

Software Engineer Off-site Problem

Introduction

Assume you are working on a new warehouse inventory management system named IMS. IMS will be responsible for the inventory tracking within physical, single site warehouses. IMS will track the named physical location of a product within the warehouse and the inventory level of each product. IMS will be deployed to busy warehouses supporting many pickers and restockers working with individual terminals and clients. Updates to inventory levels will be handled in real time to prevent pickers trying to pick a product that is out of stock.

Assumptions

- Each product will be stored at one and only one named location within the warehouse.
- Inventory adjustments may be additive (restocks) or subtractive (picks).
- No additional product information needs to be tracked beyond location and level.

Problem

In Java, implement the picking and restocking routines for the IMS system. The ***InventoryManagementSystem*** interface will be the first component to be implemented; all relevant domain objects will have to be implemented. The ***InventoryManagementSystem*** interface has already been distributed to other teams which depend on it.

InventoryManagementSystem.java

```
/**
 * Copyright (c) 1999-2016 Commerce Technologies Inc. All rights reserved.
 * Implementations of this interface including access to shared data must be
 * thread-safe.
 */
public interface InventoryManagementSystem {
    /**
     * Deduct 'amountToPick' of the given 'productId' from inventory.
     * @param productId The ID of the product to pick
     * @param amountToPick The quantity of the product to pick
     * @return TODO: to be implemented
     */
    PickingResult pickProduct(String productId, int amountToPick);
    /**
     * Add 'amountToRestock' of the given productId to inventory.
     * @param productId The ID of the product to restock
     * @param amountToRestock The quantity of the product to restock
     * @return TODO: to be implemented
     */
    RestockingResult restockProduct(String productId, int amountToRestock);
}
```

Deliverable

Please follow the submission instructions described in the email. Zip all compilable Java source code which implements the ***InventoryManagementSystem*** interface and any additional artifacts, test and domain classes used to solve the problem. It is not necessary to submit .class or IDE project files.

Constraints

- Please work on your submission independently. Feel free to email any questions you may have that come up.
- Please do not share your work.

- No external dependencies beyond the standard JDK packages and a Java test framework (e.g. JUnit, TestNG, etc.) may be used.

On-Site Expectations

If your solution warrants a follow-on discussion, be prepared to discuss broader topics of IMS and details of your design and implementation.