

## ✓ Mini Projet

### AUTEUR : KOANDA Boubacar

Le but de cet exercice étant d'extraire un sous-ensemble de données à partir d'une base de données plus grande. Supposons que nous ayons besoin de réaliser une application qui prend en entrée, des noms de lieux du Burkina Faso, avec des informations telles que la latitude/longitude. Pour ce faire, nous décidons d'extraire ces informations à partir du serveur de référencement géoname (<http://www.geonames.org/>). Nous procedons de la façon suivante :

**1 - Exporter la base de données pour le Burkina Faso : en se référant au fichier README pour identifier le code ISO correspondant au Burkina Faso.**

**2 - Téléchargez le fichier zip correspondant depuis le lien <https://download.geonames.org/export/dump/>.**

**3 - Appliquer les opérations de prétraitement et filtres nécessaires à ce fichier, pour ne garder que les colonnes correspondantes :**

- Identifiez, renommez et sauvegardez les colonnes nécessaires dans un fichier CSV nommé burkina\_location.csv. Tout d'abord il faut unzipper:

```
import zipfile
import os
import pandas as pd

# Nom du fichier zip
zip_file_name = 'BF.zip'

# Emplacement du dossier
folder_path = './' # Changez-le selon l'emplacement de votre fichier zip

# Extrait le contenu du fichier zip
with zipfile.ZipFile(os.path.join(folder_path, zip_file_name), 'r') as zip_ref:
    zip_ref.extractall(folder_path)

# Identifier toutes les colonnes nécessaires d'abord
columns = ['geonameid', 'name', 'asciiname', 'alternatenames', 'latitude', 'longitude',
            'feature_class', 'feature_code', 'country_code', 'cc2', 'admin1_code',
            'admin2_code', 'admin3_code', 'admin4_code', 'population', 'elevation',
            'dem', 'timezone', 'modification_date']

# Charger les données dans un DataFrame
data = pd.read_csv('BF.txt', sep='\t', usecols=columns, names=columns, header=None)

# Afficher les premieres colonnes pour voir
print (data.head())
print("\n")
```

	geonameid	name	asciiname	\
0	2282318	Pouéné	Pouene	
1	2285251	Léraba Occidentale	Leraba Occidentale	
2	2287216	Kéléouoro	Keleouoro	
3	2294066	White Volta	White Volta	
4	2298457	Issana Bouga	Issana Bouga	

  

	alternatenames	latitude	longitude	\
0	NaN	9.72908	-2.78660	
1	Badini,La Leraba Occidentale,Leraba,Leraba Occ...	10.28333	-5.11667	
2	Keleouoro,Keleworu,Kéléouoro,Kéléworu	9.80748	-4.05023	
3	Nakambe,Nakambé,Nakanbe,Nakanbé,Volta Blanche,...	8.70194	-0.99056	
4	Issana Bouga,Lanyung,Lanyunga	10.91667	-1.18333	

  

	feature_class	feature_code	country_code	cc2	admin1_code	admin2_code	\
0	H	STM	BF	NaN	00	NaN	
1	H	STM	BF	CI	00	NaN	
2	H	STM	BF	BF,CI	00	NaN	
3	H	STM	BF	GH	00	NaN	
4	H	STM	BF	NaN	00	NaN	

  

	admin3_code	admin4_code	population	elevation	dem	timezone	\
0	NaN	NaN	0	NaN	226	Africa/Abidjan	
1	NaN	NaN	0	NaN	283	Africa/Abidjan	
2	NaN	NaN	0	NaN	261	Africa/Abidjan	
3	NaN	NaN	0	NaN	75	Africa/Accra	
4	NaN	NaN	0	NaN	178	Africa/Accra	

  

	modification_date
0	2023-11-07
1	2023-11-07

```

2      2023-11-07
3      2023-11-08
4      2023-11-08

```

- Filtrer les colonnes Identifiants, Noms de lieux, latitudes, longitudes et les Renommer les avec les noms suivants : 'ID', 'location\_name', 'lat', 'long'

```

# Filtrer les colonnes nécessaires
filtered_data = data[['geonameid', 'name', 'latitude', 'longitude']]

# Renommer les colonnes
filtered_data.columns = ['ID', 'location_name', 'lat', 'long']

# Afficher les premières lignes pour vérification
print("\n")
print(filtered_data.head())

```

	ID	location_name	lat	long
0	2282318	Pouéné	9.72908	-2.78660
1	2285251	Léraba Occidentale	10.28333	-5.11667
2	2287216	Kéléouoro	9.80748	-4.05023
3	2294066	White Volta	8.70194	-0.99056
4	2298457	Issana Bouga	10.91667	-1.18333

- Sauvegarder ces données dans un fichier CSV, nommez-le burkina\_location.csv

```

# Sauvegarde les données dans un fichier CSV
csv_file_name = 'burkina_location.csv'
filtered_data.to_csv(csv_file_name, index=False)

print(f"Données sauvegardées dans {csv_file_name}")
print("\n")

Données sauvegardées dans burkina_location.csv

```

#### 4- Opérations sur le fichier CSV burkina\_location.csv

- **4.1** Extraction des données 'gounghin' Extraire les données du fichier burkina\_location.csv qui contiennent le nom 'gounghin'. Enregistrez ces données dans un fichier CSV nommé gounghin.csv.

```

import pandas as pd

# Chargement du fichier burkina_location.csv
df = pd.read_csv('burkina_location.csv')

# 4.1 Extraction des données 'gounghin'
gounghin_data = df[df['location_name'].str.contains('gounghin', case=False)]
gounghin_data.to_csv('gounghin.csv', index=False)

# Afficher les résultats
gounghin_data = pd.read_csv('gounghin.csv')
print(gounghin_data)

