



# Évaluation 4

**Titre du cours**

Big Data 1

**Numéro du cours**

420-J35-RO

**Programme**

Big Data

**Groupe**

478

**Prénom et nom de l'enseignant**

Abderrazak Sahraoui

**Date**

2023-10-12

## APERÇU

- Cette évaluation vise à mesurer votre habileté à
  - programmer des processus ETL en utilisant des frameworks Hadoop, MapReduce, MRjob et Python.
  - Créer et exploiter un entrepôt big data en utilisant Hive et Hadoop.
- Ne partagez pas votre copie.
- L'examen comporte 3 parties. Il dure 4 heures.
- Internet et Outils d'intelligence artificielle non autorisés.
- Documentation du cours et notes personnelles autorisées.

Barème : Partie 1 : 30%, Partie 2 : 40%, Partie 3 : 30%

## PRÉPARATION

1. Aller sur le site :
  - <https://donnees.montreal.ca/dataset/collisions-routieres#data>
2. Télécharger le fichier csv **collisions\_routieres.csv**
3. Aller sur le site :
  - <https://www.donneesquebec.ca/recherche/dataset/rapports-d-accident/resource/86cd3bcb-fe10-4ca5-992e-5bb80e5c534a>
4. Télécharger le fichier pdf **rapports-accident-documentation.pdf** qui explique les colonnes du fichier collisions\_routieres.csv

## PARTIE 1

1. Créer un programme **Ev4\_XY\_Partie1.py** MapReduce en Python pour lire les données du fichier **collisions\_routieres.csv** et pour effectuer les opérations suivantes :
2. Extraire les données des lignes correspondants à l'année **2021** et aux colonnes suivantes :
  - **No\_seq\_coll**
  - **dt\_accdn**
  - **rue\_accdn**
  - **accdn\_pres\_de**
  - **cd\_genre\_accdn**
  - **nb\_blesses\_graves**
  - **nb\_blesses\_legers**
  - **gravite**
3. Nettoyer les données en enlevant les “ qui entourent les valeurs numériques.
4. À la sortie, les données doivent être triées selon la date **dt\_accdn**.
5. Diriger les sorties de votre programme vers un fichier **Ev4\_XYPartie1.txt**

Note 1 : remplacer X dans les noms des programmes par la première lettre de votre prénom et remplacer le Y par votre nom de famille.

Note 2 : Utiliser le protocole mrjob **ByteValueProtocol** pour vos sorties afin de permettre un affichage correct des caractères accentués. Voir exemple en Annexe.

Note 3 : Si votre fichier de sortie **Ev4\_XYPartie1.txt** affiche un encodage UTF-16 LE. Enregistrer votre fichier avec un encodage UTF-8 dans VSCode. Voir procédure en annexe.

Note 4 : vous pouvez enlever manuellement la ligne d'entête du fichier **collisions\_routieres.csv**

