

LGLSI 1

Semestre 2

Programmation Python

Dhikra KCHAOU
dhikrafsegs@gmail.com

Présentation générale du cours

- ▶ **Volume Horaire:**

- ▶ 0,75 h par semaine COURS
- ▶ 1,5 h par semaine TP

- ▶ **Objectifs du cours:** Ce cours vise à permettre aux étudiants de :

- ▶ Analyser un problème donné et proposer une solution d'une manière rigoureuse et optimisée.
- ▶ Maîtriser le langage de programmation PYTHON.

Plan du cours

Chapitre 1 : Introduction à Python

Chapitre 2 : Les variables

Chapitre 3 : Les structures conditionnelles et les structures répétitives

Chapitre 4 : Les listes et les tuples

Chapitre 5 : Les ensembles et les dictionnaires



Chapitre 1: **Introduction à Python**

Plan du chapitre

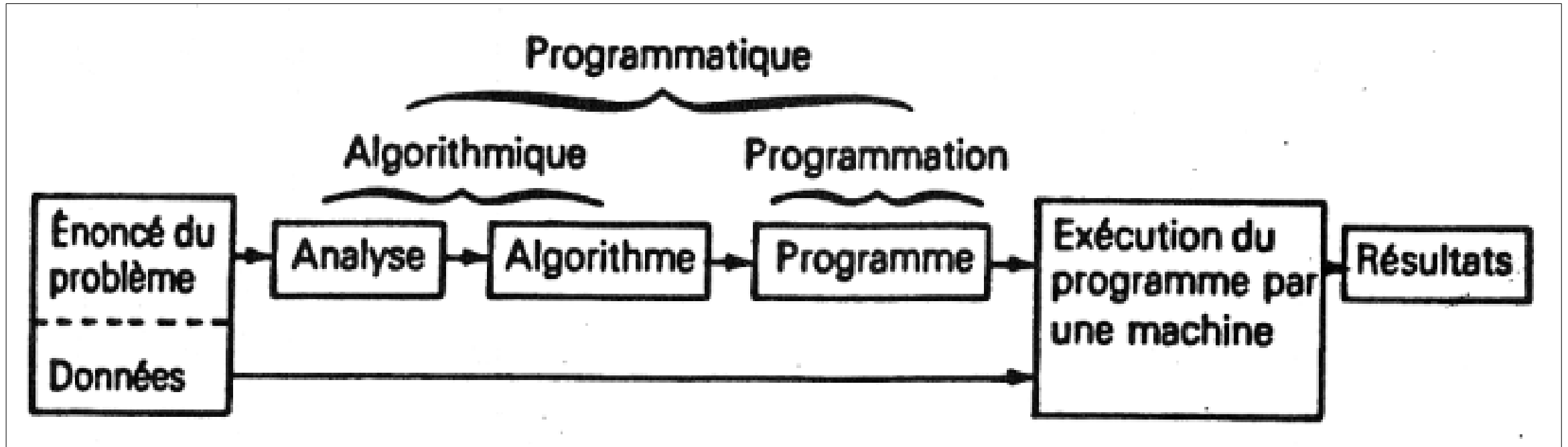
1. Notions de base de l'algorithmique
2. Présentation du langage Python
3. Installation du Python
4. Le Shell et le script de Python
5. Premier programme en python

