

TP « Personnes »

L'objectif de ce TP est d'intéresser à différentes classes susceptibles de décrire des personnes et les relations qui existent entre des personnes.

Construction de la classe *Personne*

1. Ecrire une classe *Personne* qui contienne au moins les informations suivantes : nom, prénom, année de naissance, sexe (sous la forme d'un booléen)
2. Ecrire un constructeur complet pour la classe *Personne* ainsi que les attributs nécessaires
3. Ecrire une méthode qui calcule l'âge de la personne en 2013
4. Ecrire une méthode *toString()* qui **renvoie une chaîne de caractères** ainsi constituée pour une personne de sexe masculin, née en 1943 qui s'appelle Pierre DURAND : « Monsieur Pierre DURAND est né en 1943 ». Il est très important de respecter la signature de la fonction, car cela permet d'appeler directement `System.out.println()` avec un argument de type *Personne*.
5. Tester que votre classe *Personne* fonctionne sur un ou deux exemples.

Construction d'une classe pour les personnes mariées

1. Construire une classe *PersonneMariee* reprenant les mêmes informations que la classe *Personne*. Il faudra ajouter un attribut désignant la personne avec qui la personne est mariée.
2. Ecrire un constructeur avec les mêmes arguments que celui de *Personne*. L'attribut désignant l'époux sera mis à *null*
3. Ajouter une méthode *marier()* qui prend en argument une *Personne* et qui renvoie un booléen. La méthode doit vérifier que les époux sont de sexe différent et qu'une personne ne se marie pas avec elle-même.
4. Surcharger la méthode *toString()* pour afficher le nom de l'époux (épouse). Vous utiliserez obligatoirement la méthode définie dans la classe *Personne*.
5. Tester que la classe *PersonneMariee* fonctionne en vérifiant tous les cas de figure possible.

Construction d'une classe pour les couples ayant des enfants

1. Construire une classe *PersonneFamille* reprenant les mêmes informations que la classe *PersonneMariee*. Ajoutez-y une liste d'enfants (sous la forme d'un `Vector<Personne>`)
2. Ecrivez un constructeur qui prend en paramètres les mêmes paramètres que ceux des classes écrites aux questions précédentes et qui initialise le vecteur des enfants à la liste vide
3. Ajoutez une méthode qui prend une *Personne* en argument et qui l'ajoute à la liste d'enfants. Ajoutez également cette *Personne* comme enfant de l'époux de la personne si ce n'est pas déjà fait

4. Ecrivez une méthode qui renvoie le nombre d'enfants
5. Surchargez la méthode toString pour afficher les informations sur les enfants
6. Testez le fonctionnement de la classe PersonneFamille

Relations de parenté

1. Ajoutez dans la classe *Personne* un lien vers les parents d'une personne (un attribut père et un attribut mère)
2. Modifiez la méthode toString() de la classe Personne pour afficher le nom des parents de la Personne.
3. Ecrivez une méthode pour vérifier si deux personnes passées en paramètres sont frère et/ou sœur. Il suffit de vérifier qu'ils ont les mêmes parents.
4. Montrez que toutes classes fonctionnent ensemble en vérifiant les différents cas possibles.