

TD5 – Mini-projet pandas avec Our World in Data

Python pour économistes – Université de Lille, L2 Économie

Question fil rouge. Comment importer un CSV OWID, produire des statistiques descriptives simples, tracer un graphique et rédiger une interprétation économique ?

Livrables. Un fichier `td5_nom_prenom.py`, une figure `out/gdp_top10.png`, un fichier `out/gdp_selection.csv` et une interprétation de six à huit lignes.

Objectifs du TD

- Importer un fichier CSV avec pandas et inspecter sa structure.
- Filtrer un tableau, calculer des statistiques descriptives et exporter un résultat.
- Produire un graphique lisible à partir d'un DataFrame.
- Diagnostiquer des erreurs courantes de bibliothèque, chemin, colonne et filtre vide.

Progression du TD

Les exercices 1 à 13 sont obligatoires. Ce TD est le mini-projet final : il combine chemins de fichiers, pandas, statistiques descriptives simples, graphique et interprétation. Les premiers exercices restent guidés ; les derniers demandent davantage d'autonomie.

Pathways : travailler avec un CSV OWID

Le TD lit directement un CSV publié par Our World in Data. Si la connexion internet ne fonctionne pas en salle, télécharger le CSV avec un navigateur et placer le fichier dans un dossier data.

1 <https://ourworldindata.org/grapher/gdp-per-capita-worldbank.csv>

Logique 1 – Pipeline pandas final

PSEUDO-CODE

```
1 # Objectif : mini-projet descriptif avec un CSV OWID
2 # 1. Lire le CSV
3 # 2. Inspecter lignes, colonnes et types
4 # 3. Renommer les colonnes utiles
5 # 4. Filtrer une année et quelques pays
6 # 5. Calculer moyenne, médiane, min, max
7 # 6. Tracer un graphique lisible
8 # 7. Exporter un tableau propre
9 # 8. Rédiger une interprétation prudente
```

A. Importer et inspecter

Exercice 1 – Préparer le script

OBLIGATOIRE

Créer l'arborescence suivante :

```
1 TD5/
2   td5_nom_prenom.py
```

```
3 out/
```

Dans le script, créer le dossier out.

```
1 from pathlib import Path
2
3 out_dir = Path("out")
4 out_dir.mkdir(exist_ok=True)
```

`Path("out")` désigne un dossier nommé out dans le dossier depuis lequel le script est exécuté. Si les fichiers produits ne sont pas retrouvés, vérifier le dossier courant et relancer après sauvegarde du script.

Exercice 2 – Vérifier le dossier courant

OBLIGATOIRE

Ajouter temporairement ces deux lignes au début du script, puis exécuter.

```
1 print("Dossier courant :", Path.cwd())
2 print("Le dossier out existe :", out_dir.exists())
```

Comparer le chemin affiché avec l'endroit où le fichier `td5_nom_prenom.py` a été enregistré. Supprimer ensuite ces affichages si le dossier est correct, ou les garder en commentaire si cela aide à comprendre le problème.

Exercice 3 – Lire le CSV OWID

OBLIGATOIRE

Lire le fichier depuis l'URL OWID.

```
1 import pandas as pd
2
3 url = "https://ourworldindata.org/grapher/gdp-per-capita-worldbank.csv"
4 df = pd.read_csv(url)
5
6 print(df.head())
7 print(df.shape)
8 print(df.columns)
```

Ajouter un commentaire indiquant le nombre de lignes et de colonnes. Si la connexion internet ne fonctionne pas, télécharger le CSV dans un dossier data, puis remplacer temporairement la ligne de lecture par :

```
1 df = pd.read_csv("data/gdp-per-capita-worldbank.csv")
```

Exercice 4 – Repérer la colonne de l'indicateur

OBLIGATOIRE

Avant de renommer les colonnes, afficher seulement les noms de colonnes.

```
1 print(list(df.columns))
```

Écrire en commentaire le nom exact de la colonne qui contient le PIB par habitant. Vérifier ensuite que `df.columns[-1]` correspond bien à cette colonne dans le fichier utilisé.

Exercice 5 – Renommer proprement

OBLIGATOIRE

Le nom de la colonne de valeur peut être long. On le récupère automatiquement après avoir regardé `df.columns`.

```
1 value_col = df.columns[-1]
2
3 df = df.rename(columns={
4     "Entity": "country",
5     "Code": "code",
6     "Year": "year",
7     value_col: "gdp_pc"
8 })
```

Afficher `df[["country", "year", "gdp_pc"]].head()`. **Pourquoi `df.columns[-1]` ?** Les fichiers OWID placent souvent l'indicateur dans la dernière colonne. Ici, `-1` signifie : prendre le dernier nom de colonne. Ce n'est pas une règle universelle ; c'est une vérification pratique après inspection.

Exercice 6 – Débogage pandas guidé

OBLIGATOIRE

Pour chaque situation, identifier l'hypothèse la plus probable et le premier contrôle à effectuer.

- ModuleNotFoundError: No module named 'pandas' ;
- FileNotFoundError: data/gdp-per-capita-worldbank.csv ;
- KeyError: 'gdp_pc' après le renommage ;
- le script s'exécute, mais selection est vide.

