



# Grand projet Python

PROGRAMME IPSA CYB'AIR

Charbonnet | Thomas | 15/04/2019

## Sommaire :

### INTRODUCTION

### I- NOTION NECESSAIRE A LA CONCEPTION DU CODE

### II- INTERFACE GRAPHIQUE

### III- FONCTIONS

A- Inscription

B- Affichages des membres

C- Supprimer un membre de nos fichiers

D- Recherche un membre dans nos fichiers

E- Modifier les informations d'un membre

F- Statistique

### IV- DIFFICULTE RENCONTREE ET AMELIORATION POSSIBLE

A- Difficulté rencontrée

B- Amélioration possible

### CONCLUSION

### ANNEXE

## INTRODUCTION

IPSA CYB'AIR est une association spécialisée dans l'enseignement de la programmation et de la réalisation de projet lié à programmation. Dans le cadre du grand projet python, nous avons décidé de concevoir un programme qui doit permettre de faciliter la gestion des membres de l'association. Notre programme joue le rôle d'un « annuaire » qui répertorie tous les membres de notre association. Nous allons programmer ce programme en langage python 3 et nous stockerons nos valeurs dans des fichiers txt.

Dès le début du projet, nous avons définis le but de notre programme et un cahier des charges. Selon notre cahier des charges, notre programme doit être capable de :

- enregistrer une personne dans un fichier txt
- empêcher que deux personnes portant le même prénom et nom de famille soient inscrits dans le même fichier txt
- afficher toutes les personnes inscrites avec leurs informations personnelles dans une interface graphique
- supprimer une personne de notre fichier txt
- recherche une personne dans notre fichier txt en fonction de son prénom, nom ou numéro de téléphone
- modifier les informations d'une personne se trouvant dans notre fichier txt à partir d'une interface graphique
- faire des statistiques sur les membres de notre association
- faire une surprise à l'utilisateur

Après avoir défini les objectifs du projet, nous avons défini un planning à tenir. La première séance va servir à la conception de l'interface d'enregistrement des membres et de l'affichage. La seconde séance est réservée à l'interface de suppression et de recherche des membres. La troisième séance est réservée à l'interface de statistique et de l'amélioration de l'esthétique de notre interface graphique. La quatrième séance est réservée à la correction des bugs qui gênent notre programme. La dernière séance est réservée à l'élaboration du dossier.

## I- Notion nécessaire à la conception du code

Dans le cadre de ce mini projet, nous avons dû apprendre de nombreuses choses. L'apprentissage des nouvelles notions se faisaient avant de venir en séance de grand projet.

Pour réaliser notre projet, nous avons décidé d'utiliser le langage python3. Nous avons décidé de choisir ce langage car il s'agit d'un langage assez simple à comprendre. Mais aussi, la communauté qui utilise ce langage est très grande, donc si nous avons un problème, nous n'avons qu'à aller voir les cours de la communauté ou à demander à la communauté de nous aider à résoudre notre problème.

Pour la première séance, nous avons dû apprendre à nous servir de fichier text. Ces fichiers permettent d'enregistrer des chaînes de caractères. Cela permet de ne pas perdre ces chaînes de caractères quand nous arrêtons notre programme. Afin de réaliser notre projet, nous avons dû apprendre à créer un fichier texte, à lire et à afficher les éléments qui composent notre fichier texte. Mais aussi, nous avons aussi dû apprendre à ajouter de nouveaux éléments dans notre fichier texte.

Pour la seconde séance, nous avons dû découvrir les fonctions de la bibliothèque « matplotlib ». Cette bibliothèque permet en partie à concevoir des diagrammes. Les diagrammes que nous avons conçus grâce à cette bibliothèque nous ont permis d'afficher des statistiques sur les informations que nous avons stockées dans notre fichier.

Pour la troisième séance, nous avons dû étudier les fonctionnements du « canvas ». Le « canvas » est une des trois possibilités qui permet d'afficher des éléments dans une interface graphique. Nous avons décidé d'utiliser un « canvas » dans le but d'afficher des photos. Le « canvas » nous a servi quand nous voulions afficher nos statistiques.

Pour la quatrième séance, nous avons dû chercher des solutions pour résoudre certains problèmes. Une des solutions qui a résolu beaucoup de nos problèmes était le « Toplevel() ». Le « Toplevel() » permet de concevoir des fenêtres indépendantes aux autres fenêtres.

Tous les éléments que nous venons de voir nous ont permis de concevoir un code performant avec une interface assez jolie. La recherche et la compréhension de ces éléments nous a pris beaucoup de temps. En effet, il a fallu aller sur de nombreux sites internet (comme open Classroom ou python-Django) et forum pour trouver toutes ces informations.

## II- Interface graphique

Pour concevoir nos interfaces graphiques, nous avons décidé d'utiliser la librairie Tkinter. Aujourd'hui, la librairie Tkinter est l'une des librairies les plus complètes dans la conception d'interfaces graphiques. En effet, grâce à cette librairie, nous pouvons concevoir des boutons, des checkboxes, des combobox, des labels, des entree et beaucoup d'autres choses. En plus d'être très complète, cette librairie est simple d'utilisation car elle possède des fonctions avec des noms simples et des fonctions avec des noms complexes à utiliser. En plus de tout cela, la communauté qui utilise cette librairie est très grande. Donc si nous avons un



problème, nous n'avons qu'à demander à la communauté. La réponse de la communauté est généralement très rapide.

Dans le cadre de notre projet, nous avons réalisé plusieurs interfaces graphiques. Certaines ont un rôle très important sur le bon fonctionnement du programme et d'autres servent à simplifier l'utilisation du programme. Dans cette partie, nous allons vous présenter les interfaces les plus importantes.

La première interface que nous allons voir est l'interface de base. Cette interface regroupe tous les sous programmes de notre grand programme. Nous pouvons dire que cette interface est la colonne vertébrale de notre programme. Pour réaliser cette interface, nous avons décidé d'utiliser que des boutons. Pour cela, nous avons utilisé la fonction Button présente dans la librairie Tkinter. Pour afficher l'image de fond, nous avons utilisé un label dans lequel nous avons placé le logo de notre association. Juste en dessous de ce petit paragraphe, vous pouvez voir l'interface principale.



L'interface que nous allons voir permet d'inscrire des personnes dans notre fichier texte. Cette interface possède de nombreux labels, entées et boutons. Nous avons à faire à une interface très basique mais qui reste nécessaire pour le bon fonctionnement de notre programme.

Sur votre gauche, vous pouvez observer l'interface graphique qui sert aux inscriptions.

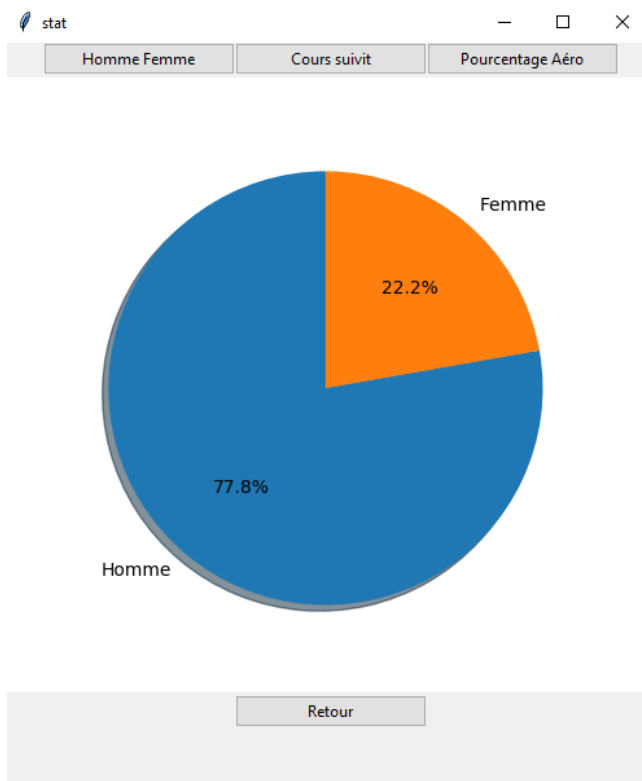
Maintenant, nous allons vous présenter l'interface qui sert à afficher toutes les personnes que nous avons inscrits dans notre fichier. Pour concevoir cette interface, nous avons dû découper notre fenêtre en deux partis. La première partie utilise la fonction « Text ». Cette fonction nous a permis de concevoir des colonnes dans lesquelles nous avons affiché les informations stockées dans notre fichier texte. Dans la seconde partie, nous avons utilisé des fonctions « Button » dans le but d'utiliser des fonctions qui affichent les membres de notre association en fonction de leur spécialité. Voici une photo de notre interface graphique qui sert à l'affichage de nos membres.

