

# Premiers pas en Python

---

Enseignant: Etienne Dagorn

Mail: [etienne.dagorn@univ-lille.fr](mailto:etienne.dagorn@univ-lille.fr)



---

**Objectif pédagogique** : approfondir la compréhension des statistiques descriptives (moyenne, médiane, variance, écart-type) à partir d'une base de données réelle sur les émissions de CO<sub>2</sub> liées à l'aviation (source : *Our World in Data*).

Le fichier `.csv` contient les données suivantes pour chaque pays :

- Le nom du pays (`Entity`)
- Le code ISO (`Code`)
- Les émissions annuelles de CO<sub>2</sub> par habitant dues à l'aviation (`Per capita total annual CO2 emissions from aviation`)

Téléchargez et placez le fichier `aviation_2024.csv` dans le même dossier que votre script Python.

## 1. Importer et explorer un fichier CSV

**Objectif de la section** : apprendre à ouvrir, lire et examiner une base de données au format `.csv` dans Python.

### Question 1 – Où Python cherche-t-il les fichiers ?

Lorsque vous demandez à Python d'ouvrir un fichier (par exemple `data.csv`), il le cherche dans le **répertoire de travail courant** (*working directory*).

- Pour connaître votre répertoire actuel :

```
import os
os.getcwd()
```

- Si le fichier n'est pas dans ce dossier, Python renverra une erreur.

- Vous pouvez changer de dossier avec la commande :

```
os.chdir("chemin/vers/votre/dossier")
```

### Exercice :

1. Affichez votre répertoire courant.
2. Changez-le vers le dossier contenant votre fichier CSV.
3. Vérifiez à nouveau que vous êtes bien au bon endroit.

## Question 2 – Importer la bibliothèque pandas

**pandas** est une bibliothèque très utilisée pour manipuler des données, comparable à Excel, mais beaucoup plus puissante. Elle permet de lire, explorer et transformer des tableaux de données appelés **DataFrames**.

### Étapes :

1. Installez **pandas** si ce n'est pas déjà fait :

```
pip install pandas
```

2. Importez-la dans votre script :

```
import pandas as pd
```

3. Lisez le fichier CSV (par exemple `data.csv`) :

```
df = pd.read_csv("data.csv")
```

4. Affichez les premières lignes :

```
df.head()
```

**Pour aller plus loin :** Consultez la documentation officielle : <https://pandas.pydata.org/docs/>

## 2. Explorer et décrire la base de données

### Question 3 – Afficher les premières lignes du fichier

Affichez les cinq premières lignes du fichier pour observer la structure de la base :

```
df.head()
```

Quels types d'informations sont présentes dans ce fichier ? Combien de colonnes et de lignes voyez-vous ?

**Question 4 – Examiner la structure du DataFrame**

Utilisez les commandes suivantes pour mieux comprendre la structure de la base :

```
df.info()
df.describe()
```

Que vous indiquent ces commandes ? Quelle est la nature des variables (texte, numérique, etc.) ?

**Question 5 – Vérifier les valeurs manquantes**

Certaines observations peuvent être manquantes. Vérifiez-le avec la commande suivante :

```
df.isna().sum()
```

Y a-t-il des valeurs manquantes dans cette base ?

**Question 6 – Identifier les colonnes principales**

Listez les noms des colonnes à l'aide de la commande :

```
df.columns
```

Expliquez à quoi correspond chaque colonne (en particulier `Entity`, `Code`, `Year`, `Per capita total annual CO2 emissions from aviation`, `time`).

**Question 7 – Explorer les émissions de CO<sub>2</sub>**

Affichez les pays qui ont les plus fortes émissions par habitant liées à l'aviation :

```
df.sort_values(by="Per capita total annual CO2 emissions from aviation", ascending=False)
```

Quels sont les 5 pays les plus émetteurs ? Les résultats vous semblent-ils surprenants ?

**Question 8 – Calculer des statistiques simples**

Calculez :

1. la moyenne et la médiane des émissions par habitant,
2. la valeur minimale et maximale.

```
df["Per capita total annual CO2 emissions from aviation"].mean()
df["Per capita total annual CO2 emissions from aviation"].median()
df["Per capita total annual CO2 emissions from aviation"].min()
df["Per capita total annual CO2 emissions from aviation"].max()
```

Interprétez ces résultats : que nous apprennent-ils sur la distribution des émissions ?

### Question 9 – Sélectionner un pays

Filtrez les données pour un pays de votre choix (par exemple, la France) :

```
df[df["Entity"] == "France"]
```

Quelle est la valeur des émissions par habitant pour ce pays ?

### Question 10 – Créer une visualisation simple

Tracez un graphique des 10 pays les plus émetteurs :

```
top10 = df.sort_values(by="Per capita total annual CO$_2$ emissions from aviation", ascending=False)
top10.plot(kind="bar", x="Entity", y="Per capita total annual CO$_2$ emissions from aviation")
```

Commentez le graphique obtenu.

