Année académique : 2024-2025 *Université Gaston Berger (UGB)*

Classe : GeIT 3 *Institut Polytechnique de Saint Louis (IPSL)*

Enseignant : Dr. Djibril MBOUP

Cours : Data Engineering/Analytics

Projet : ETL et Business Intelligence avec Apache Spark

Date Limite : 15/06/2025 avant 23h 59

Prérequis :

- Ingénierie de données avec Apache Spark.
- Gestion de bases de données SQL Server et PostgreSQL.
- Création de tableaux de bord avec Power BI.
- Automatisation des processus ETL.

Description :

Ce projet vise à construire une solution **ETL (*Extract Transform Load*)** et de **Business Intelligence** à partir de la base de données AdventureWorks de Microsoft SQL Server.

Adventureworks est une BD d'une entreprise fictive nommée Adventure Works Cycles, spécialisée dans la fabrication et la vente de bicyclettes et accessoires. Elle contient plusieurs **domaines métiers**, chacun représenté par des schémas et des tables :

Domaine	Description
Production	Gestion de la fabrication, composants, nomenclatures (BOM), produits finis.
Sales	Informations sur les commandes, clients, facturation, territoires.
HumanResources	Données sur les employés, départements, emplois, salaires.
Purchasing	Achats, fournisseurs, commandes d'achat.
Person	Informations de contact (clients, fournisseurs, employés).
Warehouse / Inventory	Suivi des stocks, emplacements d'entreposage, inventaires.

Ci-dessous le schéma Entité Association de la base dans **dbdiagram** : ici

La base contenant les données sources est installée et déjà déployée sur un conteneur Docker via le dépôt GitHub suivant : sopeKhadim/etl. Vous pouvez suivre les instructions du README.md pour l'installation.

Vous avez besoin de **DBeaver** pour éditer les en mode GUI les tables de la BD.

Objectifs du Projet

1. **Extraction des Données :** Récupérer les données depuis la base MSSQL déployée sur Docker.
2. **Transformation des Données :** Nettoyer, enrichir et formater les données pour répondre aux besoins d'analyse.
3. **Chargement des Données :** Stocker les données transformées dans un entrepôt de données PostgreSQL.
4. **Analyse BI :** Utiliser Power BI/Tableau pour analyser et visualiser les indicateurs clés issus des données.

Étapes Méthodologiques

1. Préparation de l'Environnement

- Cloner le dépôt GitHub pour récupérer le conteneur Docker MSSQL.
- Vérifier la disponibilité des tables AdventureWorks.
- Installer Apache Spark, PostgreSQL à partir d'mages docker et Power BI Desktop sur l'environnement de travail.

---

2. Extraction et Transformation des Données avec Apache Spark

- **Extraction :**
    - Se connecter à la base MSSQL dans le conteneur Docker via JDBC.
    - Extraire les données nécessaires en utilisant des requêtes SQL depuis Apache Spark.
    - Les tables à récupérer incluent :
        - Clients (Sales.Customer, Person.Person)
        - Produits (Production.Product, Production.ProductCategory)
        - Ventes (Sales.SalesOrderHeader, Sales.SalesOrderDetail)
        - Géographie (Person.Address, Person.StateProvince)

- **Nettoyage :**

- Suppression des valeurs nulles et des doublons.
- Uniformisation des formats de date et des types de données.

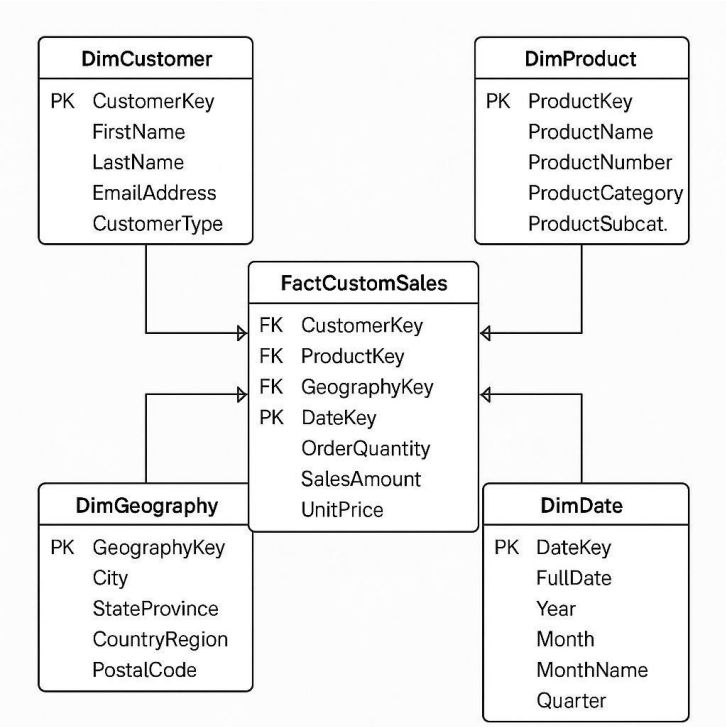
- **Enrichissement :**
  - Création de nouvelles colonnes.
  - Fusionner des tables pour obtenir des vues métiers pertinentes.
- **Agrégation :**
  - Calculer des indicateurs tels que le chiffre d'affaires par catégorie ou région.

