

1DEV1A – 1DEVL1A – Laboratoires Python**TD 02 – Variables et expressions**

Nous poursuivons notre apprentissage du langage Python en introduisant la notion de variable (permettant de stocker des données à un endroit de la mémoire) et celle d'expression. Nous apprendrons également à lancer Python en mode script et à écrire nos premiers programmes.

Objectifs :

- ▷ déclarer et initialiser des variables
- ▷ déclarer une variable référençant une valeur
- ▷ distinguer et écrire les types (chaîne, nombre entier, liste, nombre flottant)
- ▷ évaluer des expressions
- ▷ lancer un éditeur de code
- ▷ recopier et sauvegarder un script python
- ▷ exécuter un script python en ligne de commande
- ▷ adapter un script python
- ▷ utiliser les fonctions : `float`, `int`, `input`, `print`

Table des matières

1 Les variables en Python	2
2 Les expressions	2
3 Python en mode script	4
4 Vos premiers programmes	7
5 En résumé ...	8

Avertissement

Assurez vous d'avoir bien compris le TD01 *avant* d'aborder celui-ci. Votre professeur de labo peut vous aider si vous n'avez pas compris certaines notions introduites dans le TD01 : *n'hésitez pas à lui demander de l'aide* le cas échéant !

1 Les variables en Python

En Python, une *variable* est un nom symbolique permettant de faire référence à une valeur stockée en mémoire.

```
nombre_jours = 365
message = 'Bienvenue à tous'
notes_etudiants = [10,15,12,18,5,0,15]
```

Notez que le *type* de chacune des valeurs est différent : un entier, une chaîne de caractères, une liste (contenant les entiers 10, 15, etc.).

Pour connaître le type d'un objet référencé par une variable, vous pouvez utiliser l'expression : `type(nom_variable)`.

Exercice 1 Les variables en Python

Lancez Python en mode interactif et exécutez des instructions permettant d'arriver aux résultats demandés (notez l'instruction dans le cadre) :

1. Initialiser une variable `age_client` référençant le nombre 21 :
2. Initialiser une variable `nom_client` référençant la chaîne '`Charlot`' :
3. Initialiser une variable `temperature` référençant le nombre 17.5 :
4. Afficher le type de donnée référencé par `age_client` :
5. Changer la donnée référencée par `age_client` pour référencer à présent une liste contenant les nombres 1, 2 et 3 (dans cet ordre) :
6. Afficher à nouveau le type de donnée référencé par `age_client` (il a changé...) :

C

onstatez qu'une variable peut référencer des valeurs arbitraires. En Python, les valeurs ont un type, mais pas les variables. L'expression `type(variable)` donne le type de la valeur référencée.

2 Les expressions

Avec les variables, les littéraux (valeurs apparaissant telles quelles dans une expression) les opérateurs et les fonctions mises à notre disposition par Python, nous pouvons former des *expressions* de façon à calculer d'autres valeurs.

Exercice 2 Expressions

Déclarez les quatre variables suivantes :

1. `largeur` référençant le nombre 2.8,

2. `longueur` référençant le nombre 34,
3. `texte` référençant la chaîne '`le langage Python`',
4. `ma_liste` référençant une liste contenant les chaines '`A`', '`B`' et '`C`' (dans cet ordre).

(Si vous avez le moindre doute, validez ce que vous avez fait avec votre enseignant ; la suite de l'exercice en dépend.)

Devinez (puis demandez alors à Python pour confirmer) la valeur des expressions suivantes. Certaines d'entre-elles sont erronées : comprenez et indiquez pourquoi. Pour celles qui ne le sont pas, donnez la valeur de l'expression.

Expression	Raison de l'erreur ou valeur de l'expression
<code>longueur * largeur</code>	
<code>largeur**3</code>	
<code>2*longueur + 2*largeur</code>	
<code>(longueur + largeur)/2</code>	
<code>longueur + largeur/2</code>	
<code>longueur + texte</code>	
<code>longueur - len(texte)</code>	
<code>max(longueur,largeur,len(texte))</code>	
<code>ma_liste*2</code>	
<code>ma_liste+2</code>	
<code>texte[0:10]</code>	
<code>texte[0]*10</code>	
<code>largeur//10</code>	
<code>texte//10</code>	

Soyez attentif au rôle de chaque opérateur (aussi en fonction des types en présence) et à l'ordre des opérations (et l'influence des parenthèses).

Assignation

Toute expression valide peut être évaluée et son résultat stocké en mémoire et référencé par une variable. On utilise pour cela l'opérateur `=`, aussi appelé *opérateur d'assignation*. La syntaxe `nom_de_variable = expression` a pour effet de stocker la valeur de l'expression donnée, dans la variable dont le nom est donné ^a.

