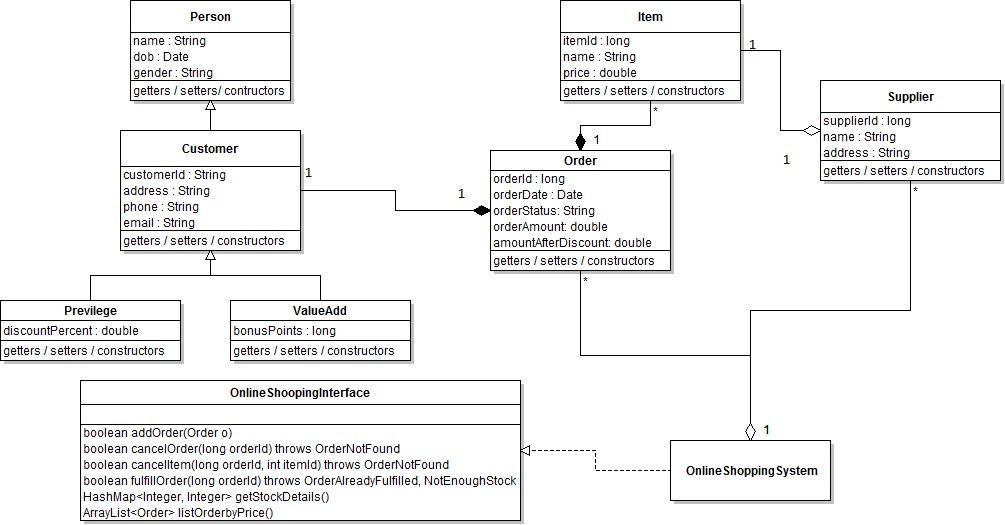
 Make sure your java project must be free from compilation error, ie. Zero compilation error. In case of any compilation error reported, the project is rejected, will not be considered for assessment.

 Copy that project once you have completed to desktop, zip the folder(*eg.*

Rohinth\_2020\_001.zip), mail it.

**As mentioned above, any logic which may be 100% correct is not valid if above points are not taken care. Hence, simply building logic does not certify us as project ready. Building exact and complete solution does.**

**Class Diagram:**



**ABC Cart** is an OnlineShoppingSystem for ordering different items through web. It consists of two types of customers;

**1. Privilege Customers:** In the case of privilege customers, a fixed percentage is provided against

each purchase. This will be automatically applied against the total order amount and find the

*amountAfterDiscount*.

2. **ValueAdd Customers** : In the case of ValueAdd customers, bonus point is allocated against each purchase, this they can used for their purchase. Bonus point is allocated in the following manner;

 If the total purchase amount is between 1000 – 2000 allocating a bonus point of 50.

 If the total purchase amount is between 3000 – 5000 allocating a bonus point of 100.

 If the total purchase amount is between 5000 – 10000 allocating a bonus point of 200.

 If the total purchase amount is greater than 10000 allocating a bonus point of 300. Against each purchase they can utilize maximum 10% of the amount as bonuspoint. (Suppose

the customer total order amount is Rs. 6000/-, they can use Rs. 600 from the bonus point) After

redeeming the points the total need to update accordingly.

**1. boolean addOrder(Order o)**

This method adds the new order to the ABCCart. After the successful processing it will set that status of Order as *In-progress* and return a true otherwise it will return false.

**2. boolean cancelOrder(long orderId) throws OrderNotFound**

This method is used to cancel an existing order by mentioning the *orderId*. It will remove the particular order from the *orderList*. If the given order doesnot extist then throw **OrderNotFound** exception. We can cancel only before fulfilling the same.

**3. boolean cancelItem(long orderId, int itemId) throws OrderNotFound**

This method is used to partial cancellation. That is we can cancel a particular item from a list of items against an order. After canceling that item update the order amount accordingly. It will throw an exception **OrderNotFound** if the *orderId* not exist.

**4. boolean fulfillOrder(long orderId) throws OrderAlreadyFulfilled, NotEnoughStock**

This method is used to fulfill a particular order from Suppliers. After fulfilling the order change the status to ***Fulfilled***. After fulfillment, the corresponding quantities need to deduct from the supplier stock. Throw the respective exception in the case of invalid scenarios. If the Order is already fulfilled, then throw **OrderAlreadyFulfilled exception.** Similarly if there is no enough enough stock available in suppliers to fulfill a request throw **NotEnoughStock** exception.

**5. HashMap<Integer, Integer> getStockDetails()**

This method is used to get the stock summary available for every item in the ABC Cart suppliers. If more than one suppliers distributing the same item, then quantity available against that particular item should be the sum of them.

**6. ArrayList<Order> listOrderbyPrice(String status)**

This method list all the orders in ABC Cart with mentioned status available in the system in the descending order of the *orderAmount.*

**NB: Write JUnit test case for all the methods in OnLineShoppingSystem (service class).**

**Zip Procedure:**

**1. In eclipse right click on your project.**

**2. Select copy, then on desktop paste it.**

**3. Now, on desktop your project folder will be available. Right click that folder and select, send to -> compressed folder.**

**4. Now you can find compressed file (zip) of your project folder.**