INSTALLATION GUIDE:

GIT REPOSITORY

https://gitlab.gbar.dtu.dk/s192675/02267-microservices
https://github.com/casperbh96/02267-microservices

REQUIRED TOOLS FOR INSTALLATION

- Java 8 JDK (https://adoptopenjdk.net/installation.html):
  - `wget -qO - https://adoptopenjdk.jfrog.io/adoptopenjdk/api/gpg/key/public | sudo apt-key add`
  - `sudo add-apt-repository --yes https://adoptopenjdk.jfrog.io/adoptopenjdk/deb/`
  - `apt-get install adoptopenjdk-8-hotspot`

- Maven
  - Mac/Linux:
    - `sudo apt install maven`
    - If that does not work out of the box you have to update the contents of the apt repositories with `sudo apt update`
  - Windows:
    - Maven can be installed here: https://maven.apache.org/download.cgi

- Docker
  - On Linux:
    - 1. Update the repositories: `sudo apt update`
    - 2. Install docker: `sudo apt install docker.io`
    - 3. Install docker-compose: `sudo apt install docker-compose`
  - On Windows/Mac (http://www.docker.com)
- Python
  - On Linux:
    - `sudo apt-get install python3`
  - On Mac:
    - `brew install pyenv`
    - `pyenv install –list | grep 3.8` (choose the version with "-dev")
    - `pyenv install 3.8-dev`
  - On Windows (https://datatofish.com/install-python/)

JENKINS BUILD SCRIPT

The whole build script is inside a Jenkinsfile that has different stages to build and test all the different microservices.

- On the first stage, we start by building all the micro services.

- On the second stage, we build and run a docker image.
- Then, we wait and make sure the docker image is up before running our tests.
- After that, we run the Junit and Cucumber tests for each of the services.
- And finally, we clean and take down image.

NEEDED SCRIPTS

Clone one of the repositories on Github or Gitlab:

```
git clone https://gitlab.gbar.dtu.dk/s192675/02267-microservices.git
git clone https://github.com/casperbh96/02267-microservices
cd 02267-microservices
```

The following script file runs the maven scripts for all the microservices:

```
mvn_install_all.sh
```

If not with the script, you can also run each of the microservices separately:

```
mvn -f CustomerMicroservice/pom.xml -D maven.test.skip=true install
mvn -f MerchantMicroservice/pom.xml -D maven.test.skip=true install
mvn -f TokenMicroservice/pom.xml -D maven.test.skip=true install
mvn -f TransactionMicroservice/pom.xml -D maven.test.skip=true install
mvn -f DTUPay/pom.xml -D maven.test.skip=true install
```

Generate the docker images for each of the microservices:

```
docker-compose build
docker-compose up -d
```

Run tests on all the services:

```
mvn -f CustomerMicroservice/pom.xml test
mvn -f MerchantMicroservice/pom.xml test
mvn -f TokenMicroservice/pom.xml test
```

Delete the docker images:

```
docker-compose down
docker-image prune -f
```

INFORMATION ON SOFTWARE TOOLS USED FOR DEVELOPMENT

- IntellJ (2019.3.1)
- Maven (3.6.3)
- Cucumber (1.2.5)
- JUnit (4.12)
- MySQL (8.0.18)
- Thorntail (2.6.0.Final)
- Jersey (2.27)
- Org.json (20190722)