



جامعة عبد المالك السعدي  
Université Abdelmalek Essadi



# Programmation Python



2023-2024

Prof : Anouar RAGRAGUI

## □ MODES D'EVALUATION

L'évaluation des connaissances peut se faire à l'aide d'exposés, contrôles continus (devoirs surveillés) , mini projet et d'un examen de fin de semestre

## □ NOTE DU MODULE

CC = Contrôle continu 30%, CTP=Contrôle TP 30% et EX = Examen 40%.

NM =Note du module =  $0,3*CC+0,3*CTP+0,6*EX$

# Plan du cours

## □ Chapitre 1: Introduction

- Chapitre 2: Les base du langage Python
- Chapitre 3: Notions avancées du langage Python



جامعة عبد المالك السعدي  
Université Abdelmalek Essadi



# Chapitre 1 : Introduction



# Plan

- Introduction
- Environnement de développement pour python
- Notion d'interpréteur et de script

- En informatique, un **langage de programmation** est une **suite de mots clés et de notations conventionnelles** permettant à un être humain de **dialoguer** avec une machine, généralement un ordinateur.
- Le langage de programmation est constitué par **un ensemble de « mots clés »** dont **chacun est associé à une action précise**.
- Il existe plusieurs **langages de programmation**, on cite par exemple C, C++, Java, PHP, Pascal, etc.
- Le **langage de programmation Python** a été créé en 1989 par Guido van Rossum, aux Pays-Bas.
- Le nom est dérivé de l'émission de télévision préférée de Guido van Rossum, «Monty Python's Flying Circus».
- La première **version** publique de ce langage a été publiée en **1991**.
- La **Python Software Fondation** est l'association qui organise le développement de Python et anime la communauté de développeurs et d'utilisateurs.



- ❑ Le **langage Python** est **multiplateforme**. C'est-à-dire qu'il fonctionne sur de **nombreux systèmes d'exploitation** : Windows, Mac OS X, Linux, Android, iOS.
- ❑ Le **langage Python** est **gratuit**. On peut **l'installer sur autant d'ordinateurs** qu'on veut (même sur des téléphones).
- ❑ C'est un **langage de haut niveau**. Il demande relativement **peu de connaissance** sur **le fonctionnement d'un ordinateur** pour être utilisé.
- ❑ C'est un **langage interprété**. Un **script Python n'a pas besoin d'être compilé** pour être **exécuté**, contrairement à des langages comme le C ou le C++.
- ❑ Le **langage Python** est **orienté objet**. C'est-à-dire qu'il est possible de concevoir en **Python** des entités qui **miment celles du monde réel** avec un certain **nombre de règles de fonctionnement et d'interactions**.
- ❑ La **syntaxe de Python** est conçue pour **être facile** à **lire** et à **comprendre**.
- ❑ La **communauté Python** est très **active** et propose **une documentation détaillée, des forums de discussion** et de nombreuses ressources **d'apprentissage en ligne**.
- ❑ Le **langage Python** est **très utilisé** en **analyse de données**

- **Développement Web** : On peut **créer** des **applications web** et des **sites web** en utilisant des **frameworks** comme **Django** et **Flask**.
- **Analyse et Visualisation de Données** : Python dispose de **puissantes bibliothèques** comme **pandas**, **NumPy** et **Matplotlib** qui vous permettent **d'analyser et de visualiser des données**.
- **Apprentissage Automatique et Intelligence Artificielle** : Python est devenu le langage de prédilection pour **les projets d'apprentissage automatique** et d'**IA**. Des bibliothèques comme **TensorFlow**, **Keras** et **scikit-learn**.
- **Développement de Jeux** : Python peut également être utilisé pour le **développement de jeux**. Des bibliothèques comme **Pygame**.
- **Applications de Bureau** : On peut **créer** des applications de bureau en utilisant des **frameworks** comme **PyQt** ou **Tkinter**. Ces **applications** peuvent avoir des **interfaces utilisateur graphiques** (GUI).



- **Programmation Réseau** : Python peut être utilisé pour **créer des applications réseau**, telles que des **scanners réseau**, des **applications de chat**, etc. La **bibliothèque socket** est couramment utilisée pour la **programmation réseau**.
- **Cybersécurité** : les tests de **pénétration**, la **recherche de vulnérabilités** et la **création d'outils de sécurité**.
- **Le calcul Scientifique** : Python est utilisé dans la **recherche scientifique** et **l'ingénierie** pour des tâches telles que les **simulations**, la **modélisation** et les **calculs complexes**.
- **Traitement du Langage Naturel (NLP)** : Python est largement utilisé dans les tâches de traitement **du langage naturel**, telles que la **génération de langage**, **l'analyse de sentiment**, la **classification de texte**, etc.
- **Internet des Objets (IoT)** : Python peut être utilisé pour **interagir** et **contrôler** des dispositifs IoT, ainsi que pour **traiter** les données qu'ils collectent.



Plus de 75 % du contenu recommandé par Netflix est généré à partir de **Machine Learning** programmé en **Python**



90 % de toutes les publications sur Facebook sont **filtrées grâce** à **l'apprentissage automatique** programmé en **Python**.



est estimé que 75 % des **systèmes d'automatisation des flux de travail** de la NASA à bord de la Station spatiale internationale **utilisent Python**.



95 % de tous les programmeurs débutants commencent avec et continuent **d'utiliser Python** comme **langage principal ou secondaire**.

- Présentation
- Caractéristiques
- Domaines d'application de Python
- Python en chiffres



On estime que la **science des données** avec **Python** deviendra le poste **le plus recherché** dans les années à venir.



Google est l'entreprise qui embauche le plus **de développeurs Python**, suivie de **près** par Microsoft.



La **science des données**, la **blockchain** et l'**apprentissage automatique** sont les compétences en **codage Python** à la croissance **la plus rapide**.



New York et San Francisco sont les principales villes du monde pour les **développeurs Python**.

