

SAÉ PYTHON :

Bilan carbone

Synthèse

Donc la première étape était de faire toutes les fonctions qui étaient proposées, je n'ai pas eu énormément de mal à les faire, ça a juste pris un peu de temps car les notions qu'il fallait maîtriser se voyaient au fur et à mesure des cours, et donc c'est pour ça qu'il était recommandé de se documenter en ligne. Pour l'affichage du menu et de l'interface utilisateur j'ai eu recours au TP7 de Python qui nous donnait de bonnes bases, pour avoir un peu d'originalité, j'ai décidé que l'utilisateur pourrait gérer les informations grâce aux listes qu'il pourrait manipuler à sa guise (plus d'informations dans la suite du rapport).

Analyse & Démonstration de compétence

La plupart des fonctions était assez simple mais pour certaines fonctions j'ai dû me documenter en ligne. Notamment pour les dictionnaires (on les a vues en cours mais c'était très tardif j'avais déjà fini mon programme à ce moment là).

J'ai d'abord été contraint d'utiliser les dictionnaire pour mes fonctions, comme:

```
def cumul_temps_activite(liste_activites, co2_minute):
    """
    Retourne le temps total passé à réaliser des activités

    Args:
        liste_activites (list): une liste d'activités

    Returns:
        float: le temps total passé à réaliser des activités
    """
    try:
        cptime = 0
        cptconso = 0
        for i in range(len(liste_activites)):
            for cle, valeur in co2_minute.items():
                if cle == liste_activites[i][3]:
                    cptime += valeur
                    cptconso += liste_activites[i][2]
            if cptime == 0:
                return 0
            return cptconso/cptime
    except:
        return None
```

Ici j'utilise le dictionnaire pour attribuer un type à un temps, grâce au dictionnaire « co2_minuttes »

```
co2_minute = {'type1': 0.87, 'type2': 0.65, 'type3': 0.96, 'type4': 0.63}
```

Et donc le programme va parcourir 2 éléments la clé du dictionnaire ainsi que ses valeurs.

Et donc je dis que si la clef est égale à l'indexe 3 de ma liste de tuple (qui correspond à la chaîne de caractère type) d'accumuler cette valeur ainsi que le temps pour renvoyer le temps en fonction de la consommation.

Et ensuite j'ai utilisé les dictionnaires dans mon programme principal pour pouvoir sauvegarder de manière dynamique des listes que l'utilisateur souhaite manipuler plus tard (pour ne pas lui faire enregistrer des fichiers CSV qu'il n'a pas besoin).

Grâce à mon dictionnaire initialisé « saved », je peux proposer à l'utilisateur de sauvegarder des listes à chaque fois qu'il exécute un programme qui retourne une liste de tuple.

```
if repinfo == 1:
    if len(saved) != 0:
        question = input("voulez vous utiliser l'une de vos liste[Y/N]"+ "\n" "si vous repondez [N]on nous allons utiliser la grande liste"+ "\n")
        if question == "Y":
            clef = input("Quelle liste voulez vous mettre ?"+ "\n")
            para = input("Mettre un prenom de la liste" + "\n")
            for keys, valeur in saved.items():
                if clef == keys:
                    valoc = valeur
                    if bc.filtre_par_prenom(valoc, para) == []:
                        print("> la recherche n'a pas abouti:"+ "\n" + " indice : Le prénom n'est pas dans la liste " + "\n" + " indice : Vous avez sur")
                    else:
                        print(f"Voici toutes les activité liée à {para}" + "\n")
                        varloc = bc.filtre_par_prenom(valoc, para)
                        print(varloc)
                        listYN = input("Voulez vous sauvgarder les information ? [Y/N]"+ "\n")
                        if listYN == "N":
                            pass
```

Ensuite, dans ce bout de code, si la longueur du dictionnaire n'est pas égale 0 cela signifie qu'il a enregistré des listes donc je lui propose si oui ou non il veut l'utiliser

