

Fiche du cours

60 h

Titre :

PYT300 - Programmation python 3

Description :

Perfectionnement orienté production : concurrence (threading, multiprocessing, asyncio), profiling & performance, tests avancés (pytest : fixtures/mocks), métaprogrammation (décorateurs avancés), patterns et bases de DSA (structures/algos). Aperçu Cython/Numba, GUI Tkinter (option), et web (intro Django/Flask) + déploiement (packaging, Docker – aperçu).

Objectifs :

- Concevoir du code concurrent & asynchrone (threads/processus/async/await).*
- Profiler & optimiser (cProfile, timeit, tracemalloc ; principes).*
- Écrire des tests avancés (fixtures, mocks, coverage, CI basique).*
- Utiliser décorateurs avancés, itérables et patterns (Strategy/Factory – intro).*
- Accélérer avec Cython/Numba (aperçu) et préparer un déploiement simple.

Chapitres :

1. Décorateurs avancés & métaprogrammation*
2. Générateurs/itérables (avancé)*
3. Concurrency : threading & multiprocessing*
4. Asyncio : tâches, async/await, I/O non bloquant*
5. Tests avancés (pytest : fixtures/mocks, coverage)*
6. Profiling & optimisation (cProfile, timeit, tracemalloc)*
7. DSA (intro) : structures & algorithmes de base*
8. Cython/Numba (aperçu)*
9. Tkinter (option)*
10. Web (intro) : Flask / Django*
11. Déploiement & bonnes pratiques (packaging, Docker – aperçu)

À la fin :

Vous serez capables de développer, tester et optimiser des applications Python performantes, d'utiliser la concurrence et l'asynchrone, et de préparer un déploiement simple. Vous saurez aussi profiler (cProfile/tracemalloc), instrumenter votre code (logs/metrics), gérer la configuration (.env) et packager avec Docker. Vous aurez les bases pour suivre des spécialités (Django, Data, Big Data, ML), avec une hygiène de code pro (pytest + fixtures, type hints, ruff/black, mini-CI) pour aborder sereinement ces parcours.