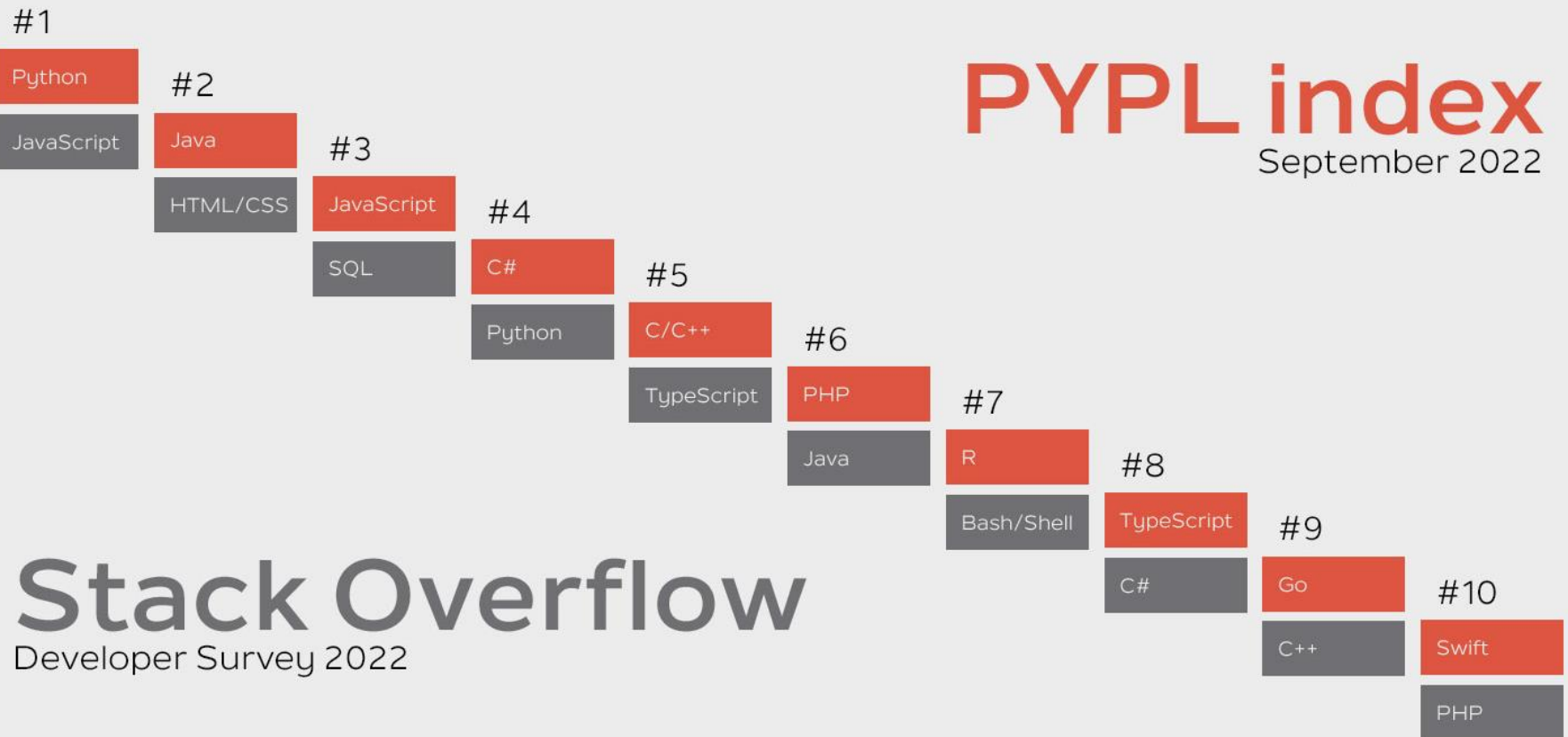


## Introduction

Environnement de développement pour python  
Notion d'interpréteur et de script

- ▶ Présentation
- ▶ Caractéristiques
- ▶ Domaines d'application de Python
- ▶ Python en chiffres

12



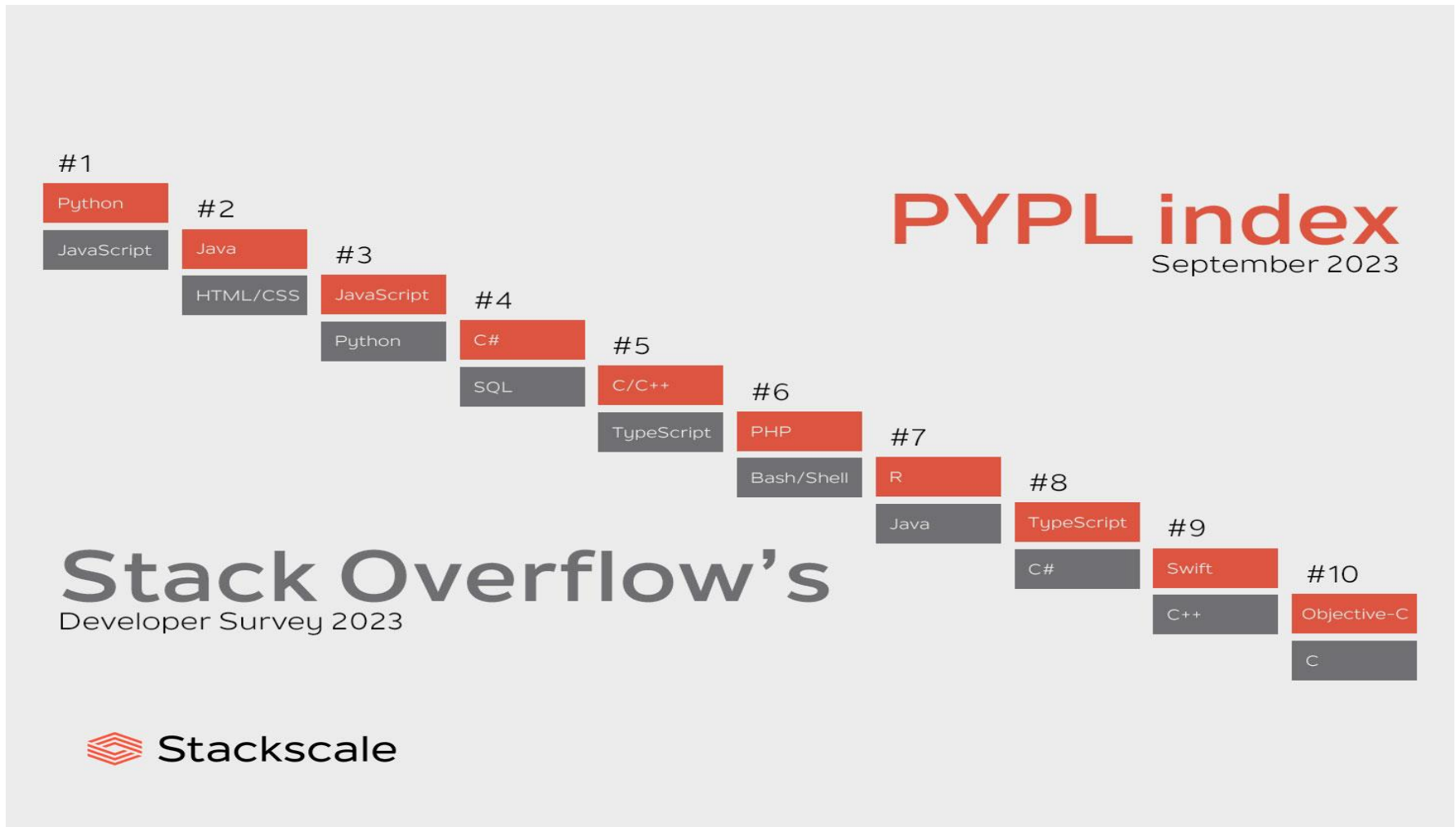
Most popular programming languages in 2022 [Ranking] ([stackscale.com](https://stackscale.com))