## 3. Statistiques descriptives et interprétation économique

### Question 11 – Décrire la distribution des émissions

À l'aide de la commande suivante, obtenez les principales statistiques descriptives sur les émissions de CO<sub>2</sub> :

```
df["Per capita total annual CO$_2$ emissions from aviation"].describe()
```

Que représentent les valeurs affichées (moyenne, écart-type, minimum, quartiles, maximum) ? Comment interprétez-vous la différence entre la moyenne et la médiane ? Cela suggère-t-il une distribution symétrique ou asymétrique ?

### Question 12 – Identifier les valeurs extrêmes

Affichez les 5 pays ayant les émissions les plus faibles et les 5 ayant les plus fortes :

```
df.sort_values(by="Per capita total annual CO$_2$ emissions from aviation", ascending=True)
df.sort_values(by="Per capita total annual CO$_2$ emissions from aviation", ascending=False)
```

Que remarquez-vous ? Les pays les plus émetteurs ont-ils des caractéristiques communes (taille, niveau de revenu, situation géographique, insularité, etc.) ?

### Question 13 – Mesurer la dispersion

Calculez la variance et l'écart-type des émissions par habitant :

```
df["Per capita total annual CO$_2$ emissions from aviation"].var()
df["Per capita total annual CO$_2$ emissions from aviation"].std()
```

Ces deux indicateurs mesurent la dispersion des données autour de la moyenne. Est-ce que les émissions par habitant varient beaucoup d'un pays à l'autre ? Que signifient des valeurs élevées de ces indicateurs dans ce contexte ? Peut-on parler d'inégalités environnementales entre pays ?

**Question 14 – Comparer un pays à la moyenne mondiale**

Sélectionnez un pays (par exemple la France) et comparez ses émissions à la moyenne mondiale.

```
mean_value = df["Per capita total annual CO$_2$ emissions from aviation"].mean()
france_value = df.loc[df["Entity"] == "France", "Per capita total annual CO$_2$ emissions from aviation"]
print(france_value - mean_value)
```

La France est-elle au-dessus ou en dessous de la moyenne mondiale ? Comment expliquer cette position par des facteurs économiques ou géographiques ?

**Question 15 – Visualiser la distribution (optionnel)**

Réalisez un histogramme des émissions par habitant :

```
df["Per capita total annual CO$_2$ emissions from aviation"].hist(bins=30)
```

Décrivez la forme de la distribution : Est-elle concentrée ? étalée ? asymétrique ? Qu'est-ce que cela nous apprend sur la répartition mondiale des émissions aériennes ?

## Rappels : notions essentielles du TP Python

### 1. Fichiers et répertoires

- **Répertoire courant** : dossier dans lequel Python cherche les fichiers.
- Vérifier le répertoire courant :

```
import os  
os.getcwd()
```

- Changer de répertoire :

```
os.chdir("chemin/vers/votre/dossier")
```

### 2. Importer et manipuler des données

- **Bibliothèque utilisée** : pandas.
- Importer la bibliothèque :

```
import pandas as pd
```

- Charger un fichier CSV :

```
df = pd.read_csv("aviation_2024.csv")
```

- Afficher les premières lignes :

```
df.head()
```

### 3. Exploration d'une base de données

- Aperçu général :

```
df.info()
```

- Statistiques descriptives rapides :

```
df.describe()
```

- Noms des colonnes :

```
df.columns
```

- Vérifier les valeurs manquantes :

```
df.isna().sum()
```

### 4. Statistiques descriptives

- Moyenne :

```
df["colonne"].mean()
```

- Médiane :

```
df["colonne"].median()
```

- Minimum et maximum :

```
df["colonne"].min()  
df["colonne"].max()
```

- Variance et écart-type :

```
df["colonne"].var()  
df["colonne"].std()
```

- Distribution détaillée :

```
df["colonne"].describe()
```

## 5. Tri, sélection et filtrage

- Trier les observations :

```
df.sort_values(by="colonne", ascending=False)
```

- Sélectionner une colonne :

```
df["colonne"]
```

- Sélectionner un pays spécifique :

```
df[df["Entity"] == "France"]
```

## 6. Visualisation simple

- Histogramme :

```
df["colonne"].hist(bins=30)
```

- Diagramme en barres :

```
top10 = df.sort_values(by="colonne", ascending=False).head(10)
top10.plot(kind="bar", x="Entity", y="colonne")
```

## 7. Interprétation économique

- **Moyenne** : niveau moyen d'émissions par habitant.
- **Médiane** : valeur centrale, moins sensible aux extrêmes.
- **Écart-type / variance** : mesure la dispersion entre pays.
- **Distribution asymétrique** : présence de pays très émetteurs tirant la moyenne vers le haut.
- **Lecture économique** : les différences reflètent à la fois le niveau de développement, la taille du pays et son exposition au transport aérien.