

Python : TD1

Exercice 1 Essayez de prédire le résultat de chacune des instructions suivantes, puis vérifiez-le dans Python :

```
1 s >>> (1+2)**3
2 s >>> "Da" * 4
3 s >>> "Da" + 3
4 s >>> ("Pa"+"La") * 2 - ("Da"*4) / 2 -5/2
5 s >>> 5//2
6 s >>> 5\%2
7 s >>> str(4) * int("3")
8 s >>> int("3") + float("3.2")
9 s >>> str(3) * float("3.2")
10 s >>> str(3/4) * 2
```

Exercice 2 Ecrire un programme, qui affiche 50 fois " Je dois ranger mon bureau " à l'aide de l'instruction for.

Exercice 3 Ecrire un programme qui demande de saisir 2 valeurs et qui affiche la plus petite des 2 valeurs.

Exercice 4 Devinez et vérifiez les résultats de :

```
1 s>>> s="Rayon_X"
2 s>>> len(s)
3 s>>> s[1:4]
4 s>>> s[-2:]
5 s>>> s[:3]
6 s>>> s[::2]
7 s>>> s[::-1]
```

Exercice 5 Comptez le nombre de voyelle dans "Python est un langage très sympa"

Exercice 6 Demander 2 nombres, que l'on affectera aux variables a et b et afficher la phrase " la moyenne de a et de b est m " dans laquelle a, b et m sont remplacés par leur valeur. m sera donné avec 2 chiffres après la virgule.

Exercice 7 Ecrire un programme qui affiche les nombres de 2 en 2 jusqu'à 100 avec un for puis avec un while.

Exercice 8 Ecrire un programme qui affiche les tables de multiplications de 2 à 9. (Utiliser 2 boucles imbriquées)

Exercice 9 Ecrivez un programme qui calcule les 50 premiers termes de la table de multiplication par 13, mais n'affiche que ceux qui sont des multiples de 7.

Exercice 10 Recherche du plus petit élément d'une liste.

Exercice 11 Le problème de la machine qui « rend la monnaie ».

Ecrivez un programme qui, pour une somme donnée en Euros, imprime le nombre (minimal) de billets et de pièces nécessaires pour la composer. Exemple : $297 = 1 \times 200 + 0 \times 100 + 1 \times 50 + 2 \times 20 + 0 \times 10 + 1 \times 5 + 1 \times 2 + 0 \times 1$

Exercice 12 A l'aide d'une boucle for, écrivez un programme qui affiche la suite de symboles suivante :

```
*
**
***
****
*****
*****
```

Exercice 13 Implémentez le tri à bulles. Lors du tri à bulles, à chaque itération de la boucle principale, les paires d'éléments voisins sont successivement comparées et interverties si besoin. La première itération de la boucle principale envoie le plus grand élément à la fin de la liste, la deuxième itération envoie le deuxième plus grand en avant dernière position etc...