- a. si une valeur s'y trouvait déjà, celle-ci est remplacée

Exercice 3 Assigner une expression à une variable

En utilisant les variables de l'exercice précédent, exécutez les instructions suivantes. Attention : certains d'entre-elles sont erronées. Inscrivez la valeur de la variable impliquée après exécution de l'instruction dans la colonne de droite, ou la raison pour laquelle l'assignation est erronée sinon.

Attribution	Valeur de la variable (ou cause de l'erreur)
<code>longueur = (5 + 3**10)/3</code>	
<code>longueur = largeur</code>	
<code>5 = longueur</code>	
<code>longueur = longueur + 5</code>	
<code>largeur = largeur/2 - 1</code>	
<code>longueur + largeur = texte</code>	
<code>texte = texte*5</code>	
<code>longueur = texte</code>	
<code>longueur = longueur +1</code>	

3 Python en mode script

Jusqu'à présent, nous avons travaillé avec Python en mode interactif. Vous allez maintenant écrire des scripts : un script est une séquence d'instructions Python, contenues dans un (ou plusieurs) fichier ; ce fichier sera ensuite à faire exécuter par Python.

Les tutoriels suivants vous guident dans l'écriture puis l'exécution de votre premier script. Lisez attentivement tout ce qui s'affiche à l'écran. Il est important de comprendre les messages éventuels ; vos enseignants sont là pour vous y aider.

Tutoriel 1 Écrire un script Python

- Lancez l'éditeur de texte **VS Code** ou **VS Codium**¹, accessibles via la recherche d'application ou via l'icône se trouvant sur le Bureau².



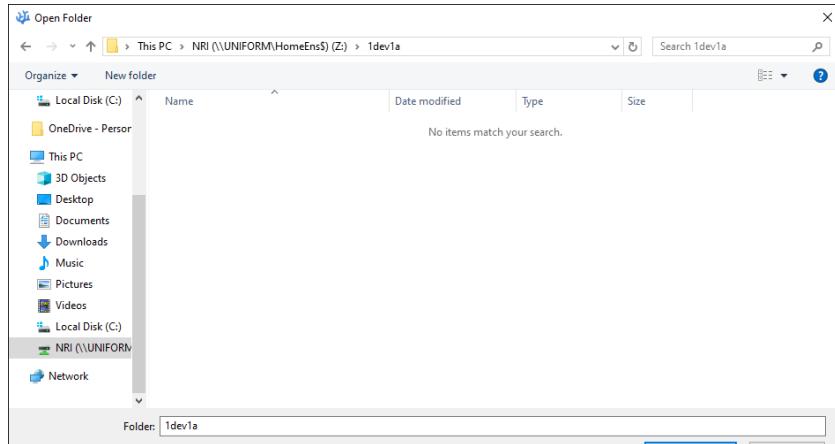
- Utiliser le menu *Open Folder*³ pour ouvrir un répertoire dédié à ce cours. À l'école, nous suggérons d'aller dans le disque Z⁴ et de créer (utilisez le menu New folder) et d'ouvrir un répertoire nommé **1dev1a**.

1. Il n'y a aucune obligation, vous pourriez même utiliser une application aussi simple que le Bloc-Notes (notepad) de Windows. Cependant s'habituer à un éditeur un peu avancé comme VS Code ou VS Codium sera bénéfique pour la suite de votre formation. On suppose dans la suite que vous utilisez cet éditeur.

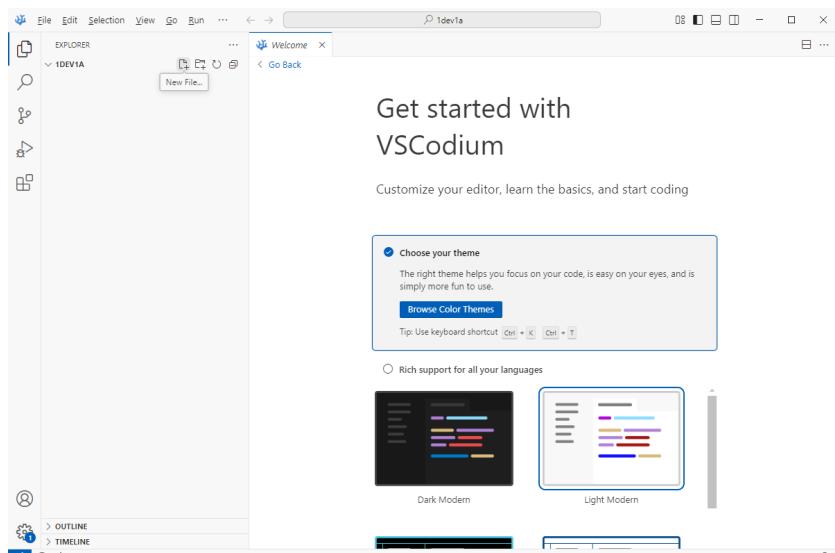
2. Alternativement, utilisez la touche Logo (windows) pour ouvrir le menu et commencez à taper : **vscod** pour trouver rapidement le programme.

