

1ere NSI – Lycée Bourg Chevreau – 2021 - 2022

Exercice 1 : Utilisation de variables / Calcul / Affichage

On souhaite coder un programme qui demande la valeur de la résistance $R_{(\Omega)}$ d'un résistor et l'intensité $I_{(A)}$ puis, connaissant la loi d'Ohm $\Rightarrow U_{(V)} = R_{(\Omega)} \times I_{(A)}$, calcule et affiche la tension $U_{(V)}$ aux bornes du résistor. (R : int / I : float)

```
Console x
>>> %Run 'S01 - Ex0 - U-I-R.py' 220
Quelle est la valeur de la résistance R(Ω) ? ->220
Quelle est la valeur de l'intensité I(A) ? ->0.045
La tension est de U = 9.90 V.
```

Exercice 2 : Variables / Calcul / Affichage

Coder un programme qui :

- demande 5 notes comprises entre [0 et 20] (float),
 - calcule au fur et à mesure la somme des notes
 - divise le total des notes par 5
- pour afficher alors la moyenne des 5 notes.

```
Console x
>>> %Run 'S01 - Ex1-0 - moyenne.py' 220
Entrez les 5 notes pour lesquelles il faut calculer la moyenne ...
Note n°0-> 10.5
Note n°1-> 12.6
Note n°2-> 8.5
Note n°3-> 14.8
Note n°4-> 9.7
La moyenne est de 11.22 / 20
```

Exercice 3 : Variables / Calcul / Affichage

Coder un programme qui :

- demande le nombre de valeurs à saisir, lequel doit être compris entre 4 et 10 sans quoi on annule la saisie et on redemande le nombre de valeurs à saisir,
- effectue la saisie des valeurs,
- affiche la moyenne, la valeur la plus basse et la valeur la plus haute.

```
Console x
>>> %Run 'S01 - Ex1-0 - moyenne.py' 220
Entrez les 5 notes pour lesquelles il faut calculer la moyenne ...
Note n°0-> 10.5
Note n°1-> 12.6
Note n°2-> 8.5
Note n°3-> 14.8
Note n°4-> 9.7
La moyenne est de 11.22 / 20
valeur n°s-> 14
La valeur minimale est 8.0
La valeur maximale est 14.0
La moyenne est 11.0
```

Exercice 4 : Utilisation de listes

Reprendre l'exercice 3 mais en utilisant une liste pour stocker les valeurs au fur et à mesure de la saisie pour afficher les valeurs en fin de programme.

```
Console x
>>> %Run 'S01 - Ex1-1 - mini-maxi-moy-liste.py' 220
Combien voulez-vous saisir de notes ? (entre 4 et 10) -> 5
Valeur n°0-> 12
Valeur n°1-> 18
Valeur n°2-> 8
Valeur n°3-> 14
Valeur n°4-> 11
Les notes sont : [12.0, 18.0, 8.0, 14.0, 11.0]
La valeur minimale est 8.0
La valeur maximale est 18.0
La moyenne est 12.6
```

Exercice 5 : Liste et codage binaire

Coder un programme qui :

- demande un nombre entier positif à l'utilisateur,
 - utilise une valeur qui va augmenter selon les puissances de 2 $\Rightarrow 1, 2, 4, 8, 16, \dots$
 - compare cette valeur avec le nombre pour déterminer s'il faut ajouter un '1' ou un '0' dans une liste de bits (comparaison avec l'opérateur : &),
 - en fin de calculs, renverse la liste de bits
- [vous pouvez utiliser la fonction : `ma_liste.reverse()`]

- affiche la valeur du nombre saisi en binaire (affiche la liste de bits)

```
Console x
>>> %Run 'S01 - Ex5 - int - binaire.py' 220
Nombre entier positif : 152
Nombre en binaire : [1, 0, 0, 1, 1, 0, 0, 0]
>>> bin(152) # vérification
'0b10011000'
```

Exercice 6 : utilisation des listes

Coder un programme Python qui

- place dans une liste, 15 valeurs aléatoires comprises entre 0 et 100,

puis affiche la liste des valeurs

- affiche la valeur minimale et la valeur maximale dans la liste ainsi que leur index (position)
- remplace dans la liste la valeur minimale par 'x' et la valeur maximale par "xx" puis affiche à nouveau la liste.

```
Console x
>>> %Run 'S01 - Ex6 - random-liste.py' 220
Valeurs avec les mini et maxi : [63, 23, 41, 32, 58, 52, 86, 33, 22, 55, 1, 24, 98, 84, 76]
Valeur mini : 1 à l'index 10
Valeur maxi : 98 à l'index 12
Valeurs sans les mini et maxi : [63, 23, 41, 32, 58, 52, 86, 33, 22, 55, 'x', 24, 'xx', 84, 76]
```

Pour information :

```
from random import randint
n = randint(0,100) # génère un nombre aléatoire entier
```