Série 7 – Interface graphique

Reprenez les classes Personne et ListePersonne de la série 6 puis créer une interface graphique (grâce à l'éditeur WYSIWYG de NetBeans et le composant graphique de Java JFrame) :

- Créez une interface graphique ressemblant à la capture d'écran suivante :

<u>*</u>	Nom: Prénom: Age:		er le contact		_ O X
	Nor	m	Prénom	Age	

Remarque : nommez vos composants avec des noms clairs. Par exemple : « btnAjouterContact » pour le bouton d'ajout ou encore « tableContacts » pour le composant graphique qui contiendra la liste des personnes, etc..

- Rajoutez un champ pour une liste de personnes dans la classe de l'interface graphique (càd : dans le code de la JFrame que vous venez de créer rajoutez un champ private ListePersonnes contacts par exemple).
- Rajoutez dans le constructeur de la classe le chargement du fichier de contacts (dé-sérialisation).
 Puis un code pour « remplir » la table avec les personnes de la liste des personnes (« peupler » la vue avec les données du modèle associé). Pour ajouter des lignes dans une table en java, vous pouvez utiliser le code suivant :

```
((DefaultTableModel)nomTable.getModel()).addRow(new String[]{
    "colonne 1", "colonne 2", "colonne 3"
});
```

Vous devez bien sur remplacer le nom de la table ainsi que les valeurs des trois colonnes par le nom, le prénom et l'âge des personnes de votre liste de personnes.

- Gérez la persistance des données en sauvegardant la liste des personnes dans un fichier (sérialisation) lorsque l'utilisateur ferme l'application.
- Gérez l'événement de clique sur le bouton « ajouter le contact ». Lorsque l'utilisateur clique sur ce bouton, vous devez reprendre le contenu des trois champs de saisie (grâce à la méthode getText() des champs nom, prénom et âge). Il faut alors créer une nouvelle personne avec ces données et l'ajouter à la liste des personnes. Il faut aussi ajouter les données du nouveau contact dans la table (avec le même code que précédement : *addRow*)