# Introduction

Environnement de développement pour python  
Notion d'interpréteur et de script

- Présentation
- Caractéristiques
- Domaines d'application de Python
- Python en chiffres

13



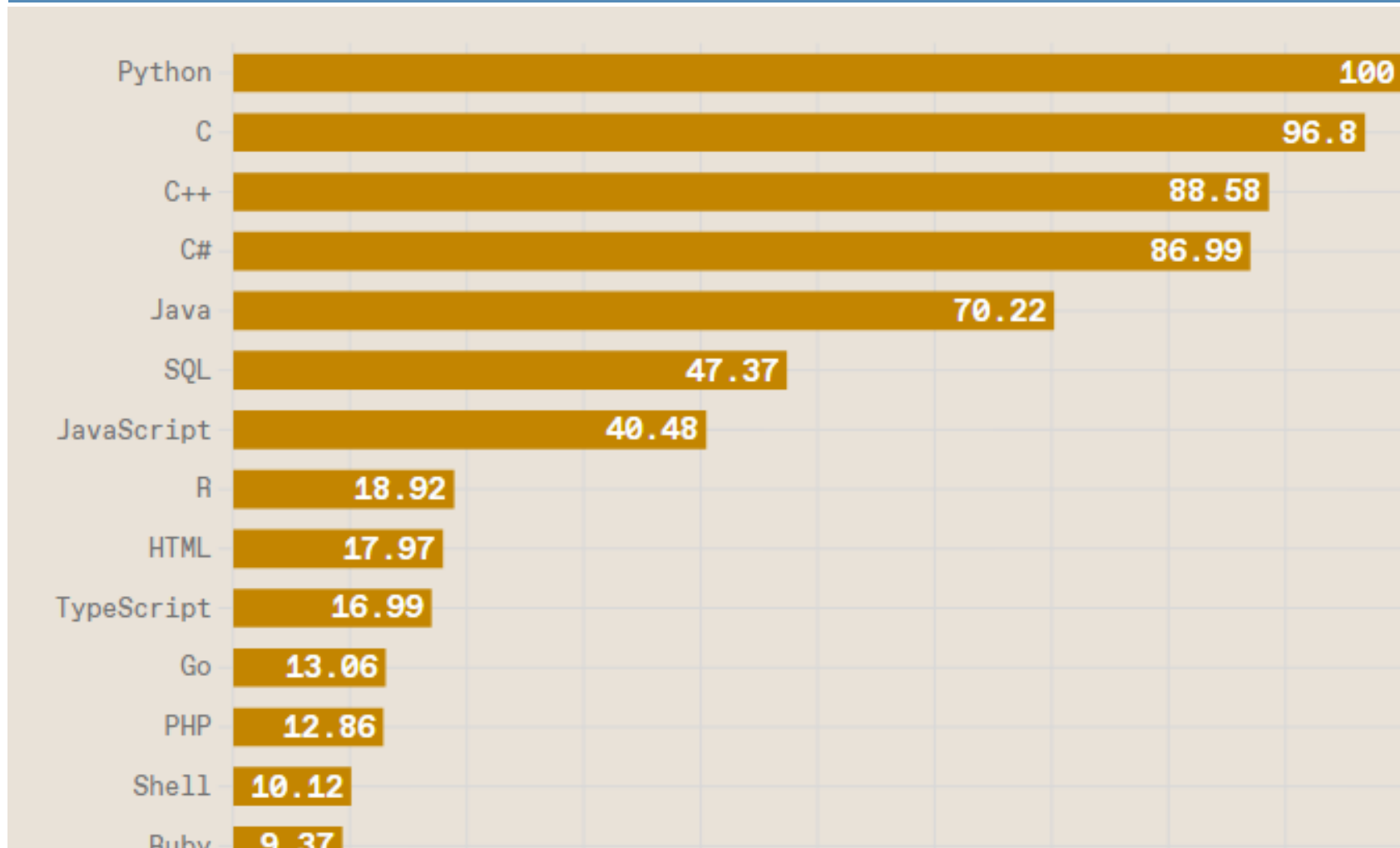
Most popular programming languages in 2023 [Ranking] ([stackscale.com](https://stackscale.com))