Et grâce à ce genre d'implémentation, on peut faire des manipulations de données rapides et précises, voici un exemple tiré par les cheveux : Je veux savoir la consommation maximale d'une liste de personnes s'appelant Magomed et Lucas et aussi de toutes les activités passées le 5 septembre uniquement de type 1

```
lucmag5sep1
voici la consommation total de co2 de la liste : lucmag5sep1 : 8724.630000000003 g
```

Donc ici j'ai dû fusionner la liste « Magomed » et la liste « Lucas », avec une liste d'activités de type 1 faite le 5 septembre (qui a dû être filtrée 2 fois avec les fonctions)(tout cela a été fait avec les options du menu par l'utilisateur), ensuite disons que je veuille enregistrer cette liste dans un fichier CSV, je choisis l'option pour, et je peux même nommer le fichier CSV à ma guise.

```

+-----+
| Choisi les options que tu désire |
+-----+
1 --> Recherche d'info
2 --> Sauvgarder les info dans un fichiers CSV
3 --> Voir vos listes sauvgarder
4 --> Comment utiliser le programme
5 --> Quitter
Entrez votre choix [1-5]
2
Vous avez choisi : [ Sauvgarder les info dans un fichiers CSV ]
Comment voulez vous appeler le fichier?
rapport
Quelle liste voulez vous mettre ?
magluc5sep1
Appuyer sur Entrée pour continuer

```

```

pythonSEA > Bilan Carbone mystérieux-20241002 > rapport.csv > data
1 Nom, Date, Consomation(g), Type
2 Lucas,2024-09-01,7.83,type1
3 Magomed,2024-09-01,58.29,type1
4 Lucas,2024-09-02,1.74,type1
5 Magomed,2024-09-02,40.02,type1
6 Lucas,2024-09-03,6.09,type1
7 Magomed,2024-09-03,19.14,type1
8 Lucas,2024-09-04,20.01,type1
9 Magomed,2024-09-04,17.4,type1
10 Abdulkerim,2024-09-05,25.23,type1
11 Abdulkerim,2024-09-05,25.23,type1
12 Adnan-Erdem,2024-09-05,28.71,type1
13 Adnan-Erdem,2024-09-05,28.71,type1
14 Alexandre,2024-09-05,3.48,type1
15 Alexandre,2024-09-05,3.48,type1
16 Anaëlle,2024-09-05,23.49,type1
17 Anaëlle,2024-09-05,23.49,type1
18 Antonin,2024-09-05,36.54,type1
19 Antonin,2024-09-05,36.54,type1
20 Arif,2024-09-05,17.4,type1
21 Arif,2024-09-05,17.4,type1
22 Arthur,2024-09-05,30.45,type1
23 Arthur,2024-09-05,30.45,type1
24 Aslam,2024-09-05,20.88,type1
25 Aslam,2024-09-05,20.88,type1
26 Asma,2024-09-05,10.44,type1
27 Asma,2024-09-05,10.44,type1
28 Bastian,2024-09-05,3.48,type1
29 Bastian,2024-09-05,3.48,type1
30 Bastien,2024-09-05,43.5,type1
31 Bastien,2024-09-05,43.5,type1
32 Enzo,2024-09-05,34.8,type1
33 Enzo,2024-09-05,34.8,type1

```

Donc toutes les conditions sont bien respectées. Donc mon programme permet une manipulation rapide et poussée de données.

Conclusion

En conclusion, cette SAÉ a été plaisante à réaliser car j'ai mis beaucoup d'efforts à sa réalisation, pour mettre « vie » à tout un programme qui fonctionne à partir de rien. Elle m'a fait me rendre compte de comment un programme doit être organisé, avec cette SAÉ je suis plus confortable à l'idée de manipuler des fichiers ou d'autres programmes (chose que je n'avais pas encore faite) et donc ça a été une bonne expérience, je pense aussi qu'il pourrait y avoir quelque amélioration sur la qualité de mon code mais ça se construit avec l'expérience