```

	ID	location_name	lat	long
0	2353306	Gounghin	12.06677	-1.42134
1	2360473	Gounghin	12.62488	-1.36398
2	2570204	Gounghin	12.31436	-1.37900
3	10342749	Gounghin	12.06667	-0.15000
4	10629032	BICIAB // Gounghin	12.35921	-1.54273
5	11257296	Gounghin Department	12.06671	-0.15484
6	11900526	Gounghin Nord	12.36120	-1.55055
7	11900528	Zone Industrielle de Gounghin	12.36631	-1.54137
8	11900619	Gounghin Sud	12.35298	-1.54342
9	11900680	Gounghin	12.35895	-1.54442

- **4.2** Extraction de la sous-partie de la base de données Extraire la sous-partie de la base de données (fichier burkina\_location.csv) où les noms des lieux commencent par les lettres entre 'A' et 'P' (ordre alphabétique).

```
# 4.2 Extraction de la sous-partie de la base de données (noms de lieux entre 'A' et 'P')
a_to_p_data = df[df['location_name'].str[0].between('A', 'P', inclusive='both')]
a_to_p_data.to_csv('A_to_P.csv', index=False)
```

```
# Afficher les résultats
```

```
A_to_P_data = pd.read_csv('A_to_P.csv')
print(A_to_P_data)
```

	ID	location_name	lat	long
0	2282318	Pouéné	9.72908	-2.78660
1	2285251	Léraba Occidentale	10.28333	-5.11667
2	2287216	Kéléouoro	9.80748	-4.05023
3	2298457	Issana Bouga	10.91667	-1.18333
4	2353290	Forêt Classée de Ziga	12.47106	-1.08644
...	...	...	...	...
8125	12640343	Cinkansé	11.11857	0.00993
8126	12640344	Kompienga	11.08169	0.72365
8127	12640603	Konglore	11.39510	0.06110
8128	12687046	Monde-So	14.20081	-1.63983
8129	12688712	Kanliyinou	12.15503	0.59881

```
[8130 rows x 4 columns]
```

- Identifiez respectivement la latitude minimale, la longitude minimale et les noms de lieux correspondants:

```
# Trouver la latitude minimale
```

```
min_lat_row = A_to_P_data.loc[A_to_P_data['lat'].idxmin()]
```

```
# Trouver la longitude minimale
```

```
min_long_row = A_to_P_data.loc[A_to_P_data['long'].idxmin()]
```

```
# Afficher les résultats
```

```
print("Latitude minimale:")
```

```
print(min_lat_row[['lat', 'location_name']])
```

```
print("\nLongitude minimale:")
```

```
print(min_long_row[['long', 'location_name']])
```

```
Latitude minimale:
lat      5.21609
location_name  Komoé
Name: 2954, dtype: object
```

```
Longitude minimale:
long      -5.65968
location_name  Banifing
Name: 1163, dtype: object
```

- **4.3 Identification des lieux avec des coordonnées spécifiques :** Identifiez les lieux dans le fichier burkina\_location.csv dont les coordonnées sont comprises entre (lat >= 11 et lon <= 0.5).

```
# Convertir les colonnes 'lat' et 'long' en types numériques
```

```
df['lat'] = pd.to_numeric(df['lat'], errors='coerce')
```

```
df['long'] = pd.to_numeric(df['long'], errors='coerce')
```

```
# 4.3 Identification des lieux avec des coordonnées spécifiques (lat >= 11 et lon <= 0.5)
```

```
specific_coords_data = df[(df['lat'] >= 11) & (df['long'] <= 0.5)]
```

```
specific_coords_data.to_csv('specific_coords.csv', index=False)
```

```
# Afficher les résultats
```

```
print("Lieux dont les coordonnées sont comprises entre (lat >= 11 et lon <= 0.5):\n")
```

```
specific_coords_data = pd.read_csv('specific_coords.csv')
```

```
print(specific_coords_data)
```

```
print("\n")
```

```
Lieux dont les coordonnées sont comprises entre (lat >= 11 et lon <= 0.5):
```

	ID	location_name	lat	long
0	2353158	Zyonguen	12.36667	-0.45000
1	2353159	Zyiliwèlè	12.38333	-2.73333
2	2353160	Zyanko	12.78333	-0.41667
3	2353161	Zouta	13.14908	-1.28197
4	2353162	Zourtenga	12.95741	-1.28745
...	...	...	...	...
9276	12640603	Konglore	11.39510	0.06110
9277	12687046	Monde-So	14.20081	-1.63983
9278	12688694	Yerfing	11.33747	-3.35025
9279	12688713	Tinonguen-Peulh	11.64924	-0.22063
9280	12720016	Yimdi	12.31014	-1.66322

```
[9281 rows x 4 columns]
```

**5 : Créer un fichier Excel et enregistrer les données extraites dans différentes feuilles.**

```
# Écrire les données 'A_to_P', 'specific_coords_data' et 'burkina_location' dans le fichier Excel  
with pd.ExcelWriter('mini_projet.xlsx') as writer:
```

```
    # Écrire les données 'burkina_location' dans la quatrième feuille  
    filtered_data.to_excel(writer, sheet_name='burkina_location', index=False)
```

```
    # Écrire les données 'gounghin' dans la première feuille  
    gounghin_data.to_excel(writer, sheet_name='gounghin', index=False)
```

```
    # Écrire les données 'A_to_P' dans la deuxième feuille  
    a_to_p_data.to_excel(writer, sheet_name='A_to_P', index=False)
```

```
    # Écrire les données 'specific_coords_data' dans la troisième feuille  
    specific_coords_data.to_excel(writer, sheet_name='Coords_Specifiques', index=False)
```