# Introduction

Environnement de développement pour python  
Notion d'interpréteur et de script

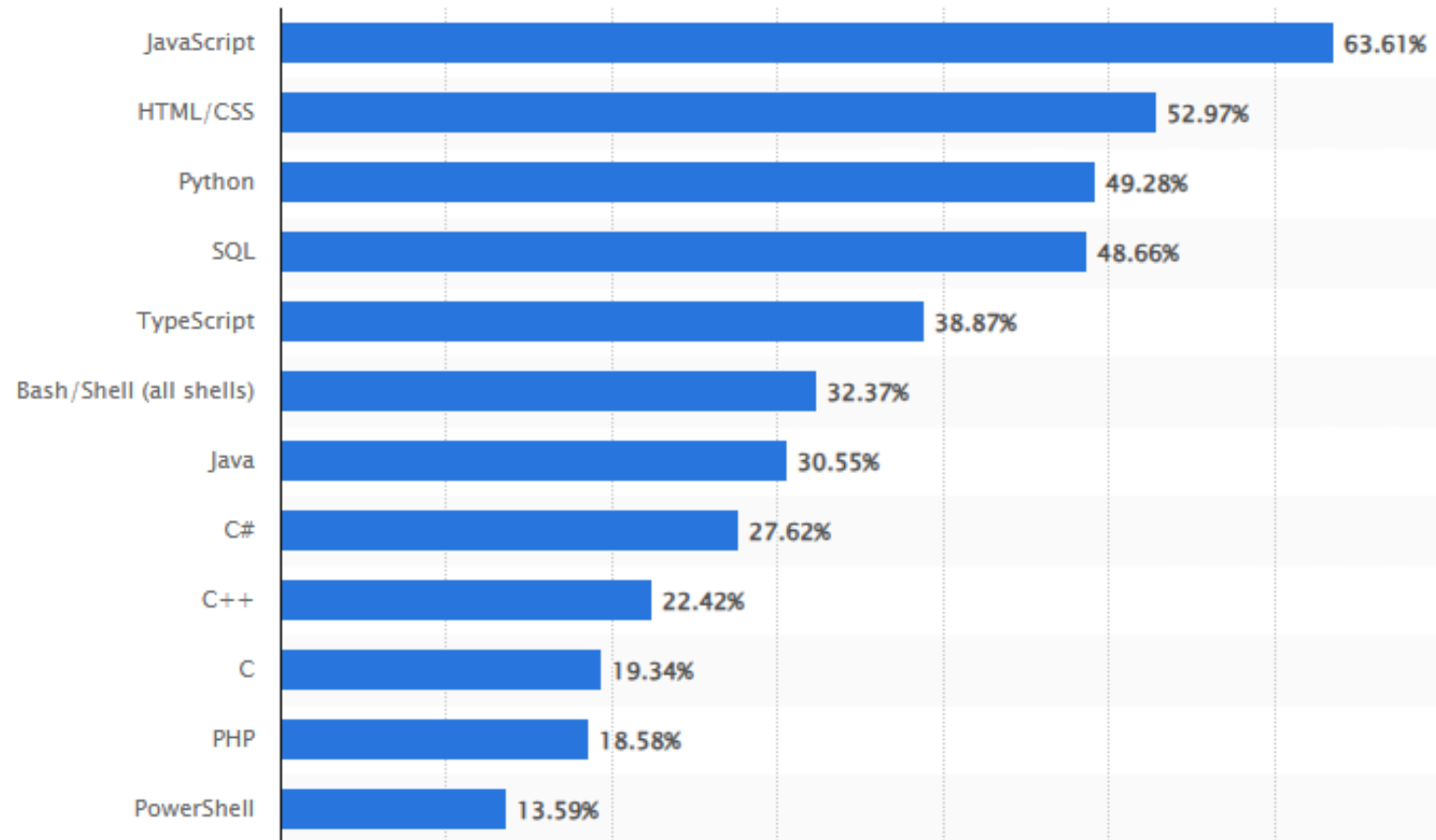
- Présentation
- Caractéristiques
- Domaines d'application de Python
- Python en chiffres

14









Top Programming Languages 2022 - IEEE Spectrum

- Présentation
- Caractéristiques
- Domaines d'application de Python
- **Python en chiffres**



- Présentation
- Caractéristiques
- Domaines d'application de Python
- Python en chiffres



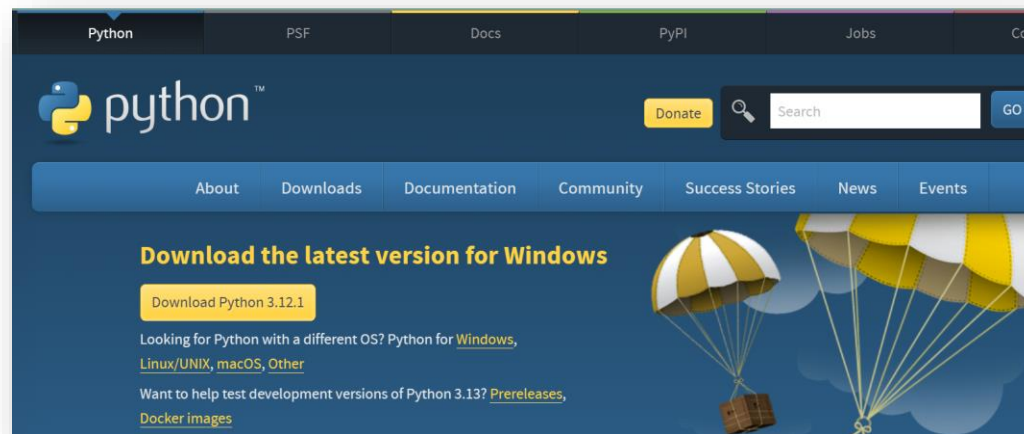
Aug 2023	Aug 2022	Change	Programming Language		Ratings	Change
1	1			Python	13.33%	-2.30%
2	2			C	11.41%	-3.35%
3	4	▲		C++	10.63%	+0.49%
4	3	▼		Java	10.33%	-2.14%
5	5			C#	7.04%	+1.64%
6	8	▲		JavaScript	3.29%	+0.89%
7	6	▼		Visual Basic	2.63%	-2.26%
8	9	▲		SQL	1.53%	-0.14%
9	7	▼		Assembly language	1.34%	-1.41%
10	10			PHP	1.27%	-0.09%

Selon l'Index TIOBE, « Python, qui est devenu un élément incontournable de la science des données, du DevOps et du développement web, est aussi désormais le langage le plus populaire dans le classement de TIOBE. Classé troisième langage le plus populaire de l'index au début de l'année 2021, Python s'est hissé à la première place en octobre 2022 ».