Charbonnet	Thomas	aéro1	homme	thomas.charbonnet@ipsa.fr	0783209696	, Catia, C++, Photoshop
Charbonnet	Philippe	aéro5	homme	philippecharbonnet@yahoo.fr	0685430158	STK, Catia, Photoshop
Lenté	Elsa	aéro1	femme	elsa.lente@ipsa.fr	850	STK, Catia, Photoshop
Bussiere	Edgar	aéro1	homme	edgar.bussiere@ipsa.fr	74123	, Catia, C++
Glachan	Luc	aéro1	homme	luc.glachan@ipsa.fr	96321	, Catia
Planell	Robin	aéro3	homme	robin.planell@ipsa.fr	753	STK, Catia, C++, Photoshop
Grelaud	Victor	aéro1	homme	victor.grelaud@ipsa.fr	74520	STK, Catia
Wernert	Anthony	aéro1	homme	anthony.wernert95@gmail.com	0687053707	, Catia, Photoshop

Les interfaces qui servent à supprimer un membre de nos fichiers et à rechercher un membre dans nos fichier sont très simple. En effet, ces interfaces utilisent des Label, des Entry, des Button (bouton) et des ComboBox.

La photo en dessous correspond à l'interface qui sert à supprimer des membres dans nos fichiers.

La photo qui est en dessous correspond à l'interface qui sert à rechercher une personne dans nos fichiers.



La dernière interface majeure de notre programme sert à afficher nos statistiques.

Pour réaliser cette interface, nous avons dû utiliser un « Toplevel() ». En faisant cela, nous avons pu afficher des photos sans problèmes. Cette interface est divisée en trois parties. La première partie est en haut. Il s'agit d'une grille qui contient trois boutons. La seconde partie est au milieu de l'interface. Il s'agit d'un « canvas » qui affiche une phot. La dernière partie est en bas. Cette partie est une grille et contient un bouton.

Vous pouvez voir sur votre gauche l'interface graphique qui affiche nos statistiques.

Dans cette partie, nous venons de voir les interfaces graphiques principales. Il y a dans notre programme beaucoup d'autre interface graphique mais ce serait trop long de tous vous les présenter. De plus, ces autres interfaces ne sont pas très intéressantes et se ressemblent toutes.

### III- FONCTIONS

#### A- Inscription

Dans l'interface graphique qui permet d'enregistrer une personne dans notre fichier, nous avons conçu deux fonctions de base.

La première se nomme « retour ». Cette fonction a juste pour but de détruire la fenêtre qui contient notre interface graphique.

```
def retour():  
    fenetre2.destroy()
```

La seconde fonction se nomme « registreur ». Cette fonction permet d'enregistrer une personne dans notre fichier texte qui se nomme « inscrist ». Cette fonction est composée de 3 parties. La première partie permet de récupérer les informations (qui se trouvent dans des entree) sur la personne que nous voulons ajouter à notre fichier texte. De plus, toutes les informations seront mises dans une grande chaîne de caractère qui se nomme T

```
def registreur():  
  
    etat = 0  
    compteurLigne = 0  
  
    A = str(entree1.get())  
    B = str(entree2.get())  
    Cbis = str(entree3.get())  
    C = str(homme.get())  
    D = str(femme.get())  
    E = str(entree5.get())  
    F = str(entree6.get())  
  
    G = str(stk.get())  
    print(G)  
    H = str(katia.get())  
    I = str(c.get())  
    J = str(photoshop.get())  
    K = str(latex.get())  
  
    entree1.delete(0,END)  
    entree2.delete(0,END)  
    entree3.delete(0,END)  
    entree5.delete(0,END)  
    entree6.delete(0,END)  
  
    T = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}\n".format(A,B,Cbis,C,D,E,F,G,H,I,J,K)
```

Après avoir fait cela, la programme va vérifier dans notre fichier qu'une autre personne enregistrer ne porte pas le même nom. Nous faisons cela dans le but d'éviter les doublons. Par exemple, dans le cas de la suppression d'une personne de nos fichiers textes, nous



risquons de supprimer la mauvaise personne si nous avons des doublons. Pour éviter un doublon, nous ouvrons notre fichier où nous stockons tous nos inscrits et nous vérifions ligne par ligne qu'il n'y a pas de personne qui porte le même prénom et nom de famille.

```
for l in FichierControle.readlines():
    r = l.split(',')
    compteurLigne = compteurLigne+1
    if A == r[0]:
        if B == r[1]:
            nomDuFichier = "inscris.txt"

            F = open('fichierListe.txt','w')
            F.write(T)
            F.write(nomDuFichier)
            F.close()

            choix()

            etat = 1

FichierControle.close()
```

Si le système trouve une personne qui porte le même nom et prénom, le système lance une fonction graph. Cette fonction chois se trouve dans ma bibliothèque qui se nomme « maBibliotheque ».

La fonction choix permet à l'utilisateur de résoudre son problème de doublon de deux manières différentes. Soit il modifie son fichier texte (fonction modifierV2) en supprimant la personne qui a le même prénom et nom que la personne qui veut ajouter. Après avoir fait cela, il enregistre la personne qui veut ajouter à notre fichier. Sinon, notre utilisateur peut modifier le prénom de la personne qui veut enregistrer (fonction modifierNom).

```
def choix():
    def retour():
        fenetre2.destroy()

    recupDeValeur = []
    F = open('fichierListe.txt','r')
    for i in F.readlines():
        recupDeValeur.append(i)
    F.close()

    L = recupDeValeur[0]
    r = L.split(',')

    fenetre2 = Tk()
    fenetre2.geometry("550x250+500+200")

    texte = "il y a déjà une personne qui se nomme "+r[0]+", " +r[1]

    label0 = Label(fenetre2, text = texte)
    label0.grid(row = 0, column = 1)

    bouton1 = Button(fenetre2, text="Modifier la base de donnée",command = modifierV2)
    bouton1.grid(row = 1, column = 1)
    bouton2 = Button(fenetre2, text="Modifier le prénom que vous voulez entrer", command = modifierNom)
    bouton2.grid(row = 2, column = 1)

    boutonRetour = Button(fenetre2, text="Retour",command = retour)
    boutonRetour.grid(row = 3, column = 0)
```

Si la machine ne trouve pas de personne qui a le même prénom et nom de famille, le programme enregistre directement la personne que nous voulons intégrer à notre fichier texte dans

```
FichierEnregistrement = open ("inscris.txt", 'a')
FichierEnregistrement.write(T)
FichierEnregistrement.tell()
FichierEnregistrement.close()
```

notre

Après avoir enregistré les nouvelles informations, la fenêtre de notre interface se détruit.

```
fenetre2.destroy()
```

## B- Affichages des membres

Pour afficher les informations dans chaque colonne, nous avons décidé de concevoir la fonction « affichage ».

```
truc = "normal"

liste = affichage(truc)
```

La fonction affichage se trouve dans ma bibliothèque qui se nomme « maBibliotheque ». Cette fonction va ouvrir notre fichier texte « inscristxt » et va lire chaque ligne. Quand il lit une ligne, il en fait une liste. Cette liste est ensuite traitée dans le but d'obtenir des variables. Après le traitement, les variables sont réparties dans 7 chaînes de caractères. La variable « truc » sert lors du traitement des variables. En effet, nous avons conçu des conditions qui sélectionnent les éléments mis dans les chaînes de caractères. Par exemple, quand truc = « normal » tous les éléments de notre fichier sont mis dans les chaînes de caractères.

```
quote1 = ""
quote2 = ""
quote3 = ""
quote4 = ""
quote5 = ""
quote6 = ""
quote7 = ""

Fichier = open("inscristxt", 'r')
for l in Fichier.readlines(): #le programme lit le fichier ligne par ligne
    cours = str()
    sexe = str()
    r = l.split(',') #la ligne devient une liste
    nom = r[0]
    prenom = r[1]
    promo = r[2]
    if r[3] == str(1):
        sexe = str("homme")
    if r[4] == str(1):
        sexe = str("femme")
    mail = r[5]
    tel = r[6]
    if r[7] == str(1):
        cours = cours + str("STK")
    if r[8] == str(1):
        cours = cours + str(",Catia")
    if r[9] == str(1):
        cours = cours + str(",C++")
    if r[10] == str(1):
        cours = cours + str(",Photoshop")
    if r[11] == str(1):
        cours = cours + str(",Latek")

    if truc == "normal":
        quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
        quote2 = quote2 + prenom + "\n"
        quote3 = quote3 + promo + "\n"
        quote4 = quote4 + sexe + "\n"
        quote5 = quote5 + mail + "\n"
        quote6 = quote6 + tel + "\n"
        quote7 = quote7 + cours + "\n"
```

Après avoir fait le traitement, nous retournons nos chaînes de caractères.

```
Fichier.close()
return(quote1,quote2,quote3,quote4,quote5,quote6,quote7)
```

