

String

Exercice 1 : Caractères spéciaux

Taper dans la console les instructions suivantes. Justifier les résultats.

- `print(len('Hello world'))`
- `print('a\nb')`
- `print('c\td')`
- `print('e\bff')`

Exercice 2 : Tracé de triangles

Q1: Ecrire un programme qui affiche à l'écran un triangle de 20 lignes remplies avec le caractère * comme ci-dessous :

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

Exercice 3 : Chaîne de caractères

Soit une chaîne de caractère : `ch='bonjour à tous'`

Ecrire un programme qui :

- Affiche la longueur de la chaîne.
- Remplace toutes les occurrences de la lettre o par la lettre a.
- Crée la liste de tous les mots de cette chaîne.

Fonctions

Exercice 4 : Analyse statistique

La fonction `randint` du module `random` permet de générer un nombre entier aléatoire.

- Importer la fonction `randint` en mémoire vive.
- Consulter l'aide de cette fonction

Q1. Quelle instruction permet de reproduire le tirage aléatoire d'un lancer de dé ?

On souhaite un programme qui réalise 30 000 lancers de dé, et qui compte le nombre d'occurrences de chacun des résultats. Il affichera la fréquence statistique de chaque résultat.

- Ecrire ce programme.
- Q2.** L'affichage est-il conforme à la théorie ?
- Faire de même avec cette fois un lancer de deux dés.

Exercice 5 : Suivi de compte bancaire

L'objectif de cet exercice est un programme de suivi de solde bancaire. Ce programme s'appuie sur trois fonctions :

- fonction « solde » : affiche le solde du compte
- fonction « debit(somme à retirer) » : retire la somme du compte, et affiche le solde modifié
- fonction « credit(somme à ajouter) » : ajoute la somme au compte, et affiche le solde modifié

Le corps du programme est décrit ci-dessous :

```

1 Solde=1000
2 n='init'
3 while n!=0:
4     n=int(input('faite votre choix : '))
5     if n==1:
6         solde()
7     elif n==2:
8         debit(int(input('quel débit ?')))
9     elif n==3:
10        credit(int(input('quel crédit ?')))
11    elif n==0:
12        print('fin de programme')
13    else:
14        print('choix non valide')

```

Partie 1 : Compréhension du programme

- Q 1:** Quelles sont les variables utilisées par ce programme ?
Q 2: Décrire les opérations réalisées à la ligne 8.
Q 3: Quelle est l'utilité de la structure conditionnelle ici ?
Q 4: Quelle est l'utilité de la structure « while » ?
Q 5: Quelle est la différence entre « Solde » et « solde » ?

Partie 2 : Finalisation

- Q 6:** Rédiger les fonctions « solde », « credit », « debit ».
Q 7: Vérifier le bon fonctionnement en exécutant le programme.

Exercice 6 : Moyenne et variance

Le fichier notes.txt contient les notes obtenues par les élèves de PCSI à un DS.

Pour charger le fichier sous forme d'une liste, utiliser le programme lecture.py

- Q1:** Ecrire le programme qui crée une liste L d'entiers, correspondant aux notes contenues dans le fichier.

```
| print(L)
```

Il est possible d'afficher un graphe des notes (éléments de la liste) en fonction de l'index dans la liste :

```
| plot(L)
```

La courbe représentée n'a aucune signification : les grandeurs tracées ne sont pas continues. On préférera :

```
| plot(L, 'o')
```

Il est aussi possible de tracer l'histogramme associé à la répartition de ces notes :

```
| hist(L, range(20))
```

On cherche à extraire les données statistiques de cette liste.

- Q2:** Spécifier et écrire une fonction moyenne qui, à une liste d'entiers, renvoie la moyenne de ces entiers. Vérifier cette fonction sur la liste L.
Q3: Spécifier et écrire une fonction ecart_type qui, à une liste d'entiers, renvoie l'écart type de ces entiers. Vérifier cette fonction sur la liste L.