- Pour **installer Python**, il suffit de télécharger la dernière version qui correspond à votre système d'exploitation (Windows, Linux/UNIX, ou Mac) à l'adresse :

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



- **Remarque:**
  - Il est toujours recommandé d'utiliser la dernière version stable de **Python**, car elle comprend des **améliorations** de **performances**, de **sécurité** et de **fonctionnalités**.
  - On peut vérifier la version actuelle sur le site officiel de Python

- Pour écrire un programme en utilisant un langage de programmation, on utilise un logiciel appelé **Environnement de développement intégré (IDE)**.
- Ce logiciel nous permet **d'éditer les programmes**, **détecter les erreurs** et faire la **transformation en langage machine**.
- Il existe de nombreux **environnements de développement intégrés (IDE)** pour **Python**, chacun ayant ses propres **fonctionnalités** et **avantages**. Voici quelques-uns des IDE les plus connus pour Python :



- **PyCharm** (JetBrains) : Il est réputé pour sa **convivialité**, ses **fonctionnalités** avancées de débogage, son intégration avec des outils de gestion de versions, et son support pour le **développement web** avec **Django**.



- **Visual Studio Code** (Microsoft) : Il propose des extensions pour **Python** qui offrent un développement complet.



- **Jupyter Notebooks** : Jupyter est **une application Web open-source**. Il est souvent utilisé pour la **science des données** et **l'apprentissage machine**.



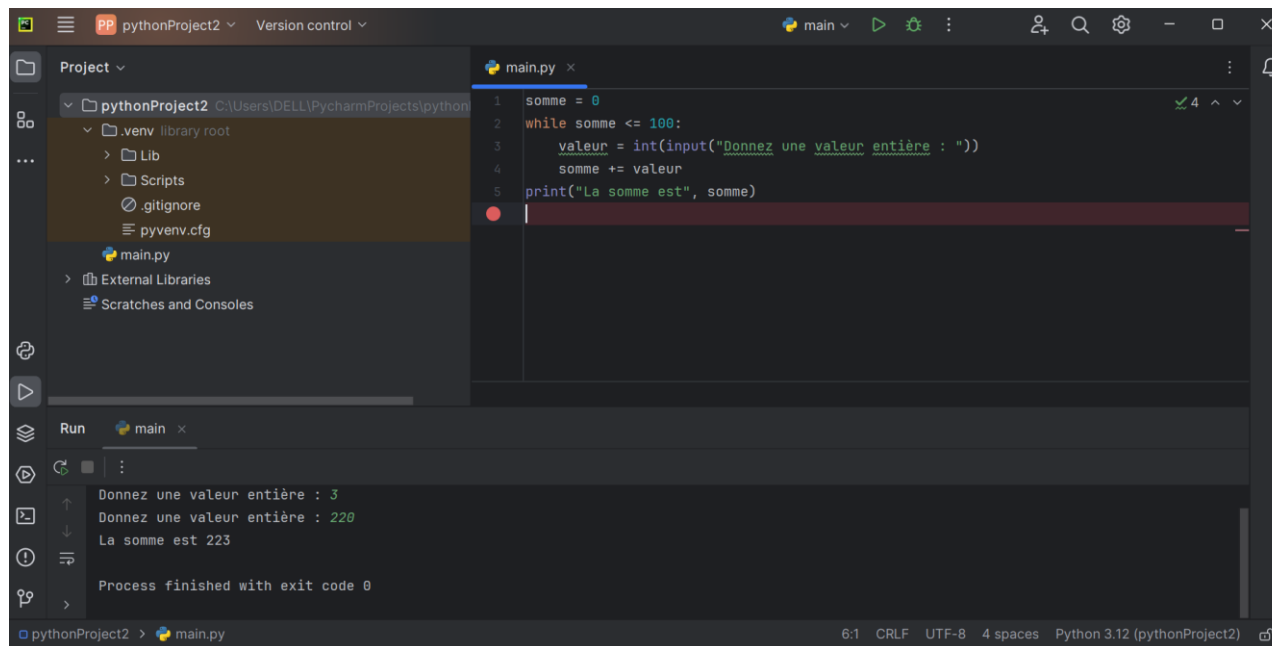
- **Spyder** : Un IDE conçu spécifiquement pour les scientifiques et les ingénieurs. Il propose des fonctionnalités avancées pour **le calcul numérique** et la **visualisation de données**.

- **Installation**
- Environnement de développement pour Python
- Les versions de Python

19

□ Pour **installer PyCharm**, il suffit de cliquer sur le lien :

Lien : <https://www.jetbrains.com/fr-fr/edu-products/download/#section=pycharm-edu>



- **Python** est **un langage de programmation populaire** qui a connu plusieurs **versions** au fil du temps.
- Les versions majeures de **Python** sont généralement notées avec un format **X.Y**, où:
  - ▣ **X** représente le numéro de **version principal**.
  - ▣ **Y** représente le numéro de **version secondaire**.
- Chaque **version** apporte des **améliorations**, des **fonctionnalités nouvelles** et des **corrections de bugs**.
- Voici quelques-unes des **versions majeures de Python** :
  - ▣ **Python 1.x** : Les premières versions de Python ont été publiées dans les années 1990. Ces versions initiales ont jeté les bases du langage.
  - ▣ **Python 2.x** : Cette série a été très populaire et a connu plusieurs versions, dont la dernière était Python 2.7. Cependant, Python 2 a atteint sa fin de vie le 1er janvier 2020, et il est fortement recommandé d'utiliser Python 3 pour tous les nouveaux projets.
  - ▣ **Python 3.x** : La série Python 3 a été introduite pour corriger certaines lacunes de conception de Python 2. Elle a apporté des améliorations significatives au langage. La dernière version stable de Python 3 était Python 3.10.