## PARTIE 2

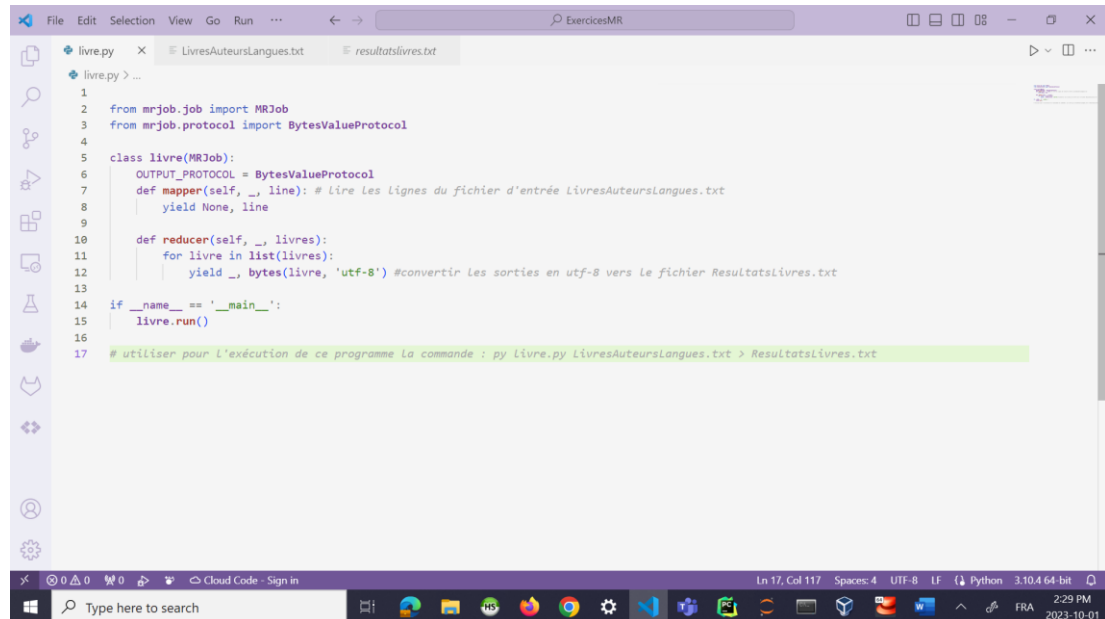
1. Se connecter sur la machine virtuelle **Hortonworks** avec **maria\_dev** et créer un dossier **Ev4** dans le dossier **/tmp**.
2. Transférer les fichiers **Ev4\_XYPartie1.txt**, **genres\_collisions.txt** et **collisions2.txt** vers le dossier **/tmp/Ev4**
3. Créer un script HiveQL **Ev4\_XYPartie4Creation.hql**
  - 1) Créer dans ce script une base de données **Ev4\_XY**
  - 2) Créer dans ce script une table **Ev4\_XY\_Collisions1**
  - 3) Charger dans **Ev4\_XY\_Collisions1** les données du fichier **Ev4\_XYPartie1.txt**
  - 4) Créer dans ce script une table **Ev4\_XY\_Genres**
  - 5) Charger dans **Ev4\_XY\_Genres** les données du fichier **genres\_collisions.txt**.
  - 6) Créer dans ce script une table **Ev4\_XY\_Collisions2**. Cette table ressemble à la table **Ev4\_XY\_Collisions1**. Les colonnes **nb\_blesses\_graves** et **nb\_blesses\_legers** sont remplacées par une colonne **blesses** de type **map<string,int>** le premier paramètre représente une clé qui peut avoir les valeurs **graves** ou **legers**. Le deuxième paramètre représente le nombre de blessés.
  - 7) Charger dans **Ev4\_XY\_Collisions2** les données du fichier **collisions2.txt**.

## PARTIE 3

1. Créer un script de requêtes d'interrogation **Ev4\_XYPartie3Requetes.hql** pour les requêtes suivantes :
  1. Afficher sans doublons les valeurs de la colonne **gravite**.
  2. Afficher les collisions dont la **gravité est Mortel**
  3. Compter les collisions dont la **gravité est Mortel**
  4. Grouper et compter les collisions **par genre de collision (cd\_genre\_accdn)**
  5. Afficher la jointure des tables **Ev4\_XY\_Collisions1** et **Ev4\_XY\_Genres**
  6. Afficher les collisions ayant des **blessés graves** dans la table **Ev4\_XY\_Collisions1**