---

4. Chargement dans l'Entrepôt

- Utiliser Spark pour créer et charger les données transformées dans PostgreSQL.
- Organiser les données en tables Fact et Dimension pour une modélisation en étoile.

  Le schéma ci-dessous montre le modèle de données Sales.



  - **Dimensions:** DimCustomer, DimProduct, DimGeography, DimDate

```
- ============================================

-- Dimension Customer

- ============================================


CREATE TABLE DimCustomer (
```

```
    CustomerKey INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    EmailAddress VARCHAR(50),
    CustomerType VARCHAR(15)
);
- ============================================
-- Dimension Product
- ============================================
CREATE TABLE DimProduct (
    ProductKey INT PRIMARY KEY,
    ProductName VARCHAR(100),
    ProductNumber VARCHAR(25),
    ProductCategory VARCHAR(50),
    ProductSubcategory VARCHAR(50)
);
- ============================================
-- Dimension Geography
- ============================================
CREATE TABLE DimGeography (
    GeographyKey INT PRIMARY KEY,
    City VARCHAR(30),
    StateProvince VARCHAR(50),
    CountryRegion VARCHAR(50),
    PostalCode VARCHAR(15)
);

- ============================================
-- Dimension Date
- ============================================
```

```
CREATE TABLE DimDate (
    DateKey INT PRIMARY KEY,
    FullDate DATE,
    Year INT,
    Month INT,
    MonthName VARCHAR(10),
    Quarter INT
);
```

- **Fact :** fact_sales table

```
- ============================================
-- Create FactCustomSales table (custom fact table)
-- ============================================
CREATE TABLE [dbo].[FactCustomSales](
    [CustomerKey] [int] NOT NULL,
    [ProductKey] [int] NOT NULL,
    [GeographyKey] [int] NOT NULL,
    [DateKey] [int] NOT NULL,
    [OrderQuantity] [smallint] NOT NULL,
    [SalesAmount] [money] NOT NULL,
    [UnitPrice] [money] NOT NULL,
    CONSTRAINT [PK_FactCustomSales] PRIMARY KEY CLUSTERED (
        [CustomerKey], [ProductKey], [GeographyKey], [DateKey]
    ),
    CONSTRAINT [FK_FactCustomSales_DimCustomer] FOREIGN KEY
([CustomerKey]) REFERENCES [dbo].[DimCustomer]([CustomerKey]),
    CONSTRAINT [FK_FactCustomSales_DimProduct] FOREIGN KEY
([ProductKey]) REFERENCES [dbo].[DimProduct]([ProductKey]),
    CONSTRAINT [FK_FactCustomSales_DimGeography] FOREIGN KEY
([GeographyKey]) REFERENCES
[dbo].[DimGeography]([GeographyKey]),
```

```
    CONSTRAINT [FK_FactCustomSales_DimDate] FOREIGN KEY
([DateKey]) REFERENCES [dbo].[DimDate]([DateKey])

) ON [PRIMARY];
```

---

5. Analyse BI avec Power BI

- Importer les tables PostgreSQL dans Power BI.
- Créer des tableaux de bord dynamiques pour les indicateurs suivants :
    - **Ventes :** Analyser l'évolution des ventes par année, trimestre, mois ou jour. Analyser le chiffre d'affaires par période, les ventes par catégorie.
    - **Clientèle :** Répartition des ventes par segment, Fidélité client.
    - **Géographie :** Ventes par région et par ville.
    - **Performance :** Taux de croissance, Comparaison des ventes par période.

Livrables Attendus

1. **Code ETL avec Spark :** Script Scala/Spark pour extraire, transformer et charger les données.
2. **Modèle de Données :** Schéma relationnel entre les tables PostgreSQL.
3. **Rapport BI :** Tableau de bord Power BI avec visualisation des indicateurs.
4. **Documentation :** Instructions pour déployer l'ETL et utiliser les tableaux de bord.

Important : Les livrables doivent être partagés dans un repos GitHub accessible.