Indices : environnement Python, dossier courant, nom exact des colonnes, orthographe exacte des pays.

B. Filtrer et décrire

Exercice 7 – Sélectionner une année récente

OBLIGATOIRE

Créer un tableau latest avec la dernière année disponible dans le fichier, puis retirer les valeurs manquantes.

```
1 latest_year = df["year"].max()
2 latest = df.loc[df["year"] == latest_year].copy()
3 latest = latest.dropna(subset=["gdp_pc"])
4 print(latest_year)
5 print(latest.shape)
```

Exercice 8 – Sélectionner quelques pays

OBLIGATOIRE

Créer une liste de pays et filtrer le tableau.

```
1 pays_selection = [
2     "France", "Germany", "Spain", "Poland",
3     "United States", "China", "India", "Brazil"
4 ]
5 selection = latest.loc[latest["country"].isin(pays_selection)].copy()
6 print(selection[["country", "gdp_pc"]])
```

Exercice 9 – Diagnostiquer une sélection vide

OBLIGATOIRE

Modifier temporairement un nom de pays, par exemple "Germany" en "Germanie", puis relancer le filtre. Observer ce qui disparaît du tableau.

Ajouter ensuite ces deux contrôles :

```
1 print(len(selection))
2 print(sorted(latest["country"].unique())[:10])
```

Remettre le nom correct avant de continuer. L'objectif est de comprendre qu'un filtre peut fonctionner techniquement tout en produisant une sélection incomplète.

Exercice 10 – Statistiques descriptives simples

OBLIGATOIRE

Sur selection, calculer :

- a. moyenne ;
- b. médiane ;
- c. minimum ;
- d. maximum ;
- e. proportion de pays au-dessus de la moyenne de la sélection.

On peut utiliser `mean()`, `median()`, `min()` et `max()`.

C. Graphique et export

Logique 2 – Passer du tableau au graphique

PSEUDO-CODE

```
1 # Objectif : représenter les dix valeurs les plus élevées
2 # Entrée : tableau latest sans valeurs manquantes
3 # 1. Trier le tableau par PIB par habitant décroissant
4 # 2. Garder les dix premières lignes
5 # 3. Tracer une barre par pays
6 # 4. Ajouter titre, axes et rotation des étiquettes
7 # 5. Sauvegarder la figure dans le dossier out
8 # Sortie : fichier gdp_top10.png
```

Exercice 11 – Graphique du top 10

OBLIGATOIRE

Créer un graphique en barres des dix pays les plus élevés dans latest.

```
1 import matplotlib.pyplot as plt
2
3 top10 = latest.sort_values("gdp_pc", ascending=False).head(10)
4 ax = top10.plot(kind="bar", x="country", y="gdp_pc", legend=False)
5 ax.set_xlabel("Pays")
6 ax.set_ylabel("PIB par habitant")
7 ax.set_title(f"Top 10 du PIB par habitant - {latest_year}")
8 plt.xticks(rotation=45, ha="right")
9 plt.tight_layout()
10 plt.savefig(out_dir / "gdp_top10.png", dpi=300)
```

Exercice 12 – Exporter un tableau propre

OBLIGATOIRE

Exporter selection dans out/gdp_selection.csv avec seulement country, code, year, gdp_pc.

```
1 selection[["country", "code", "year", "gdp_pc"]].to_csv(  
2     out_dir / "gdp_selection.csv",  
3     index=False  
4 )
```

Exercice 13 – Interprétation finale

OBLIGATOIRE

Ajouter six à huit lignes de commentaire :

- a. année étudiée ;
- b. pays le plus élevé dans la sélection ;
- c. pays le plus faible dans la sélection ;
- d. moyenne et médiane de la sélection ;
- e. comparaison de la France à la moyenne ;
- f. limite : choix des pays ;
- g. limite : le PIB par habitant ne mesure pas toutes les dimensions du développement.

D. Extension facultative

Pour aller plus loin 1 – Pour les étudiants plus rapides

Choisir au moins deux questions si le mini-projet est terminé.

- a. Tracer uniquement les pays de la sélection.
- b. Comparer la France à l'Allemagne, aux États-Unis et à la Chine.
- c. Créer une colonne above_selection_mean.
- d. Exporter le top10 dans out/gdp_top10.csv.
- e. Tracer l'évolution du PIB par habitant de la France depuis 1990.
- f. Ajouter le monde "World" si présent dans le fichier.
- g. Remplacer latest_year par une année choisie manuellement.
- h. Calculer le rapport entre le maximum et le minimum de la sélection.
- i. Importer aussi l'espérance de vie depuis le fichier CSV OWID life-expectancy.csv.
- j. Extension avancée : fusionner PIB par habitant et espérance de vie pour une même année.

À retenir

- Avant de traiter un CSV, il faut inspecter ses colonnes, ses dimensions et quelques lignes.
- Un DataFrame se filtre avec une condition booléenne et se copie quand on crée un extrait de travail.
- Les chemins de sortie doivent être construits et vérifiés pour retrouver les fichiers produits.
- Une interprétation descriptive doit citer l'année, la sélection et les limites du résultat.