# ANNEXE 1

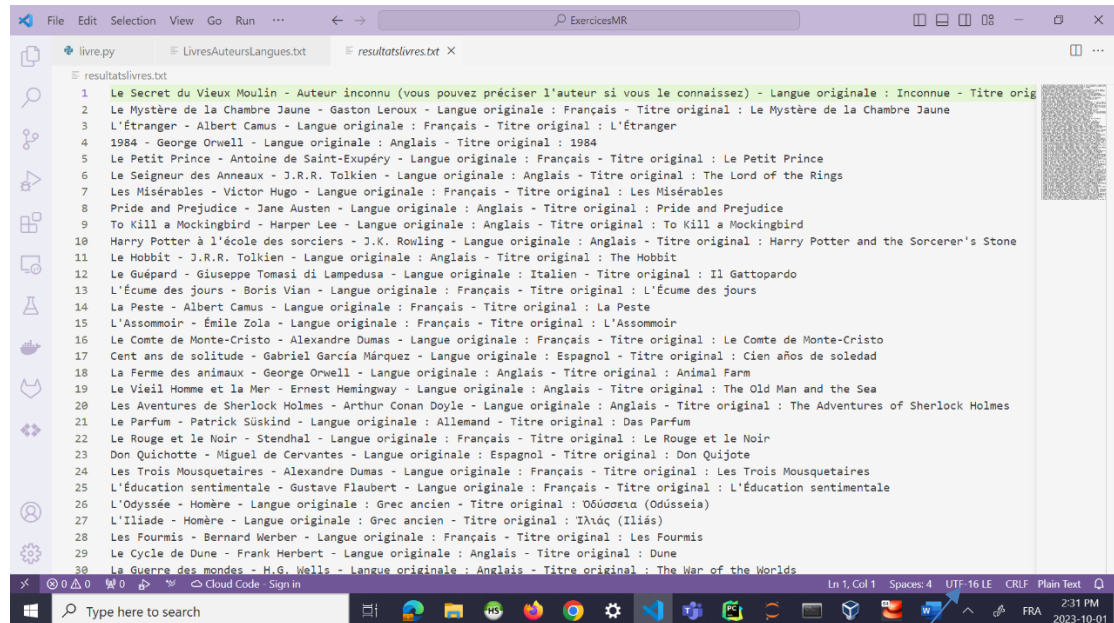
Utiliser le protocole mrjob ByteValueProtocol pour vos sorties afin de permettre un affichage correct des caractères accentués. Voir exemple ci-dessous :



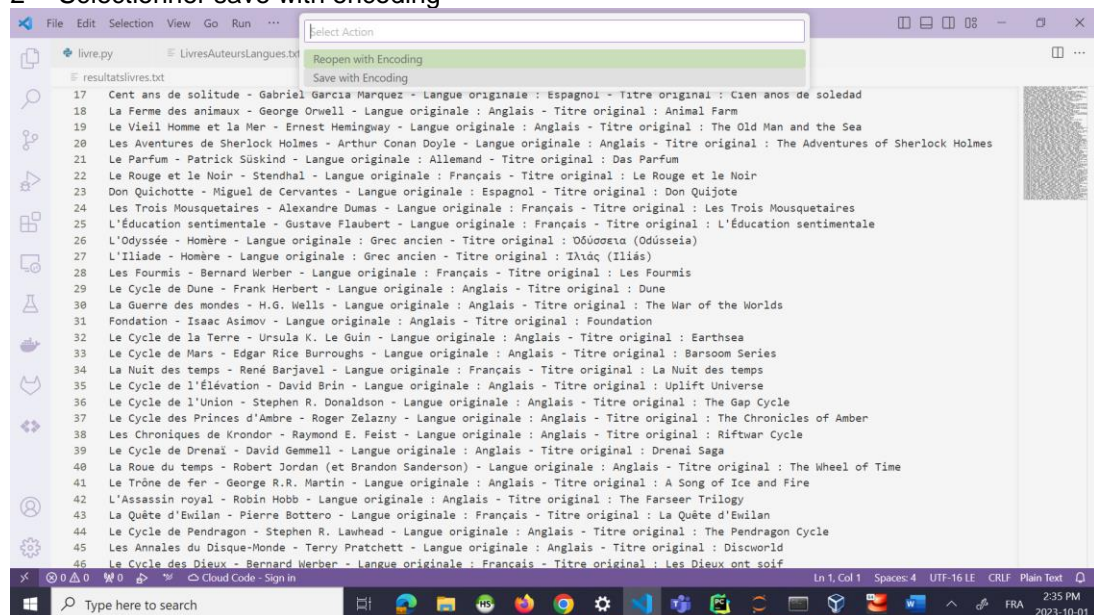
```
1  livre.py x LivresAuteursLangues.txt resultatslivres.txt
2  livre.py > ...
3  1
4  2 from mrjob.job import MRJob
5  3 from mrjob.protocol import BytesValueProtocol
6  4
7  5 class livre(MRJob):
8  6     OUTPUT_PROTOCOL = BytesValueProtocol
9  7     def mapper(self, _, line): # Lire Les lignes du fichier d'entrée LivresAuteursLangues.txt
10  8         yield None, line
11  9
12  10     def reducer(self, _, livres):
13  11         for livre in list(livres):
14  12             yield _, bytes(livre, 'utf-8') #convertir Les sorties en utf-8 vers Le fichier ResultatsLivres.txt
15  13
16  14 if __name__ == '__main__':
17  15     livre.run()
18  16
19  17 # utiliser pour l'exécution de ce programme la commande : py livre.py LivresAuteursLangues.txt > ResultatsLivres.txt
```