Notions de base de l'algorithmique

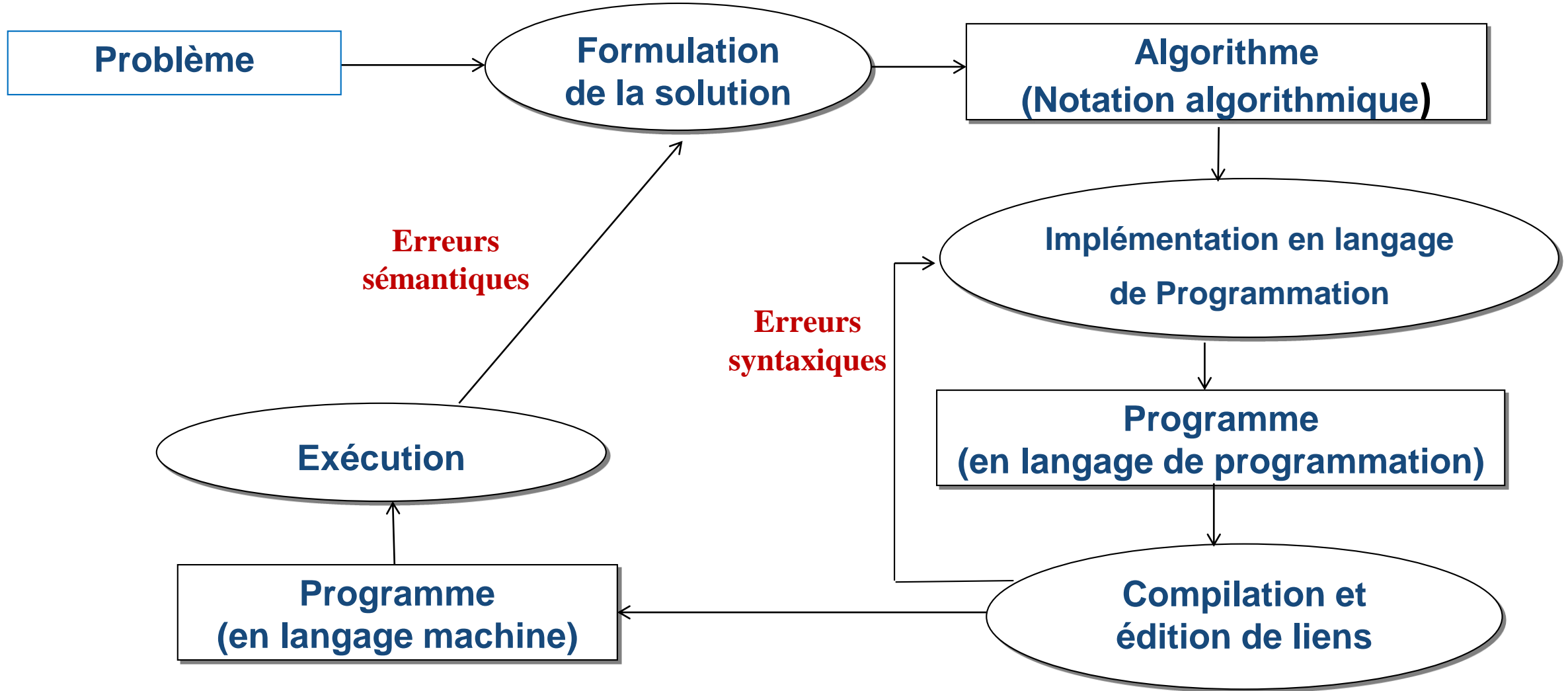
- ▶ **Un algorithme** suite de raisonnements ou d'**opérations** de calculs permettant de trouver la solution à un problème en un nombre fini d'étapes.
- ▶ **Un programme** : ensemble d'instructions permettant de définir les étapes élémentaires permettant de résoudre un problème. Le programme est **écrit en langage de programmation** et **sera chargé en mémoire** et **exécuté par la machine**.
- ▶ **La programmation**: regroupe l'ensemble des activités que le programmeur exerce pour générer des programmes fiables et comporte plusieurs étapes : **la réalisation (production du programme), analyse de performance (étude de complexité), le test et la maintenance de programmes.**

Étapes d'élaboration d'un programme

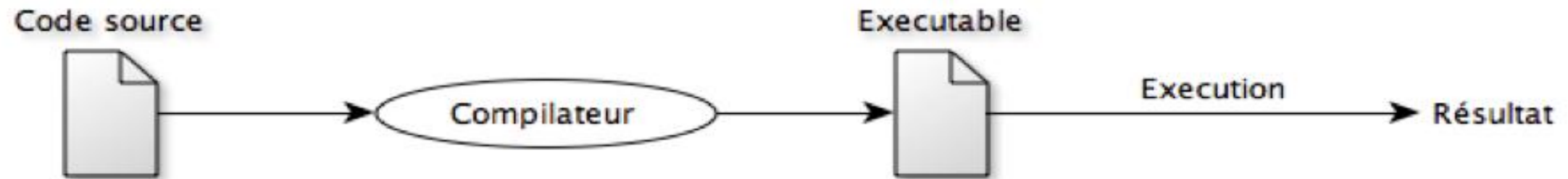
Où se trouve la **réalisation** d'un programme dans un langage de programmation dans **le cycle de développement d'un programme** ??



Processus de développement d'un programme

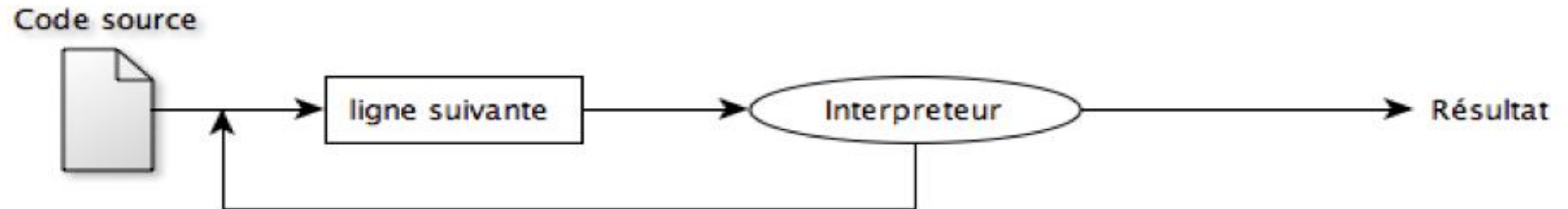


Compilation et interprétation



Compilation:

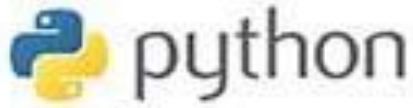
1. le code source est envoyé en intégralité au compilateur.
2. le compilateur traduit le code source en code machine.
3. le code machine est exécuté pour obtenir le résultat



Interpretation:

1. une ligne du code source est envoyée à l'interpreteur.
2. l'interpreteur traduit la ligne en résultat.
3. et demande une nouvelle ligne du code.

Langages de programmation



Langages de programmation

- L'index TIOBE est un indice qui mesure la popularité des langages de programmation. Il est publié chaque mois par la société TIOBE Software et se base sur plusieurs facteurs pour classer les langages de programmation en fonction de leur popularité.