3. Le fait d'utiliser *Open Folder* plutôt que *Open File* vous permet d'accéder facilement, depuis l'éditeur, à tous les fichiers de ce répertoire. Pour information : les mots *folder* et *directory* et *répertoire* et *dossier* désignent la même chose.

4. À l'ESI, le disque Z est un disque réseau, auquel vous pourrez accéder de toutes les machines de l'école, contrairement aux fichiers sur le Bureau.



- Créez un nouveau fichier (*file*, en anglais) dans ce répertoire grâce au bouton idoine sur la zone de gauche (il apparaît en y amenant le curseur de la souris, voir screenshot)⁵. Nommez ce fichier `aire_carre.py`. L'*extension* `.py` indique qu'il s'agit d'un fichier Python. Normalement vous obtenez un écran avec, sur la gauche, le nom du fichier nouvellement créé et, sur la droite, une grande zone vide prête à accueillir du texte.



- Inscrivez ensuite le code suivant dans la fenêtre :

```

1 # Mon premier script Python !
2 #
3 # Un programme calculant l'aire du carré dont la
4 # longueur du côté est entrée au clavier par l'utilisateur
5
6 print(" **** Bienvenue ! **** ")
7 x = input("Entrez la valeur du côté du carré : ")
8 x = int(x)
9 print("L'aire du carré vaut :", x**2)

```

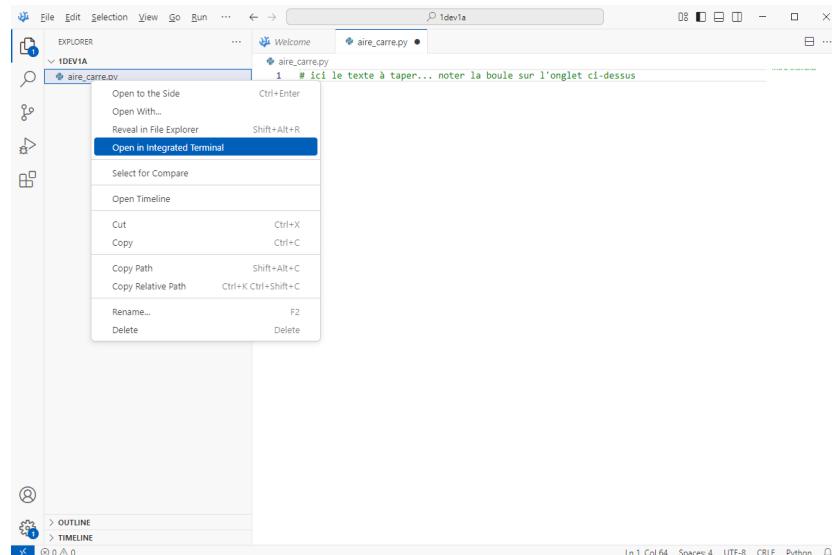
- Sauvez le fichier grâce au raccourci (quasi) universel `Ctrl+S`⁶.

Tutoriel 2 Exécuter le script

Pour exécuter le script, il faut l'exécuter en ligne de commande (ou « Terminal ») :

-
5. Une autre possibilité est d'utiliser le menu New file, ou le raccourci `Ctrl+N`, mais l'expérience est légèrement différente.
 6. Dans cet éditeur, une petite boule située dans l'onglet, en haut de la zone d'édition, indique si le fichier n'est pas sauvé. Vous pouvez aussi activer l'Auto-Save si vous le souhaitez.

- ☒ Dans VS Code⁷, faites un clic-droit sur le nom du fichier (dans la zone de gauche), et choisissez « Open in integrated terminal ».



- ☒ Un panneau s'ouvre en bas de l'écran. Tapez l'incantation suivante :

```
python aire_carre.py
```

- ☒ À ce moment le programme doit⁸ s'exécuter, vous souhaiter la bienvenue et vous inviter à taper un nombre. Faites-le.

