---Manuel d’utilisateur---

Contact

**A. Attributs**

->nom : Champ attribue au nom de famille du contact

->prénom : Champ attribue au prénom du contact

->telFixe : Champ attribue au téléphone fixe du contact

->telPortable : Champ attribue au téléphone portable du contact

->email : Champ attribue à l’email du contact

->id : a pour rôle de déterminer l’ID propre à un contact grâce à la position de la variable

**B. Méthodes**

Deux(2) constructeurs avec et sans arguments les accesseurs et mutateurs pour les champs notes ci-dessus et Afficher Contact méthode permettant l’affichage d’un contact.

Répertoire

**A. Attributs**

->nom : Champ destine au nom du répertoire

**B. Méthodes**

**1. AjouterContact()**

Elle prend un argument un objet de la classe contact et déclare plusieurs chaines de caractère chacune d’elles correspondant à un champ de notre classe contact après la saisie des différents champs on instancie le paramètre d’entrée l’objet de la classe contact avec les chaines de caractère saisies.

**2. FiletoList()**

Méthode destine à récupérer le fichier regroupant tous les contacts du répertoire en une liste.

**3. AjouterRepFile()**

Méthode destine à récupérer une liste regroupant tous les contacts du répertoire et les enregistrer dans un fichier.

Chacun des contacts constituent une ligne dans le fichier et c’est l’ensemble des lignes qui constituent le répertoire.

**4. AjouterRepertoire()**

Dans la plupart des méthodes comme celle-ci vous allez surement remarque que l’on fait appel à la méthode FiletoList() pour récupérer le contenu du fichier étant donné qu’à l’ouverture d’un fichier avec FileWriter le contenu du fichier précèdent est écrasé c’est pourquoi nous fesons appel à cette méthode FiletoList().

Et puis on fait appel à la méthode AjouterContact() pour remplir le répertoire avec les contacts saisis.

Enfin AjouterRepFile() pour enregistrer la liste des contacts symbolisant le répertoire dans un fichier

**5. ViderRepertoire()**

Pour vider le répertoire on se contente juste d’ouvrir le fichier et il effacera totalement le contenu précèdent.

**6. ModifierContact()**

Dans le cas où le numéro saisi existe dans le répertoire un nouveau contact sera ajoute et l’ancien supprime au cas échéant sortie.

**7. SupprimerContact()**

Dans le cas où le numéro saisi existe dans le répertoire l’ancien est tout simplement supprime au cas échéant sortie.

**8. RechercherContact()**

Cette méthode se base sur la fonction indexOf() qui retourne la position de l’élément prise en paramètre sur une chaine de caractère s’il est trouvé et au cas échéant renvoie -1.

**9. RepVide()**

Cette méthode est utilisée uniquement dans le cas où le répertoire est vide.

**10. TestDoublon**

Notifie et Renvoie True si le numéro existe déjà dans le répertoire et false dans le cas échéant.

-START-

Le Menu consiste l’arborescence du programme et tant que l’utilisateur

Ne saisit pas 0 la méthode Menu renvoie true et false dans le cas contra

Ire pour permettre la sortie dans le programme.