Selon TIOBE 2023

Programming Language		Ratings
	Python	13.42%
	C	11.56%
	C++	10.80%
	Java	10.50%
	C#	6.87%
	JavaScript	3.11%
	Visual Basic	2.90%
	SQL	1.48%
	PHP	1.41%
	MATLAB	1.26%

C'est quoi Python?

- ▶ Langage de programmation de **haut niveau** inventé en 1991 par Guido van Rossum.
- ▶ Version actuelle de python 3.12

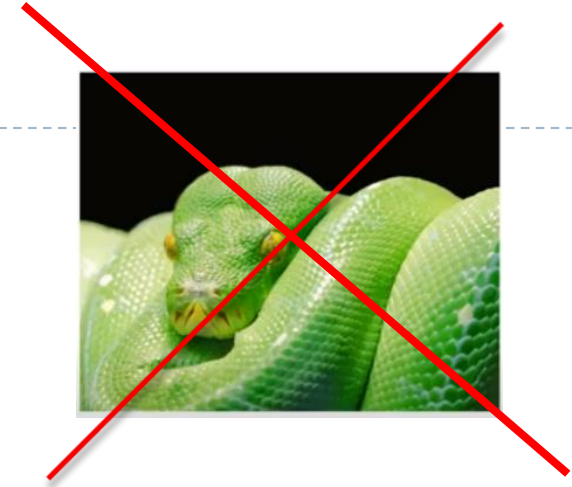


<https://www.python.org/downloads/>

Active Python Releases			
For more information visit the Python Developer's Guide .			
Python version	Maintenance status	First released	End of support
3.13	prerelease	2024-10-01 (planned)	2029-10
3.12	bugfix	2023-10-02	2028-10
3.11	bugfix	2022-10-24	2027-10
3.10	security	2021-10-04	2026-10
3.9	security	2020-10-05	2025-10
3.8	security	2019-10-14	2024-10

C'est quoi Python?

► D'où provient le nom de **Python** ??



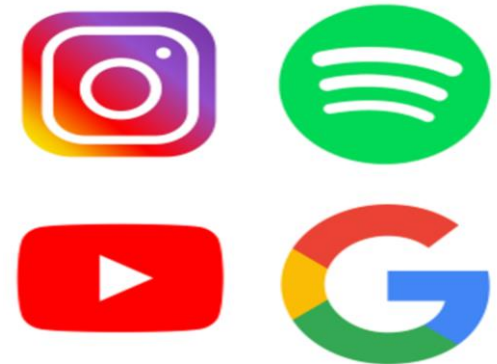
► Le nom **Python** vient d'un hommage à la série télévisée Monty Python's dont G. van Rossum est fan.



Monty Python's Flying Circus

C'est quoi Python?

- ▶ Utilité (liste non exhaustive):
 - ▶ **Applications de divertissement et jeux vidéo** : des applications comme Youtube, Netflix et spotify sont développées en python
 - ▶ **Science et analyse de données** : python propose une multitude de bibliothèques pour représenter facilement des données
 - ▶ **Applications Web**
 - ▶ **Applications bureautiques**

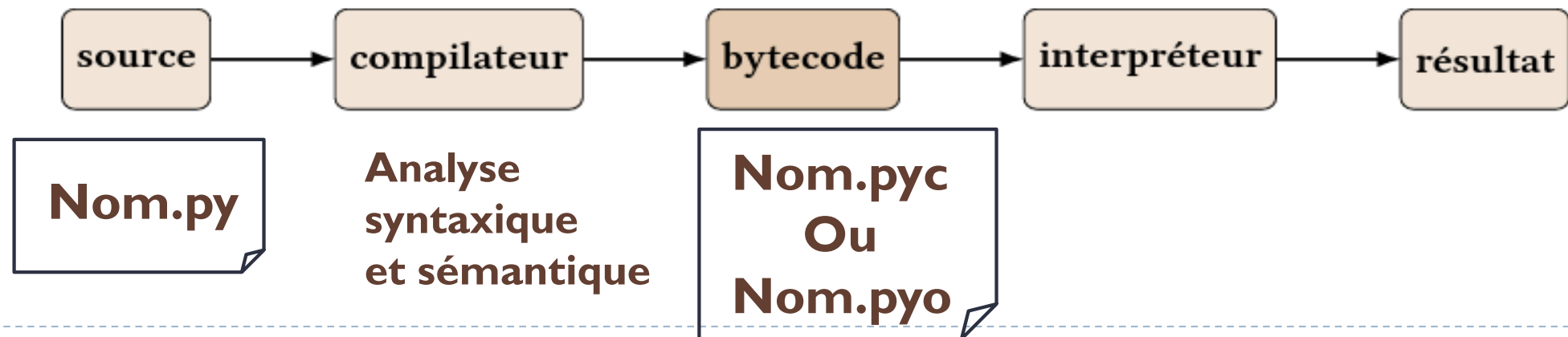


Caractéristiques de Python

- ▶ **Multiplateforme** : fonctionne sur de nombreux systèmes d'exploitation : Windows, Mac OS X, Linux, Android, iOS, depuis les mini-ordinateurs Raspberry Pi jusqu'aux supercalculateurs.
- ▶ **Gratuit** : sous licence libre, nous pouvons le télécharger, l'utiliser, le modifier, ...
- ▶ **Un langage de haut niveau** : se rapproche du langage naturel.
- ▶ **Un langage interprété** : nécessite un interpréteur pour s'exécuter.
- ▶ **Un langage orienté objet** : il est possible de concevoir en Python des entités qui encapsulent les données et les traitements à faire.