## ANNEXE 2

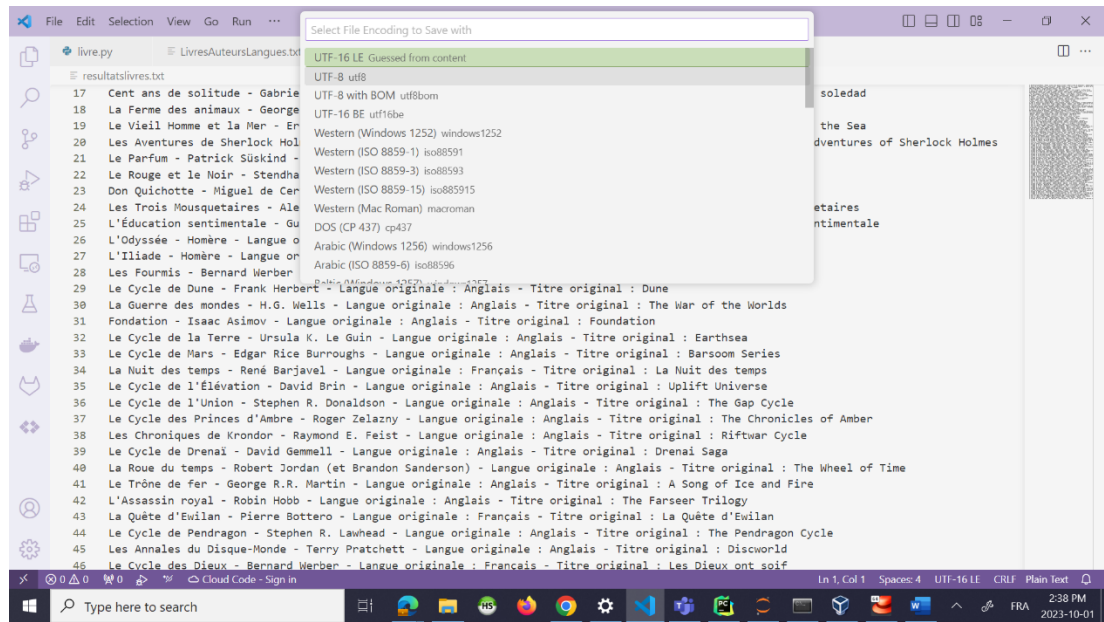
Si votre fichier de sortie LivresPartie1\_eqX.txt affiche un encodage UTF-16 LE. Enregistrer votre fichier avec un encodage UTF-8 dans VSCode.



