


Activité 1 : Débuter avec le logiciel *Thonny* |

Jusqu'à maintenant, nous avons écrit des programmes Python dans l'environnement de jeu *Py-Rates*. Pour vraiment programmer, on utilise un logiciel dédié à la programmation appelé *Environnement de Développement Intégré* (*IDE* en anglais). Nous allons utiliser par la suite *Thonny*, un IDE spécialement conçu pour débiter en Python.

1. Étape 1. Lancer le logiciel *Thonny*.
 - (a) Cliquer sur l'icône  en bas de l'écran de votre ordinateur.
 - (b) Dans le menu qui s'est ouvert, cliquer dans l'encart de recherche et entrer le mot *thonny* et lancer la recherche.
 - (c) La fiche de *Thonny* apparaît. Cliquer sur "Épingler à la barre des tâches" pour ajouter un raccourci à la barre des tâches.
 - (d) Cliquer ensuite sur "Ouvrir".
2. Étape 2. Paramétrer l'affichage de *Thonny*.
 - (a) Mettre la fenêtre en plein écran.
 - (b) Afficher l'inspecteur de variables (dans le menu `Affichage >> Variables`).
3. Étape 3. Découvrir l'organisation spatiale de *Thonny*.
 La fenêtre de *Thonny* est divisée en trois parties principales :
 - En haut à gauche, un cadre réservé à l'écriture de véritables programmes.
 - En bas à gauche, la console Python réservée l'exécution d'instructions simples.
 - À droite l'inspecteur de variables et l'assistant de programmation.
4. Étape 4. Utiliser la console Python.
 Dans la console, on peut voir l'ensemble de caractères `>>>` appelé *invite de commande* (*prompt* en anglais). Le prompt indique au programmeur ou à la programmeuse qu'il ou elle peut écrire une instruction en Python qui sera ensuite exécutée par l'ordinateur.
 - (a) Dans la console, cliquer juste à droite du prompt.
 - (b) Entrer l'expression Python `1 + 3`.
 - (c) Appuyer sur la touche `Entrée` pour lancer l'exécution.
 - (d) Compléter ci-dessous le résultat obtenu.

```
>>> 1 + 3
...
>>>
```

Le prompt a réapparu en dessous du résultat indiquant qu'une nouvelle instruction peut être entrée.

- (e) Exécuter dans la console les instructions suivantes et ajouter ci-dessous les résultats obtenus.

```
>>> 2 * 2 * 2
...
>>> 2 ** 3
...
>>> (1 + 6)/2
...
```

Que signifie `2 ** 3`? _____

Activité 2 : Types et valeurs de base en Python avec *Thonny* |

Le langage Python offre quatre types de valeur de base résumés dans le tableau ci-dessous.

Type	Permet de représenter en Python ...	Exemples
<code>int</code>	... les nombres entiers positifs et négatifs.	<code>-1</code> , <code>3</code> , <code>321</code>
<code>float</code>	... les nombres décimaux.	<code>1.5</code> , <code>-0.1</code> , <code>2.0</code>
<code>str</code>	... du texte (chaînes de caractères).	<code>'Bonjour !'</code> , <code>"vert"</code>
<code>bool</code>	... les booléens (Vrai et Faux).	<code>True</code> , <code>False</code>

Remarques.

- Attention, le séparateur décimal en Python n'est pas la virgule mais le point !
- Une valeur textuelle, appelée en informatique *chaîne de caractères*, est bordée par des guillemets simples ou doubles.
- les mots-clés Python représentant Vrai (`True`) et Faux (`False`) commencent par une majuscule.

En procédant comme dans l'activité 3., évaluer chacune des expressions Python suivantes dans la console Python. Compléter au fur et à mesure le tableau ci-dessous en indiquant la valeur obtenue ainsi que son type.

N°	Expression Python	Valeur	Type
1	<code>2 + 5</code>		
2	<code>2 * 1.5</code>		
3	<code>30 / 5</code>		
4	<code>2 / 3</code>		
5	<code>2 ** 8</code>		
6	<code>'Bon' + 'jour'</code>		
7	<code>'hello' + 'hello' + 'hello'</code>		
8	<code>3 * 'hello'</code>		
9	<code>3 * 5</code>		
10	<code>3 * '5'</code>		
11	<code>1 == 2</code>		
12	<code>3 < 5</code>		

Remarques.

- L'expression `30 / 5` donne une valeur du type _____ alors que la division tombe juste.
- Dans les expressions 6, 7, 8 et 10, les opérateurs `+` et `*` ont un comportement particulier sur les valeurs de type `str` : on parle de *concaténation*.
- Les expressions 11 et 12 sont des tests.
- On peut interpréter l'expression `1 == 2` comme une question posée à l'ordinateur : « Est-ce que le nombre 1 est égal au nombre 2? ». L'ordinateur répond évidemment _____ ! Noter que l'opérateur de comparaison d'égalité est un double égal !
- De même, proposer une interprétation pour l'expression `3 < 5`.
« Est-ce que _____
_____ ». L'ordinateur répond évidemment _____ !