Python : langage interprété

- ▶ Technique mixte : compilation + interprétation
 1. Le programme source est tout d'abord compilé pour générer le bytecode (.pyc ou .pyo).
 2. Le bytecode est par la suite interprété.
- ▶ Le bytecode (forme intermédiaire) est portable sur tout ordinateur muni de la machine virtuelle Python.



Installation de Python: un éditeur de code ou un IDE

- ▶ Un **éditeur de texte/code** est un outil qui permet d'écrire et de modifier le code avec certaines fonctionnalités telles que la coloration syntaxique et le formatage du code.
 - ▶ **Exemples:** Visual Studio Code, Sublime Text, GNU Emacs, ...
 - ➡ Il fournissait moins de fonctionnalités que l'IDE.
- ▶ Un **Environnement de développement intégré IDE** sont des plates-formes dédiées pour écrire, compiler, exécuter, tester et déboguer un code Python.
 - ▶ **Exemples:** IDLE, PyCharm, ...

Environnement de développement intégré IDE



IDLE



PyCharm



Atom



PyDev



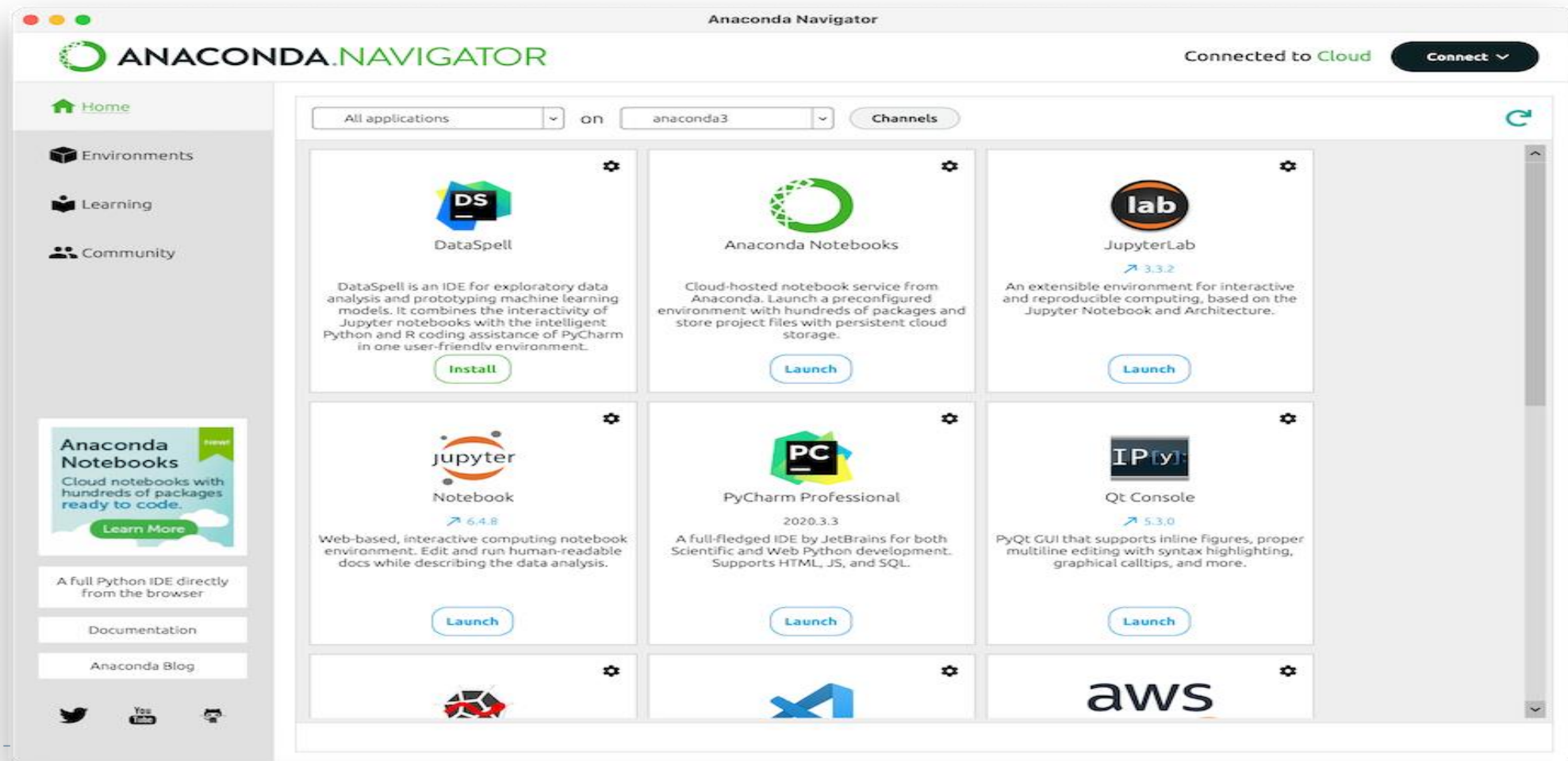
Jupyter Notebook



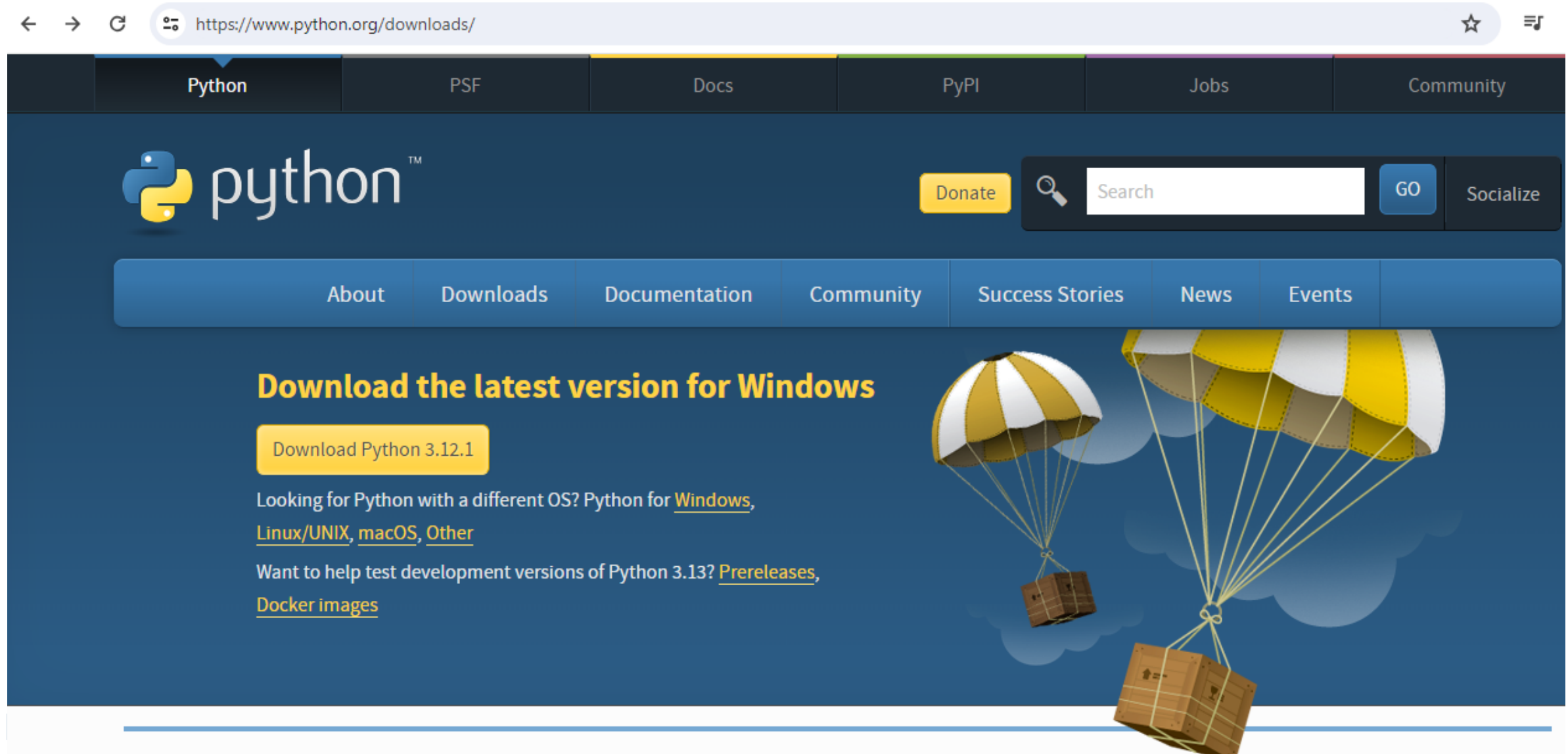
Spyder

Anaconda

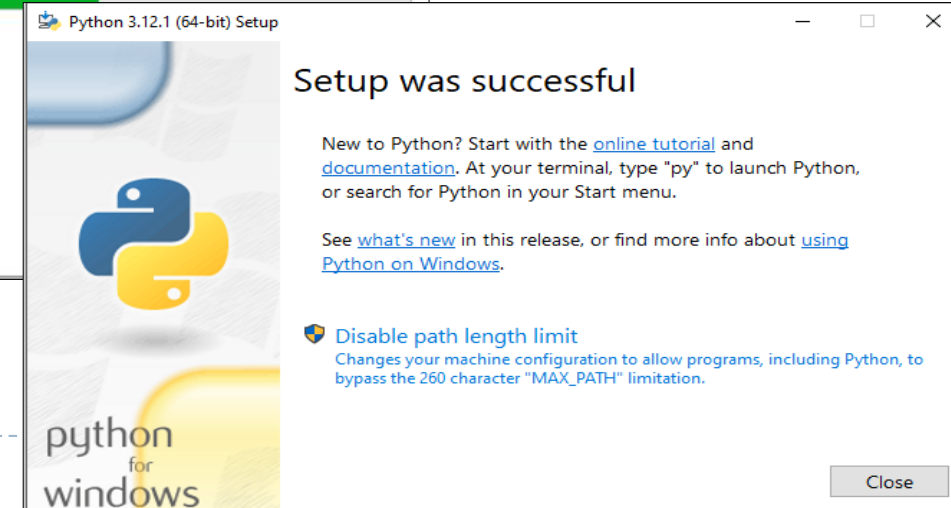
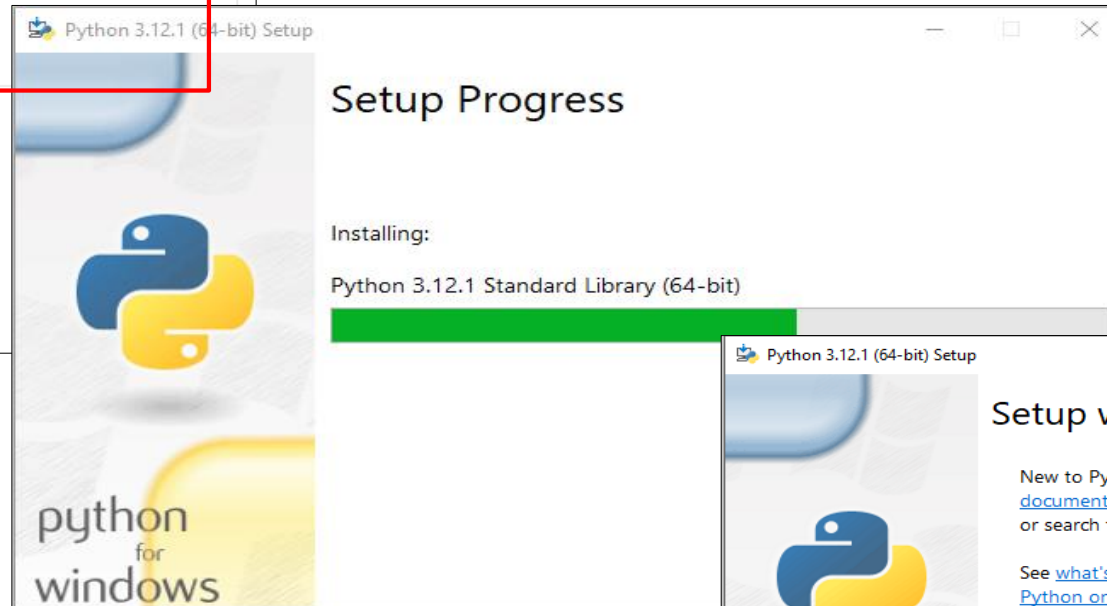
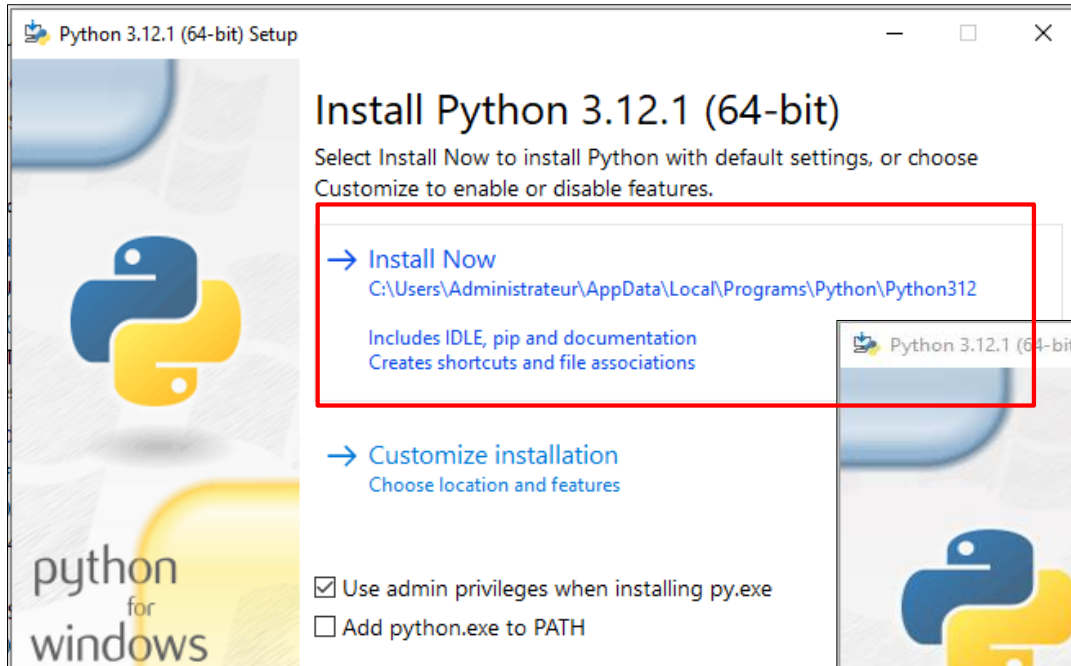
► <https://www.anaconda.com/>