Après avoir fait cela, la fonction affichage prend fin. Les éléments que nous retournons sont mis dans une liste qui se nomme « liste ». Les éléments dans « liste » sont répartis dans des chaînes de caractères puis chaque chaîne de caractère est mise dans une ligne de notre tableau. Nous obtenons cela :

Charbonnet	Thomas	aérol	homme	thomas.charbonnet@ipsa.fr	0783209696	,Catia,C++,Photoshop
Samou	Mélanie	aérol	femme	melanie.samou@ipsa.fr	12345678	,Photoshop
Charbonnet	Philippe	aéro5	homme	philippecharbonnet@yahoo.fr	0685430158	STK,Catia,Photoshop
Lenté	Elsa	aérol	femme	elsa.lente@ipsa.fr	850	STK,Catia,Photoshop
Bussiere	Edgar	aérol	homme	edgar.bussiere@ipsa.fr	74123	,Catia,C++
Glachan	Luc	aérol	homme	luc.glachan@ipsa.fr	96321	,Catia
Planell	Robin	aéro3	homme	robin.planell@ipsa.fr	753	STK,Catia,C++,Photoshop
Grelaud	Victor	aérol	homme	victor.grelaud@ipsa.fr	74520	STK,Catia
Wernert	Anthony	aérol	homme	anthony.wernert95@gmail.com	0687053707	,Catia,Photoshop

Si nous voulons afficher que des personnes qui ont les mêmes spécialités, nous utilisons des fonctions qui changent la valeur de « truc ». Par exemple, si nous utilisons la fonction STK, la valeur de truc devient : « STK ». En faisant cela, la condition truc == « normal » n'est plus remplie. Par contre, la condition truc == STK est remplie. Après avoir fait cela, une autre condition arrive est demande que la 8 ième valeur de notre liste soit égale à 1. Si la condition est remplie, seules les personnes de notre registre qui ont la spécialité STK sont enregistrer dans notre fichier.

```
if truc == "STK":
    if r[7] == str(1):
        quote1 = quote1 + nom + "\n"
        quote2 = quote2 + prenom + "\n"
        quote3 = quote3 + promo + "\n"
        quote4 = quote4 + sexe + "\n"
        quote5 = quote5 + mail + "\n"
        quote6 = quote6 + tel + "\n"
        quote7 = quote7 + cours + "\n"
```

Nous obtenons cela si nous utilisons la fonction STK :

Charbonnet	Philippe	aéro5	homme	philippecharbonnet@yahoo.fr	0685430158	STK,Catia,Photoshop
Lenté	Elsa	aérol	femme	elsa.lente@ipsa.fr	850	STK,Catia,Photoshop
Planell	Robin	aéro3	homme	robin.planell@ipsa.fr	753	STK,Catia,C++,Photoshop
Grelaud	Victor	aérol	homme	victor.grelaud@ipsa.fr	74520	STK,Catia

## C- Supprimer un membre de nos fichiers

Cette fonctionnalité du programme est l'une des fonctionnalités sur laquelle nous avons eu le plus de mal à concevoir. En effet, nous ne savions pas comment faire pour supprimer une ligne de notre fichier. Cette fonctionnalité utilise la fonction « priseValeur ».

Cette fonction prend les valeurs dans les deux entrées de notre interface graphique.

```
def priseValeur():
    information1 = entre1.get()
    information2 = entre2.get()
    supress(information1,information2)
    label3 = Label(fenetreDel, text = "Personne supprimer de notre fichier")
    label3.grid(row = 3, column = 1)
```

Après avoir pris les deux valeurs, cette fonction appelle une fonction qui se trouve dans « maBibliotheque » et qui se nomme « supress ». La fonction « supress » prend en paramètre deux valeurs.

```
def supress(information1,information2):
    F = open("inscris.txt","r")
    maListe = []
    for l in F.readlines():
        ligne = l.split(",")
        maListe.append(ligne)
    for i in range(0,len(maListe)):
        ligne = maListe[i]
        if information1 == ligne[0]:
            if information2 == ligne[1]:
                print(ligne)
                maListe.remove(maListe[i]) |
                break
    F.close()
    F2 = open("inscris.txt","w")
    for i in range (0,len(maListe)):
        T = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}".format(maListe[i][0],
        F2.write(T)
        print(T)
    F2.close()
```

La fonction « supress » commence par ouvrir le fichier « inscris » et lis chaque ligne. Chaque ligne est transformée en liste et chaque liste est mis dans une liste qui se nomme « maListe ». Après avoir fait cela, le programme regarde la valeur se trouvant à la première et seconde position de chaque liste (ces listes se trouvent dans ma liste). Si le programme trouve une correspondance avec les deux paramètres de la fonction « supress », la liste qui contient les deux éléments est supprimé de « maListe ». Après avoir fait cela, la boucle for s'arrête. Après avoir fait cela, le programme ouvre de nouveau le fichier inscris , supprimer tous ce qui se trouve dans ce fichier et réécrit chaque liste qui se trouve dans « maListe ». Après avoir réalisé cette action, la fonction « supress » prend fin.

## D- Recherche un membre dans nos fichiers

Cette fonctionnalité permet à notre programme de retrouvé une personne en fonction de son nom, prénom ou nom de famille. Dans un premier temps, notre programme exécute la fonction « premierTraitement ». Cette fonction va d'abord regarder sur quel critère est basé notre recherche. Ensuite, cette fonction va regarder si l'élément que nous recherchons est présent dans plusieurs lignes de notre fichier texte. Si cela est le cas, la valeur d'un compteur augmente. Après avoir fait cela, le programme va lancer la fonction « modifListeComboBox() ». Cette fonction va modifier les éléments qui sont dans notre comboBox. Peu après, le programme va rentrer dans une fonction « if else ». La condition de la fonction est la valeur du compteur.

Si le compteur est >1 , le programme va demander à l'utilisateur s'il n'a pas une autre information sur la personne recherchée. Si c'est le cas, l'opérateur rentre une autre information. En faisant cela, le programme arrive à retrouver la personne recherchée. Le programme trouve la personne recherchée, un fenetre s'ouvre et affiche toutes les informations que nous avons sur la personne recherchée. Par contre, si l'opérateur n'a pas

d'autre information sur la personne recherchée, il affiche toutes les personnes qui ont la même informations (celle qui a été introduit lors de la première recherche).

Si le compteur = 1, le programme ouvre une fenêtre grâce à la fonction « infoMembre » qui affiche toutes les informations sur la personne recherché

Si le compteur = 0, le programme dit à l'opérateur qu'il ne trouve personne dans notre fichier qui a cette caractéristique.

```
def rechercherUnPersonne():
    value = C1.get()
    premiereInformation = str(e1.get())

    compteur = premierTraitement(value,premiereInformation)

    modifListeComboBox(value)

    if compteur > 1:
        fenetre.destroy()
        interfaceSecondTraitement()

    elif compteur == 1 :
        fenetre.destroy()
        Fichier2 = open("recherche.txt",'r')### j'ouvre le fichier dans lequel j'ai enregistrer toutes les correspondances
        information = Fichier2.readline()
        a = information.split(',')
        Fichier2.close()
        infoMembre(a)

    else:
        fenetre.destroy()
        pasDeCorrespondance(premiereInformation)
```

## E- Modifier les informations d'un membre

Cette fonction permet de modifier des informations que nous avons sur une personne déjà enregistré dans notre fichier. Cela permet de ne pas à avoir à supprimer une personne de nos fichiers si nous voulons juste modifier des informations sur cette personne. Cette fonction est très simple à comprendre. Pour commencer, la fonction prend tous les éléments que nous avons mis dans notre interface graphique. Ensuite, le programme ouvre notre fichier texte, récupère chaque ligne, transforme les lignes en liste et ajoute chaque liste dans une grande liste. Après avoir fait cela, le programme cherche dans la grande liste une personne qui a le même prénom et nom de famille que la personne que nous voulons enregistrer. Quand le programme cette personne, il supprime cette personne de la grande liste. Pour finir, le programme ouvre notre fichier avec tous les inscrits, il supprime tous ce qu'il y a dedans et il réécrit tous les éléments de notre grande liste dans notre fichier qui doit contenir tous les inscrits.



```
def modifier():
    A = str(entre1.get())
    B = str(entre2.get())
    Cbis = str(entre3.get())
    C = str(homme.get())
    D = str(femme.get())
    E = str(entre5.get())
    F = str(entre6.get())
    G = str(stk.get())
    H = str(katia.get())
    I = str(c.get())
    J = str(photoshop.get())
    K = str(latex.get())
    T = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}\n".format(A,B,Cbis,C,D,E,F,G,H,I,J,K)
    F = open("inscris.txt","r")
    maListe = []
    for l in F.readlines():
        ligne = l.split(",")
        maListe.append(ligne)
    for i in range(0,len(maListe)):
        ligne = maListe[i]
        if A == ligne[0]:
            if B == ligne[1]:
                maListe.remove(maListe[i]) ### on supprime la ligne que l'on veut supprimer dans notre liste
                break
    F.close()
    F2 = open("inscris.txt","w")### on réécrit notre fichier avec notre nouvelle liste
    F2.write(T)
    for i in range (0,len(maListe)):
        T2 = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}".format(maListe[i][0],maListe[i][1],
            maListe[i][2],maListe[i][3],
            maListe[i][4],maListe[i][5],
            maListe[i][6],maListe[i][7],
            maListe[i][8],maListe[i][9],
            maListe[i][10],maListe[i][11])
    F2.write(T2)
    F2.close()
```

