

Prérequis Python avancé

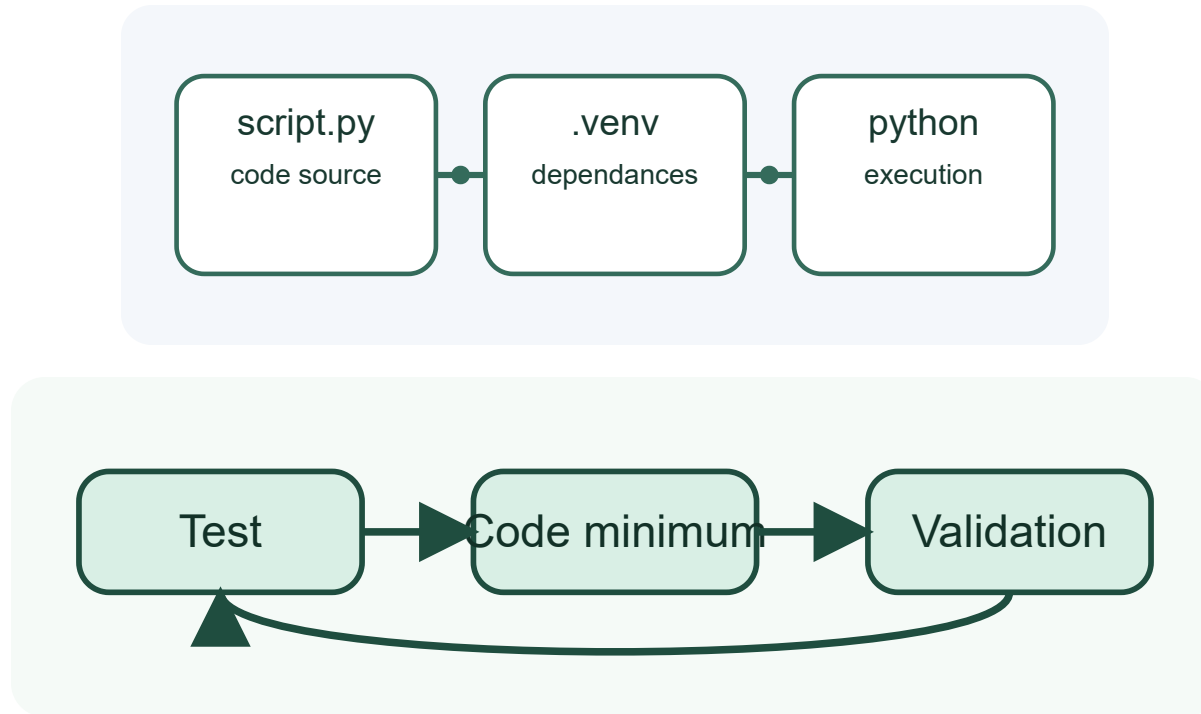
Socle commun avant d'attaquer l'automatisation, les tests et l'outillage



Changelog — V0.0.2

- 10/03/2026 04:56 — Création du module Python avancé et ajout du deck de prérequis.

Repères visuels du module



Runtime & environnement

- Savoir créer un environnement virtuel avec `python -m venv .venv`
- Toujours isoler les dépendances du système
- Vérifier rapidement le contexte avec `python --version` et `pip list`

```
python -m venv .venv
source .venv/bin/activate
python -m pip install --upgrade pip
```

Bases syntaxiques indispensables

- Comprendre `list`, `dict`, `set`, `tuple`
- Maîtriser les compréhensions sans perdre en lisibilité
- Utiliser `pathlib`, `dataclasses`, `enum`, `typing`

```
from dataclasses import dataclass
```

```
@dataclass(slots=True)
```

```
class User:
```

```
    login: str
```

```
    active: bool = True
```

Types & contrats

- Annoter les fonctions publiques avec `typing`
- Préférer les retours explicites aux structures implicites
- Faire tourner un vérificateur statique tôt (`mypy` , `pyright`)

```
def find_user(logins: list[str], target: str) -> str | None:  
    for login in logins:  
        if login == target:  
            return login  
    return None
```

Erreurs & exceptions

- Ne pas masquer une exception utile avec un `except Exception` trop large
- Enrichir l'erreur quand on remonte de couche
- Libérer les ressources avec `with`

```
from pathlib import Path

def load_config(path: Path) -> str:
    try:
        return path.read_text(encoding="utf-8")
    except OSError as exc:
        raise RuntimeError(f"Impossible de lire {path}") from exc
```

Structure de projet minimale

- `src/` pour le code applicatif
- `tests/` pour les tests
- `pyproject.toml` pour la configuration outillage
- `README.md` pour l'exécution locale

```
project/  
  pyproject.toml  
  src/app/__init__.py  
  src/app/service.py  
  tests/test_service.py
```

Checklist avant d'aller plus loin

- Environnement virtuel reproductible
- Imports propres et code formaté
- Typage des points d'entrée importants
- Tests unitaires lancés localement