

# Tp9 r1.1

Programmez l'exercice du TD 8. Rappel de l'énoncé :

*On veut écrire une application permettant de gérer une bibliothèque personnelle. Pour chaque livre, on a besoin des informations suivantes :*

- le titre
- le nom de l'auteur
- l'année de parution
- le nombre de pages

1- Proposez une structure de donnée adaptée à la gestion d'un nouveau type : le type **Livre**

2- Écrivez un programme proposant le menu suivant :

- afficher l'ensemble des livres de la bibliothèque
- ajouter un nouveau livre
- rechercher un livre (par le titre)
- quitter

Pensez à utiliser / créer vos propres modules

Vous pouvez utiliser les "listes" de Python directement. Rappel du cours :

```
import random

if __name__ == "__main__":
    li : list[int]
    li = [] #crée une liste vide

    for i in range(0,10) :
        li.append(random.randint(0, 100)) #ajoute un nombre en fin de liste

    print(li[len(li)-1]) #affiche la dernière valeur de la liste
```

# programme

```
class livre:
    titre: str
    nomAuteur: str
    anneeParution: int
    nbPages: int

menu_options = {
    1 : "Afficher l'ensemble des livres de la bibliothèque",
    2 : "Ajouter un nouveau livre",
    3 : "Rechercher un livre (par le titre)",
    4 : "Quitter"
}

def print_menu():
    """
    Affiche le menu des options
    entrées: aucune
    sorties: aucune
    """
    for key in menu_options.keys():
        print(key, '-- ', menu_options[key])

def afficher_livres(table: list[livre], n: int):
    """
    procédure permettant d'afficher les livres de la bibliothèque de livres renvoie un message d'erreur si le tableau est vide
    entrées: table(list[livre]) : tableau de livres, n(int) : nombre de livres dans la bibliothèque
    sorties: aucune
    """
    if n == 0: # Si le tableau est vide on affiche un message
        print("La bibliothèque est vide")
        return
    for livre in table: # Parcours le tableau et affiche les livres
        print("Titre: ", livre.titre)
        print("Auteur: ", livre.nomAuteur)
        print("Année de parution: ", livre.anneeParution)
        print("Nombre de pages: ", livre.nbPages)
        print("-----") # Ligne vide pour séparer les livres
```

Je commence dans mon programme par créer une **classe Livre** pour me permettre de l'utiliser par la suite dans mon tableau de livres.

Ce dernier est composé de:

- ★ **titre** : entier
- ★ **nomAuteur** : chaîne
- ★ **annéeParution** : int
- ★ **nbPages** : int

Par la suite, je défini mes options de menu dans une **bibliothèque** qui se fait afficher grave à la procédure **print\_menu**.

Je définis la première **procédure** à utiliser dans mon programme. **L'affichage des livres** ne renvoyant rien est considéré comme un procédure de ce fait. Ici cette dernière permet d'**afficher les livres présents dans la bibliothèque de livre** et si cette dernière est vide un **message d'erreur** s'affiche indiquant à l'utilisateur qu'il n'y a pas de livres à afficher.

```

def ajouter_livre(table: list[livre], n:int):
    """
    fonction permettant d'ajouter un livre à la bibliothèque
    entrées: table(list[livre]) : tableau de livres, n(int) : nombre de livres dans la bibliothèque
    sorties: n(int) : nombre de livres dans la bibliothèque indenté de 1
    """
    nouveau_livre = livre() # Création d'un nouveau livre
    nouveau_livre.titre = input("Entrez le titre du livre: ")
    nouveau_livre.nomAuteur = input("Entrez le nom de l'auteur: ")
    nouveau_livre.anneeParution = int(input("Entrez l'année de parution: "))
    nouveau_livre.nbPages = int(input("Entrez le nombre de pages: "))
    table.append(nouveau_livre) # Ajout du livre au tableau
    n+=1 # Incrémentation du nombre de livres
    return n

def rechercher_livre(table: list[livre]):
    """
    fonction permettant de rechercher un livre dans la bibliothèque par son titre
    entrées: table(list[livre]) : tableau de livres
    sorties: id(int) : position du livre dans le tableau
    """
    titre = input("Entrez le titre du livre à rechercher: ") # Demande le titre du livre à rechercher
    for id in range(len(table)): # Parcours le tableau
        if table[id].titre == titre: # Si le titre du livre est trouvé on affiche le livre
            print("Titre: ", table[id].titre)
            print("Auteur: ", table[id].nomAuteur)
            print("Année de parution: ", table[id].anneeParution)
            print("Nombre de pages: ", table[id].nbPages)
            print("-----")
            return id
    return -1 # Si le livre n'est pas trouvé, retourne -1

```

Par la suite, j'ai défini une **fonction** permettant de faire **saisir un livre dans ma bibliothèque** à l'utilisateur. Chaque **caractéristique du livre est saisie** par l'utilisateur et à la fin de sa saisie le **nombre de livres de la bibliothèque est incrémenté de 1** et ce dernier est **retourné dans le programme principal**.