IDLE : Integrated Development and Learning Environment



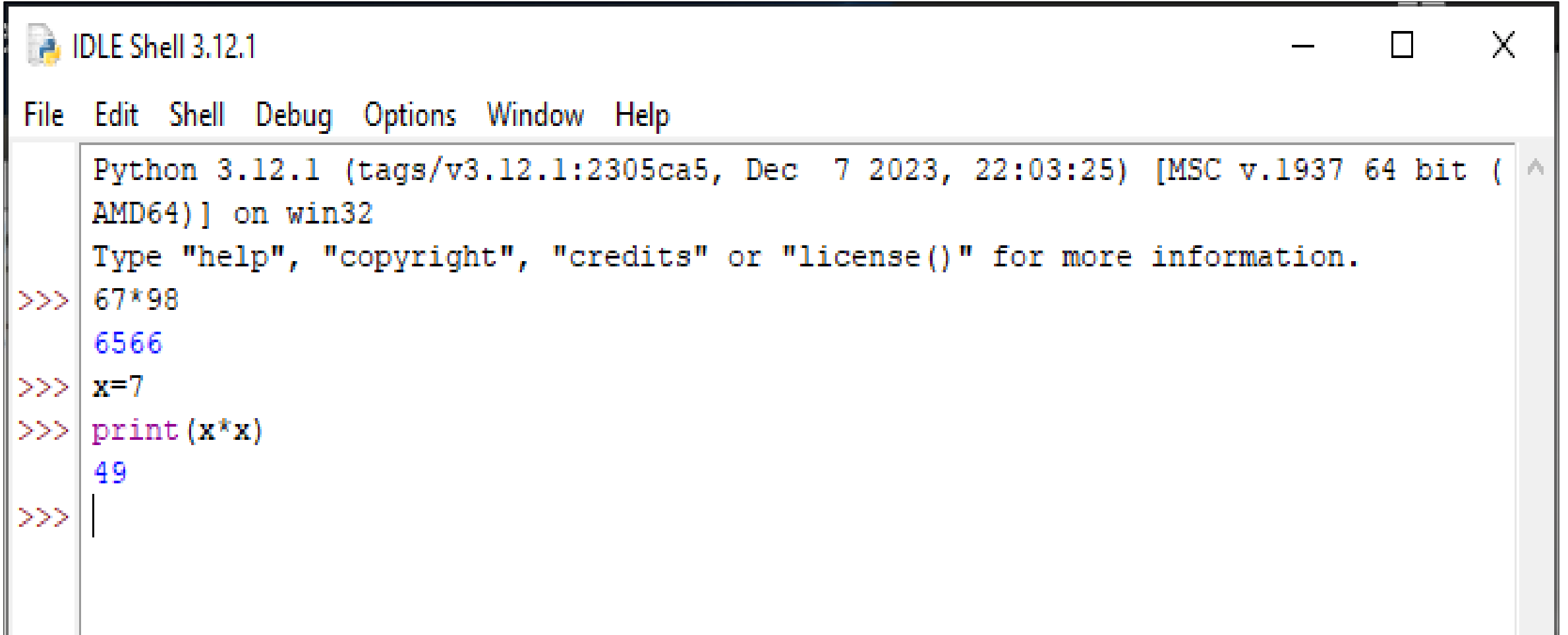
Installation de l'IDLE Python



Le Shell de Python

- ▶ Un shell est un interpréteur de commandes interactif permettant d'interagir avec l'ordinateur.
- ▶ Le triple chevron `>>>` est l'invite de commande (prompt en anglais) de l'interpréteur Python.
- ▶ Tapez une instruction puis validez cette commande en appuyant sur la touche Entrée.

Utilisez le Shell comme une calculatrice



```
Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 67*98
6566
>>> x=7
>>> print(x*x)
49
>>> |
```

L'interpréteur Python

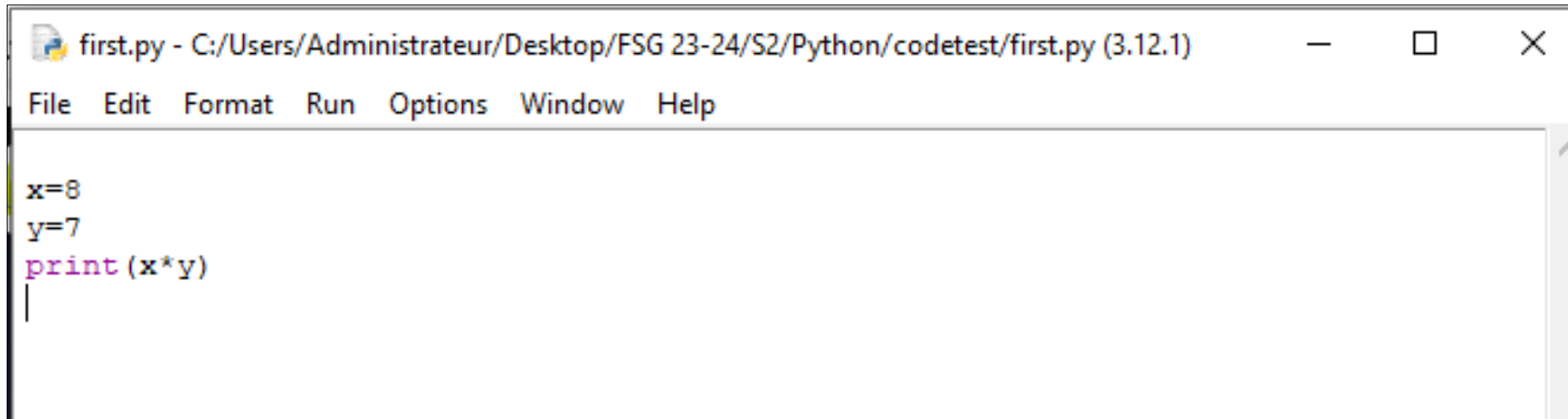
- ▶ À ce stade, vous pouvez entrer une autre commande ou bien quitter l'interpréteur Python, :
 - ▶ soit en tapant la commande `exit()` puis en validant en appuyant sur la touche Entrée,
 - ▶ soit en pressant simultanément les touches Ctrl et D sous Linux et Mac OS ou Ctrl et Z puis Entrée sous Windows
- ▶ **En résumé**, l'interpréteur fonctionne comme suit:
 - ▶ `>>>` instruction python
 - ▶ Résultat