## F- Statistique

Cette fonctionnalité utilise une fonction du nom de graph. Cette fonction se trouve aussi dans ma librairie « maBibliotheque ». Cette fonction va d'abord commencer par lire ligne par ligne les lignes du fichier « inscris ». Chaque ligne sera mise dans une liste (les valeurs de la liste changent à chaque fois que les lignes changent) . Après avoir fait cela, le programme prend certaines valeurs de la liste et augmente certains compteurs.

```
Fichier = open ("inscris.txt", 'r')
for l in Fichier.readlines():
    liste = l.split(',')
    if liste[3] == str(1):
        nombreHomme = nombreHomme + 1
    if liste[4] == str(1):
        nombreFemme = nombreFemme + 1
    if liste[7] == str(1):
        nombreSTK = nombreSTK + 1
    if liste[8] == str(1):
        nombreCatia = nombreCatia + 1
    if liste[9] == str(1):
        nombreC = nombreC + 1
    if liste[10] == str(1):
        nombrePhotoshop = nombrePhotoshop + 1
    if liste[11] == str(1):
        nombrelatex = nombrelatex + 1
    if liste[2] == "aéro1":
        nombreAero1 = nombreAero1 + 1
    if liste[2] == "aéro2":
        nombreAero2 = nombreAero2 + 1
    if liste[2] == "aéro3":
        nombreAero3 = nombreAero3 + 1
    if liste[2] == "aéro4":
        nombreAero4 = nombreAero4 + 1
    if liste[2] == "aéro5":
        nombreAero5 = nombreAero5 + 1
Fichier.close()
```

Après avoir obtenues les valeurs des compteurs finaux, nous introduisons ces valeurs dans des diagrammes. Nous arrivons à concevoir ces diagrammes grâce à la bibliothèque « matplotlib ».

```

name = ['Homme', 'Femme']
data = [nombreHomme,nombreFemme]
explode=(0, 0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat1.png')
plt.close()
plt.show()

name = ['STK', 'CATIA','C++','Photoshop','latek']
data = [nombreSTK,nombreCatia,nombrePhotoshop,nombreC, nombrelatek]
explode=(0,0,0,0,0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat2.png')
plt.close()
plt.show()

name = ['Aéro1', 'Aéro2','Aéro3','Aéro4','Aéro5']
data = [nombreAero1,nombreAero2,nombreAero3,nombreAero4, nombreAero5]
explode=(0,0,0,0,0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat3.png')
plt.close()
plt.show()

```

La fonction va faire des captures de chaque diagramme. Ces captures sont enregistrées dans le fichier qui contient tous les bibliothèques et fichiers textes. Après avoir fait ces captures, la fonction s'arrête. Le programme va ensuite afficher la capture qui se nomme « stat1 ». Si nous voulons afficher une autre capture, nous avons juste à appuyer sur des boutons qui exécuter des fonctions qui changent la valeur de « nomDeLaPhoto ».

```

img = PhotoImage(file=nomDeLaPhoto)
canf1 = Canvas(p2, bg = 'blue', width = 500, height = 475)
canf1.create_image(-80,0,anchor = tk.NW, image = img)
canf1.image = img
canf1.grid(column = 2,row= 1)

```

## IV- DIFFICULTE RENCONTREE ET AMELIORATION POSSIBLE

### A- Difficulté rencontrée

Lors de la réalisation de ce projet, nous avons dû faire face à de nombreuses difficultés. Cela allé de la simple erreur de syntaxe jusqu'à la panne sèche pour la conception de certaines fonctions. Mais nous avons eu 2 grosses difficultés.

Une des fonctions qui nous a posé le plus de problème était la fonction qui permettaient de supprimer une personne de nos fichiers. En effet, nous avons tenté plusieurs fonctions mais à chaque fois, les résultats obtenus n'étaient pas ceux que nous voulions. Mais à force de travailler, nous avons réussi à concevoir une fonction qui remplit cette fonction. Grâce à cette fonction, nous avons pu concevoir d'autres fonctions sans problèmes.

Un autre problème que nous avons eu était lié aux checkBoXs qui se trouvaient dans la fonctionnalité inscription et modification. En effet, quand notre programme était lancé directement, les checkBoXs marchaient sans problèmes. Quand nous avons relié nos checkBoXs au programme de base, nous avons eu de nombreux problèmes. Par exemple,

les checkBoxs se sélectionnaient tous seul ou les valeurs des checkBoxs n'étaient pas prises. Nous avons réussi à résoudre ce problème en utilisant une fenêtre Toplevel().

Les deux éléments que nous venons de vous présenter nous ont bien ralenti sur l'avancée du projet.

### B- Amélioration possible

Le programme que nous venons de vous présenter est loin d'être terminé. Nous voulons faire 4 grandes améliorations.

Premièrement, nous voulons remplacer les fichiers textes par des bases de données.

Deuxièmement, nous voulons que notre programme respecte les règles de la PEP8.

Troisièmement, nous voulons retravailler l'interface graphique.

Pour finir, nous voulons simplifier le code. En effet, il y a certaines parties du code que nous répétons plusieurs fois.

## CONCLUSION

Durant nos séances de Grand Projet, nous avons pu apprendre de nouvelles choses qui nous seront très utiles plus tard. Cela m'a permis de créer un projet sur lequel j'ai pu m'investir, ce fut une super expérience. Notre projet était de concevoir un programme qui répertorie tous les membres de notre association. La programmation fut plutôt facile à mettre en œuvre même si j'ai dû faire face à quelques problèmes. De plus, selon moi, ce projet est loin d'être terminé. Mais au final, le programme marche parfaitement et remplit ses objectifs.

## ANNEXE

Programme de base :

```

from tkinter import*
from inscription import*
from affichageUnMembre import*
from rechercheUnMembre import*
from supprimerUnMembre import*
from statistiqueUnMembre import*

fenetre = Tk()
fenetre.geometry("850x800+0+0")
fenetre.title("Cyber")

background_image=PhotoImage(file="cyber.gif")
background_label = Label(fenetre, image=background_image)
background_label.place(x=0, y=-11, relwidth=1, relheight=1)

button1 = Button(fenetre, text = "Inscription", command = register,width=50 )
button1.place( x=275, y = 300)
button2 = Button(fenetre, text = "Affichage des inscri", command = show,width=50 )
button2.place( x=275, y = 350)
button3 = Button(fenetre, text = "Supprimer un membre", command = delate, width=50 )
button3.place( x=275, y = 400)
button4 = Button(fenetre, text = "Rechercher un membre", command = alpha, width=50 )
button4.place( x=275, y = 450)
button5 = Button(fenetre, text = "Statistique",width=50, command = statistique)
button5.place( x=275, y = 500)
button6 = Button(fenetre, text="Quitter", command = quit,width=50)
button6.place( x=275, y = 550)

```

Programme d'inscription :

```

from tkinter import*
import tkinter as tk
from maBibliotheque import*

T = "0"
G = 1

def register():
    def retour():
        fenetre2.destroy()
    def registeur():

        etat = 0
        compteurLigne = 0

        A = str(entre1.get())
        B = str(entre2.get())
        Cbis = str(entre3.get())
        C = str(homme.get())
        D = str(femme.get())
        E = str(entre5.get())
        F = str(entre6.get())

        G = str(stk.get())
        print(G)
        H = str(katia.get())
        I = str(c.get())
        J = str(photoshop.get())
        K = str(latex.get())

        entre1.delete(0,END)
        entre2.delete(0,END)
        entre3.delete(0,END)
        entre5.delete(0,END)
        entre6.delete(0,END)

        T = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}\n".format(A,B,Cbis,C,D,E,F,G,H,I,J,K)

        FichierControle = open ("inscris.txt", 'r')

