

Université Sultan Moulay Slimane Faculté Polydisciplinaire Béni Mellal

Département INFORMATIQUE (MIP)

Filière : Science de données et sécurité des systèmes d'information

Compte Rendu de TP01 – Python pour la science des données

Date : 4 février 2024

Réaliser par : MAFTOUH Omar

Introduction :

L'objectif du TP01 est de renforcer ce que nous avons appris pendant les cours. On va se concentrer sur des idées comme les boucles (quand on répète des actions), les façons de trier des choses, comment travailler avec des listes (comme une liste de courses), et comment utiliser des dictionnaires (une sorte de liste spéciale). On va pratiquer tout ça pour bien comprendre comment ça marche !

Exercice 01 : Tri par insertion et Nombres Premiers

Créez un programme Python pour classer une liste d'entiers dans l'ordre croissant en utilisant le tri par insertion. Ensuite, identifiez et retirez les nombres premiers de la liste d'origine (L) pour les placer dans une nouvelle liste (P). Cet exercice permet de pratiquer le tri d'une liste et la détection des nombres premiers en Python.

```
# Initialisation deux listes vides L & P
L = []
P = []

# Proposer a l'utilisateur de saisir le nombre d'elements qu'il doit etre dans
la list
n = int(input("Saisir le nombre d'éléments :"))

# Remplir la liste L par nombre d'element n
for i in range(1 , n+1) :
    element = int(input(f"Saisir l'elemnt nombre {i} : "))
    # La remplissage de la liste L par éléments
    L.append(element)
print(f"Votre Liste est : {L[:]}")

# Le trie par insertion sur la liste L
for i in range(1 , len(L)) :
    cle = L[i]
    j = i - 1
    while j >= 0 and cle < L[j]:
        temp = L[j]
        L[j] = L[j + 1]
        L[j + 1] = temp
        j -= 1

# La liste après le trie par insertion
print(f"La liste triée : {L[:]}")

# Identification du les nombre premier dans la liste P & supprimer apartir la
list L
i = 0
while i < len(L) :
    nombre = L[i]
```

```

    if nombre > 1 :
        premier = True
        for j in range(2 , nombre):
            if nombre % j == 0 :
                premier = False
                break
        if premier :
            P.append(nombre)
            L.pop(i)
        else :
            i += 1
    else :
        i += 1
# L'affichage de la liste P & L
print(f"La liste P : {P[:]}")
print(f"La liste L : {L[:]}")

```

Résultat du programme :

```

Saisir le nombre d'éléments :5
Saisir l'elemnt nombre 1 : 2
Saisir l'elemnt nombre 2 : 9
Saisir l'elemnt nombre 3 : 7
Saisir l'elemnt nombre 4 : 3
Saisir l'elemnt nombre 5 : 39
Votre Liste est : [2, 9, 7, 3, 39]
La liste triée : [2, 3, 7, 9, 39]
La liste P : [2, 3, 7]
La liste L : [9, 39]

