TD N°11 : tri, fusion de listes

1. **Fusion de deux listes**
	1. Définir une structure (dictionnaire) représentant une personne et contenant le nom, l’âge et le sexe.
	2. Écrire un programme qui à partir de deux listes de personnes ordonnées suivant le nom crée une troisième liste ordonnée suivant le nom des personnes des deux premières listes.

TP  N°11: Le jeu du tri

Le but de ce TP est d’implémenter les différents algorithmes de tri vu en cours sur une liste désordonnée de dictionnaires. Chaque tri pourra être fait selon une clé saisie par l’utilisateur (nom de l’artiste, titre de l’album, année de sortie, durée calculée au préalable – voir script donné ci-dessous -). Les fonctions renvoient une liste triée.

* Tri par sélection (liste, cle)
* Tri par insertion (liste, cle)
* Tri à bulles (liste, cle)
* Tri rapide (liste, cle) (facultatif)

Vous pourrez utiliser la liste créée au TP 9 à partir de votre fichier ou utiliser cette base créée par ChatGPT.

# Liste de dictionnaires représentant des albums de musique
albums = [
 {
 "artiste": "The Beatles",
 "titre\_album": "Abbey Road",
 "annee\_sortie": 1969,
 "morceaux": [
 {"titre\_morceau": "Come Together", "duree": 260},
 {"titre\_morceau": "Something", "duree": 183},
 {"titre\_morceau": "Maxwell's Silver Hammer", "duree": 207}
 ]
 },
 {
 "artiste": "Pink Floyd",
 "titre\_album": "The Dark Side of the Moon",
 "annee\_sortie": 1973,
 "morceaux": [
 {"titre\_morceau": "Speak to Me", "duree": 90},
 {"titre\_morceau": "Breathe", "duree": 163},
 {"titre\_morceau": "Time", "duree": 413}
 ]
 },
 {
 "artiste": "Nirvana",
 "titre\_album": "Nevermind",
 "annee\_sortie": 1991,
 "morceaux": [
 {"titre\_morceau": "Smells Like Teen Spirit", "duree": 301},
 {"titre\_morceau": "In Bloom", "duree": 255},
 {"titre\_morceau": "Come as You Are", "duree": 219}
 ]
 },
 {
 "artiste": "Michael Jackson",
 "titre\_album": "Thriller",
 "annee\_sortie": 1982,
 "morceaux": [
 {"titre\_morceau": "Wanna Be Startin' Somethin'", "duree": 363},
 {"titre\_morceau": "Thriller", "duree": 357},
 {"titre\_morceau": "Beat It", "duree": 258}
 ]
 },
 {
 "artiste": "Led Zeppelin",
 "titre\_album": "Led Zeppelin IV",
 "annee\_sortie": 1971,
 "morceaux": [
 {"titre\_morceau": "Black Dog", "duree": 295},
 {"titre\_morceau": "Rock and Roll", "duree": 220},
 {"titre\_morceau": "Stairway to Heaven", "duree": 482}
 ]
 }
]

# Calcul de la durée des albums
def calculDuree(albums) :
 for album in albums:
 dureeTotale = 0
 for morceau in album['morceaux']:
 dureeTotale += morceau["duree"]
 album["duree\_totale"] = dureeTotale

# Affichage de la liste des albums pour vérification
def affichageAlbums(albums) :
 for album in albums:
 print(f"Artiste: {album['artiste']}")
 print(f"Titre de l'album: {album['titre\_album']}")
 print(f"Année de sortie: {album['annee\_sortie']}")
 if "duree\_totale" in album.keys():
 print(f"Duree Totale: {album['duree\_totale']}")
 print("Morceaux:")
 for morceau in album['morceaux']:
 print(f" {morceau['titre\_morceau']} : {morceau['duree']}")
 print()

affichageAlbums(albums)
print('----------------------------')
calculDuree(albums)
print('----------------------------')
affichageAlbums(albums)

Facultatif : vérification du générateur de nombres aléatoires

Écrire un programme permettant de tirer aléatoirement N valeurs entre 0 et 1, de diviser l’intervalle [0, 1] en NbInt intervalles égaux et de compter le nombre de valeurs appartenant à chacun des intervalles.

Si le générateur fonctionne correctement les valeurs devraient être réparties de façon uniforme entre les différents intervalles.