```

```

for l in FichierControle.readlines():
    r = l.split(',')
    compteurLigne = compteurLigne+1
    if A == r[0]:
        if B == r[1]:
            nomDuFichier = "inscris.txt"

            F = open('fichierListe.txt','w')
            F.write(T)
            F.write(nomDuFichier)
            F.close()

            choix()

            etat = 1

FichierControle.close()

if etat == 0: ### enregistrer le nouveau membre
    FichierEnregistrement = open("inscris.txt", 'a')
    FichierEnregistrement.write(T)
    FichierEnregistrement.tell()
    FichierEnregistrement.close()

fenetre2.destroy()

fenetre2 = Toplevel()
fenetre2.geometry("280x300+500+200")
fenetre2.title("Fiche d'inscription")

label0 = Label(fenetre2, text = "Fiche d'inscription")
label0.grid(row = 0, column = 1)

```

```

label0 = Label(fenetre2, text = "Fiche d'inscription")
label0.grid(row = 0, column = 1)

label1 = Label(fenetre2, text = "Nom:")
label1.grid(row = 1, column = 0)
label2 = Label(fenetre2, text = "Prénom:")
label2.grid(row = 2, column = 0)
label3 = Label(fenetre2, text = "Année:")
label3.grid(row = 3, column = 0)
label4 = Label(fenetre2, text = "Sexe:")
label4.grid(row = 4, column = 0)
label5 = Label(fenetre2, text = "E mail:")
label5.grid(row = 5, column = 0)
label6 = Label(fenetre2, text = "Téléphone:")
label6.grid(row = 6, column = 0)
label7 = Label(fenetre2, text = "Cours suivi:")
label7.grid(row = 7, column = 0)

entrel = Entry(fenetre2)
entrel.grid(row = 1, column = 1)
entre2 = Entry(fenetre2)
entre2.grid(row = 2, column = 1)
entre3 = Entry(fenetre2)
entre3.grid(row = 3, column = 1)
entre5 = Entry(fenetre2)
entre5.grid(row = 5, column = 1)
entre6 = Entry(fenetre2)
entre6.grid(row = 6, column = 1)

```



```

stk = IntVar()
Checkbutton1 = Checkbutton(fenetre2, text="STK",variable = stk)
Checkbutton1.grid(row = 8, column = 0)
katia = IntVar()
Checkbutton2 = Checkbutton(fenetre2, text="Catia",variable = katia)
Checkbutton2.grid(row = 8, column = 1)
c = IntVar()
Checkbutton3 = Checkbutton(fenetre2, text="C++",variable = c)
Checkbutton3.grid(row = 8, column = 2)
photoshop = IntVar()
Checkbutton4 = Checkbutton(fenetre2, text="Photoshop",variable = photoshop)
Checkbutton4.grid(row = 9, column = 1)
latex = IntVar()
Checkbutton5 = Checkbutton(fenetre2, text="Latex",variable = latex)
Checkbutton5.grid(row = 9, column = 2)

homme = IntVar()
Checkbutton6 = Checkbutton(fenetre2, text="Homme",variable = homme)
Checkbutton6.grid(row = 4, column = 1)
femme = IntVar()
Checkbutton7 = Checkbutton(fenetre2, text="Femme",variable = femme)
Checkbutton7.grid(row = 4, column = 2)

button1 = Button(fenetre2, text="Retour", command = retour)
button1.grid(row = 10, column = 0)
button2 = Button(fenetre2, text="Enregistrer",command = registeur)
button2.grid(row = 10, column = 2)

```

Programme d’affichage:

```

from tkinter import *
from maBibliotheque import*

def show():
    root = Tk()
    root.title("Cyber.affichageDesMembres")

    truc = "normal"
    liste = affichage(truc)

    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]

    def destroy():
        root.destroy()

    def vider():
        text2.delete(1.0,END)
        text3.delete(1.0,END)
        text4.delete(1.0,END)
        text5.delete(1.0,END)
        text6.delete(1.0,END)
        text7.delete(1.0,END)
        text8.delete(1.0,END)

```

```

def STK():
    vider()
    truc = "STK"
    liste = affichage(truc)
    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]
    text2.insert(END, quote1)
    text3.insert(END, quote2, 'color')
    text4.insert(END, quote3, 'color')
    text5.insert(END, quote4, 'color')
    text6.insert(END, quote5, 'color')
    text7.insert(END, quote6, 'color')
    text8.insert(END, quote7, 'color')

def Katia():
    vider()
    truc = "Katia"
    liste = affichage(truc)
    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]
    text2.insert(END, quote1)
    text3.insert(END, quote2, 'color')
    text4.insert(END, quote3, 'color')
    text5.insert(END, quote4, 'color')
    text6.insert(END, quote5, 'color')
    text7.insert(END, quote6, 'color')
    text8.insert(END, quote7, 'color')

```

```

def Photoshop():
    vider()
    truc = "Photochop"
    liste = affichage(truc)
    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]
    text2.insert(END, quote1)
    text3.insert(END, quote2, 'color')
    text4.insert(END, quote3, 'color')
    text5.insert(END, quote4, 'color')
    text6.insert(END, quote5, 'color')
    text7.insert(END, quote6, 'color')
    text8.insert(END, quote7, 'color')

```

```

def Latex():
    vider()
    truc = "Latex"
    liste = affichage(truc)
    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]
    text2.insert(END, quote1)
    text3.insert(END, quote2, 'color')
    text4.insert(END, quote3, 'color')
    text5.insert(END, quote4, 'color')
    text6.insert(END, quote5, 'color')
    text7.insert(END, quote6, 'color')
    text8.insert(END, quote7, 'color')

```

```

def c():
    vider()
    truc = "c"
    liste = affichage(truc)
    quote1 = liste[0]
    quote2 = liste[1]
    quote3 = liste[2]
    quote4 = liste[3]
    quote5 = liste[4]
    quote6 = liste[5]
    quote7 = liste[6]
    text2.insert(END, quote1)
    text3.insert(END, quote2, 'color')
    text4.insert(END, quote3, 'color')
    text5.insert(END, quote4, 'color')
    text6.insert(END, quote5, 'color')
    text7.insert(END, quote6, 'color')
    text8.insert(END, quote7, 'color')

text2 = Text(root, height=30, width=20)
text2.insert(END, quote1,) #on affiche la chaine
text2.grid(row = 1, column = 1)

text3 = Text(root, height=30, width=20)
text3.insert(END, quote2, 'color')
text3.grid(row = 1, column = 2)

```

```

text4 = Text(root, height=30, width=20)
text4.insert(END, quote3, 'color')
text4.grid(row = 1, column = 3)

text5 = Text(root, height=30, width=20)
text5.insert(END, quote4, 'color')
text5.grid(row = 1, column = 4)

text6 = Text(root, height=30, width=30)
text6.insert(END, quote5, 'color')
text6.grid(row = 1, column = 5)

text7 = Text(root, height=30, width=20)
text7.insert(END, quote6, 'color')
text7.grid(row = 1, column = 6)

text8 = Text(root, height=30, width=30)
text8.insert(END, quote7, 'color')
text8.grid(row = 1, column = 7)

button1 = Button(root, text = "STK", width=22, command = STK)
button1.grid(row = 2, column = 2)

button2 = Button(root, text = "Photoshop", width=22, command = Photoshop)
button2.grid(row = 2, column = 3)

button3 = Button(root, text = "C++", width=22, command = c)
button3.grid(row = 2, column = 4)

button4 = Button(root, text = "Catia", width=34, command = Katia)
button4.grid(row = 2, column = 5)

button5 = Button(root, text = "Latex", width=22, command = Latex)
button5.grid(row = 2, column = 6)

button6 = Button(root, text = "Retour", width=22, command = destroy)
button6.grid(row = 2, column = 1)

button7 = Button(root, text = "Vider", width=22, command = vider)
button7.grid(row = 2, column = 7)

root.mainloop()

```

### Programme de suppression:

```

from maBibliotheque import* ### on importe la fonction supress
from tkinter import*
import tkinter as tk

def delate():
    def priseValeur():
        information1 = entre1.get()
        information2 = entre2.get()
        supress(information1, information2)
        label3 = Label(fenetreDel, text = "Personne supprimer de notre fichier")
        label3.grid(row = 3, column = 1)
    def back():
        fenetreDel.destroy()

    fenetreDel = tk.Tk()
    fenetreDel.geometry("450x250+500+200")
    fenetreDel.title("Supprimer un membre")

    label0 = Label(fenetreDel, text = "Veuillez entrer le nom et le prénom de la personne visée")
    label0.grid(row = 0, column = 1)

    label1 = Label(fenetreDel, text = "Nom:")
    label1.grid(row = 1, column = 0)
    label2 = Label(fenetreDel, text = "Prénom:")
    label2.grid(row = 2, column = 0)

    entre1 = Entry(fenetreDel)
    entre1.grid(row = 1, column = 1)
    entre2 = Entry(fenetreDel)
    entre2.grid(row = 2, column = 1)

    entre1 = Entry(fenetreDel)
    entre1.grid(row = 1, column = 1)

    button1 = Button(fenetreDel, text="Retour", command = back)
    button1.grid(row = 10, column = 0)
    button2 = Button(fenetreDel, text="Supprimer", command = priseValeur)
    button2.grid(row = 10, column = 2)

