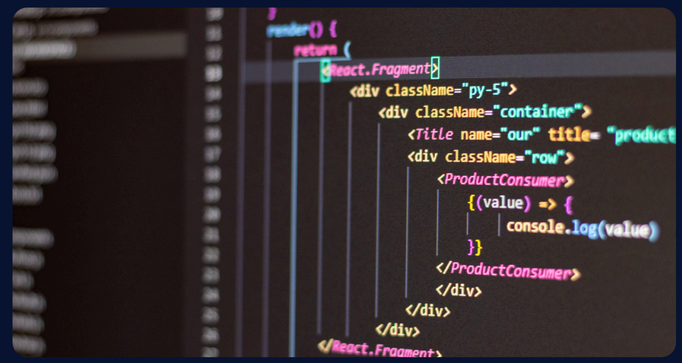


Modules et Packages

Laffet takwa

Qu’est-ce qu’un module en Python ?

Un module est simplement un fichier Python (.py) qui contient du code (fonctions, classes, variables…) que tu peux réutiliser dans d’autres fichiers.
==> C’est une brique de code réutilisable.

Création d’un module simple

Crée un fichier nommé maths_utils.py :

```python
# fichier : maths_utils.py

def addition(a, b):
    return a + b

def multiplication(a, b):
    return a * b

PI = 3.14159
```

Maintenant, dans un autre fichier Python (par exemple main.py), tu peux importer ton module :

```python
# fichier : main.py

import maths_utils

print(maths_utils.addition(4, 5))
print(maths_utils.multiplication(3, 2))
print(maths_utils.PI)
```

Différentes manières d’importer un module

Python permet plusieurs formes d’importation selon le besoin.
a) Importer tout le module

```
import maths_utils
print(maths_utils.addition(2, 3))
```

Différentes manières d’importer un module

b) Importer une fonction précise

```
from maths_utils import addition
print(addition(2, 3))
```

Différentes manières d’importer un module

c) Importer plusieurs fonctions

```
from maths_utils import addition, multiplication
```

Différentes manières d’importer un module

d) Importer tout le contenu

```
from maths_utils import *
```

Différentes manières d’importer un module

e) Donner un alias à un module

```
import maths_utils as mu
print(mu.addition(4, 5))
```

Le module intégré math

Python possède de nombreux modules intégrés qu’on peut utiliser sans installation.

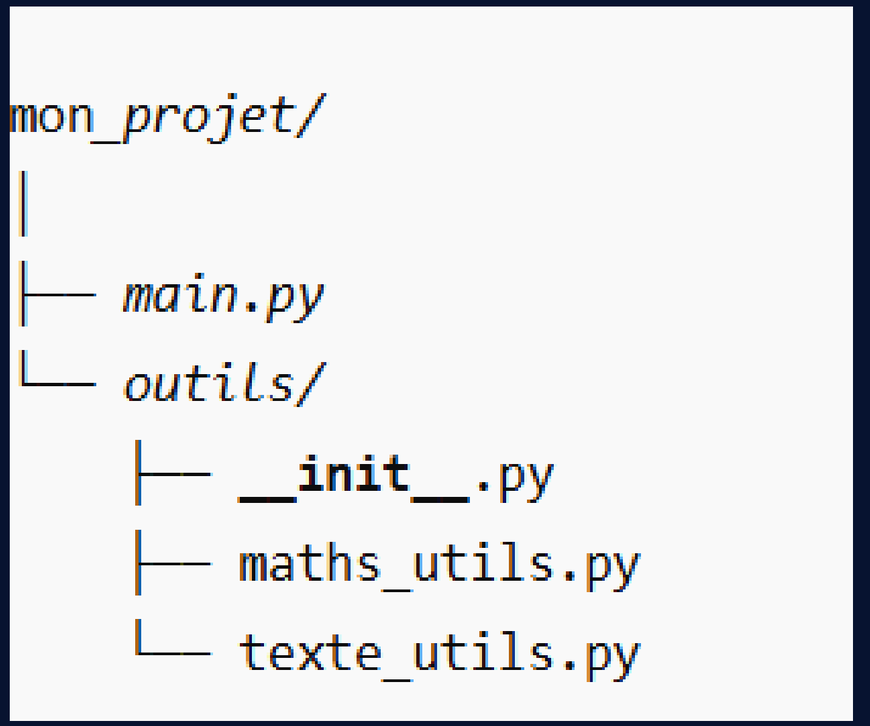
```
import math


print(math.sqrt(16))    # racine carrée → 4.0
print(math.pow(2, 3))   # puissance → 8.0
print(math.pi)          # 3.141592653589793
```

math est un module standard livré avec Python.

Qu’est-ce qu’un package ?

Un package est un dossier contenant plusieurs modules (fichiers .py) organisés ensemble.
 C’est une manière d’organiser de gros projets.

Le fichier __init__.py (même vide) indique à Python que ce dossier est un package.



