POINTS IMPORTANTS DU CHAPITRE 2 :DATA PROCESSING

- les étapes impliquées dans le prétraitement des données sont : la gestion des valeurs manquantes, la fusion des ensembles de données, l'application de fonctions, les agrégations et le tri.
- Pour importer les differntes fonctions et types de données requis à partir despark.sql one utilise :
  - **from pyspark.sql import SparkSession**
  - **import pyspark.sql.functions as F**
  - **from pyspark.sql.types import ***
- Dans Spark, nous pouvons traiter les valeurs nulles en les remplaçant par une valeur spécifique ou en supprimant les lignes/colonnes contenant des valeurs nulles.
  - **remplacement avec : replace()**
  - **suppression avec : drop()**
- Un sous-ensemble d'une trame de données peut être créé, en fonction de plusieurs conditions dans lesquelles nous sélectionnons quelques lignes, colonnes ou données avec certains filtres en place.
- le processus de filtrage des enregistrements :
  - **Select : on utilise select()**
  - **Filter : on utilise filter()**
  - **Where : on utilise where()**
- Tout type d'agrégation peut être simplement divisé en trois étapes, dans l'ordre suivant :
  - **Diviser**
  - **Appliquer**
  - **Combiner**

  nous agrégeons egalement les données en utilisant les fonctions groupBy()
- il existe différents types d'opérations sur des groupes d'enregistrements
  - **Moyenne**
  - **Maximum**
  - **Mini**
  - **Somme**
- Nous pouvons collecter des valeurs d'une liste de deux manières différentes :
  - **Collect List : elle fournit toutes les valeurs dans l'ordre d'occurrence d'origine (elles peuvent également être inversées)**

• Collect Set : elle fournit que les valeurs uniques

- User-Defined Functions (UDFs) : sont des routines programmables par l'utilisateur qui agissent sur une ligne.

  Pour définir les propriétés d'une fonction définie par l'utilisateur, l'utilisateur peut utiliser certaines des méthodes définies dans cette classe :

  - **asNonNullable() : met à jour UserDefinedFunction en non nullable.**
  - **AsNondeterministic() : met à jour UserDefinedFunction sur non déterministe.**
  - **withName(name: String): met à jour UserDefinedFunction avec un nom donné.**

- Pandas UDFs sont beaucoup plus rapides et efficaces, en termes de temps de traitement et d'exécution, par rapport aux standard Python UDFs.
  La principale différence entre eux est que UDF Python est exécuté ligne par ligne et, parconséquent, n'offre vraiment pas l'avantage d'un framework distribué. Cela peut prendreplus de temps, par rapport à un Pandas UDF, qui s'exécute bloc par bloc et donne des résultats plus rapides.

  La seule différence dans l'utilisation reside dans la declaration.

- PySpark offre un moyen très pratique de fusionner et de faire pivoter vos valeurs de dataframe, selon les besoins.

- Pivoting dans PySpark permet de créer simplement une vue pivot du cadre de données pour des colonnes spécifiques.

- Window Functions or Windowed Aggregates : dans PySpark permet d'effectuer certaines opérations sur des groupes d'enregistrements appelés "dans la fenêtre". Il calcule les résultats pour chaque ligne dans la fenêtre.

- PySpark prend en charge trois types dela fenêtreles fonctions:

  • Agrégations

  • Classement

  • Analytique