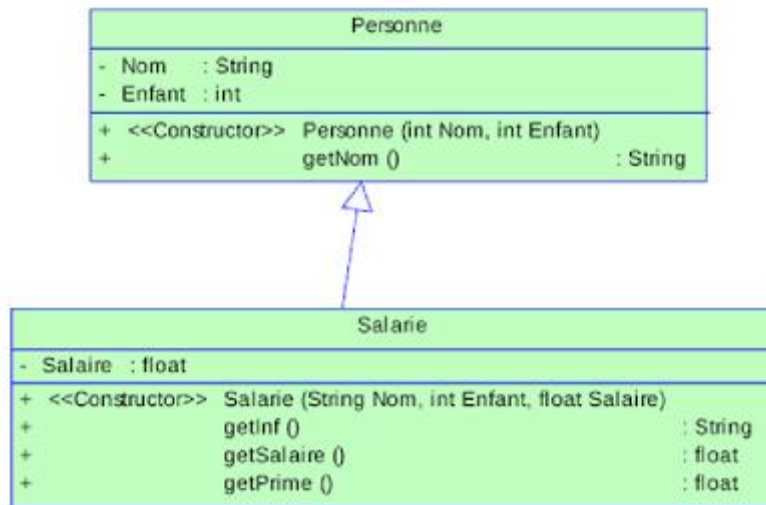


+Institut Supérieur des Etudes Technologiques de Gafsa
Département de Technologies de L'informatique

Matière : P.O.O	TD POO HERITAGE	Date : Décembre 2014
Enseignante : RHAHLIA AMEL		
Classe : DSI 2- RSI 2		

Exercice 1

Soit le diagramme de classes d'UML suivant :



1. Créer les classes "Personne.java" et "Salarie.java", sachant que la prime est calculée à la base de la formule suivante : $\text{prime} = (5 * \text{Salaire} * \text{Nombre d'enfant}) / 100$
2. Ecrire un programme "test1.java" pour tester vos classes

Exercice 2

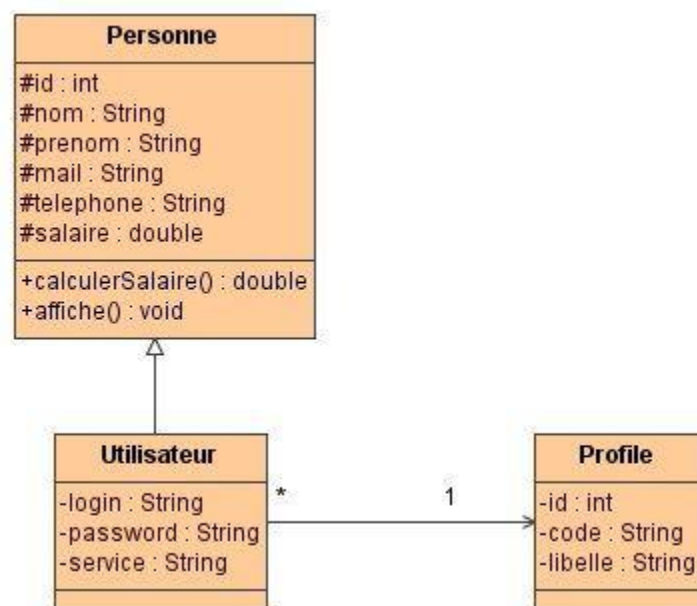
On modélise une application devant servir à l'inventaire d'une bibliothèque. Elle devra traiter des documents de nature diverse : des livres, des dictionnaires, et autres types de documents qu'on ne connaît pas encore précisément mais qu'il faudra certainement ajouter un jour (articles, bandes dessinées...). Tous les documents possèdent un numéro d'enregistrement et un titre. A chaque livre est associé, en plus, un auteur et un nombre de pages, les dictionnaires ont, eux, pour attributs supplémentaires une langue et un nombre de tomes. On veut manipuler tous les articles de la bibliothèque au travers de la même représentation : celle d'un document.

1. Définissez les classes Document, Livre et Dictionnaire. Définissez pour chacune un constructeur permettant d'initialiser toutes ses variables d'instances.
2. Définissez une classe Bibliothèque réduite à une méthode main permettant de tester les classes précédentes (ainsi que les suivantes).
3. Définissez la classe ListeDeDocuments permettant de créer une liste vide de documents, puis y adjoindre une fonction permettant d'ajouter un document.
4. Dans la classe ListeDeDocuments définissez une méthode tousLesAuteurs() qui affiche la liste des numéros des documents de la liste avec, pour chacun, l'éventuel auteur.
5. Redéfinissez la méthode toString() dans la classe Document ainsi que dans les classes Livre et Dictionnaire et qui renvoie une chaîne de caractères décrivant un document, un livre ou un dictionnaire. Ajoutez alors dans la classe ListeDeDocuments une méthode tousLesDocuments() qui affiche consécutivement la description de tous les documents.
6. Proposez quelques lignes de codes à ajouter à la classe Bibliothèque afin de tester la classe ListeDeDocuments.

Exercice3

Le directeur des systèmes d'information de la société COLAPS souhaite développer un module pour la gestion des utilisateurs de son service, pour cela il vous a fait appel pour réaliser cette tâche.

Le diagramme de classe a été établi par un analyste afin de mettre en place une base de données sous ORACLE ou MySQL :



1. Développer les classes ci-dessus en JAVA dans un package de votre choix.

NB : l'identifiant est auto incrément.

2. Redéfinir la méthode **calculerSalaire ()** et la méthode **affiche()** dans la classe Utilisateur.

Sachant que :

✓ Le manager aura une augmentation de 10% par rapport à son salaire normal,

✓ Le directeur général aura une augmentation de 40% par rapport à son salaire normal.

3. Créer les profiles :

- Chef de projet (CP),
- Manager (MN),
- Directeur de projet (DP),
- Directeur des ressources humaines (DRH),
- Directeur général (DG),

4. Créer des utilisateurs avec les différents profiles métiers.

5. Afficher la liste des managers.