```

Exercice 02 : Manipulation de Texte en Python

```
# Initialisation du text
text = "La filière D3SI de la FPBM, est une filière d'excellence du nouveau
système lancé par le ministère de l'enseignement supérieur. En plus de la
filière D3SI, nous trouvons la filière I3DIA, nous trouvons la filière I3DIA
et la filière IOR"

# Diviser le text on utiliser la fonction split et stoker la resultat dans
liste mots
mots = text.split()

# Initialiser la liste mots_traites
mots_traites = []

# Vérifier si une mot continet vergule ou point a la fin
for i in mots :
    if i[-1] == '.' or i[-1] == ',' :
        motsSontVergule = i.replace('.', '').replace(',', '')
        mots_traites.append(motsSontVergule)

# Afficher les mots traitez
print(f"Les mots Traiter : {mots_traites[:]}")

# Convertir chaque mot de la liste mots_traites en miniscule et stocker dans
mots_min
mots_min = []
for i in mots_traites :
    motsMiniscule = str(i)
    mots_min.append(motsMiniscule.lower())

# Afficher mots_min
print(f"les mots en miniscule : {mots_min[:]}")

# Trier les les mots de la liste mots_in et stoker le résultat dans une
nouvelle liste mots_min_len
mots_min_len = []
motsMinLen = sorted(mots_min , key=len , reverse=True)
mots_min_len.append(motsMinLen)
print(f"les mots min len : {mots_min_len[:]}")

# Inialiser le dictinnaire frequence
frequence = {}

# Initialiser un compteur qui compter le nombre de frequence et stocker dans
le dictinnaire frequence
cp = 0
```

```

for i in range(0 , len(mots_min)) :
    for j in range(0 , len(mots_min)):
        if mots_min[i] == mots_min[j] :
            cp += 1
        frequence[mots_min[i]] = cp
        cp = 0
print(f"le dictionnaire frequence : {frequence}")

# Initialiser la liste mots_min_frequence
mots_min_frequence = []

# Trier les mots de la liste mots_min par sorted & stocker dans
mots_min_frequence
motsMinSorted = sorted(mots_min , key=lambda x : (-frequence[x] , x))
mots_min_frequence.append(motsMinSorted)
print(f"Mots min Frequence : {mots_min_frequence}")

```

Résultat du programme :

```

Les mots Traiter : ['FPBM', 'supérieur', 'D3SI', 'I3DIA']
les mots en miniscule : ['fpbm', 'supérieur', 'd3si', 'i3dia']
les mots min len : [['supérieur', 'i3dia', 'fpbm', 'd3si']]
le dictionnaire frequence : {'fpbm': 1, 'supérieur': 1, 'd3si': 1, 'i3dia': 1}
Mots min Frequence : [['d3si', 'fpbm', 'i3dia', 'supérieur']]

```

Exercice 03 : Analyse de Texte en Python

Créez un programme Python qui examine un texte saisi par l'utilisateur et génère des statistiques sur les mots utilisés. Les résultats, tels que le nombre total de mots, la fréquence de chaque mot, la longueur moyenne des mots, et le mot le plus long, seront enregistrés dans un dictionnaire nommé D. Cet exercice vise à pratiquer la manipulation de dictionnaires et l'analyse simple de texte en Python.

```
# Initialisation du dictionnaire D
D = {
    "nombre_mots": 0,
    "frequence_mots": {},
    "longueur_moyenne": 0,
    "mot_le_plus_long": ""
}

# Demander à l'utilisateur d'entrer un texte
text = input("Entrez un texte : ")

# Compter le nombre de mots dans le texte
nombreMots = text.split()
D['nombre_mots'] = len(nombreMots)

# Calculer la fréquence de chaque mot dans le texte saisi
for mot in nombreMots:
    if mot in D['frequence_mots']:
        D['frequence_mots'][mot] += 1
    else:
        D['frequence_mots'][mot] = 1

# Calculer la longueur moyenne des mots
D['longueur_moyenne'] = sum(len(mot) for mot in nombreMots) / D['nombre_mots']

# Trouver le mot le plus long
D['mot_le_plus_long'] = max(nombreMots, key=len)

# Afficher le dictionnaire résultant
print(D)
```

Résultat du programme :

Entrez un texte : Lorem ipsum dolor sit amet

```
{  
'nombre_mots': 5,  
'frequence_mots': {  
'Lorem': 1,  
'ipsum': 1,  
'dolor': 1,  
'sit': 1,  
'amet': 1  
},  
'longueur_moyenne': 4.4,  
'mot_le_plus_long': 'Lorem'  
}
```

Conclusion :

En résumé, ces exercices pratiques en Python ont consolidé nos compétences essentielles. Du tri de listes à l'analyse de texte avec des dictionnaires, nous avons renforcé notre compréhension des fondamentaux du langage. Ces exercices diversifiés ont enrichi notre expérience et amélioré notre confiance dans l'utilisation de Python pour résoudre une variété de problèmes.