Expliquons le fonctionnement du code ci-dessus :

- ▷ Lors de l'exécution du script, Python exécute toutes les instructions du fichier l'une à la suite de l'autre, en commençant par la première.
- ▷ Les lignes 1 à 4 contiennent des *commentaires* : ces lignes sont ignorées par l'interpréteur Python.
- ▷ À la ligne 6 la fonction `print` est utilisée pour afficher une message de bienvenue à l'utilisateur.
- ▷ À la ligne 7, la fonction `input` est utilisée pour demander un nombre à l'utilisateur par une lecture au clavier⁹. Le résultat de la lecture est alors stocké sous la forme d'une chaîne de caractères dans la variable `x`.
- ▷ Comme la fonction `input` renvoie toujours une chaîne de caractères, il est nécessaire de convertir cette chaîne en nombre pour pouvoir faire nos calculs : c'est le but de la ligne 8.
- ▷ Finalement, la ligne 9 affiche la valeur de l'aire (qui est simplement x^2), précédé d'un message explicatif.

Les lignes 7 et 8 du script auraient pu être écrites plus simplement :

```
x = int(input("Entrez la valeur du côté du carré : "))
```

De même que la fonction `int()`, la fonction `float()` permet de convertir une chaîne de caractères en nombre à virgule. L'expression `float(x)` déclenchera une erreur si `x` contient autre chose que des chiffres ou un `,` servant de virgule.

7. Voyez avec votre enseignant pour d'autres techniques permettant d'ouvrir un terminal. Par exemple : depuis l'explorateur de fichiers de Windows, dans le répertoire du fichier, un clic-droit suivi du menu "Git Bash here" ouvrira également un terminal.

8. Et si ce n'est pas le cas, il faut comprendre pourquoi et régler cela.

9. techniquement : lecture sur l'entrée standard. Vous re-verrez ce concept dans d'autres cours.

4 Vos premiers programmes

Exercice 4 Le volume d'un cube

L'objectif est d'écrire un script, s'inspirant du précédent `aire_carre.py`, de façon à calculer le volume ¹⁰ d'un cube de côté `x`, où `x` est un nombre lu au clavier. Pour ce faire, nous vous demandons de suivre les étapes suivantes :

1. Créez un nouveau fichier `XXX.py` ou vous remplacerez `XXX` par un nom évoquant la fonction du script,
2. Copiez-collez le code du fichier `aire_carre.py`,
3. Adaptez les commentaires du début du fichier (commenter correctement son code est important),
4. Adaptez l'appel à la fonction `input` de façon à demander la longueur de l'arête d'un cube à l'utilisateur,
5. Adaptez le `print` final.

Exercice 5 L'aire et le périmètre d'un cercle

Même question qu'à l'exercice précédent, mais on souhaiterait cette fois disposer d'un script permettant de calculer à la fois l'aire et le périmètre d'un cercle de rayon `r` lu au clavier ¹¹.

Le script doit afficher l'aire du cercle, passer à la ligne, afficher le périmètre du cercle. A vous de voir comment adapter correctement les scripts déjà écrits pour arriver à ce résultat.

Exercice 6 L'aire d'un rectangle

Même question qu'à l'exercice précédent, mais on souhaiterait cette fois disposer d'un script permettant de calculer l'aire d'un rectangle de longueur `l` et de largeur `L`, où `l` et `L` sont lues au claviers ¹².

Exercice 7 Temps en secondes

Écrivez un script qui demande un nombre d'heures, un nombre de minutes et un nombre de secondes et qui affiche le nombre de secondes totales.

Par exemple : si l'utilisateur entre 2 heures, 10 minutes et 27 secondes, le programme affiche 7827. En effet 2 heures donnent 7200 secondes, 10 minutes sont 600 secondes auxquelles il faut ajouter les 27 secondes : $7200 + 600 + 27 = 7827$.

10. Rappel : le volume V d'un cube de côté x est donné par $V = x^3$

11. Rappel : l'aire A d'un cercle de rayon r est donnée par $A = \pi r^2$ tandis que le périmètre P est donné par $P = 2\pi r$, vous pouvez utiliser l'approximation $\pi \approx 3.14$

12. Rappel : l'aire d'un rectangle de longueur l et de largeur L est donné par $A = l \cdot L$.

5 En résumé ...

Principaux points de matière du TD

Voici les principaux points abordés lors de ce TD. Vous devez absolument être à l'aise avec ceux-ci avant d'aborder la prochaine séance d'exercice.

1. Déclarer et manipuler des variables en Python, utiliser la fonction `type()` pour connaître leur type.
2. Écrire et manipuler des expressions.
3. Écrire et exécuter un script Python, utiliser les fonctions `print()` et `input()` pour afficher des données à l'écran et lire des données au clavier.