### Exercice : Gestionnaire de Contacts en Python avec Persistance dans un Fichier Texte

1. **Création de la classe** Contact **:**
   * Créez une classe Contact avec les attributs nécessaires pour représenter un contact (nom, prénom, email, téléphone).
   * Implémentez un constructeur (\_\_init\_\_) pour initialiser ces attributs.
   * Créez les getters et setters en utilisant des propriétés pour chacun des attributs.
   * Ajoutez une méthode \_\_str\_\_() pour afficher les informations du contact sous forme de chaîne de caractères.
2. **Gestion des Contacts en Mémoire :**
   * Créez une classe GestionnaireContact qui gère une liste de contacts en mémoire.
   * Implémentez une méthode ajouter\_contact() pour ajouter un nouveau contact à la liste.
   * Implémentez une méthode afficher\_contacts() pour afficher tous les contacts dans la liste.
   * Créez une méthode rechercher\_contact() pour rechercher un contact par son nom.
   * Implémentez une méthode modifier\_contact() pour modifier les informations d'un contact existant.
   * Créez une méthode supprimer\_contact() pour supprimer un contact de la liste.
3. **Persistance des Contacts dans un Fichier Texte :**
   * Modifiez la classe GestionnaireContact pour qu'elle puisse sauvegarder la liste des contacts dans un fichier texte (contacts.txt).
     + Implémentez une méthode sauvegarder\_contacts() qui écrit tous les contacts dans le fichier texte.
     + Implémentez une méthode charger\_contacts() qui lit les contacts depuis le fichier texte et les charge dans la liste.
   * Chaque contact doit être enregistré sur une ligne distincte dans le fichier texte, avec les attributs séparés par un délimiteur (par exemple, une virgule).
4. **Interaction avec l'Utilisateur :**
   * Créez une classe Main avec une méthode main() pour permettre à l'utilisateur d'interagir avec le gestionnaire de contacts via la console.
   * Écrivez le code pour que l'utilisateur puisse :
     + Ajouter un nouveau contact.
     + Consulter la liste de tous les contacts.
     + Rechercher un contact par son nom.
     + Modifier un contact existant.
     + Supprimer un contact.
     + Sauvegarder les contacts dans le fichier texte après chaque opération CRUD.
     + Charger les contacts depuis le fichier texte au démarrage de l'application.
   * Ajoutez une option pour quitter l'application.