

Introduction à Python pour l'analyse données

Python pour data scientists

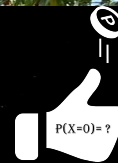


Table des Matières



Introduction

- ❑ Qu'est-ce que Python en Analyse de données?
- ❑ Installer Python via Anaconda
- ❑ Structures de données essentielles
- ❑ Projet pratique : Nettoyage de fichiers CSV simples avec Python



Introduction



Pourquoi ce cours ?

- ✓ **La maîtrise de l'analyse de données** est devenue incontournable. (Science, finance, marketing, Économie, Médecine, IA, ...)
- ✓ **Python est le langage de référence** pour le traitement et l'analyse de données

Ce que vous allez apprendre

- ✓ Bases de Python appliquées à la data
- ✓ **NumPy** : calcul numérique performant
- ✓ **Pandas** : manipulation de données tabulaires
- ✓ **Matplotlib & Seaborn** : visualisation claire et impactante

À qui s'adresse ce cours ? ?

- Étudiants, professionnels ou curieux de la data
- Aucune expertise préalable nécessaire
- Idéal pour **poser les bases solides** en Data Science



Table des Matières



- ❑ Introduction

- ❑ **Qu'est-ce que Python en analyse de données?**

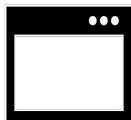
- ❑ Installer Python via Anaconda
- ❑ Structures de données essentielles
- ❑ Projet pratique : Nettoyage de fichiers CSV simples avec Python



Qu'est-ce que Python en Analyse de données?



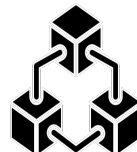
*Un langage puissant,
simple et universel*



*L'écosystème Python
Data Science*



*Une communauté
vivante et contributive*



*Python, le couteau
suisse de la donnée*



Interprété : accessible
même aux débutants

Typage dynamique

Compatible : Windows,
macOS, Linux, cloud & local

Pandas : manipulation de
données tabulaires

NumPy : calcul scientifique
rapide et efficace

Matplotlib / Seaborn :
visualisation

Scikit-learn : machine
learning

Jupyter / Colab / Anaconda :
environnement interactif

Documentation : abondante,
support communautaire actif

Énorme bibliothèque : de
packages open source

Développement : facile de vos
propres librairies

Connexion à toutes sources :
CSV, Excel, SQL, JSON, APIs,
Web, Big Data

Intégré dans des géants tech :
Instagram, Google, Netflix



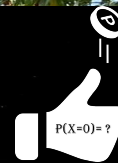
Table des Matières



- ❑ Introduction
- ❑ Qu'est-ce que Python en Analyse de données?

❑ **Installer Python via Anaconda**

- ❑ Structures de données essentielles
- ❑ Projet pratique : Nettoyage de fichiers CSV simples avec Python



Installer Python via Anaconda



Voir le tutoriel

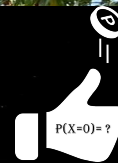
<https://github.com/openDataSenegal/BaseDeDonnees/blob/main/IntroductionBasesDeDonneesNoSQLElasticsearch.ipynb>



Table des Matières



- ❑ Introduction
- ❑ Qu'est-ce que Python en Analyse de données?
- ❑ Installer Python via Anaconda



❑ Structures de données essentielles

- ❑ Projet pratique : Nettoyage de fichiers CSV simples avec Python



Structures de données essentielles



Types Simples : immutables (leur valeur ne peut être changée après création).

- ❑ **int** : Nombre entier, **Exemple** a = 10
- ❑ **float** : Nombre à virgule flottante, **Exemple** : b = 3.14
- ❑ **str** : Chaîne de caractères (texte), **Exemple** : nom = "Alice"
- ❑ **bool** : Booléen (vrai/faux), **Exemple** : actif = True
- ❑ **NoneType** : Valeur nulle (absence de valeur)

Dictionnaires (dict) : Paires clé-valeur

- ❑ Idéal pour structurer et organiser l'information
- ❑ Flexible, dynamique et très utilisé en Data Science
- ❑ Exemple : etudiant = {'nom': 'Jean', 'age': 22}

Listes (list) : Collections ordonnées et modifiables

- ❑ Contient différents types : entiers, chaînes, autres listes
- ❑ Très utilisées pour stocker des séries de données
- ❑ Exemple : fruits = ['pomme', 'banane', 'orange']

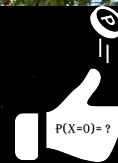
Tuples (tuple) : Collections immuables

- ❑ Plus performants que les listes
- ❑ Idéal pour représenter des données fixes (coordonnées, identifiants)
- ❑ Exemple : coordonnees = (45.75, -73.58)

Table des Matières



- ❑ Introduction
- ❑ Qu'est-ce que Python en Analyse de données?
- ❑ Installer Python via Anaconda
- ❑ Structures de données essentielles



❑ **Projet pratique : Nettoyage de fichiers CSV simples avec Python**





- ❖ Utilisation des bons types de données
- ❖ Optimisation des requêtes
- ❖ Utilisation des filtres
- ❖ Tuning des ressources système



Voir le tutoriel

<https://github.com/openDataSenegal/Open-Learning-Data-Science/blob/main/Python/IntroductionAPythonPourAnalyseDeDonnees.md>



Conclusion

?????

