

**Candidate Skills and Knowledge Assessment**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Welcome!**

As part of our hiring process we have created a Candidate Skills and Knowledge Assessment. There are two exercises designed to test your skill in the development environment as well as your analytical skills. We understand the different skill levels of candidates and will evaluate accordingly.

We hope you have fun with these exercises. Ensure to read through the entire exercise before starting.

**Please complete the exercises within twenty-four hour (24-hrs) and e-mail to:** [**Rennae.baker@cimcorp.com**](mailto:Rennae.baker@cimcorp.com)

For questions or additional information you can e-mail [Rennae.baker@cimcorp.com](mailto:Rennae.baker@cimcorp.com)

**Guidelines:**

Exercise 1 should be done with the selected development environment(s). If you are comfortable in both, you may select to do both even if not selected.

If you do not have access to the selected development environment, make a draft of the program. Programs made in other development environments can also be sent as text files. Please indicate what environment was used.

JAVA:

Eclipse 4.6.3 using Java

Visual Studio .NET:

Version 2015 using VB.Net or C#.Net

Exercise 2 will not require a development environment.

***Thank you for taking the time to complete this assessment.***

Exercise 1: ABC Order Analysis

**Objective:**

Read the CSV file Orderlines.csv and create a new CSV file, ABC.csv, with sum of ordered products by product number ordered by total quantity descending.  
  
When completed the ABC.csv file can be opened in Microsoft Excel and be used to make a bar chart that looks similar to “Figure 1”.

**Instructions:**

Attached is the file Orderlines.csv, which contains data about orders. Each row of information in the file equals to one product order line. Each row contains three data items, separated by a semicolon. These items are:

466142;"Product\_01083";1

Ordered quantity

(Numeric)

Order number

(Numeric)

Product number

(Alpha-Numeric)

Make an ABC-analysis from the aforementioned data. Write a program that reads the rows of the Orderlines.csv file and sums the ordered quantity of products, by product and then saves the result in the file called ABC.csv. Each row in the ABC file contains the product number and the sum of the ordered products for that product number.

The products should be sorted from most ordered product, to least ordered product. Example of what the start of the file should look like:

"Product\_25157";3588

"Product\_69130";2967

"Product\_69132";2757

**Tests:**

1. Manually verify some of the entries in the ABC.csv file to ensure correct sums.
2. Import the data in Excel and verify grid shows products, values with quantity High to Low (with no sorting in Excel) similar to Figure 1.
3. Optionally create your own Orderlines.csv and test.

Figure 1

**Considerations:**

The structure of the program should be easily modified. In regards to, at least the following sections (You are not required to implement these):

* The reading of the order line data should be easily exchangeable to be read from a database or an xml-document.
* It should be easy to change the save location and type of document saved; e.g. a database or an xml-document.
* You are not required to do error handling.

**Please Submit:**

* The time it took to complete the Exercise (Code Time and Test Time can be separated).
* Test Results.
* Project Files.
* ABC.csv file.

**Congratulations - Exercise 1 is Complete!**

**Exercise 2: Customer Pallet Planning**

**Objective:**

Design a program that gets the customer’s order as input, as in how many crates of product the customer has ordered, and decides how pallets are configured. The output should include; number of Pallets, stacks on each pallet, and crates in each stack.

**Instructions:**

Let us assume that we are making a picking system that collects and stacks plastic crates onto pallets. One pallet can have at most four stacks, and a stack has a maximum height of 1200 mm. The customer has crates of three different heights, with all the same footprint. The heights are as follows: 150 mm, 180 mm and 250 mm. A given product is always in the same crate. All crates can be stacked on each other, and there are no limitations as to what order the products can be stacked.

Briefly document how you would implement a program like this, no coding is expected or required.

**Considerations:**

* As few pallets as possible are used to complete the order.
* The height difference between pallets, and stacks on a pallet, cannot exceed 300 mm.

**Please Submit:**

* The time it took to complete the Exercise
* Microsoft Word, PDF or Text document with design concept.

**Congratulations - Exercise 2 is Complete!**