```

### Programme recherche :

```
from tkinter import*
from tkinter.ttk import *
from maBibliotheque import*

information = ["nom","prenom","tel"]

def alpha():
    def retour():
        fenetre.destroy()
    def rechercherUnPersonne():
        value = C1.get()
        premiereInformation = str(e1.get())

        compteur = premierTraitement(value,premiereInformation)

        modifListeComboBox(value)

        if compteur > 1:
            fenetre.destroy
            interfaceSecondTraitement()

        elif compteur == 1 :
            fenetre.destroy()
            Fichier2 = open("recherche.txt",'r')### j'ouvre le fichier dans lequel j'ai enregistrer toutes les correspondances
            information = Fichier2.readline()
            a = information.split(',')
            Fichier2.close()
            infoMembre(a)

        else:
            fenetre.destroy()
            pasDeCorrespondance(premiereInformation)
```

```
fenetre = Tk()
fenetre.geometry("400x250+500+200")
fenetre.title("Recherche")

l1 = Label(fenetre, text = "trouver un membre")
l1.grid(row = 0, column = 1)

C1=Combobox (fenetre,values=information)
C1.grid(row=1,column=0)
e1 = Entry(fenetre)
e1.grid(row = 1, column = 1)
b2 = Button(fenetre, text="Rechercher",command = rechercherUnPersonne)
b2.grid(row = 1, column = 2)

button1 = Button(fenetre, text="Retour", command = retour)
button1.grid(row = 2, column = 0)
```

### Programme statistique :



```

import matplotlib.pyplot as plt
from tkinter import*
import tkinter as tk
from maBibiliotheque import*

nomDeLaPhoto = "stat1.png"

def statistique():
    def goBack():
        fenetre.destroy()
    def a1():
        img = PhotoImage(file="stat1.png")
        canf1 = Canvas(p2, bg = 'blue', width = 500, height = 475)
        canf1.create_image(-80,0,anchor = tk.NW, image = img)
        canf1.image = img
        canf1.grid(column = 2,row= 1)

    def a2():
        img = PhotoImage(file="stat2.png")
        canf1 = Canvas(p2, bg = 'blue', width = 500, height = 475)
        canf1.create_image(-80,0,anchor = tk.NW, image = img)
        canf1.image = img
        canf1.grid(column = 2,row= 1)

    def a3():
        nomDeLaPhoto = "stat3.png"
        img = PhotoImage(file=nomDeLaPhoto)
        canf1 = Canvas(p2, bg = 'blue', width = 500, height = 475)
        canf1.create_image(-80,0,anchor = tk.NW, image = img)
        canf1.image = img
        canf1.grid(column = 2,row= 1)

graph()

```

```

fenetre = Toplevel()
fenetre.geometry("500x580+500+200")
fenetre.title("stat")

p1=PanedWindow(fenetre)
p1.grid(row=0, column=0)
p2=PanedWindow(fenetre)
p2.grid(row=1, column=0)
p3=PanedWindow(fenetre)
p3.grid(row=2, column=0)

button1 = Button(p1, text = "Homme Femme",width=23, command = a1)
button1.grid(row = 1, column = 1)
button2 = Button(p1, text = "Cours suivit",width=23, command = a2)
button2.grid(row = 1, column = 2)
button3 = Button(p1, text = "Pourcentage Aéro",width=23, command = a3 )
button3.grid(row = 1, column = 3)

img = PhotoImage(file=nomDeLaPhoto)
canf1 = Canvas(p2, bg = 'blue', width = 500, height = 475)
canf1.create_image(-80,0,anchor = tk.NW, image = img)
canf1.image = img
canf1.grid(column = 2,row= 1)

button4 = Button(p3, text = "Retour",width=23, command = goBack )
button4.grid(row = 0, column = 0)

```

## Ma bibliothèque (maBibliotheque) :

```
from tkinter import *
from tkinter.ttk import *
import tkinter as tk
import matplotlib.pyplot as plt

def suppress(information1,information2):
    F = open("inscris.txt","r")
    maListe = []
    for l in F.readlines():
        ligne = l.split(",")
        maListe.append(ligne)
    for i in range(0,len(maListe)):
        ligne = maListe[i]
        if information1 == ligne[0]:
            if information2 == ligne[1]:
                print(ligne)
                maListe.remove(maListe[i]) ### on supprime la ligne que l'on veut supprimer dans notre liste
                break
    F.close()
    F2 = open("inscris.txt","w")### on réécrit notre fichier avec notre nouvelle liste
    for i in range (0,len(maListe)):
        T = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}".format(maListe[i][0],maListe[i][1],
                                                                    maListe[i][2],maListe[i][3],maListe[i][4],
                                                                    maListe[i][5],maListe[i][6],maListe[i][7],
                                                                    maListe[i][8],maListe[i][9],maListe[i][10],
                                                                    maListe[i][11])
        F2.write(T)
        print(T)
    F2.close()
```

```
def premierTraitement(value,premiereInformation): ### Cela permet de faire un premier trie dans notre fichier
    compteur = 0
    F = open ("inscris.txt", 'r')
    F2 = open ("recherche.txt", 'w')
    for l in F.readlines():
        listel = l.split(',')
        if value == "nom":
            if premiereInformation == listel[0]:
                F2.write(l)
                compteur = compteur + 1
        elif value == "prenom":
            if premiereInformation == listel[1]:
                F2.write(l)
                compteur = compteur + 1
        else:
            if premiereInformation == listel[2]:
                F2.write(l)
                compteur = compteur + 1
    F2.close()
    F.close()
    return(compteur)

def modifierListeComboBox(value): ### Cela permet de modifier les éléments que contient notre comboBox
    information = ["nom","prenom","tel"]

    if value == "nom":
        information.remove(information[0])
    elif value == "prenom":
        information.remove(information[1])
    else:
        information.remove(information[2])
    FichierListe = open ("fichierListe.txt", 'w')
    T = "{0},{1}\n".format(information[0],information[1])
    FichierListe.write(T)
    FichierListe.close()
```

```

def infoMembre(a):
    def back():
        fenetre2.destroy()
    def modifier():
        A = str(entre1.get())
        B = str(entre2.get())
        Cbis = str(entre3.get())
        C = str(homme.get())
        D = str(femme.get())
        E = str(entre5.get())
        F = str(entre6.get())
        G = str(stk.get())
        H = str(katia.get())
        I = str(c.get())
        J = str(photoshop.get())
        K = str(latex.get())
        T = "(0),(1),(2),(3),(4),(5),(6),(7),(8),(9),(10),(11)\n".format(A,B,Cbis,C,D,E,F,G,H,I,J,K)
        F = open("inscris.txt","r")
        maliste = []
        for l in F.readlines():
            ligne = l.split(",")
            maliste.append(ligne)
        for i in range(0,len(maliste)):
            ligne = maliste[i]
            if A == ligne[0]:
                if B == ligne[1]:
                    maliste.remove(maliste[i]) ### on supprime la ligne que l'on veut supprimer dans notre liste
                    break
        F.close()
        F2 = open("inscris.txt","w")### on réécrit notre fichier avec notre nouvelle liste
        F2.write(T)
        for i in range (0,len(maliste)):
            T2 = "(0),(1),(2),(3),(4),(5),(6),(7),(8),(9),(10),(11)".format(maliste[i][0],maliste[i][1],
                                                                           maliste[i][2],maliste[i][3],maliste[i][4],
                                                                           maliste[i][5],maliste[i][6],maliste[i][7],
                                                                           maliste[i][8],maliste[i][9],maliste[i][10],
                                                                           maliste[i][11])
            F2.write(T2)
        F2.close()

