

# Développement avancé avec Python

---

**Ronan Amicel**

([ronan@amicel.net](mailto:ronan@amicel.net))

**PLB Formation – 9-11 décembre 2019 – Levallois-Perret**

Licence : [CC-BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/)

## Module 1 : Packager un projet Python

- Packager son projet
  - Application vs bibliothèque
  - `setup.py`, `setup.cfg` et `pyproject.toml`
  - outils de *scaffolding*
- Gérer les dépendances de son application
  - `setup.py` vs `requirements.txt`
  - `pip-tools`, `pipenv`, `poetry`
- Distribuer son projet
  - Distribution source vs distribution binaire
  - Créer des *wheels*
  - Publier sur PyPI
  - Générer une version autonome de son application
- Compiler son code Python
  - Motivations
  - Principaux outils : `Cython`, `Numba`, `nuitka`

## Module 2 : Manipulations avancées des primitives en Python

- Décorateurs
  - décorateur simple
  - décorateur paramétré
  - décorateurs multiples
  - décorateur et context manager à la fois
- Classes personnalisées
  - génération dynamique (*class factory*)
  - métaclasses
  - décorateurs de classes
  - `__init_subclass__`
- Générateurs et coroutines
- Monkey patching
  - patcher à la main
  - patcher avec `unittest.mock.patch`
- Introspection du code

## Module 3 : Gestion des bases de données avec Python

- Vue d'ensemble des bases de données supportées
  - bases relationnelles
  - bases non relationnelles
- Faire du SQL « à la main » avec la DB API
- Utiliser SQLAlchemy
  - Core : la boîte à outils SQL
  - ORM : projeter ses tables vers des objets Python
- Migrations avec Alembic

## Module 4 : Concurrence et parallélisme

- Programmation non bloquante avec les threads
  - Le GIL et les limites des threads
- Exploiter plusieurs cœurs avec le multiprocessing
- Entrées-sorties asynchrones avec `asyncio`

## Module 5 : Programmation graphique

- Différentes solutions : Tkinter, Qt, GTK, wxWidgets et Kivy
  - cas d'usage
  - avantages et inconvénients
- Mise en pratique
  - création des éléments d'interface
  - mise en page (*layout*)
  - réaction aux actions de l'utilisateur

## Module 6 : Développement web avec Python

- Frameworks les plus populaires : Django, Flask, Pyramid...
- Principes
  - le protocole HTTP
  - le cycle requête / réponse
  - les normes WSGI et ASGI
- Mise en œuvre avec Flask