

Un Python réinstallable sans internet

Pourquoi le Python système est le mauvais endroit où travailler, ce qu'est réellement un environnement virtuel, et comment transporter une installation fonctionnelle vers un portable de terrain qui n'a jamais vu d'index de paquets.

Pierre Robentz Cassion

Leçon 1 sur 8

Python pour les données de programme — Un environnement qui tiendra

Ce que couvre cette leçon

- L'échec que cette leçon prévient
- Ne travaillez pas dans le Python système
- Ce qu'est un environnement virtuel
- uv, et pourquoi il compte davantage ici qu'ailleurs
- Installer là où il n'y a pas d'internet
- Épingler aussi la version de Python
- Vérifier l'environnement avant de lui faire confiance
- Ce qu'il faut remettre à un collègue
- Ce qui vient ensuite

L'échec que cette leçon prévient

- Une analyse s'exécute sur votre portable.

Ne travaillez pas dans le Python système — Exemple

```
error: externally-managed-environment
```

```
x This environment is externally managed
```

```
+-> To install Python packages system-wide, try apt install  
    python3-xyz, where xyz is the package you are trying to  
    install.
```

Ne travaillez pas dans le Python système

- Cette erreur est justifiée et ne doit pas être contournée — Il existe une option pour passer outre

Ce qu'est un environnement virtuel — Shell

```
python3 -m venv .venv  
source .venv/bin/activate      # Windows : .venv\Scripts\activate  
python -m pip install pandas
```

Ce qu'est un environnement virtuel

- **Par projet, non par machine.** Deux analyses exigeant des versions de pandas différentes coexistent sans que l'une. . .
- **Jetable.** Si un environnement se retrouve dans un mauvais état, supprimez le répertoire et reconstruisez-le. Rien de. . .

uv, et pourquoi il compte davantage ici qu'ailleurs — Shell

Installer uv lui-meme, une fois, sur une machine effectivement connectee.

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

```
uv init muac-analysis
```

```
cd muac-analysis
```

```
uv add pandas openpyxl
```

```
uv run python -c "import pandas; print(pandas.__version__)"
```

uv, et pourquoi il compte davantage ici qu'ailleurs — Shell

```
uv sync
```

```
# lit uv.lock, installe exactement ces versions
```

Installer là où il n'y a pas d'internet — Shell

```
mkdir wheels
```

```
uv pip download pandas openpyxl --dest wheels
```

Installer là où il n'y a pas d'internet — Shell

```
uv venv
```

```
uv pip install --no-index --find-links wheels pandas openpyxl
```

Installer là où il n'y a pas d'internet

Les wheels sont spécifiques à une plateforme. Une wheel téléchargée sous macOS ne s'installera pas sur un portable Windows, et une wheel construite pour Python 3.12 ne s'installera pas dans 3.11. Téléchargez sur une machine correspondant à la cible, ou passez explicitement `-python-platform` et `-python-version`.

Épingler aussi la version de Python — Exemple

```
# pyproject.toml
[project]
name = "muac-analysis"
requires-python = ">=3.11"
dependencies = ["pandas>=2.2", "openpyxl>=3.1"]
```

Épingler aussi la version de Python — Shell

```
uv python install 3.12  
uv python pin 3.12
```

Vérifier l'environnement avant de lui faire confiance — En Python

```
import sys
import pandas as pd

print(sys.executable)           # doit se trouver dans .venv, non dans /usr/bin
print(sys.version)
print(pd.__version__)
```

Vérifier l'environnement avant de lui faire confiance — En Python

```
import sys

assert sys.version_info >= (3, 11), f"exige Python 3.11+, execute {sys.version}"
```

Ce qu'il faut remettre à un collègue

À versionner	À ne pas versionner
pyproject.toml	.venv/
uv.lock	wheels/
README.md avec les deux commandes	--pycache--/

Ce qu'il faut remettre à un collègue — Exemple

```
uv sync
```

```
uv run python scripts/indicator_table.py --input data/raw/muac.csv
```

Ce qui vient ensuite

- L'environnement est reproductible.

Lire la leçon complète, avec le code exécutable

[https://data-analysis.cassion.dev/fr/cours/python-for-programme-data/
python-01-environment](https://data-analysis.cassion.dev/fr/cours/python-for-programme-data/python-01-environment)