```

```

fenetre2 = Toplevel()
label0 = Label(fenetre2, text = "Fiche d'inscription")
label0.grid(row = 0, column = 1)

label1 = Label(fenetre2, text = "Nom:")
label1.grid(row = 1, column = 0)
label2 = Label(fenetre2, text = "Prénom:")
label2.grid(row = 2, column = 0)
label3 = Label(fenetre2, text = "Année:")
label3.grid(row = 3, column = 0)
label4 = Label(fenetre2, text = "Sexe:")
label4.grid(row = 4, column = 0)
label5 = Label(fenetre2, text = "E mail:")
label5.grid(row = 5, column = 0)
label6 = Label(fenetre2, text = "Téléphone:")
label6.grid(row = 6, column = 0)
label7 = Label(fenetre2, text = "Cours suivi:")
label7.grid(row = 7, column = 0)

entre1 = Entry(fenetre2)
entre1.grid(row = 1, column = 1)
entre2 = Entry(fenetre2)
entre2.grid(row = 2, column = 1)
entre3 = Entry(fenetre2)
entre3.grid(row = 3, column = 1)
entre5 = Entry(fenetre2)
entre5.grid(row = 5, column = 1)
entre6 = Entry(fenetre2)
entre6.grid(row = 6, column = 1)

stk = IntVar()
Checkbutton1 = Checkbutton(fenetre2, text="STK",variable = stk)
Checkbutton1.grid(row = 8, column = 0)
katia = IntVar()
Checkbutton2 = Checkbutton(fenetre2, text="Katia",variable = katia)
Checkbutton2.grid(row = 8, column = 1)
c = IntVar()
Checkbutton3 = Checkbutton(fenetre2, text="C++",variable = c)
Checkbutton3.grid(row = 8, column = 2)
photoshop = IntVar()
Checkbutton4 = Checkbutton(fenetre2, text="Photoshop",variable = photoshop)
Checkbutton4.grid(row = 9, column = 1)
latex = IntVar()
Checkbutton5 = Checkbutton(fenetre2, text="Latex",variable = latex)
Checkbutton5.grid(row = 9, column = 2)

```

```

homme = IntVar()
Checkbutton6 = Checkbutton(fenetre2, text="Homme",variable = homme)
Checkbutton6.grid(row = 4, column = 1)
femme = IntVar()
Checkbutton7 = Checkbutton(fenetre2, text="Femme",variable = femme)
Checkbutton7.grid(row = 4, column = 2)

button1 = Button(fenetre2, text="Retour", command = back)
button1.grid(row = 10, column = 0)
button2 = Button(fenetre2, text="Modifier", command = modifier)
button2.grid(row = 10, column = 2)

entre1.insert(0,a[0])
entre2.insert(0,a[1])
entre3.insert(0,a[2])
entre5.insert(0,a[5])
entre6.insert(0,a[6])

if a[3] == str(1):
    homme.set(True)
if a[4] == str(1):
    femme.set(True)
if a[7] == str(1):
    stk.set(True)
if a[8] == str(1):
    katia.set(True)
if a[9] == str(1):
    c.set(True)
if a[10] == str(1):
    photoshop.set(True)
if a[11] == str(1):
    latex.set(True)

```

```

def true():

def secondTraitement():
    value = C2.get()
    secondeInformation = str(e2.get())
    compteur2 = 0
    Fichier3 = open("recherche.txt","r")
    for l in Fichier3.readlines():
        liste1 = l.split(',')
        if value == "nom":
            if secondeInformation == liste1[0]:
                a = liste1
                compteur2 = compteur2 + 1
                break
            elif value == "prenom":
                if secondeInformation == liste1[1]:
                    a = liste1
                    compteur2 = compteur2 + 1
                    break
            else:
                if secondeInformation == liste1[2]:
                    a = liste1
                    compteur2 = compteur2 + 1
                    break
    Fichier3.close()
    if compteur2 == 1:
        infoMembre(a)
    else:
        ligne1 = "Nous n'avons aucune correspondance avec cette nouvelle information"
        label10.config(text="%s" %ligne1)
        compteur = 0

FichierListe = open ("fichierListe.txt", 'r')
a = FichierListe.readline()
information=a.split(',')
FichierListe.close()

```

```

fenetre3 = Tk()

fenetre3V1=PanedWindow(fenetre3)
fenetre3V1.grid(row=0, column=0)
fenetre3V2=PanedWindow(fenetre3)
fenetre3V2.grid(row=1, column=0)

l1 = Label(fenetre3V1, text = "Veuillez entrer une autre information sur la personne recherché")
l1.grid(row = 0, column = 0)

C2=Combobox (fenetre3V2,values=information)
C2.grid(row=1,column=0)
e2 = Entry(fenetre3V2)
e2.grid(row = 1, column = 1)
b2 = Button(fenetre3V2, text="Rechercher", command = secondTraitement)
b2.grid(row = 1, column = 2)

b3 = Button(fenetre3V2, text="Quitter", command = quit)
b3.grid(row = 10, column = 0)

label0 = Label(fenetre3V2, text = "...")
label0.grid(row = 2, column = 1)

```

```

def false():
    fenetre3 = Tk()
    quote1 = """ """
    quote2 = """ """
    quote3 = """ """
    quote4 = """ """
    Fichier = open ("recherche.txt", 'r')
    for l in Fichier.readlines(): #le programme lit le fichier ligne par ligne
        a = l.split(',')#la ligne devient une liste
        b = a[0]
        c = a[1]
        d = a[2]
        quote1 = quote1 + b + "\n"#on ajoute la première valeur de la liste
        quote2 = quote2 + c + "\n"
        quote3 = quote3 + d + "\n"
    Fichier.close()

    text2 = Text(fenetre3, height=30, width=20)
    text2.insert(END, quote1, 'color')#on affiche la chaine de caractère dans la partie text2 de l'interface graphique
    text2.grid(row = 1, column = 1)

    text3 = Text(fenetre3, height=30, width=20)
    text3.insert(END, quote2, 'color')
    text3.grid(row = 1, column = 2)

    text4 = Text(fenetre3, height=30, width=20)
    text4.insert(END, quote3, 'color')
    text4.grid(row = 1, column = 3)

```

```

def interfaceSecondTraitement():
    fenetre2 = Tk()

    label0 = Label(fenetre2, text = "Avez vous une autre information sur la personne recherché?")
    label0.grid(row = 0, column = 1)

    button1 = Button(fenetre2, text="OUI", command = true)
    button1.grid(row = 1, column = 0)
    button2 = Button(fenetre2, text="NON",command = false)
    button2.grid(row = 1, column = 2)

def pasDeCorrespondance(premiereInformation):

    fenetre2 = Tk()

    label1 = Label(fenetre2, text = "...")
    label1.grid(row = 0, column = 1)
    label2 = Label(fenetre2, text = "...")
    label2.grid(row = 1, column = 1)

    texte1 = "nous n'avons aucune personne qui se nomme "
    texte2 = " dans notre répertoire"

    texte3 = "Vous avez peut être fait une faute d'orthographe en rentrant le nom"

    ligne1 = texte1+str(premiereInformation)+texte2
    label1.config(text="%s" %ligne1)
    label2.config(text="%s" %texte3)

    button1 = Button(fenetre2, text="Quitter", command = quit)
    button1.grid(row = 2, column = 0)
    button2 = Button(fenetre2, text="Retour")
    button2.grid(row = 2, column = 2)

```



```

def affichage(truc):
    quote1 = ""
    quote2 = ""
    quote3 = ""
    quote4 = ""
    quote5 = ""
    quote6 = ""
    quote7 = ""

    Fichier = open("inscris.txt", 'r')
    for l in Fichier.readlines(): #le programme lit le fichier ligne par ligne
        cours = str()
        sexe = str()
        r = l.split(',') #la ligne devient une liste
        nom = r[0]
        prenom = r[1]
        promo = r[2]
        if r[3] == str(1):
            sexe = str("homme")
        if r[4] == str(1):
            sexe = str("femme")
        mail = r[5]
        tel = r[6]
        if r[7] == str(1):
            cours = cours + str("STK")
        if r[8] == str(1):
            cours = cours + str(",Catia")
        if r[9] == str(1):
            cours = cours + str(",C++")
        if r[10] == str(1):
            cours = cours + str(",Photoshop")
        if r[11] == str(1):
            cours = cours + str(",Latak")

        if truc == "normal":
            quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
            quote2 = quote2 + prenom + "\n"
            quote3 = quote3 + promo + "\n"
            quote4 = quote4 + sexe + "\n"
            quote5 = quote5 + mail + "\n"
            quote6 = quote6 + tel + "\n"
            quote7 = quote7 + cours + "\n"

```

```

        if truc == "STK":
            if r[7] == str(1):
                quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
                quote2 = quote2 + prenom + "\n"
                quote3 = quote3 + promo + "\n"
                quote4 = quote4 + sexe + "\n"
                quote5 = quote5 + mail + "\n"
                quote6 = quote6 + tel + "\n"
                quote7 = quote7 + cours + "\n"

        if truc == "Katia":
            if r[8] == str(1):
                quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
                quote2 = quote2 + prenom + "\n"
                quote3 = quote3 + promo + "\n"
                quote4 = quote4 + sexe + "\n"
                quote5 = quote5 + mail + "\n"
                quote6 = quote6 + tel + "\n"
                quote7 = quote7 + cours + "\n"

        if truc == "c":
            if r[9] == str(1):
                quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
                quote2 = quote2 + prenom + "\n"
                quote3 = quote3 + promo + "\n"
                quote4 = quote4 + sexe + "\n"
                quote5 = quote5 + mail + "\n"
                quote6 = quote6 + tel + "\n"
                quote7 = quote7 + cours + "\n"

        if truc == "Photochop":
            if r[10] == str(1):
                quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
                quote2 = quote2 + prenom + "\n"
                quote3 = quote3 + promo + "\n"
                quote4 = quote4 + sexe + "\n"
                quote5 = quote5 + mail + "\n"
                quote6 = quote6 + tel + "\n"
                quote7 = quote7 + cours + "\n"

        if truc == "Latex":
            if r[11] == str(1):
                quote1 = quote1 + nom + "\n" #on ajoute la première valeur de la liste
                quote2 = quote2 + prenom + "\n"
                quote3 = quote3 + promo + "\n"
                quote4 = quote4 + sexe + "\n"
                quote5 = quote5 + mail + "\n"
                quote6 = quote6 + tel + "\n"
                quote7 = quote7 + cours + "\n"

Fichier.close()