Exemple de contenu

outils/maths_utils.py

```
def addition(a, b):
    return a + b
```

Exemple de contenu

outils/texte_utils.py

```
def majuscules(texte):
    return texte.upper()
```

Exemple de contenu

main.py

```python
from outils import maths_utils, texte_utils

print(maths_utils.addition(2, 3))      # 5
print(texte_utils.majuscules("python"))  # PYTHON
```

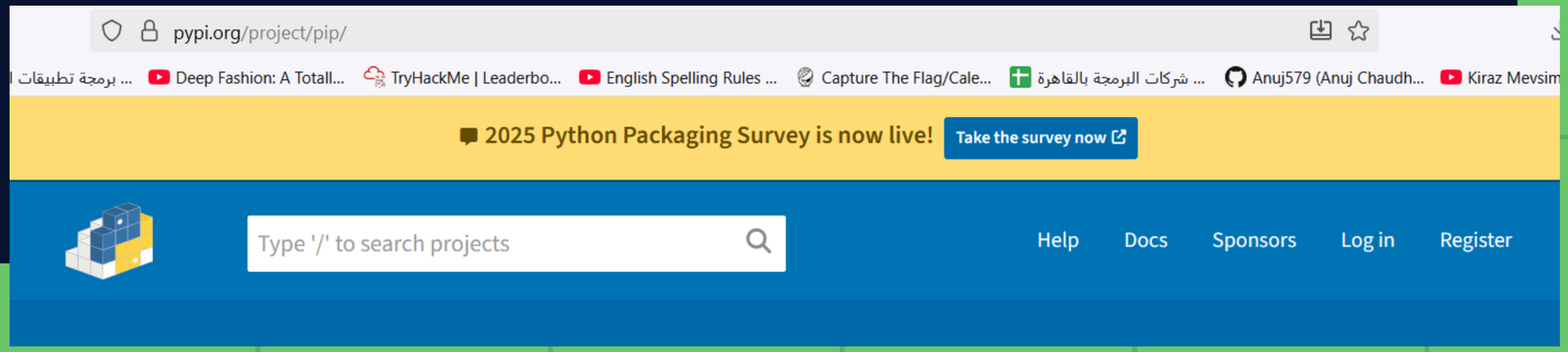
Packages intégrés et externes

Python possède :

- Des modules intégrés (ex: math, os, datetime, random, json, etc.)
- Des packages externes que tu peux installer avec pip.

Installer un package avec pip

Qu’est-ce que pip ?
pip signifie Package Installer for Python.
 C’est un gestionnaire de paquets : il sert à installer, mettre à jour ou désinstaller des bibliothèques Python disponibles sur PyPI (Python Package Index).



Vérifier si pip est installé

Dans ton terminal :

```bash
pip --version
```

Si tout va bien, tu verras quelque chose comme :

```csharp
pip 24.2 from ... (python 3.12)
```

Installer un package

Exemple : installer le package requests

```
pip install requests
```

Utiliser le package installé

```python
import requests


reponse = requests.get("https://api.github.com")
print(reponse.status_code)  # 200
```

Mettre à jour un package

```
pip install --upgrade requests
```

Désinstaller un package

```
pip uninstall requests
```

Lister les packages installés

```
pip list
```

Créer ton propre package installable

Si tu veux partager ton code avec d'autres, tu peux créer ton package Python.

setup.py contient les métadonnées du package :

```
from setuptools import setup, find_packages

setup(
    name='mon_package',
    version='1.0',

packages=find_packages(),
)
```

```
mon_package/
│
├── setup.py
└── mon_module/
    ├── __init__.py
    └── fonctions.py
```

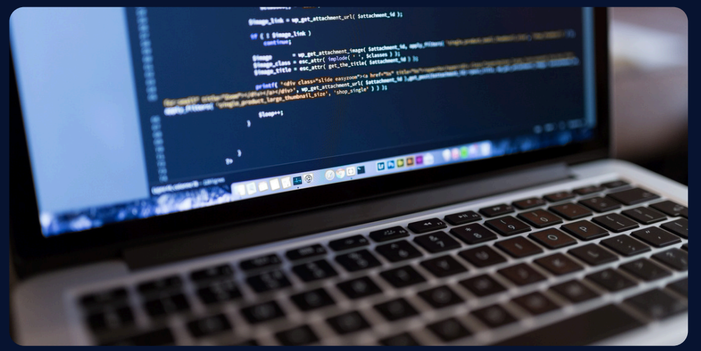
Ensuite, tu peux installer ton package localement :`pip install .`

Exercices

Exercice :
Crée un projet structuré comme suit :

```
mon_projet/
 │
 ├──── main.py
 └──── outils/
       ├──── __init__.py
       ├──── maths_utils.py
       └──── texte_utils.py
```

Dans maths_utils.py, ajoute une fonction carre(x)
 Dans texte_utils.py, ajoute une fonction inverser(texte)
 Dans main.py, importe ces fonctions et affiche leurs résultats.



Thank You