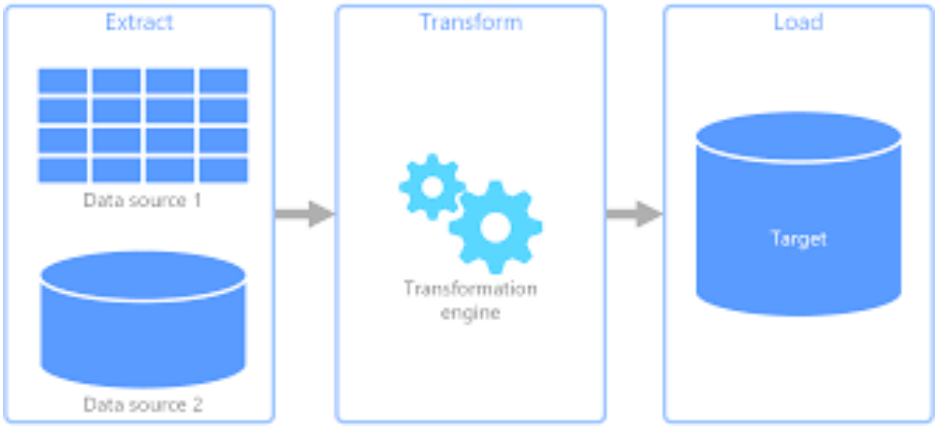
Ce projet est cofinancé par l'Union européenne

Extraction Transform and Load

ou

ETL



source: microsoft learn

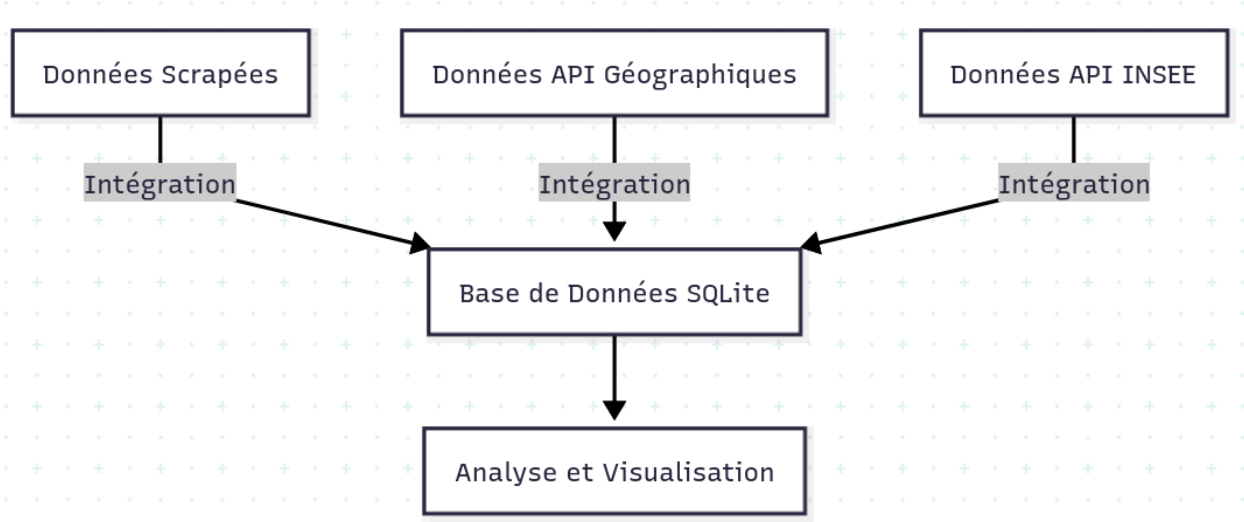
INTRODUCTION

Un peu de contexte

Dans les parties précédentes, vous avez été chargé de collecter des données depuis différentes sources et de les intégrer dans une base de données SQLite. Il est maintenant souhaitable de constituer une base de données avec des informations sur les entreprises des différentes communes. La base de données devrait inclure des informations telles que :

- Commune
- Nom de l'entreprise
- SIREN de l'entreprise
- Adresse de l'entreprise
- Code postal et ville
- Secteur d'activité
- Capital
- Dirigeants
- Nombre de salariés
- Coordonnées GPS de la commune
- Population de la commune

Les données requises sont disponibles, mais malheureusement pas dans la même source ni au même format. Sur la base de diverses sources, vous avez été chargé de créer la base de données souhaitée. Une partie des données sources a déjà été récupérée dans les modules précédents de scraping et d'APIs. Le diagramme ci-dessous montre le contexte, les parties dans la suite de ce kit vont servir à préparer l'industrialisation des collectes et transformations.