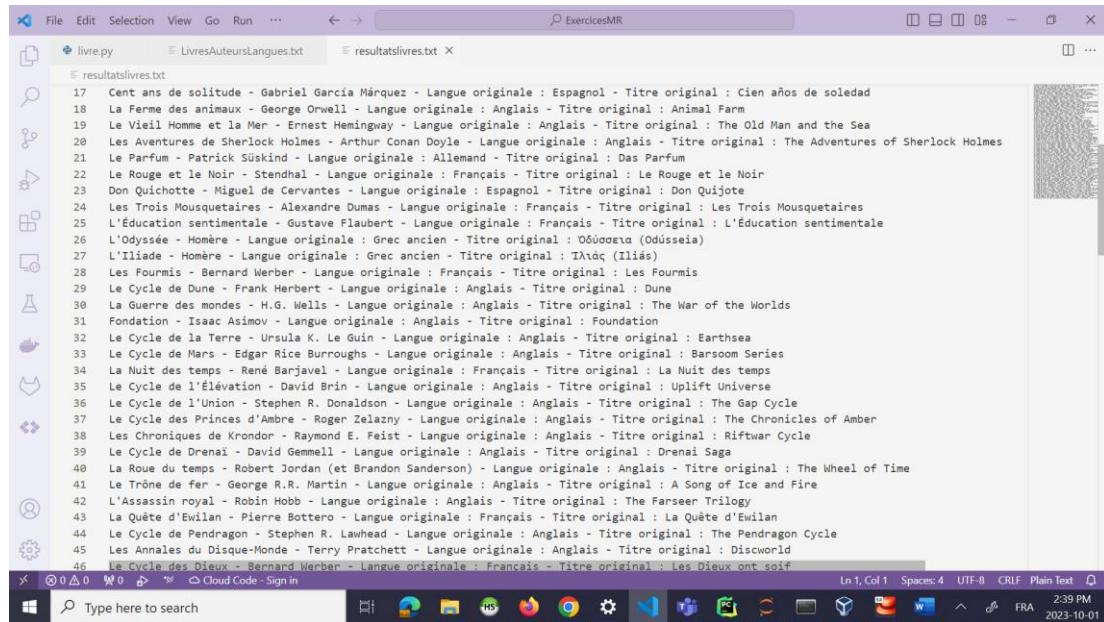
- 1- Cliquer sur le texte UTF-16 LE qui se trouve sur la barre violette en bas de l'écran
- 2- Sélectionner save with encoding



- 3- Sélectionner utf-8



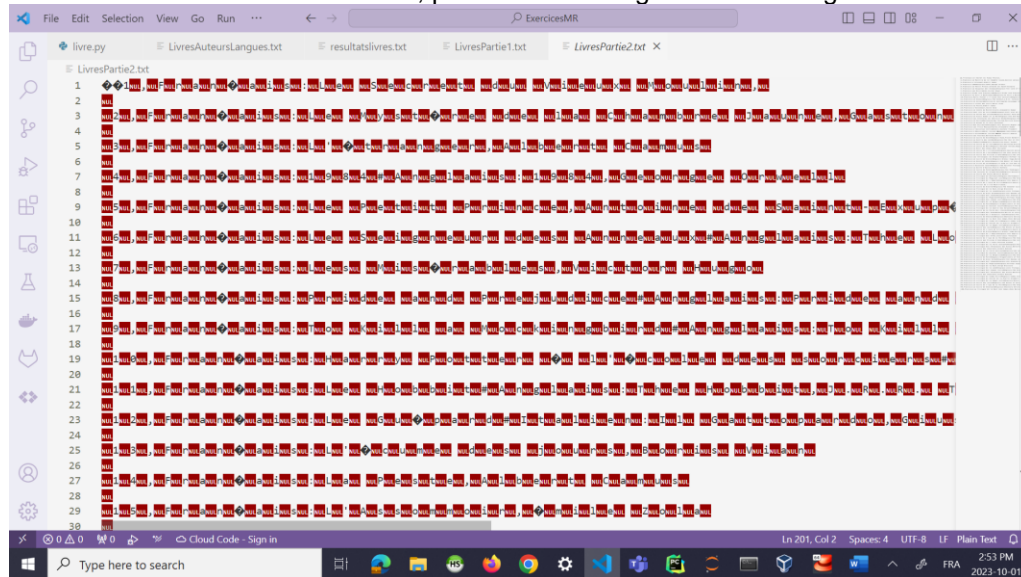
Et votre fichier prendra l'encoding utf-8



