

## CH4 Exercices chaînes et listes

### **Exercice 1 :**

Donner le code ASCII des deux chaînes de caractères Python ci-dessous :

- a. « bonjour tout le monde »
- b. « programmer en Python »

### **Exercice 2 :**

L'objectif de cet exercice est d'étudier la conversion de format d'un fichier texte et l'impact sur la taille et sur le contenu du fichier.

1. Créer un fichier Word (ou openoffice) dans lequel vous tapez les mots « le petit ». Enregistrer le fichier au format .docx et noter la taille du fichier obtenu.
2. Enregistrer ce même fichier au format texte brut (.txt) en sélectionnant autre codage, UTF-8. Comparer la taille des deux fichiers. Comment expliquer cette différence ?
3. Copier-coller un texte contenant des caractères spéciaux dans un fichier Word. L'enregistrer en .docx et en .txt et comparer les tailles de fichiers.
4. Enregistrer le fichier précédent en choisissant le codage ASCII. Fermer et ouvrir ce fichier. Que s'est-il produit ?

### **Exercice 3**

Soit la liste : `ma_liste=[1,2,3,4,5,6,7,8,9]`

- a) Supprimer le deuxième élément de la liste
- b) Afficher le troisième élément de la nouvelle liste
- c) Insérer un 2 en quatrième position de la nouvelle liste
- d) Soustraire 3 à l'avant-dernier élément
- e) Ajouter 10 à la fin de la liste
- f) Afficher la nouvelle liste
- g) Trier la nouvelle liste puis l'afficher

### **Exercice 4**

Voici deux scripts.

- a. Quel est l'avantage du script 2 par rapport au script 1 ?
- b. Le script 2 fonctionne-t-il si on supprime sa première ligne ? A quoi sert cette première ligne ?
- c. Sur le même modèle que le script 2, créer la liste [1 , 3 , 6 , 10 , 15]

```

1. a = 2
2. V = [a]
3. b = a**2
4. V.append(b)
5. c = b**3
6. X = V + [c]
7. print(X)

```

```

1. V = []
2. a = 2
3. V.append(a)
4. a = a**2
5. V.append(a)
6. a = a**3
7. V.append(a)
8. print(V)

```

### Exercice 5

Écrire un programme qui demande à l'utilisateur un numéro de mois (de 1 à 12), puis affiche : « Vous avez sélectionné le mois de : xxxxxx » (avec xxxxxx janvier si l'utilisateur a choisi 1, août si l'utilisateur a choisi 8, etc...).

### Exercice 6

1. Créer par compréhension les listes suivantes :

a. Les multiples de 5 jusqu'à 60

b.  $X = [0.1, 0.2, \dots, 2]$

c.  $Y = \left[ \frac{1}{0.1}, \frac{1}{0.2}, \dots, \frac{1}{2} \right]$

2. a. Quelle est la valeur de couples à la fin de l'exécution du programme suivant ?

```

lettres = ['a', 'b', 'c']
nombres = [1, 5]
couples = [(c, n) for c in lettres for n in nombres]

```

b. Proposer un programme qui permet de construire couples en utilisant des boucles bornées.

### Exercice 7

Écrire un algorithme de permutation en utilisant les listes. L'utilisateur entre une liste, et le programme crée une nouvelle liste où la première valeur passe à la dernière place.

### Exercice 8

Voici une fonction mystere.

On appelle cette fonction avec l'instruction `mystere([2, 5, 6, 8], [1, 4, 7, 8, 9])`. Quel est le résultat renvoyé ? Trouver le résultat avant de tester sur la machine !

```

def mystere(liste1, liste2):
    liste=[]
    i, j = 0, 0
    while (i < len(liste1)) and (j < len(liste2)):
        if liste1[i] < liste2[j]:
            liste.append(liste1[i])
            i = i + 1
        else:
            liste.append(liste2[j])
            j = j + 1
    return liste

```