Et pour finir dans mes fonctions et procédures, ma dernière **fonction** permet de **chercher un livre par son titre** dans la bibliothèque. Si ce **livre est trouvé on retourne sa place** dans la bibliothèque à l'utilisateur et les **autres informations** de ce livre. **Sinon un -1 est renvoyé** au programme affichant un **message d'erreur**.

```

if __name__ == '__main__':
    table: list[livre] = [] # Déclaration du tableau
    n:int
    n=0
    place: int
    option = str # Déclaration de la variable pour l'option choisie

    while True:
        print_menu() # Affiche le menu
        option = ''
        try:
            option = input("Entrez votre choix: ") # Lit le choix de l'utilisateur
        except:
            print("Entrez un nombre valide")

        if option == '1':
            afficher_livres(table,n)
        elif option == '2':
            n=ajouter_livre(table,n)
        elif option == '3':
            place=rechercher_livre(table)
            if place == -1:
                print("Le livre n'est pas dans la bibliothèque")
            else:
                print("Le livre se trouve à la place ", place+1 )

        elif option == '4':
            break

```

Le programme principal lui gère majoritairement les **options du menu** de plus du **retour négatif de la recherche de livre**.

## jeux d'essais

```

1 -- Afficher l'ensemble des livres de la bibliothèque
2 -- Ajouter un nouveau livre
3 -- Rechercher un livre (par le titre)
4 -- Quitter
Entrez votre choix: 1
La bibliothèque est vide

```

Ici nous voyons la gestion d'une **bibliothèque vide** renvoyant le **message d'erreur** à l'utilisateur.

```

Entrez votre choix: 2
Entrez le titre du livre: pride and prejudice
Entrez le nom de l'auteur: Austen
Entrez l'année de parution: 1813
Entrez le nombre de pages: 342
1 -- Afficher l'ensemble des livres de la bibliothèque
2 -- Ajouter un nouveau livre
3 -- Rechercher un livre (par le titre)
4 -- Quitter
Entrez votre choix: 2
Entrez le titre du livre: Frankenstein
Entrez le nom de l'auteur: Sheiley
Entrez l'année de parution: 1818
Entrez le nombre de pages: 352
1 -- Afficher l'ensemble des livres de la bibliothèque
2 -- Ajouter un nouveau livre
3 -- Rechercher un livre (par le titre)
4 -- Quitter
Entrez votre choix: 2
Entrez le titre du livre: The Bloody Chamber
Entrez le nom de l'auteur: Carter
Entrez l'année de parution: 1979
Entrez le nombre de pages: 176

```

Nous traitons dans ce jeu d'essai **l'ajout de trois livres** avec leurs propres caractéristiques.

```

Entrez votre choix: 1
Titre: pride and prejudice
Auteur: Austen
Année de parution: 1813
Nombre de pages: 342
-----
Titre: Frankenstein
Auteur: Sheiley
Année de parution: 1818
Nombre de pages: 352
-----
Titre: The Bloody Chamber
Auteur: Carter
Année de parution: 1979
Nombre de pages: 176
-----

```

Ici nous avons **l'affichage de la bibliothèque** quand elle n'est pas vide.

```
Entrez votre choix: 3
Entrez le titre du livre à rechercher: The Bloody Chamber
Titre: The Bloody Chamber
Auteur: Carter
Année de parution: 1979
Nombre de pages: 176
Le livre se trouve à la place 3
```

La **recherche d'un livre présent** dans la bibliothèque et l'**affichage des informations** de ce dernier en plus de son **emplacement dans le tableau**.

```
1 -- Afficher l'ensemble des livres de la bibliothèque
2 -- Ajouter un nouveau livre
3 -- Rechercher un livre (par le titre)
4 -- Quitter
Entrez votre choix: 3
Entrez le titre du livre à rechercher: zfbjh
Le livre n'est pas dans la bibliothèque
```

Pour finir la **recherche d'un livre non présent** dans la bibliothèque **renvoyant un message d'erreur** à l'utilisateur.