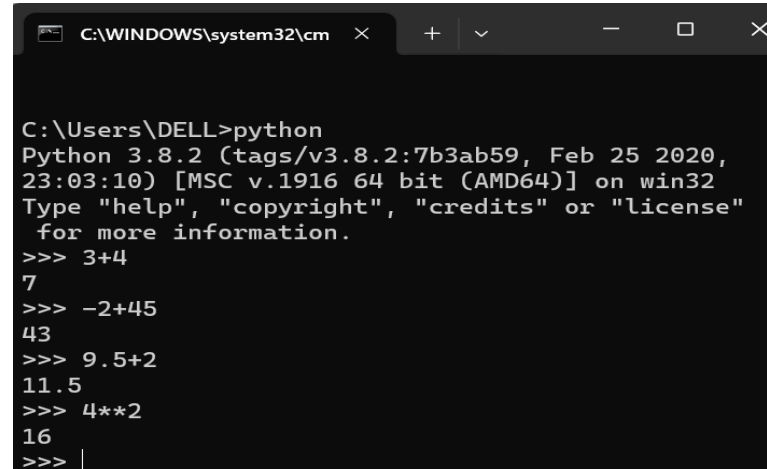
- Ce cours est basé sur **la version 3** de **Python**, qui est désormais le standard.
- **Remarque:** Il existe quelques différences importantes entre Python 2 et Python 3:
  - ▣ **Le mot-clé print / la fonction print()** : En Python 2 print est un mot-clé du langage. Il s'utilise ainsi sans parenthèse. Par contre en Python 3, print() est une fonction. Ainsi, si vous n'utilisez pas de parenthèse, Python vous renverra une erreur.
  - ▣ **Division d'entiers** En Python 3, la division de deux entiers, se fait avec l'opérateur / renvoie systématiquement un float. Il est également possible de réaliser une division entière avec l'opérateur //. En Python 2, la division de deux entiers avec l'opérateur / correspond à la division entière.
  - ▣ **Encodage et utf-8** : En Python 3, vous pouvez utiliser des caractères accentués dans les commentaires ou dans les chaîne de caractères. Ce n'est pas le cas en Python 2. Si un caractère accentué est présent dans votre code, cela occasionnera une erreur. Pour éviter ce genre de désagrément, ajoutez la ligne suivante en tout début de votre script :  
`# coding : utf -8`

- **Python** est **un langage interprété**, c'est-à-dire que **chaque ligne de code** est **lue** puis **interprétée** afin d'être **exécutée** par **l'ordinateur**.
- **L'interpréteur Python** est un **système interactif** dans lequel on peut entrer des **commandes**, que **Python exécutera** au moment où on validera la commande en appuyant sur la touche Entrée.
- Il existe de nombreux autres **langages** interprétés comme **Perl** ou **R**
- Le **gros avantage** de ce type de **langage** est qu'on peut immédiatement **tester une commande** à l'aide de **l'interpréteur**, ce qui est très utile pour **débugger** (c'est-à-dire trouver et corriger les éventuelles erreurs d'un programme).
- **L'interpréteur** présente vite des **limites** dès lors que l'on veut **exécuter une suite d'instructions** plus complexe.
- Toutes les **séquences d'instructions** écrites disparaissent irrémédiablement dès que nous **fermons** l'interpréteur.

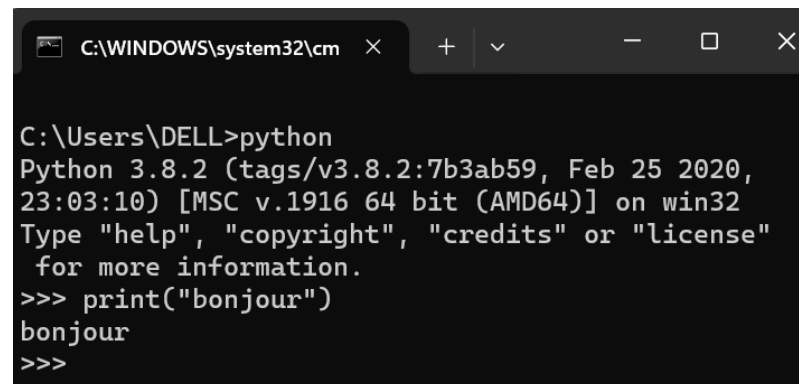
- ▶ Notion d'interpréteur
- ▶ Notion de script
- ▶ Création de script
- ▶ Exécution de script

23

## □ Exemple:



```
C:\WINDOWS\system32\cmd
C:\Users\DELL>python
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020,
23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
for more information.
>>> 3+4
7
>>> -2+45
43
>>> 9.5+2
11.5
>>> 4**2
16
>>>
```



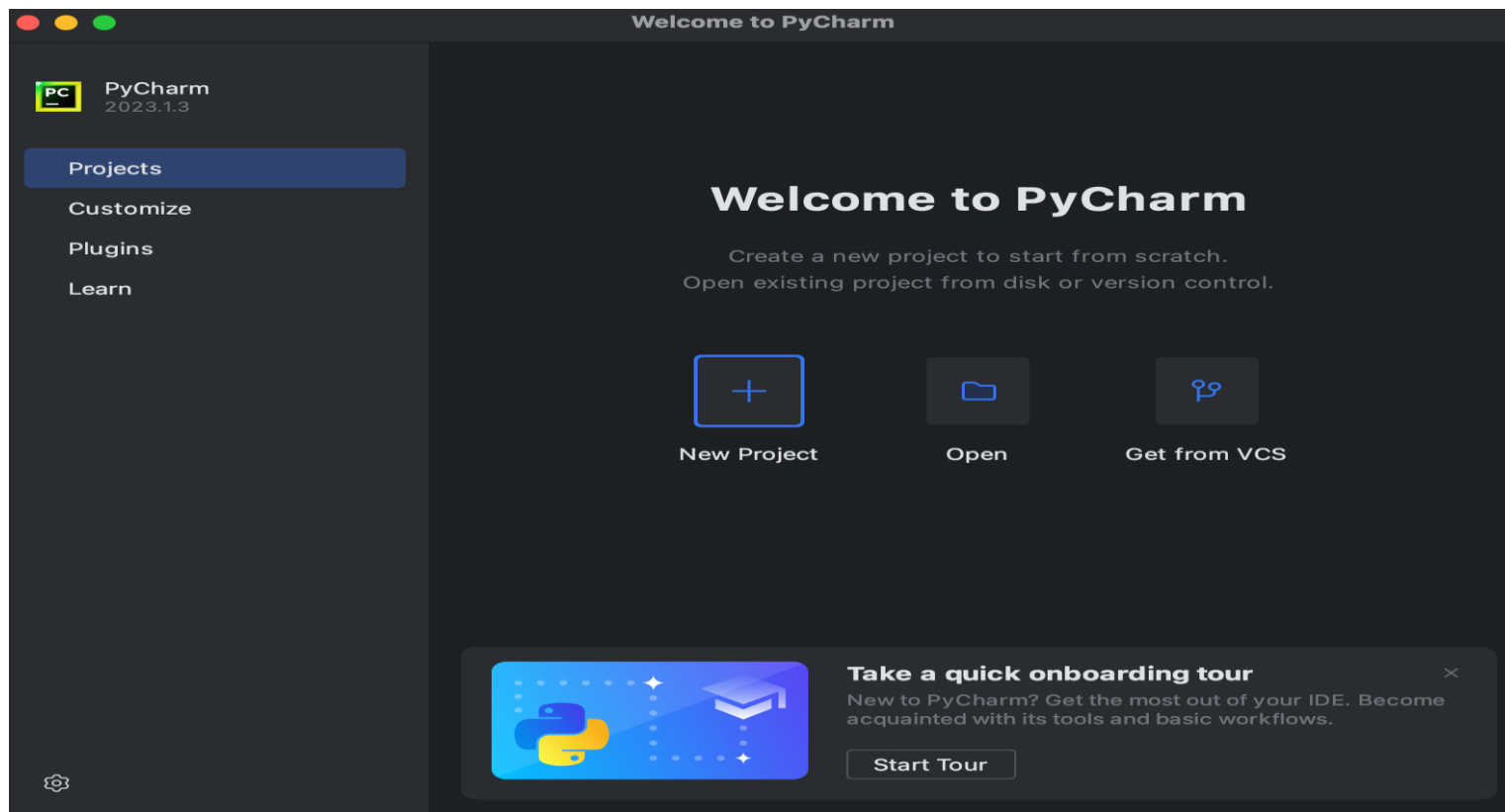
```
C:\WINDOWS\system32\cmd
C:\Users\DELL>python
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020,
23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
for more information.
>>> print("bonjour")
bonjour
>>>
```

