

TP n°1 d'informatique

J'ai fait le premier TP sous forme d'un script python complet avec des menus.

Télécharger le script ici.

```
"""
TP1 : Programme réalisé pour le Tp 1 d'informatique en MPSI, ce programme
calcule des sommes, moyennes et variances
"""

def main():
    """Fonction principale qui retourne toutes les opérations sur la liste"""
    print(
        "1) Somme\n2) Moyenne\n\
3) Variance 1\n4) Variance 2\n5) Collatz\n6) Nombres premiers\n\
7) N-ieme premier\n8) Recherche d'un élément dans une liste\n\
9) Maximum d'une liste\n10) Comptage des éléments d'une liste\n\
11) Comptage des éléments d'une liste v2 (entiers naturels)"
    ) # Récupère un liste de termes saisis par l'utilisateur
    reponse = int(input(">>> "))
    if reponse in (1, 2, 3, 4):
        liste_texte = input(
            "Donner une liste de nombres séparés par des espaces\n>>> "
        )
        liste = list(map(int, (x for x in liste_texte.split() if x.isdigit())))

        if reponse == 1:
            print("Somme :", somme(liste), "\n---")
            main()

        elif reponse == 2:
            print("Moyenne :", moyenne(liste), "\n---")
            main()
```

```

elif reponse == 3:
    print("Variance 1 :", variance1(liste), "\n---")
    main()

elif reponse == 4:
    print("Variance 2 :", variance2(liste), "\n---")
    main()

elif reponse == 5:
    entier = int(input("Donner un nombre entier\n>>> "))
    print("Collatz :", collatz(entier), "\n---")
    main()

elif reponse == 6:
    n_max = int(input("Liste des n premiers nombre premiers\n>>> "))
    print(n_max, "premiers nombre premiers :", n_premiers(n_max), "\n---")
    main()

elif reponse == 7:
    n = int(
        input("Couple constitué de n et du n-ième nombre premier\n>>> ")
    )
    retour = "(" + str(n) + ", " + str(n_premiers(n)[-1]) + ")" + "\n---"
    print(retour)
    main()

elif reponse == 8:
    list_input = input("Liste séparé par des espace\n>>> ")
    liste = list(map(str, (x for x in list_input.split())))
    element = input("Chercher quoi dans la liste\n>>> ")

    if recherche(liste, element):
        print(
            element,
            "est dans la liste à la position",
            recherche(liste, element),
            "\n---",
        )
    elif not recherche(liste, element):
        print(element, "n'est pas dans la liste\n---")

elif reponse == 9:
    list_input = input("Liste séparé par des espace\n>>> ")
    liste = list(map(int, (x for x in list_input.split())))

```

```

print(maximum(liste), "\n---")
main()

elif reponse == 10:
    list_input = input("Liste séparé par des espace\n>>> ")
    liste = list(map(str, (x for x in list_input.split())))
    print(compte(liste), "\n---")
    main()

# elif reponse == 11:
#     list_input = input(
#         "Liste séparé par des espace (entier naturels)\n>>> "
#     )
#     liste = list(map(int, (x for x in list_input.split())))
#     print(compte2(liste), "\n---")
#     main()

def somme(liste):
    """Retourne la somme des éléments de la liste"""
    somme_liste = 0
    for i in liste:
        somme_liste += i
    return somme_liste

def moyenne(liste):
    """Retourne la moyenne des éléments de la liste"""
    moyenne_liste = somme(liste) / len(liste)
    return moyenne_liste

def variance1(liste):
    """Retourne la variance des éléments de la liste"""
    variance1_liste = somme(
        [(x - moyenne(liste)) ** 2 for x in liste]) / len(liste)
    return variance1_liste

def variance2(liste):
    """Retourne la variance des éléments de la liste avec une autre méthode"""
    variance2_liste = (somme([x**2 for x in liste]) / len(liste)) + (
        (somme(liste) ** 2) / len(liste)
    )

```

```

return variance2_liste

def collatz(entier):
    """Fonction de collatz"""
    if entier % 2 == 0:
        image = entier / 2
    else:
        image = 3 * entier + 1
    return image

def premier(n):
    """Dit si un nombre est premier"""
    d = 0
    for i in range(2, n):
        if n % i == 0:
            d += 1
    if d < 1:
        return n
    return None

def n_premiers(n_max):
    """Retourne la liste des n premiers nombres premiers"""
    liste_n_premiers = []
    n = 2
    while len(liste_n_premiers) < n_max:
        if premier(n) is not None:
            liste_n_premiers.append(premier(n))
            n += 1
    return liste_n_premiers

def recherche(liste, element):
    """
    Renvoie true et la liste des indices correspondants si le nombre est
    dans la liste sinon false
    """
    trouve = []
    for i in range(len(liste)):
        if liste[i] == element:
            trouve.append(i)
    return trouve

```

```

def maximum(liste):
    """Renvoie le maximum d'une liste"""
    if not liste:
        return None
    n_max = liste[0]
    num_n_max = 0
    for i in range(len(liste)):
        if liste[i] > n_max:
            n_max = liste[i]
            num_n_max = i
    return n_max, num_n_max

def compte(liste):
    """Renvoie le nombre de chaque éléments d'une liste"""
    d = {}
    for e in liste:
        if e not in d:
            d[e] = 1
        else:
            d[e] += 1
    return d

# def compte2(liste):
#     """je sais même pas"""
#     T = [0] * (maximum(liste)[0] + 1)
#     for k in liste:
#         T[k] += 1
#     return T

if __name__ == "__main__":
    main()

```

Voir le PDF
