Programming Test Instructions

This document is for candidates to the developer positions. The only purpose is to define requirements for a programming test. There are sections named “what to do” and “how to do”. Both of them are discussed below.

# What to do

You are required to write simple project for people management. There are 2 required methods – addPerson and calculateAverageAge. Code needs to be split into 2 layers – business and data access. Methods are defined in the business layer, but inside they HAVE TO use data access layer. This fact is not mentioned in the texts later.

**AddPerson** method simply stores person in memory and is there until JVM finishes. Every call with valid parameters adds one record. This method accepts parameters **name**, **age** and **gender**. Implementation takes these parameters and stores person if it is valid. Validation rules are the following:

* Name must be in format “First Last” (examples: John Seaman, Caroline Horse)
* Age must be not negative integer (examples: 0, 1, 2…)
* Gender is one of the following – MAN, WOMAN
* If age is more than 50, then first name cannot be John (for example John Seaman cannot have 55 years, but can have 35 years)
* If gender is WOMAN, then first name must end with “ko” (for example Tomoko Seaman is valid for WOMAN, Caroline Seaman is NOT valid for WOMAN)

Objects which don’t satisfy the constraints cannot be stored.

**CalculateAverageAge** is method which accepts string parameter called subname. This method finds all people who has name which contains sub string equal to subname and returns average of their age.

Example:

Following objects are in memory:

* John Seaman – 30 – MAN
* Peter Seaman – 40 – MAN
* Joko Uehara – 20 – WOMAN

Now **calculateAverageAge(“Seaman”)** = 35, **calculateAverageAge(“Jo”)** = 25 and **calculateAverageAge(“e”)** = 30.

# How to do

You are given the maven project. Please use your favorite IDE and work with it. There is a java class call **PersonController** which is the starting point. After the test is finished, please run mvn clean, remove all IDE generated files, create a zip archive and send it to the agreed e-mail address. If you have anything you couldn’t finish, because of time limit, please sum it up to the readme.txt file (for example “I would make it more efficient by…”).

How to design data access layer? The interface for data access layer is up to you. Requirement is that all components must be testable as units. Implementation must store all objects in the memory.

What about comments and unit tests? They play significant part in this test, please put your best effort to proper comment and unit test everything.