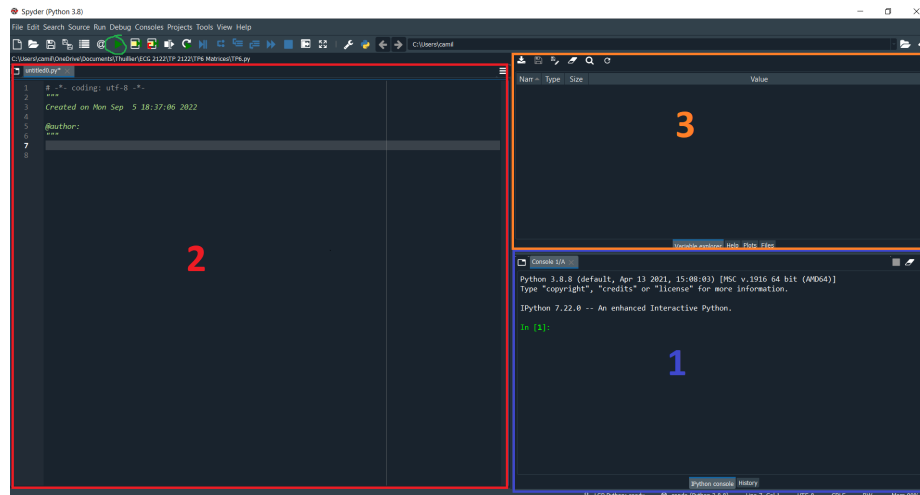


TP 1 : (Ré-)Introduction à Python

1 Présentation de Python

On utilisera au cours de cette année le logiciel Spyder pour accéder à Python.

Dans ce TP on va chercher à s'approprier l'environnement **Python**. Cet environnement se décompose en 3 parties :



Partie 1 : La console (ou interpréteur) est une interface interactive que l'on peut utiliser comme simple calculatrice. L'invite est le symbole `In[]` derrière lequel on peut écrire des calculs ou des instructions. S'il y en a plusieurs, on les sépare par un point-virgule. L'appui sur la touche **Entrée** provoque l'exécution des instructions indiquées sur la ligne.

Partie 2 : L'éditeur de texte est un espace dans lequel on peut écrire des programmes ou des fonctions (dont le texte est appelé code ou script) qui pourront être enregistrés, puis exécutés dans la console. L'éditeur, contrairement à la console, permet d'enregistrer les scripts/programmes écrits pour les utiliser lors d'une autre session.

Partie 3 : Ce dernier espace permet de retrouver la valeur des variables que l'on aura déjà utilisées dans la case **Variable explorer** ainsi que les graphiques que l'on aura tracés dans la case **Plots**.

2 Découverte de la console et commandes de base.

2.1 Calculs sur les nombres réels

- Evaluer dans la console de manière successive, les instructions suivantes :

$$2 + 4 \quad 5 - 3 \quad 4 * 7 \quad 8/2 \quad 5 ** 2$$

En déduire l'interprétation de ces symboles.

- Dans la console, vérifier que rien n'est inscrit devant l'invite de commande `In[...]`. Appuyer alors plusieurs fois sur la touche $\uparrow$ puis sur la touche $\downarrow$. A quoi servent ces touches ?

Exercice 1. Calculer avec Python :

$$A = 7 - 4 \frac{\left(\frac{5}{9} - \frac{1}{3}\right) \left(3 - \frac{1}{2}\right)}{\frac{7}{9} - 3}; \quad B = 2 \times \frac{3-5}{3 \times 4} + 2^3 \times \frac{\frac{7}{9}}{\frac{6}{4}}.$$

2.2 Fonctions prédéfinies

Attention. Les fonctions usuelles ainsi que les constantes usuelles $\pi, e, \dots$ ne sont pas directement accessibles. Il faut importer une *bibliothèque* pour les utiliser.

Une des bibliothèques au programme est la bibliothèque **numpy**. Elle permet d'avoir accès aux fonctions usuelles, aux constantes usuelles ainsi que de manipuler des matrices ce que l'on verra plus tard dans l'année. Dans ce TP nous nous contentons de voir la partie "calculs numériques".

Comme pour toute bibliothèque, il faut commencer par l'importer pour pouvoir l'utiliser. On l'importe via la commande :

```
import numpy as np
```

Cela nous donne ainsi accès à toutes les constantes et fonctions définies dans **numpy**. Pour les appeler, il faudra utiliser le préfixe **np**.

Par exemple, on peut utiliser π via la commande `np.pi`

On a également accès aux fonctions usuelles, à savoir :

- `np.exp` pour l'exponentielle
- `np.log` pour le logarithme népérien
- `np.sqrt` pour la racine carrée
- `np.abs` pour la valeur absolue.
- `np.cos`, `np.sin` et `np.tan` pour les fonctions trigonométriques.