- Comme tout **langage informatique**, on peut **enregistrer les instructions** dans un **fichier**, que l'on appelle communément un **script** (ou **programme**) **Python**.
- Un **script Python** est un fichier texte contenant des instructions en langage Python.
- Les **scripts** ont généralement l'extension de fichier **.py** (par exemple, **monscript.py**).
- **Remarque :**
  - ▣ Dans un **script**, tout ce qui suit le caractère **#** est ignoré par **Python** jusqu'à la fin de la ligne et est considéré comme un commentaire.
  - ▣ Le passage d'une instruction à l'autre se fait **par retour à la ligne** ou en insérant un **point-virgule ' ; '**

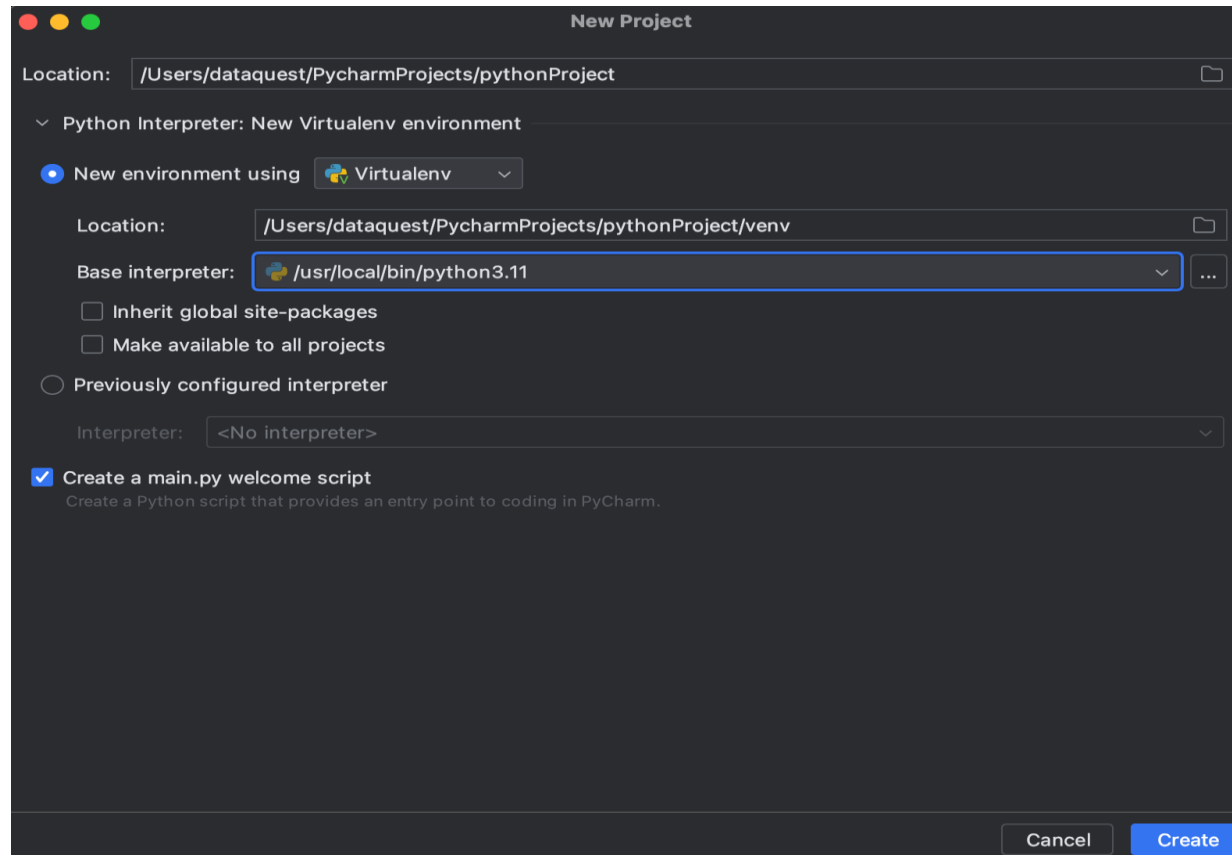


25

## ❑ Exemple de création d'un projet:



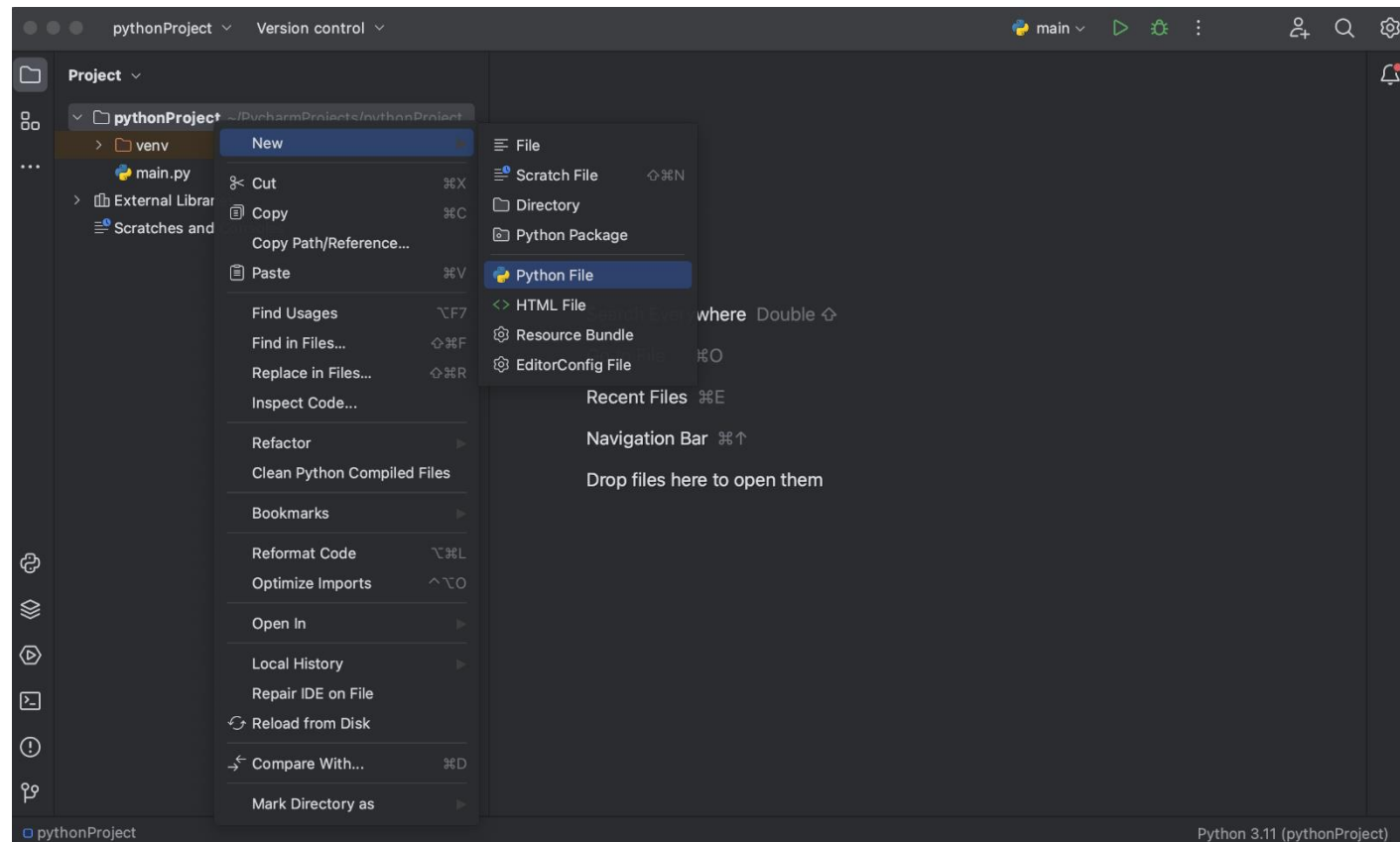
