

Évaluation ITII25-Info-LangageJava

DOCUMENTATION

Groupe :

DACHEUX Corentin, DUVAL Damien, RAMIREZ
SANTILLAN Brenda, DESJARDIN Paul

2024

Lien GIT du projet :

https://github.com/damienduval691/TP_NOTE_RAMIREZ_DESJARDIN_DACHEUX_DUVAL

1) Installation et configuration de notre programme

Installation des éléments liés à Docker

Pour commencer, vous devrez vous assurer que le logiciel Docker est correctement installé et lancé sur votre ordinateur.

Une fois le logiciel lancé, ouvrez IntelliJ puis ouvrez notre projet. Vous trouverez alors plusieurs dossiers dans « src/main/java/fr/itii25 ». Vous y trouverez trois dossiers correspondant à chacune des options qui nous ont été proposées pour cette évaluation, ainsi qu'un dossier « install » permettant d'installer les composants nécessaires dans votre Docker.

Dans un premier temps, il faut parcourir ce dossier jusqu'au fichier nommé « InstallationBaseDeDonnees », puis lancer le programme qui s'y trouve. Les images des bases de données dont nous avons besoin seront alors installées dans votre docker. Attention, en cas d'erreur, merci de rentrer manuellement les commandes suivantes dans un terminal sur IntelliJ :

- Téléchargement de la base de données PostGRE :
 - **docker pull postgres**
- Téléchargement de la base de données MySQL :
 - **docker pull sakiladb/mysql**

Une fois ces deux images installées sur votre ordinateur, il vous suffira de trouver le fichier nommé « InstallationBaseDeDonnees » dans le dossier « install » afin de lancer les deux bases de données. Exécutez ce fichier si le premier essai ne fonctionne pas. Merci de rentrer manuellement les deux commandes suivantes :

- Lancement du conteneur PostGRE :
 - **docker run -rm -it -p 5432:5432 -d POSTGRES_USER="user" -e POSTGRES_PASSWORD="admin" postgres:latest**
- Lancement du conteneur MySQL :
 - **docker run -rm -it -p 3306:3306 -d sakiladb/mysql:latest**

Pour vérifier le bon lancement des deux conteneurs, utilisez la commande « docker ps » qui doit afficher un docker pour chaque base de données.

Pour conclure, une fois ces commandes correctement exécutées, deux bases de données sont en train de fonctionner dans l'outil Docker. Celles-ci sont utilisées dans les options 2 et 3 pour simuler le transfert de données depuis une base de données sources vers une base de données de destination.

2) Utiliser notre programme

Utiliser l'option 1

Prérequis :

- Environnement JAVA 23
- Logiciel IntelliJ

Guide :

1. Exécutez le programme. <src/main/java/fr/itii25/option1/main/Utilisation.java>
2. Dans le terminal, saisissez un message autre que « fin ». Votre message sera ensuite affiché sous la forme « Données reçues : [Votre message] ».
3. Entrez le message « fin » pour arrêter le programme. Le message « Tâche Réceptrice arrêtée. » sera alors affiché.

Utiliser l'option 2

Prérequis :

- Environnement JAVA 23
- Le logiciel Docker et IntelliJ
- Une connexion internet

Guide :

1. Suivez le guide d'installation.
2. Exécutez le programme. <src/main/java/fr/itii25/option2/main/Utilisation.java>
3. (Optionnel) Consultez votre base de données Postgres afin d'y apercevoir la table « Actor » issue de la base de données Sekila.
4. Saisissez « fin » pour fermer le programme. La table « Actor » sera affichée.

Utiliser l'option 3

Prérequis :

- Environnement JAVA 23
- Le logiciel Docker et IntelliJ
- Une connexion internet

Guide :

1. Suivez le guide d'installation.
2. Exécutez le programme. [src/main/java/fr/itii25/option3/main/Utilisation.java](#)
3. Choisissez une option parmi les suivantes :
 - a. Entrez « 1 » pour importer une table de la base de données Sekila dans la base de données Postgres.
 - b. Entrez « 2 » pour consulter une table la base de données Postgres.
 - c. Entrez « 3 » pour supprimer une table parmi la liste de la base de données Postgres.
 - d. Entrez « fin » pour fermer le programme.

FERMER NOTRE PROGRAMME**Fermer les conteneurs Docker :**

Une fois que vous avez fini d'utiliser le programme, entrez la commande dans le terminal suivante afin de libérer les conteneurs :

docker stop \$(docker ps -q)