

# Résumé du travail rendu par Claude

---

Le programme est complet et fonctionnel. Voici un résumé de ce qui est implémenté pour chaque étape du sujet.

## Structure de données

Deux listes Python — `personnes` (sous-listes `[nom, prénom, sexe, date]`) et `liens` (sous-listes `[num_enfant, num_parent]`).

## Les 10 étapes

1. `ajouter_personne()` — construit et appende la sous-liste, retourne l'indice attribué
2. `afficher_personnes()` — parcours simple avec formatage tabulaire
3. `selectionner_personne()` — recherche linéaire nom+prénom, retourne l'indice ou -1
4. `ajouter_lien()` — appende le couple d'entiers dans liens
5. `obtenir_ascendants()` — remontée récursive, retourne une liste d'entiers sans doublons
6. `obtenir_descendants()` — descente récursive, même principe
7. `obtenir_freres_soeurs()` — trouve les parents communs puis leurs enfants
8. `afficher_par_ordre_alpha()` — tri par insertion sur (nom, prénom)
9. `afficher_par_age()` — tri par insertion sur la date convertie en `AAAAMMJJ`
10. `menu_principal()` + fonctions `action_*` — boucle menu avec saisie validée, séparant bien lecture/traitement/affichage

## Bonnes pratiques respectées

Commentaires, doc de chaque paramètre et du type retourné, pas d'entrées/sorties dans les fonctions de traitement, `return` unique en fin de fonction, aucun `return` dans une boucle.