Software Engineering
MSc in Engineering in Computer Science
Sapienza Università di Roma

Programming test – 21 July 2021 – Duration 150 minutes

Name _____

Surname _____

Matricola _____

☐ *OK* _____

☐ *KO* _____

On the instructor machine (IP address 192.168.50.83), on the port 61616, it is deployed a JMS provider. The JMS provided is Apache ActiveMQ (version 5.16.1).

Each student can interact with a topic named `dynamicTopics/students`, in which messages are published. Each message has a property `student` of type *String,* `name` of type *String* (corresponding to the name of a stock) and a body text (to be freely composed by the student). The `student` should be set by each student as her/his own *matricola* (INFOSTUD number).
Therefore, the student Massimo Mecella with matricola 6666666 should set the property `student` equal to '6666666' on all messages published by him.

In addition, the student is provided with a standalone `jar` file which creates a database, if the option passed in the command line is `create`, or it queries and prints all the tuples in the database, if the option is `show`. The pathname of the database is passed as option in the command line (it is suggested to use `/home/biar/database`, but any other pathname is fine). The database has a single table `stocks` with two columns `id` and `name` both of type *String*.
The following command line shows how to launch it:
```
% java -cp DBManager-1.0-SNAPSHOT.jar
it.sapienza.softeng.exam_20210721_dbmanager.DBManager
/home/biar/database create
```
or
```
% java -cp … /home/biar/database show
```

The following snippet might be used in the `pom.xml` for using SQLite:
```
<dependency>
    <groupId>org.xerial</groupId>
    <artifactId>sqlite-jdbc</artifactId>
    <version>3.28.0</version>
```

```
</dependency>
```

The student is required to develop:

- a publisher of the stocks, which continuously select a stock from the database, and sends a text message in the topic. The text message should set the property `student` as the matricola of the student, and `name` as the name of the stock selected from the database. As an example, the student Massimo Mecella who has selected from the database the record [12, Oracle], should publish the message {6666666, Oracle, …}. Messages should be published continuously at a given frequency (a few seconds) and the selection process should be chosen by the student (it can be random, it can be a scan over all the records of the database and then reiterate, etc.)

- a subscriber of the stocks, which subscribes to the topic `dynamicTopics/students` and receives **all and only those messages sent by the publisher** of the student and print them in the console output. It is important to note that, as the topic is shared among all students, you potentially might receive all messages, conversely you should filter them out based on your matricola (suggestion: use a selector over the property `student` of the messages).