



## Python avancé : programmation scientifique

LY002

Durée: 4 jours

### Public :

Développeurs en programmation scientifique.

### Objectifs :

Maîtriser les bibliothèques de programmation scientifique en Python  
Explorer et analyser des données  
Programmer des algorithmes et calculer de manière efficace en Python

### Connaissances préalables nécessaires :

Avoir suivi la formation "Python Initiation" ou avoir un niveau équivalent et avoir une pratique régulière du langage Python

### Programme :

#### Python avancé

Rappels concepts fondamentaux Python (architecture objet du langage, type de données, structure de données)  
Bonnes pratiques / Design de code : PEP8, structuration script et modules, POO, notion de packaging

**Atelier : Ecriture d'un code de base et transformation en paquet.**

Mise en oeuvre de `__init__`, `PYTHONPATH`, `site-packages`,  
`pyvenv`, `.pyc`, ...

Fonctionnalités avancées de Python : gestionnaire de contexte, liste par compréhension, itérateur, générateur

**Atelier : Rappels sur les types d'écritures en python : transformation d'un**

`for` en compréhension, utilisation des générateurs, suppression des `for`, ...

Evaluation de performances : outil de profilage (timing et mémoire)

**Atelier : présentation de quelques outils : `timeit`, `profler`, `memory_profler`,**

module `pstats`,

Limite des performances du langage Python natif

**Atelier : analyse de performances avec le mode profiling de `pycharm`**



# Phirio

---

## Manipulation de tableaux et opérations mathématiques avec NumPy

---

Manipulation de tableaux et opérations mathématiques avec NumPy. Durée : 1 jour  
Structure de donnée : les numpy array (différences avec les listes python, définition, type de données, allocation mémoire, shape, reshape)  
Création de tableau (random, à partir de données existantes, ...)  
Accès aux éléments du tableau : indexation simple, slicing et striding, indexation avancée (par tableaux d'entiers ou de booléens)  
Opérations sur les tableaux : opérations basiques, broadcasting, méthodes spécifiques (min, max, mean, std...), travailler à partir de formules mathématiques.  
Entrées/sorties avec Numpy  
Explication de l'optimisation des opérations sur les tableaux avec Numpy : vectorisation, allocation mémoire, binding C++, compilation sur architecture dédiée, optimisation des ressources du processeur (CPU/SIMD optimisation)  
Bannir les boucles for

Atelier : comparaison de performances avec Python natif sur la manipulation/calculs de gros tableaux

Profilage/timing

Atelier : analyse de l'empreinte mémoire avec memory\_profiler

---

## Calcul scientifique avec SciPy

---

Présentation générale de librairie : integrate, linalg, fftpack, signal, ndimage, odr, optimize, spatial  
Quelques exemples : Fit de courbe, Algèbre linéaire, Fourier, Traitement du signal

Atelier : matrices. Calcul de valeurs propres. Calcul de la matrice de passage.

Atelier : mise en oeuvre d'un système Lotka-Volterra avec matplotlib

Atelier : mise en oeuvre d'un système de pendule. Exploitation des fonctions d'intégration.

---

## Exploration et analyse de données avec Pandas

---

Structure de données : DataFrame, Serie  
Fonctionnalités essentielles : Attributs, statistiques descriptives, type de données, ...  
Indexation et sélection de données : basique, accès par attribut, slicing, sélection par label, par position, par fonction callable, méthodes de sélection (isin, where), sélection par masque

Atelier : analyse de fichiers de températures mondiales : extraction des données, découpage, structuration.

Opérations mathématiques sur les DataFrames : concaténation, fusion, regroupement  
Travailler avec des données textuelles : gestion de chaînes, encodage, gestion des accents, caractères diacritiques  
Travailler avec des données manquantes : stratégie de remplissage, interpolations.

Atelier : nettoyage, préparation et regroupement des données précédentes. Mise en oeuvre des bfill, ffill, ... Gestion des nan.

Travailler avec des séries temporelles : les timestamps, calculs, gestion (origines, échelles, water-marking)

Atelier : gestion d'une fenêtre temporelle distante sur les données précédentes : calculs de moyenne, écart-type, min et max.



# Phirio

---

Outils d'entrée/sortie (csv, HDF5, ...)

Atelier : jointures multiples avec enrichissement, de fichiers csv vers un fichier HDF5

## Accélération de code Python

---

Présentation générale et rapide des outils existants pour accélérer les codes Python (Numba, Cython, Pythran)

Présentation de pyarrow pour le passage à l'échelle.

Passage à l'échelle et parallélisme avec Dask

Atelier : Réalisation d'un enrichisseur de données sur Dask entre fichiers parquets sur S3