

TP 1 – PRISE EN MAIN DE PYTHON

Préambule :

1. Ouvrir une session sur l'ordinateur en entrant Identifiant / Mot de passe ;
2. Lancer Spyder en double-cliquant sur l'icône du bureau ;

Objectifs de ce premier TP :

- Prendre en main l'environnement de travail Python que vous utiliserez pendant vos deux années de prépa ;
- Découvrir le langage Python en mode interactif;
- Découvrir le langage Python en mode exécution de fichier ;
- Comprendre les types de variables utilisés en Python.

Exercice 1 : évaluation interactive d'expressions

Dans la console Python (*IPython Console*) en bas à droite de Spyder, écrire les commandes suivantes puis valider par entrée. Ecrire le résultat renvoyé dans la colonne droite du tableau.

»2	
»2.5	
»2,5	
»2+3	
»2<3	
»2==3	
»2=3	
»2==(10/5)	
»10/3	
»10.0/3	
»float(10)/3	
»toto	
»"toto"	
»"toto"2	
»"toto"+2	
»"toto"+"2"	
»"t"<"T"	
»"toto"<"Truc"	
»false	
»False	
»True and False	
»not True	
»(2<3) and (3<4)	
»(2<3) and False	
»"toto" and (2<3)	
»("t"<"T") and (2<3)	

Exercice 2 : les opérateurs

On continue de dialoguer avec l'interpréteur Python : tapez les instructions ci-dessous et écrivez les résultats obtenus (les réponses de l'interpréteur). Essayez de prévoir la réponse et si elle est différente, de comprendre pourquoi la réponse qui vous paraît logique n'a pas été celle de l'ordinateur).

» 20 + 1	
» 20/3	
» 20//3	
» 20% 3	
» 5.45*10	
» 2**4	
» (3+2)*5	
» 3+2*5	
» "bonjour"	
» 'bonjour'	
» "il fait " + "beau"	
» "il fait " + "beau" + " et chaud"	
» "bonjour"*5	

Exercice 3 : les premiers messages d'erreur

Certaines instructions ne sont pas acceptées par l'interpréteur python et provoquent l'affichage d'un message d'erreur. Il faut lire attentivement tous les messages d'erreur (même s'ils sont en anglais) car ces messages nous apprennent comment fonctionne le langage et nous aident à corriger nos instructions et nos programmes. Tapez les instructions suivantes, observez les erreurs obtenues et expliquez l'erreur dans le champ de réponse :

» 20/0	
» 20@3	
» "bonjour"/3	
» "bonjour"+ 5	
» (3+2))*5	
» (3+2*5	

Exercice 4 : les types de données

Pour connaître le type d'une donnée, on peut utiliser le mot `type` :

» type(3)	
» type(1.5)	
» type(3.0)	
» type("bonjour")	
» type("3")	
» type(3+1.5)	
» type(2==b)	

Exercice 5 : opérateurs et types de données

Sur quels types de données peut-on utiliser les opérateurs `+`, `*`, `/`, `//`, `%`, `**` ?

Quel est le type du résultat ? Répondre en complétant ci-dessous. Vous pouvez faire des tests en tapant d'autres instructions dans l'interpréteur python.

(Exemple : `3+7.1` est accepté par l'interpréteur python, donc l'opérateur `+` s'applique à un entier et à un flottant, et le résultat est de type flottant.)

Opérateur	types de données utilisables	type du résultat
+		
*		
/		
//		
%		
**		

Exercice 6 : le transtypage

On peut parfois transformer une donnée d'un certain type, en une donnée d'un autre type. Cela s'appelle le transtypage (ou cast en anglais). Compléter le tableau suivant :

Donnée de départ	de quel type est la donnée de départ	instruction à taper	type du résultat obtenu
3.0		<code>int(3.0)</code>	
3.5		<code>int(3.5)</code>	
3		<code>float(3)</code>	
4		<code>str(4)</code>	
"3"		<code>int("3")</code>	
"3.5"		<code>float("3.5")</code>	
"3.5"		<code>int("3.5")</code>	
"bonjour"		<code>int("bonjour")</code>	

