Master bio-info : Java

Année 2011-2012

TD 3

Exercice 1 [Classe Personne]

1. Écrivez une classe `Personne` (constructeur et attributs) permettant de décrire complètement une personne, sachant que l'on souhaite avoir autant d'informations que dans la phrase suivante :
"LeBlanc Pierre, sexe masculin, né(e) en 1965, célibataire" ou "LeNoir Bernadette, sexe féminin, né(e) en 1945, marié(e)"
2. On veut ajouter une méthode `age` qui calcule l'âge approximatif d'une personne en fonction de l'année courante. Pour cela,
 - Ajoutez un attribut `anneeCourante`, et ajouter une méthode `incrémenterAnneeCourante()`.
 - Écrivez la méthode `age()`.
3. Ajoutez à la classe `Personne` un attribut `conjoint` et examinez les conséquences que cela peut avoir sur l'ensemble du code.
Attention : on ne peut-être marié qu'à une seule personne!
4. Ajoutez une méthode `boolean marier(Personne p)` qui permet de marier une personne à une autre. On vérifiera que les deux mariés ont plus de 18 ans, sinon le mariage n'aura pas lieu et on retournera `false`.
5. Écrivez une méthode `boolean marieAvec(Personne p)`, telle que `p1.marieAvec(p2)` retourne `true` si les personnes `p1` et `p2` sont mariés ensemble. Écrivez une autre méthode `boolean marieAvec(Personne p1, Personne p2)`. Que peut-on en dire? Quelle méthode `marieAvec` vous paraît préférable?
6. Écrivez un autre constructeur qui créera une personne et la mariera en même temps à la personne donnée en argument.
7. Ajoutez une méthode `boolean divorcer()` qui annule le précédent mariage s'il y avait lieu, retourne `false` si la personne n'était pas mariée, et `true` sinon.
8. Écrivez une méthode `Personne enfanter(String prenom, Char sexe)` qui créera une nouvelle personne de même nom que la mère, de sexe masculin ou féminin selon la lettre ('M' ou 'F') donnée en argument.
Attention : A l'heure actuelle, seule une femme peut enfanter!

9. Soit la classe `Compte`, permettant de gérer un compte en banque :

```
class Compte {
    private int solde;

    public Compte() { solde = 0; }
    public int getSolde() { return solde;}
    public void credit(int montant) { solde += montant; }
    public void debit(int montant) { solde -= montant; }
}
```

On suppose maintenant que toute personne a un attribut `compte` de type `Compte`. Complétez le code suivant (dans `Personne`).

```
//Vole a p sa monnaie. Modifie la monnaie de p.
    public void vole(Personne p){ ... }

//Donne sa monnaie a p. Modifie la monnaie de p.
    public void donne(Personne p){ ... }

//Depose la monnaie sur le compte
    public void deposeMonnaie(){ ... }

//Retire de l'argent du compte pour avoir de la monnaie
    public void retireMonnaie(int montant){ ... }
```