```

```

Fichier.close()
return(quote1,quote2,quote3,quote4,quote5,quote6,quote7)

def modifierV2():
    recupDeValeur = []
    F = open('fichierListe.txt','r')
    for i in F.readlines():
        recupDeValeur.append(i)
    F.close()
    T = recupDeValeur[0]
    nomDuFichier = recupDeValeur[1]

    liste1 = T.split(',')
    maListe = []
    F = open(nomDuFichier,'r')
    for l in F.readlines():
        ligne = l.split(",")
        maListe.append(ligne)
    for i in range(0,len(maListe)):
        ligne = maListe[i]
        if liste1[0] == ligne[0]:
            if liste1[1] == ligne[1]:
                maListe.remove(maListe[i]) ### on supprime la ligne que l'on veut supprimer dans notre liste
                break
    F.close()

    F2 = open(nomDuFichier,"w")### on réécrit notre fichier avec notre nouvelle liste
    F2.write(T)
    for i in range(0,len(maListe)):
        T2 = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}".format(maListe[i][0],maListe[i][1],
                                                                    maListe[i][2],maListe[i][3],maListe[i][4],
                                                                    maListe[i][5],maListe[i][6],maListe[i][7],
                                                                    maListe[i][8],maListe[i][9],maListe[i][10],
                                                                    maListe[i][11])
        F2.write(T2)
    F2.close()

```

```

def modifNom():
    liste = []
    F = open('fichierListe.txt','r')
    L = F.readline()
    F.close()
    recupDeValeur = L.split(',')
    print(recupDeValeur)

    def modifierV3():
        newName = entre666.get()
        fenetreJeNeSaisPlusCombien.destroy()
        recupDeValeur[1] = newName
        T2 = "{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11}".format(recupDeValeur[0],recupDeValeur[1],
                                                                    recupDeValeur[2],recupDeValeur[3],
                                                                    recupDeValeur[4],recupDeValeur[5],
                                                                    recupDeValeur[6],recupDeValeur[7],
                                                                    recupDeValeur[8],recupDeValeur[9],
                                                                    recupDeValeur[10],recupDeValeur[11])

        F = open('inscris.txt','a')
        F.write(T2)
        F.close()

    fenetreJeNeSaisPlusCombien = Tk()
    fenetreJeNeSaisPlusCombien.geometry("550x250+500+200")
    label0 = Label(fenetreJeNeSaisPlusCombien, text = "Pouvez vous modifier le prénom")
    label0.grid(row = 0, column = 1)

    entre666 = Entry(fenetreJeNeSaisPlusCombien)
    entre666.grid(row = 1, column = 1)

    entre666.insert(0,recupDeValeur[1])

    button1 = Button(fenetreJeNeSaisPlusCombien, text="Enregistrer",command = modifierV3)
    button1.grid(row = 2, column = 1)

```

```

def choix():
    def retour():
        fenetre2.destroy()

    recupDeValeur = []
    F = open('fichierListe.txt','r')
    for i in F.readlines():
        recupDeValeur.append(i)
    F.close()

    L = recupDeValeur[0]
    r = L.split(',')

    fenetre2 = Tk()
    fenetre2.geometry("550x250+500+200")

    texte = "il y a déjà une personne qui se nomme "+r[0]," ",r[1]

    label0 = Label(fenetre2, text = texte)
    label0.grid(row = 0, column = 1)

    button1 = Button(fenetre2, text="Modifier la base de donnée",command = modifierV2)
    button1.grid(row = 1, column = 1)
    button2 = Button(fenetre2, text="Modifier le prénom que vous voulez entrer", command = modifNom)
    button2.grid(row = 2, column = 1)

    buttonRetour = Button(fenetre2, text="Retour",command = retour)
    buttonRetour.grid(row = 3, column = 0)

```

```

def graph():
    nombreHomme = 0
    nombreFemme = 0
    nombreSTK = 0
    nombreCatia = 0
    nombreC = 0
    nombrePhotoshop = 0
    nombrelatek = 0
    nombreAero1 = 0
    nombreAero2 = 0
    nombreAero3 = 0
    nombreAero4 = 0
    nombreAero5 = 0

    Fichier = open("inscris.txt", 'r')
    for l in Fichier.readlines():
        liste = l.split(',')
        if liste[3] == str(1):
            nombreHomme = nombreHomme +1
        if liste[4] == str(1):
            nombreFemme = nombreFemme +1
        if liste[7] == str(1):
            nombreSTK = nombreSTK + 1
        if liste[8] == str(1):
            nombreCatia = nombreCatia + 1
        if liste[9] == str(1):
            nombreC = nombreC + 1
        if liste[10] == str(1):
            nombrePhotoshop = nombrePhotoshop + 1
        if liste[11] == str(1):
            nombrelatek = nombrelatek + 1
        if liste[2] == "aéro1":
            nombreAero1 = nombreAero1 + 1
        if liste[2] == "aéro2":
            nombreAero2 = nombreAero2 + 1
        if liste[2] == "aéro3":
            nombreAero3 = nombreAero3 + 1
        if liste[2] == "aéro4":
            nombreAero4 = nombreAero4 + 1
        if liste[2] == "aéro5":
            nombreAero5 = nombreAero5 + 1
    Fichier.close()

```

```

name = ['Homme', 'Femme']
data = [nombreHomme,nombreFemme]
explode=(0, 0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat1.png')
plt.close()
plt.show()

name = ['STK', 'CATIA', 'C++', 'Photoshop', 'latek']
data = [nombreSTK,nombreCatia,nombrePhotoshop,nombreC, nombrelatek]
explode=(0,0,0,0,0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat2.png')
plt.close()
plt.show()

name = ['Aéro1', 'Aéro2', 'Aéro3', 'Aéro4', 'Aéro5']
data = [nombreAero1,nombreAero2,nombreAero3,nombreAero4, nombreAero5]
explode=(0,0,0,0,0)
plt.pie(data, explode=explode, labels=name, autopct='%1.1f%%', startangle=90, shadow=True)
plt.axis('equal')
plt.savefig('stat3.png')
plt.close()
plt.show()

```

## RESUME :

Comme nous venons de la voir dans ce compte rendu, le langage python est très simple et permet de réaliser de grandes choses. Notre programme est composé d'une interface graphique qui permet à l'opérateur d'utiliser notre programme. Il s'agit de la partie visible du programme. La partie invisible de notre programme est la partie algorithmique. Cette partie utilise des algorithmes plus ou moins complexes qui permettent au programme de répondre au besoin de l'utilisateur. Le programme que nous avons conçu répond au cahier des charges que nous avons fixé et est opérationnel. Pour finir, le projet que nous venons de réaliser nous a permis d'apprendre de nombreuses choses et nous a donné envie de continuer à découvrir le monde de la programmation.

Mot clés: Python3, Tkinter, checkBoxs, comboBox, entrey, fichier.txt, Button, Text

As we just saw in this review, the Python language is very simple and allows you to do great things. Our program consists of a graphical interface that allows the operator to use our program. This is the visible part of the program. The invisible part of our program is the algorithmic part. This part uses more or less complex algorithms that allow the program to meet the needs of the user. The program we have designed meets the specifications we have set and is operational. Finally, the project that we have just realized has allowed us to learn many things and made us want to continue to discover the world of programming.

Key words: Python3, Tkinter, checkBoxs, comboBox, entrey, fichier.txt, Button, Text

## SOURCE:

<https://python-django.dev/page-creer-graphiques-scientifiques-python-apprendre>

<http://tkinter.fdex.eu/doc/toplww.html>

<https://openclassrooms.com/fr/courses/235344-apprenez-a-programmer-en-python/234859-creez-des-interfaces-graphiques-avec-tkinter>

<https://python-django.dev/page-tkinter-interface-graphique-python-tutoriel>

<http://tkinter.fdex.eu/>

[https://www.python-course.eu/tkinter\\_text\\_widget.php](https://www.python-course.eu/tkinter_text_widget.php)

<https://openclassrooms.com/fr/courses/235344-apprenez-a-programmer-en-python/232431-utilisez-des-fichiers>