- ❑ On doit Spécifier ensuite l'emplacement du projet, l'environnement et l'interpréteur Python.



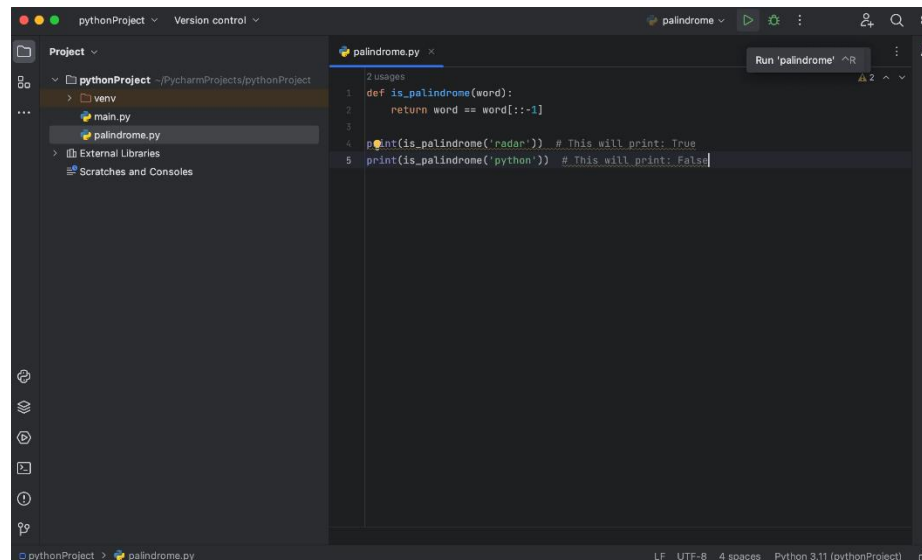
- ▶ Notion d'interpréteur
- ▶ Notion de script
- ▶ **Création de script**
- ▶ Exécution de script

27

## □ Un nouveau script Python



- À partir de la ligne de commande/terminal :
  - Ouvrez un terminal ou une ligne de commande.
  - Utilisez la commande Python suivie du nom du fichier pour exécuter le script.
  - **Exemple:** `>Python monScript.py`
- IDE (Environnement de Développement Intégré) :
  - Si vous utilisez un IDE tel que PyCharm, vous pouvez ouvrir votre script et exécuter le code directement depuis l'interface utilisateur.



- Jupyter Notebook : vous pouvez exécuter une cellule de code en appuyant sur Shift + Enter.