

## **DM Java à me rendre au plus tard le vendredi 29 janvier 2021**

L'exercice a pour objectif de réaliser un convertisseur de devises en se basant sur des framework que vous avez/allez rencontrer dans l'entreprise. Le projet a les contraintes suivantes :

- Le cœur de l'application se base sur un main Spring Boot avec une classe implémentant CommandLineRunner. Exemple ici : <https://www.baeldung.com/spring-boot-console-app>
- La gestion des dépendances est assurée par Maven, possible de créer un pom.xml depuis le site <https://start.spring.io>
- Me rendre l'ensemble dans un zip avec des scripts permettant de compiler et exécuter le programme sous Linux(ou Mac) et Windows. Vous pouvez faire la supposition que Maven(mvn) et Java sont dans le path donc résolu sans chemin absolu.

Ne pas hésiter à utiliser les profils Spring et d'y définir les variables dont vous aurez besoin.

Concernant le code proprement dit, voilà les différentes étapes à implémenter :

### **1. Compilation et exécution du projet**

- A la racine du projet, mettre à disposition deux scripts compil.bat et execute.bat pour compiler et exécuter sous Windows.
- A la racine du projet, mettre à disposition deux scripts compil.sh et execute.sh pour compiler et exécuter sous Linux/Mac.

En cas d'échec des scripts, j'importerais le projet dans un IDE. Vous n'aurez cependant pas les points correspondant à cette partie.

### **2. Import des taux de conversion dans une base de données en mémoire**

Le site suivant propose une API REST renvoyant les données json dont nous avons besoin. Voilà l'URL appelée :

<https://api.exchangeratesapi.io/latest>

Vous pourrez par exemple utiliser la classe RestTemplate du framework Spring pour récupérer les taux de conversion en utilisant le verbe GET sur ce même URL. Un exemple simple et didactique ici :

<https://www.baeldung.com/rest-template>

Pour ceux souhaitant rester à la pointe du progrès, ne pas hésiter à utiliser la nouvelle API WebClient. Le choix de l'API utilisé est libre et n'apporte aucun point supplémentaire.

Votre mission est d'utiliser un POJO avec le mapping adéquat comme dans l'exemple du lien. Vous allez ensuite exploiter l'information du POJO pour alimenter une base de données en mémoire. A vous de définir la database et les tables pertinentes. Concernant l'usage d'une base de données mémoire vous pourrez vous inspirer de l'exemple vu en cours.

### **3. Affichage d'une sortie console du contenu de la DB**

Dans cette étape, l'objectif est de proposer une sortie console propre mettant en évidence les données de la table.

### **4. Afficher quelques exemples de conversions de devises A l'aide des taux**

Effectuer quelques exemples de conversion de devise en vous appuyant sur des données extraites de la DB.

### **5. Proposer sur l'entrée console à l'utilisateur de réaliser une conversion d'une devise dans une autre, ou de quitter l'application.**

Proposer une entrée console afin que je puisse effectuer des conversions d'une devise dans une autre. L'option de quitter l'application doit également m'être proposée.

Lorsque vous êtes satisfait de votre réalisation vous pouvez me l'envoyer par mail à l'adresse [cpiriou@fr.ibm.com](mailto:cpiriou@fr.ibm.com). Le DM est à me rendre au plus tard le 29 janvier 2021.