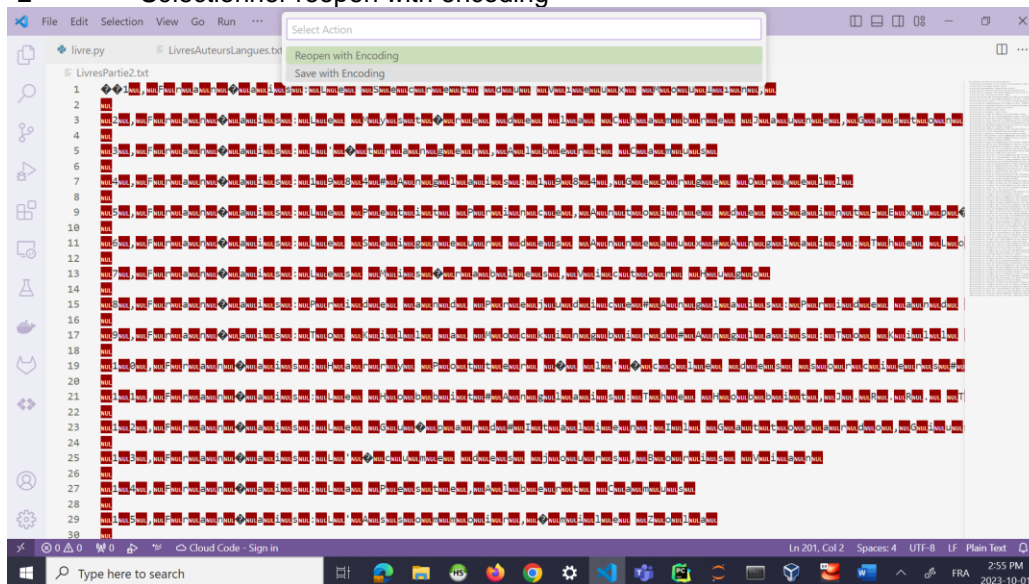


## ANNEXE 3

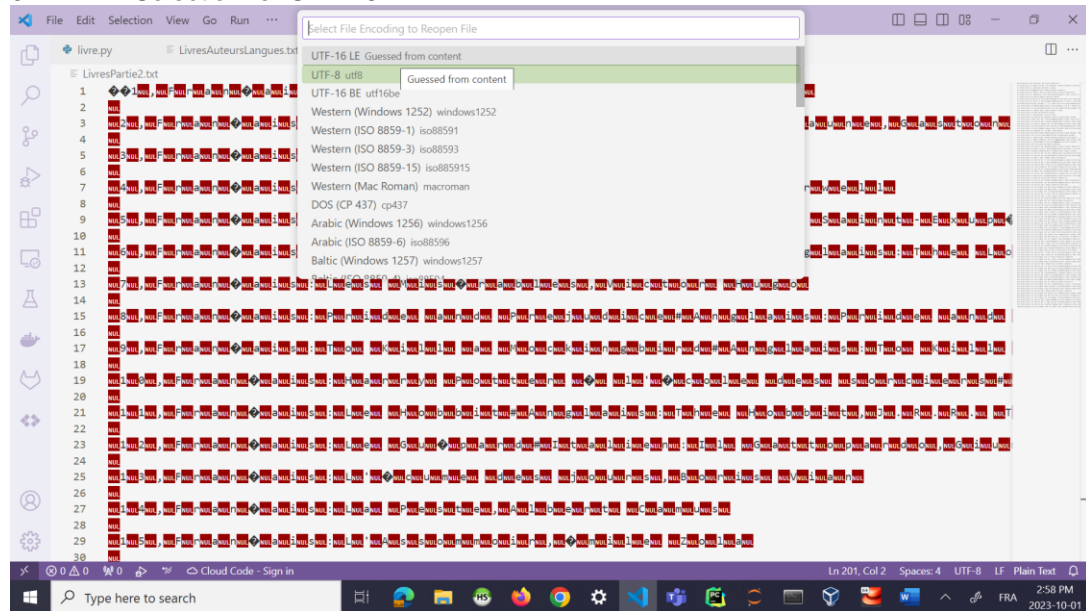
Si votre fichier de sortie LivresPartie3\_eqX.txt affiche un encodage UTF-8 mais n'affiche pas correctement le contenu, passer en affichage avec encodage UTF-16 LE.



- 1- Cliquer sur le texte UTF-8 qui se trouve sur la barre violette en bas de l'écran
- 2- Sélectionner reopen with encoding



### 3- Sélectionner UTF-16 LE



### Votre fichier s'affiche correctement

