

## BRIEF PROJET

### *Analyser les ventes d'une PME*

Projet pour la préparation à la journée de sélection pour le parcours de formation Data Engineer.

## Objectifs visés

- Créer et mettre en œuvre une architecture avec deux services : un d'exécution de scripts et un autre de stockage de données
- Explorer et qualifier un jeu de données et en expliquer ses caractéristiques
- Créer une base de données cohérente avec les données qualifiées
- Importer des données à partir de fichiers distants
- Réaliser un premier niveau d'analyses de données avec SQL
- Stocker les résultats des analyses

## Durée estimée

1 journée.

## Énoncé du brief

### Scénario

Vous êtes technicien·ne système et vous assistez un·e data engineer, au sein d'une PME de services numériques. Un client de votre entreprise souhaite mieux comprendre la dynamique de ses ventes afin d'améliorer leur prise de décision stratégique.

Dans un premier temps, votre responsable data engineer vous demande de :

- réaliser une analyse préalable afin de mieux cadrer le projet final. Pour cela, le client vous a fourni [un extrait de leur jeu de données](#) des ventes sur 30 jours et des données relatives à ses [produits](#) et à ses [magasins](#) distribués dans plusieurs villes de France.
- concevoir une architecture en deux services :
  - un premier service pour l'exécution des scripts dans le langage de programmation de votre choix.
  - un deuxième service pour le stockage des données en base de données avec SQLite.
- implémenter l'architecture conçue :
  - créez les fichiers Dockerfile et Docker Compose
  - assurez la bonne exécution de l'architecture

Vous devez prendre connaissance des données partagées et en comprendre les principales caractéristiques pour créer le schéma de la base de données avec ces tables et relations. Une fois le schéma créé vous devez :

- créer une base de données avec ses tables associées afin d'importer les données partagées par le client dans la base de données
- effectuer une première analyse des ventes avec SQL
- stocker les résultats des analyses .

Comme le fichier de données sur les ventes s'actualise en temps réel, vous devez vous assurer que les lignes de données à importer ne sont pas déjà stockées en base de données.

## L'organisation de votre travail

### Concevoir l'architecture

Créer sous forme de schéma une architecture qui identifie :

1. Le service d'exécution des scripts et le service de stockage de données avec :
  - a. le nom du service
  - b. l'objectif du service
2. les entrées et sorties de communication entre les services indiquant
  - a. les ports exposés
  - b. le sens de la communication

### Réaliser l'architecture

1. Créez le Dockerfile pour créer l'image du service qui exécutera les scripts. Prenez en compte ce qui suit :
  - a. trouvez une image de base avec l'environnement d'exécution que vous avez choisi. Par exemple, si vous décidez d'écrire les scripts en Python, votre image de base pourrait être celle-ci : [python - Official Image | Docker Hub](#)
  - b. créez un script "hello-world" dans le langage de programmation de votre choix et tester sa bonne exécution
2. Créer le fichier Docker compose qui orchestre le lancement des deux services. Prenez en compte ce qui suit :
  - a. Pour lancer le service qui stocke les données, utilisez une image Docker avec SQLite3 (vous pouvez en trouver une sur Docker Hub) et assurez-vous qu'elle fonctionne lorsque vous la lancez localement (par exemple, vous pouvez accéder à la ligne de commande SQLite dans le conteneur).
  - b. Assurez-vous que le service qui exécute les scripts s'exécute après le démarrage du service qui stocke les données.

### Réaliser une analyse des données préalable

1. Créez le schéma de la base de données :
  - a. les tables définies par un nom de table, les attributs de la table et la clé primaire sous forme de diagramme
  - b. les relations entre les tables inclus sur le diagramme
  - c. n'oubliez pas de créer les tables nécessaires pour pouvoir ajouter les résultats des analyses à la base de données.
2. Intégrez dans le script "hello-world" (vous pouvez le renommer si vous voulez) la logique pour créer les éléments suivantes :

- a. La base de données
  - b. Les tables
3. Intégrez dans le script (ou en créant un autre) la logique pour importer les données partagées par le client dans les tables de la base de données :
  - a. Collecter les fichiers de données directement à partir des URLs partagées par le client. Vous devrez effectuer les requêtes HTTP.
  - b. Pour les données de vente, créez une condition dans le script pour n'importer que les nouvelles données qui ne sont pas déjà dans la base de données.
4. Créez les requêtes SQL pour répondre aux questions clés sur les ventes de l'entreprise :
  - a. le chiffre d'affaires total,
  - b. les ventes par produit.
  - c. les ventes par région.
5. Stockez les résultats des requêtes SQL dans la base de données :
  - a. Écrivez les résultats des requêtes SQL sur les tables créées dans la base de données en ajoutant la logique d'importation sur le script du point 3.
6. Testez à plusieurs reprises l'exécution du code que vous créez

## Ressources associées

- [La conteneurisation, qu'est-ce que c'est ?](#)
- [Qu'est-ce que la conteneurisation ? | IBM](#)
- [Overview | Docker Documentation](#)
- [Docker Hub](#)
- [Dockerfile: the building block of the Docker ecosystem - IONOS](#)
- [Les requêtes HTTP expliquées simplement](#)
- [Exemple de requêtes HTTP avec Python](#)
- Les données client
  - [CSV produits](#)
  - [CSV magasins](#)
  - [CSV ventes](#)

## Livrables

- Le schéma de l'architecture conçu
- Le schéma de la base de données
- Le Dockerfile
- Le fichier yaml du Docker Compose
- Le(s) script(s) d'exécution pour la collecte, transformation, et import des données
- Une fiche synthèse des résultats d'analyse obtenus (point 4.a, 4.b, 4.c).
- Une vidéo de démonstration, en enregistrant l'écran de l'ordinateur, du projet réalisé
  - La vidéo doit commencer en montrant la base de données créée mais vide, pour ensuite aller vers le terminal, lancer les services et observer les logs d'exécution. Finalement la vidéo doit montrer la base de données avec les données importées.

- La vidéo doit enregistrer l'intégralité de l'écran afin que la date et l'heure de l'ordinateur soient incluses dans l'enregistrement.

Le.a candidat.e apporte la vidéo de démonstration de son projet et les autres livrables sur un support (usb / url en ligne) pour pouvoir présenter sa réalisation au jury technique.