Exercice 7 : variables et affectation

Taper les instructions suivantes dans l'ordre indiqué et compléter le tableau.

instruction	Quelle est la variable ?	Quelle est la donnée (ou résultat de l'évaluation) ? De quel type est cette donnée ?
<code>age = 20</code>		
<code>age = 30</code>		
<code>prenom = "Perceval"</code>		
<code>taille_m = 1.75</code>		
<code>taille_cm = 100*taille_m</code>		
<code>age = age + 3</code>		
<code>phrase1 = "bonjour " + prenom</code>		
<code>phrase1 = phrase1 + " !"</code>		
<code>phrase2 = prenom + " a " + str(age) + " ans."</code>		

Pour afficher le contenu d'une variable dans l'interpréteur python, il suffit de taper le nom de cette variable. Taper les instructions suivantes et compléter :

instruction	Quelle valeur contient la variable ?
<code>age</code>	
<code>prenom</code>	
<code>prenom = "Lancelot"</code>	
<code>prenom</code>	
<code>age=40</code>	
<code>age</code>	
<code>age=22</code>	
<code>age+=1</code>	
<code>age</code>	
<code>phrase1</code>	
<code>phrase2</code>	
<code>profession</code>	
<code>profession="etudiant(e)"</code>	
<code>profession</code>	

Exercice 8 : la fonction print()

Exécuter les instructions suivantes dans une session python interactive et noter les réponses données par python.

<pre>x=3 print(x) y=8 print(x+y) z=x+y print(z) z=2 print(z) res=x+y*z print(res) res=x-y**z print(res) print(res/x)</pre>	<pre>x=3 y=8 z=x+y print(z) print("resultat=z") print("resultat="+z) print("resultat="+str(z)) print("resultat=",z) print("resultat=%d" % z) print("%d+%d=%d"%(x,y,z)) resultat=2.5 print("resultat=%d"% resultat) print("resultat=%f"% resultat)</pre>
--	---

Exercice 9 : introduction à la saisie interactive

La fonction `input` est une fonction d'entrée : Python affiche une question à l'attention de l'utilisateur et attend que l'utilisateur tape sur des touches du clavier en réponse à cette question. L'utilisateur doit terminer sa saisie par la touche "Entrée" pour que le programme continue son exécution. La donnée saisie est de type chaîne de caractères.

Dans l'interpréteur, tapez les instructions suivantes :

instruction	commentaire
<code>print("bonjour")</code>	
<code>input "bonjour"</code>	
<code>input("bonjour")</code>	
<code>input()</code>	
<code>input("entrez votre age:")</code>	
<code>age=input("entrez à votre age:")</code>	
<code>print(age)</code>	
<code>type(age)</code>	
<code>age=age+1</code>	
<code>print("votre age est %s" %age)</code>	
<code>print("votre age est %d" %age)</code>	
<code>int(age)</code>	
<code>print("votre age est %d" %age)</code>	
<code>age = age + 1</code>	
<code>age</code>	

Exercice 10 : les booléens

Le type booléen ou `bool` est un type de données. Un booléen ne peut prendre que 2 valeurs : vrai (`True`) ou faux (`False`). Les majuscules sont importantes. Les opérateurs qui s'appliquent aux booléens s'appellent opérateurs logiques. Ceux sont : `not`, `and`, `or`. D'autres opérateurs s'appliquent à des données numériques ou chaînes de caractères et produisent un résultat booléen : les opérateurs de comparaison.

Taper les instructions suivantes dans l'interpréteur et compléter le tableau :

instruction	résultat obtenu + commentaire ou explication
<code>1==1</code>	
<code>1=1</code>	
<code>1!=3</code>	
<code>1==1.0</code>	
<code>(3+7)==(5*2)</code>	
<code>(1==1) and (2<2)</code>	
<code>(1==1) and (2<=2)</code>	
<code>(2==15) or (3>2)</code>	
<code>1<4<15</code>	
<code>valeur = (1==1)</code>	
<code>type(valeur)</code>	
<code>not(valeur)</code>	
<code>True and False</code>	
<code>True or False</code>	
<code>not(False)</code>	