Comme ce sont des fonctions Python, il ne faut pas oublier les `()` au moment de les appliquer, par exemple `np.sqrt(7)`

Traduire en Python l'expression mathématique : $\sqrt{e^3 + 2^4 - \ln(2)}$

Exercice 2. Calculer la valeur absolue et la racine carrée des nombres suivants :

$$A = \frac{2^6}{3} - \frac{5}{2}; \quad B = \frac{131}{\sqrt{2}} + 10^3 - \frac{5}{49}; \quad C = \frac{5 + 2 \cos\left(\frac{\pi}{7}\right)}{3}$$

3 Les variables

Lorsqu'on programme en Python, il sera souvent utile de garder en mémoire certains résultats. Pour cela on utilise des *variables*. Une *variable* est une case mémoire de l'ordinateur à laquelle on a donné un nom.

Pour affecter une valeur à une variable, on écrit son nom suivi du symbole `=` puis de la valeur que l'on veut donner. Lorsqu'une variable est créée, elle est visible dans la fenêtre *Variable explorer*.

- Evaluer dans la console `a`. Noter le message d'erreur obtenu.

- Evaluer maintenant `a=5` puis de nouveau `a`. Comparer et expliquer ce résultat avec le précédent.

- Evaluer `5*a+1`. Noter précisément ce qui est affiché en retour.

- Evaluer `a=5*a+1`. Regarder ce que cela modifie dans Variable explorer. Expliquer le fonctionnement de cette affectation et comparer l'affichage obtenu avec le précédent.

- Prédire, sans l'évaluer, le contenu de la variable `a` à la fin de la séquence suivante :

`a=1, a=np.sqrt(36*a+13), a=a/2, a=np.sqrt(a+4), a=8`

Vérifier alors votre résultat en évaluant cette séquence. Expliquer le résultat obtenu.

- Evaluer la séquence `a=3, b=7, a==b, a=b, a==b`. Noter les résultats obtenus.

- A quoi correspondent les opérateurs `=` et `==` en Python ? On expliquera les résultats précédents.

- Prédire le résultat de la commande `a=5*a+1`.

Exercice 3. On considère trois variables a , b et c et on effectue la suite des affectations ci-dessous :

`In[1] a=1, b=2, c=3`

`In[2] a=2*b+5`

`In[3] c=b, b=a`

`In[1] a=c-b+2`

Compléter le tableau de valeurs des variables a, b, c .

Variables	a	b	c
Affectation 1			
Affectation 2			
Affectation 3			
Affectation 4			

Exercice 4. Proposer une suite d'instruction qui permet d'échanger les valeurs de deux variables **a** et **b**.

4 Editeur de texte

4.1 Première utilisation

Taper dans la console a deux inconvénients : l'enregistrement n'est pas possible, et si plusieurs lignes d'instructions ont été tapées, les modifications ne sont pas aisées. Pour enchaîner plusieurs instruction, l'éditeur est un outil plus approprié.

L'éditeur reconnaît le langage Python : il met en couleur les mots-clef, un décalage de début de ligne appelé *indentation* se fait automatiquement lorsqu'on commence une boucle ou un test, il ferme automatiquement les parenthèses ouvertes...

- Placer vous dans l'éditeur.
- Dans cette partie, taper une première ligne de commande **a=5**. Sur la ligne en dessous, taper la commande **b=8**.
- Dans le menu "Fichier", choisir "Enregistrer sous" afin de sauvegarder ce document dans un dossier TP1, sous le nom *monPremierProg*.
- Pour exécuter votre programme cliquer sur l'icône "Play" (triangle vert entouré dans l'image de début du TP). Cette icône permet à la fois d'enregistrer votre fichier et de l'exécuter.

Remarque : Pensez à sauvegarder très régulièrement vos fichiers.

- Dans la console, évaluer **a** puis **b**. Quel est le contenu de ces variables ?

4.1.1 Affichage

Dans la console, un calcul est automatiquement affiché, et pour afficher une variable nommée **a** (par exemple), il suffit de taper dans la console :

```
In[...] a
```

Mais dans l'éditeur, il faut utiliser la commande **print(...)**

Par exemple, il faudra taper **print(3+4)** pour afficher le résultat de 3+4 ou **print(a)** pour afficher le contenu de la variable **a**.

4.1.2 Commentaires

Pour gagner en clarté, il est recommandé de commenter son script. Notamment, si le script est utilisé dans différentes séances. On peut par exemple expliquer ce que fait un programme ou bien ce que contient une variable,...

Pour insérer des commentaires, il suffit de taper **#** et ce qui suit sur la ligne ne sera pas lu par l'ordinateur (donc pas exécuté)