



PROGRAMMER EN PYTHON INSTALLATION ET PRISE EN MAIN

Généralités

Python est un langage de programmation qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées. Il est particulièrement répandu dans le monde scientifique, et possède de nombreuses extensions destinées au calcul numérique. La version conseillée est python 3.6 à la date d'écriture de ce guide. Par ailleurs, pour la rédaction de cette ressource, la suite logicielle utilisée est Anaconda, avec les environnements Spyder ou Jupyterlab Anaconda.

Cette version contient toutes les bibliothèques scientifiques utiles pour Python. Après s'être laissé guider pour l'installation d'Anaconda, ouvrir Anaconda et cliquer sur Spyder ou Jupyterlab.

En ce qui concerne le choix entre ces deux environnements.

Jupyterlab possède une présentation plus conviviale (par exemple, une séparation automatique du programme en cellules indépendantes), et permet une entrée plus simple dans la programmation Python. Mais, certains types de réseaux informatiques de lycées sont incompatibles avec le lancement d'une session de Jupyterlab (une telle session nécessite d'ouvrir un site local, ce que les sécurités de réseau ne permettent pas toujours de faire). D'autre part, les fichiers issus de Jupyterlab sont du type .ipynb et ne sont pas lisibles par les autres environnements. Cela limite les possibilités de diffusion et d'utilisation des ressources extérieures, qui ne sont pas du type .ipynb. Ce n'est pas dommageable si l'on souhaite se limiter à l'utilisation des scripts de cet ensemble de ressources mais cela limitera les possibilités d'évolution.

Spyder possède une présentation un peu moins conviviale (par exemple, la séparation en cellules doit se faire au moyen de #%%) mais il a l'avantage de ne pas nécessiter de site local et il est donc forcément compatible avec les réseaux informatiques de lycées. D'autre part, il fournit des fichiers du type .py qui sont compatibles avec d'autres environnements (Pyzo...) et permet ainsi une diffusion et une utilisation des ressources extérieures plus directes.

Ainsi, selon les contraintes de réseaux ou les volontés individuelles, on pourra utiliser au choix l'un des deux environnements conseillés, Jupyterlab ou Spyder. Dans les deux cas, les fiches et les scripts qui suivent couvrent l'ensemble des attendus des programmes du lycée.

Retrouvez éduscol sur :



Jupyterlab

Pour Jupyterlab, ouvrir une fenêtre de travail « Notebook », qui est divisée en une succession de blocs :

- entrée (In) : pour entrer le programme ;
- sortie (Out) : pour utiliser un programme et voir les résultats qu'il fournit.

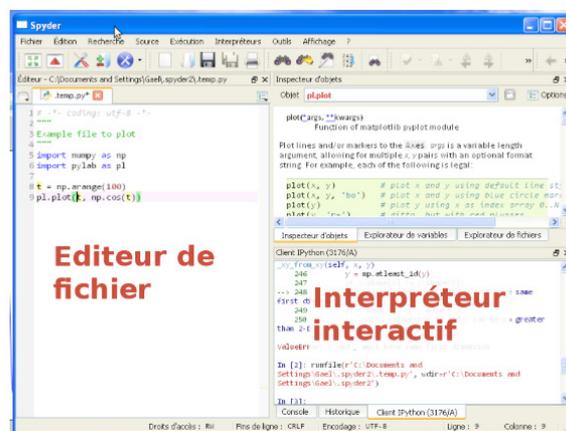
On peut demander à l'interpréteur de travailler successivement sur les différentes cellules (parties) d'un code. Cela permet de visualiser plus facilement l'avancée du programme et de déceler plus facilement les éventuelles erreurs grâce à la flèche (qui fait exécuter le programme contenu dans la cellule).

Spyder

Pour Spyder, l'espace de travail fait apparaître plusieurs blocs, dont deux principaux :

- l'éditeur de fichier sert à rédiger le programme ;
- l'interpréteur interactif permet d'utiliser le programme et de voir les résultats qu'il fournit.

Remarque : le bloc de l'éditeur peut être séparé en cellules indépendantes (on place `%%` en début et en fin de cellule). Cela permet de faire interpréter uniquement la cellule sélectionnée.



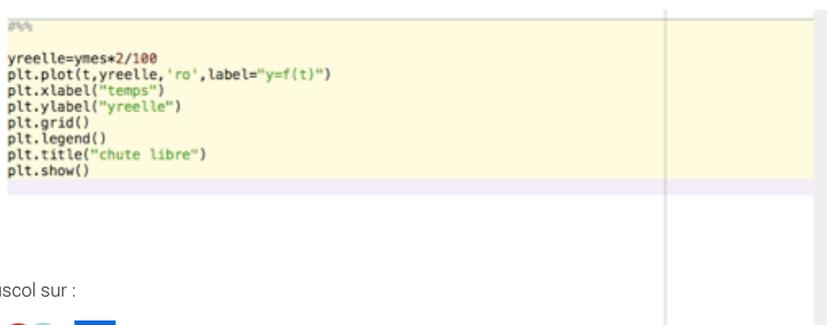
Pour exécuter un script ou une cellule de script, l'utilisateur peut cliquer sur l'un des boutons suivants :



Le premier fait exécuter tout le script.

Le deuxième fait exécuter seulement la cellule sélectionnée qui est jaunie. Cette dernière utilisation est préférable dans un premier temps pour suivre l'avancée du script.

Dans les deux cas, le résultat est visible dans la fenêtre de l'interpréteur, comme l'illustre l'exemple ci-dessous.



Retrouvez éducol sur :

