

### **Development Technologies:**

- ➔ Spring 4, Hibernate 4, MySql 5, Tomcat 8, Java 8, Maven
- ➔ Your implementation should be TDD using TestNG, Mockito, AssertJ

### **Submission Notes:**

- ➔ Submission of task email should be sent to [husnain@easy2employ.com](mailto:husnain@easy2employ.com) with the subject format like:  
[ASSIGNMENT\_NAME]\_[YOURNAME]\_[JOBPOSITION]
- ➔ In the email body, you need to submit github repository url (make sure to have a public git repository)
- ➔ Your github repository should have README file of how to configure your project and also explaining the overall architecture of your application.
- ➔ Your code should be well documented as per Java documentation styles and should be clear enough to understand the purpose of implemented function.

### **Questions:**

If you have any questions related to task, you can shoot an email right away to [husnain@easy2employ.com](mailto:husnain@easy2employ.com)

### **Problem:** Product&Warehouse&Inventory (PWI)

Company A has two brands Brand1 and Brand2 where each brand has multiple products with multiple sizes. Company has their offices in following locations:

- ➔ USA

- ➔ Ireland
- ➔ Netherlands
- ➔ Dubai
- ➔ Australia
- ➔ Italy
- ➔ Pakistan
- ➔ Mexico

In future, they can start a new office in some other location as well. They have warehouses where they manage their inventory. Let's suppose they have warehouses in USA, Ireland, Netherlands, Dubai, Australia, Italy. Each warehouse can have all products or some products and they manage their inventory according to the warehouse inventory. There are some technical terms being used in Company A which are:

**In Stock:** Number of products physically available.

**Available Quantity:** Number of products which can be sold (This can be different than In Stock quantity since in stock is physical quantity but all physical quantity might not be available for next order to process)

**In Transit:** Quantity ordered by the Company A from some 3<sup>rd</sup> party company and 3<sup>rd</sup> party company has shipped the quantity but it's not yet received by Company A.

**MOQ:** Minimum order quantity means if Company A wants to purchase this particular product, a minimum of this much quantity needs to be ordered.

**QPB:** Quantity Per Box is the quantity of the product that can be added in 1 box

**Reorder Point:** Minimum quantity when system should alert to place an order for this product.

A product can be of 3 types:

i). Finished Product

ii). Component

iii). Packaging Material

Components are not sellable and only internal to the company and same goes for packaging material. Only Finished product can be sold via Ecommerce Company A has. Each product, component and packaging material can have different sizes.

For example:

Product A can be available in 3 sizes. 10 ml, 20 ml, 30 ml

OR

Component A can be available in 2 sizes: Small, Large etc

Task:

Company A requires to implement a solution so that they could manage their products, components, packaging material and their sizes and their inventory per warehouse and per office.

For example:

Product A can be in USA and Ireland warehouses only and USA let's suppose has 3 warehouses: Warehouse1, Warehouse2 and Warehouse3 where as Ireland has 2 warehouses Warehouse4, Warehouse5. And the inventory of Product A is given as follows according to the Office and warehouse.

**USA**

<b>Warehouse</b>	<b>Item</b>	<b>Size</b>	<b>In Stock</b>	<b>Avl. Qty</b>	<b>In Transit</b>	<b>MOQ</b>	<b>QPB</b>	<b>Reorder Point</b>
Warehouse1	ProductA	10 ml	10	8	5	50	6	15
Warehouse2	ProductA	20 ml	900	100	1000	1000	12	100
Warehouse3	ProductA	30 ml	150	90	80	50	36	45

Warehouse3	ProductA	10 ml	100	50	30	45	50	100
Warehouse2	ProductA	30ml	40	32	12	34	43	9

## Ireland

Warehouse	Item	Size	In Stock	Avl. Qty	In Transit	MOQ	QPB	Reorder Point
Warehouse4	ProductA	10 ml	10	7	5	1	4	6
Warehouse4	ProductA	30 ml	100	100	0	5	36	10
Warehouse5	ProductA	10 ml	10	0	100	10	8	10

Your task is to design database so that this purpose could be fulfilled and implement backend restful services to manage their inventory. Make sure that your backend services are a complete package so that these could just be integrated with any front end and it starts adding a product, warehouse, office, sizes and inventory etc that fulfills this purpose. You have to implement following endpoints:

1. Add/Edit/Delete Item (Product, Component, Packaging Material)
2. Add/Edit/Delete Warehouse
3. Set Item Quantity in Single/All Warehouse(s) of Company or any Office(s).
4. View Item Quantity in Single/All Warehouse(s) of Company or any Office(s).
5. See all available sizes of any Item

**Note: Make sure to implement test cases (unit, system) in your development cycle as well.**