INF8200

Travail Pratique 1

préparé par Jean-Francois Rajotte

Hiver 2024

Consignes de remise du travail

Le travail doit être remis au plus tard le **4 mars (23h55)**.

Important: Pour chaque jour de retard, vous perdrez 5% de votre note. Après 7 jours, votre résultat sera 0. Pas de période de grâce une fois le délai écoulé. Votre remise doit être composé de deux fichiers: * Un rapport **Markdown** ou **Word** contenant les instructions pour réaliser le TP ainsi que les résultats demandés. * Un script python (pyspark) qui vous a permis d'obtenir les résultats.

Critères d'évaluation

- La présentation du rapport en général et le script pyspark 4/40
- Implémentation, instructions et réponses aux questions 36/40 Ce TP est noté sur 40 et compte pour 30% de votre note finale.

Objectif

Le but est d'utiliser votre cluster Spark sur Kubernetes pour créer un scprit ETL de base. Comme simulation d'une source externe, vous aurez un script python contenant une fonction `read_data_source`. Cette fonction vous retournera un DataFrame de données simulées des dépenses d'usagers d'un service infonuagique. Il vous faudra ensuite manipuler ces données et les mettre dans une table SQL (*Temporary views*) pour permettre d'envoyer des requêtes SQL proposées par vos collègues.

Concrètement, vous devrez implementer les étapes suivantes, en présumant que votre cluster est dans son état par défaut tel que décrit dans le 3e cours Spark (vous n'avez pas a installer de notebook Jupyter). Un début de script est partagé contenant la fonction de création du dataframe de données initiales, vous devez ajouter le code nécessaire pour obtenir les résultats demandés.

- En preparation pour augmenter les capacités de votre cluster pour l'ETL, vous allez ajouter un worker. **Note**: C'est étape se fait à l'extérieure de votre script pyton.
- Vous allez ensuite soumettre (`spark-submit`) un votre script pyspark qui
  - créera un DataFrame de dépenses
  - Mettra les donnees dans une table Spark
  - Fera un rapport à partir de commandes SQL données

En plus du script, votre rapport de TP doit contenir un document **markdown** ou **Word** donnant les étapes a suivre pour obtenir les résultats demandés. Tout le code Spark doit etre dans un seul fichier `.py`. Le fichier ne doit **pas** être un notebook, mais un script python qui sera appelé par la commande `spark-submit` à partir de votre spark master.

Pour préparer votre TP, il faudra apporter quelques modifications a votre cluster tel qu'expliqué ci-dessous.

Tâche no. 1: Modification de votre cluster (2 pts)

Une fois votre cluster Spark Kubernetes installe, vos instructions devront donner la commande pour ajouter un *worker* dans votre cluster.

Tâche no. 1.5 (point bonus): réparer pyspark-shell (2 pts)

Il vous sera pratique de tester des commandes dans avec pyspark shell, mais malheurement il y a un bug. Pour le decouvrir, connectez-vous à votre Spark master et essayez de démarrer une session pyspark shell. **Question**: Quel est le bug? Trouvez la solution dans ce liens et appliquez-là.

Tâche no. 2: Script pyspark (30 pts)

Votre rapport doit contenir toutes les commandes nécessaire pour démarrer le traitement pyspark.

Votre script pyspark doit faire les étapes suivantes * Aller chercher le dataframe des dépenses des utilisateurs (déjà fait dans le code partagé) * Ajouter une colonne `Total` contenant le total des dépenses (i.e. la sommme des colonnes, rangée par rangée) * Montrer les 20 premières lignes du Dataframe de l'étape precedente * Créer une table `depenses` pour pouvoir utiliser les script SQL de vos collaborateurs * Envoyez les commandes suivantes de vos collaborateurs créant un rapport sur les données contenant: * Le nombre d'entrées de votre table résultante de votre ETL * `SELECT COUNT(Storage) FROM depenses` * La moyenne de la somme total des dépenses * `SELECT AVG(Total) FROM depenses` * La liste des dépenses (incluant la sommes) de l'utilisateur dont le `userID` est votre code permanent. * `SELECT * FROM depenses WHERE userID = <userID>`

Les sorties de toutes ces étapes doivent se trouver dans votre rapport de TP. Un simple copier-coller est suffisant.

Tâche no. 4: reduire votre cluster a sa taille initiale (2 pts)

Une fois tout terminé, enlever un worker à votre cluster Spark pour le remmetre dans son état original.