

Programmation en Python et utilisation de Thonny

1. Présentation de l'éditeur Thonny

Thonny est un **logiciel libre et gratuit** permettant l'écriture et l'exécution de programmes écrits en Python. Il a été créé à destination des élèves et étudiants découvrant la programmation par des professeurs de l'institut de Sciences Informatiques de l'université de Tartu, en Estonie.

Il est téléchargeable gratuitement [ici](#), en version Windows, Mac ou Linux, et contient une version autonome de Python, qui ne perturbe en rien l'ordinateur sur lequel on installe le logiciel.

Présentation de l'interface graphique de Thonny

Comme tout bon *environnement de développement intégré* ou autrement dit **IDE**, Thonny propose plusieurs outils pour aider à la programmation en Python. Nous ne présenterons ici que ceux nécessaires à une initiation à Python.



- La **zone d'exécution**, aussi appelée **Shell** ou **console**, est une zone *interactive* où une instruction Python est exécutée directement après avoir été tapée. Cette zone est souvent utilisée pour *tester une instruction ou série d'instructions*, mais est aussi l'endroit où sera exécuté un **script Python**. Attention, dans le **Shell**, il n'est pas possible de **revenir en arrière et de modifier ce qui a été tapé**. En cas d'erreur, il faudra recommencer !
- La **zone d'édition** ou zone de **scripts**, est une zone dans laquelle on peut taper des séries d'instructions Python, et les sauvegarder sous la forme d'un fichier d'extension `.py`. Ces instructions ne seront pas exécutées tant que l'utilisateur n'aura pas demandé explicitement cette exécution.
- Pour *exécuter un script*, il faudra appuyer, soit sur **la flèche verte** de la barre de menu, soit sur la touche `F5`, soit par l'intermédiaire du menu `Run>Run current script`. Le résultat de l'exécution du script sera affiché dans le **shell**.
- Les zones situées à droite de l'éditeur (fenêtre des variables et assistant d'édition), contiennent des informations qui peuvent être utiles pour analyser un programme qui ne fonctionne pas.

**Mais c'est en anglais**

Oui, l'interface de Thonny est en anglais, oui il n'y a une possibilité pour la passer en français, non je ne la donnerai pas ici, car l'anglais utilisé est très simple, à vous de vous débrouiller !

2. Premiers pas avec Thonny

**Exercice 1 : Prise en main****Énoncé**

1. Créer un dossier `Maths` dans votre dossier personnel.
2. Créer un dossier `Algorithmique` dans le dossier `Maths`.
3. Ouvrir le logiciel Thonny.
4. Créer un script vide.
5. Tapez le code suivant, en **respectant scrupuleusement** ce qui est écrit :

```

1  nom = input("Quel est votre nom ?")
2  age = int(input("Quel est votre age ?"))
3  majeur = age >= 18
4  if majeur :
5      majorite = age - 18
6      message = f"Bonjour {nom}, vous avez {age} ans et êtes majeur depuis {majorite} ans."
7  else :
8      majorite = 18 - age
9      message = f"Bonjour {nom}, vous avez {age} ans et serez majeur dans {majorite} ans."
10 print(message)

```

6. Enregistrer ce fichier sous le nom `AP_Algo_1.py` dans le dossier `Algorithmique`.
7. Exécutez le script et répondez aux questions posées dans le **shell**. Que se passe-t-il ? Recommencez plusieurs fois, en essayant différentes réponses. *Pour les questions suivantes, vous pouvez vous aider de la fenêtre des variables. Si elle n'est pas présente, vous pouvez l'afficher par le menu `View>Variables`.*
8. À quoi sert l'instruction `input` en ligne 1 ? Quel est le type de la variable `nom` ?
9. Quel est le type de la variable `age` ? Pourquoi n'est-elle pas du même type que `nom` ?
10. Quel est le type de la variable `majeur` ? Que cela signifie-t-il ?
11. Quel est le type de la variable `majorite` ? Est-ce toujours le cas ?
12. Quel est le type de la variable `message` ? Qu'a-t-elle de spécial ?

Solution

A venir

**Le calcul avec Python**

Python est capable de calculer avec des nombres entiers (type `int`) et des nombres flottants (type `float`). Les opérations de bases sont gérées ainsi :

- les opérations $(+)$, $(-)$, (\times) et (\div) s'écrivent respectivement `+`, `-`, `*`, `/` ;
- (x^n) s'écrit `x**n` ;
- Le reste de la division euclidienne de (a) par (b) s'obtient par `a%b` (lu « a modulo b ») ;
- le quotient de la division euclidienne de (a) par (b) s'obtient par `a//b`.

? Exercice 2 : Compléter un programme

Enoncé

1. Créer un nouveau script, et enregistrez-le dans le dossier `Algorithmique` sous le nom `AP_Algo_2.py`.
2. Tapez le code suivant dans le script :

```
1  x = 2
2  y = 3
3  z = x+y+x*y
4  z = z**2
5  y = y/2
```

3. Exécutez-le. Qu'obtient-on ?
4. Compléter ce programme afin qu'une variable booléenne `t` teste si `z` est pair.

Solution

A venir !

? Exercice 3 : Se méfier !

Enoncé

On considère l'algorithme suivant en pseudo-code, ainsi que le tableau d'état des variables correspondant:

$$\begin{array}{|c|} \hline A & \leftarrow 12 \\ B & \leftarrow 4 \\ C & \leftarrow \frac{A}{B} \\ D & \leftarrow \frac{A \times 7}{B \times 7} \\ T & \leftarrow C=D \\ \hline \end{array}$$

A	B	C	D

1. Compléter à la main le tableau d'état des variables. Que doit contenir la variable `T` ?
2. Créer un script vide et l'enregistrer dans le dossier `Algorithmique` sous le nom `AP_Algo_3.py`.
3. Coder l'algorithme en Python.
4. Exécutez-le. Le résultat est-il conforme à vos attentes ? Corriger si besoin le code.
5. Quel est le type des variables `C` et `D` ? Comment l'expliquer ?

Solution

A venir !