Le travail dans les parties qui vont suivre s'effectuera en îlot en mode "projet" pour cela il est important de définir des bases communes de travail.

En plus de ce qui sera défini dans les pages suivantes de ce kit, il est recommandé de mettre en place des pratiques inspirées des méthodes agiles afin de pouvoir avancer en groupe et de pouvoir échanger plus facilement sur les problématiques rencontrées.

PARTIE 0

Préparation du projet

OBJECTIF

- Créer des groupes de travail
- Mise en place des outils de travail
- Définition du modèle de données

FORMAT

- Travail en îlot
- ~ 1.5 heures

LIVRABLES

- Un dépot github pour chaque groupe, contenant la documentation demandées

0.1 — Quelques premiers pas

Avant de commencer à travailler en îlot, il est important de se synchroniser sur ce qu'il y a à faire et de connaître les ressources à disposition.
Dans cette partie, un état des lieux va être effectué:

- Par groupe, mettez en commun vos jeux de données collectés, faites un schéma des modèles de données que vous avez.
- Définissez ensemble, un modèle de données dans lequel les différentes données collectées seront intégrées.

Conservez bien les schémas, ils font partie du livrable.

0.2 — Mise en place d'un dépot github

Créer un dépôt github dans lequel vous inviterez:

- Les membres de votre îlot
- Les formateurs

En groupe avant de mettre des fichiers dans le dépôt, définissez:

- L'arborescence des fichiers, afin de séparer les données du code et de la documentation et de conserver les dépendances dans un fichier requirements.txt
- Le fichier .gitignore pour ignorer le dossier contenant les données et autres fichiers indésirables.

Ajoutez les schémas de données à votre dépôt et écrivez un README.md.

RESSOURCES

Pour définir votre modèle (à compléter avec les ressources du kit3):

- https://mermaid.js.org/
- https://d2lang.com/
- https://www.drawio.com
- https://excalidraw.com/

Petits rappels sur gitignore

- https://www.w3schools.com/git/git_ignore.asp
- https://www.freecodecamp.org/news/gitignore-file-how-to-ignore-files-and-folders-in-git/

PARTIE 1

ETL & ELT

OBJECTIF

- Améliorer votre compréhension des processus ETL et ELT.
- Améliorer votre compréhension des Data Lakes vs Data Warehouses.

FORMAT

- Travail en ilot
- ~ 1.5 heures

1.1 — Petite introduction

Les Data Lakes et les Data Warehouses sont deux systèmes importants pour stocker des données. Les données stockées dans ces systèmes peuvent être utilisées à des fins d'analyse. Il existe deux approches principales pour construire ces systèmes :

- ETL (Extract, Transform, Load)

- ELT (Extract, Load, Transform)

Ces méthodes d'intégration transfèrent des données d'une source vers une destination souhaitée. Selon l'objectif commercial et le système choisi (Data Lake ou Data Warehouse), les données peuvent être extraites à partir de différentes sources.

Pour combiner les avantages des Data Lakes et des Data Warehouses, une alternative est le Data Lakehouse.

Sources for further reading:

- https://www.integrate.io/blog/etl-vs-elt/
- https://en.wikipedia.org/wiki/Extract,_transform,_load
- https://www.talend.com/resources/data-lake-vs-data-warehouse/
- https://en.wikipedia.org/wiki/Data_warehouse
- https://en.wikipedia.org/wiki/Data_lake

1.2 — Encore un peu de préparation

Faites une comparaison des processus ELT et ETL et discutez de celui qui est le plus approprié pour un Data Warehouse ou un Data Lake. Mettez à jour la documentation dans le dépôt GitHub en documentant les jobs (ou séquence de traitement) nécessaires pour passer des jeux de données collectés à un jeu de données structuré avec votre modèle de données commun dans une base de données.

Implémentez seulement un des jobs, les autres seront implémentés dans le kit suivant.