Interpréteur Python

- ▶ Problèmes lors de l'utilisation de l'interpréteur:
 - ▶ Difficulté de retourner en arrière pour modifier une ligne
 - ▶ Pas de sauvegarde du travail
- ➔ L'interpréteur présente des limites lorsqu'on veut exécuter une suite d'instructions plus complexe.
- ➔ La solution est d'enregistrer ces instructions dans un fichier, que l'on appelle communément un script(ou programme) Python.

Premier programme en python

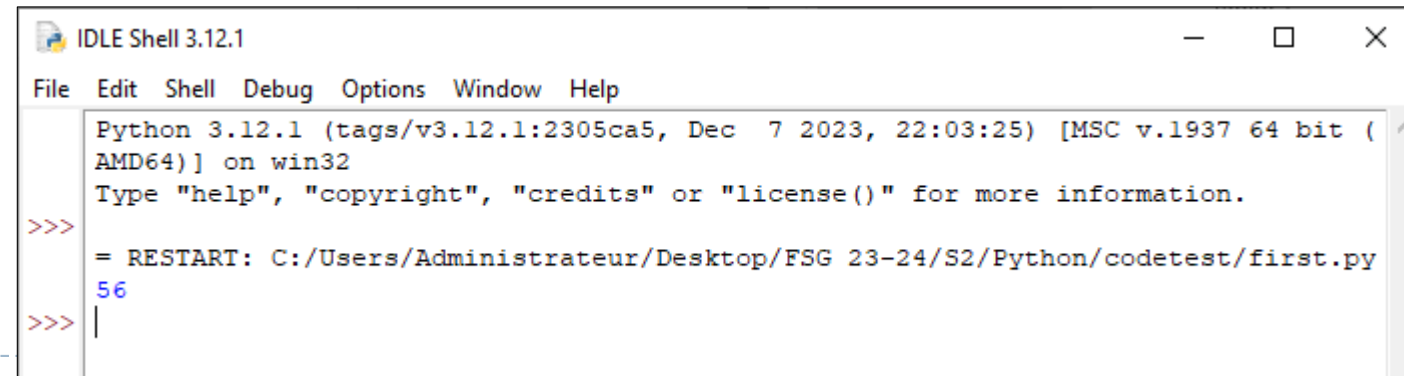
- Dans l'interpréteur de commande, File → new File



```
first.py - C:/Users/Administrateur/Desktop/FSG 23-24/S2/Python/codetest/first.py (3.12.1)
File Edit Format Run Options Window Help

x=8
y=7
print(x*y)
|
```

- Enregistrer le script (first.py)
- Run → Run Module



```
IDLE Shell 3.12.1
File Edit Shell Debug Options Window Help

Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Administrateur/Desktop/FSG 23-24/S2/Python/codetest/first.py
56
>>> |
```

Commentaire

- ▶ Un commentaire sert à :
 - ▶ Expliquer ce qui se passe dans une portion de code
 - ▶ Documenter qui a écrit le code ou d'autres informations
 - ▶ Désactiver une ligne de code

```
# commentaire sur une seule ligne
```

Commentaire sur une seule ligne #

```
"""
```

```
commentaire sur plusieurs lignes
```

```
"""
```

Commentaire sur plusieurs lignes """ """

Notion de bloc d'instructions et d'indentation

In C

```
#include <stdio.h>
int main()
{ //Beginning of Block 1
  int a;
  printf("Enter the easiest programming language: ");
  scanf("%d", &a);
  if (a == "python")
  { //Beginning of Code Block 2
    printf("Yes! You are right");
  } //End of Code Block 2
  else
  { //Beginning of Code Block 3
    printf("Nope! You are wrong");
  } //End of Code Block 3
  return 0;
} //End of Block 1
```

In Python

```
a = input('Enter the easiest programming language: ')

if a == 'python':
    print('Yes! You are right')
else:
    print('Nope! You are wrong')
```

Additional space before print is used to indicate a new block of code. Here, both the print are indented 2 spaces.

Ce décalage est appelé indentation, et l'ensemble des lignes indentées constitue un bloc d'instructions



Démonstration