

# 1. Programmation en java

TP associé au cours 02\_formation\_Java\_-\_POO

## 1.1. Gestion des employés d'une entreprise

Dans une société, un **employé** est décrit par les membres suivants :

### Attributs :

Nom ; Age ; salaire.

### Constructeurs :

- par défaut,
- par copie (on ne s'en occupera pas pour le moment),
- avec trois paramètres : nom ; age ; salaire.

### Méthodes :

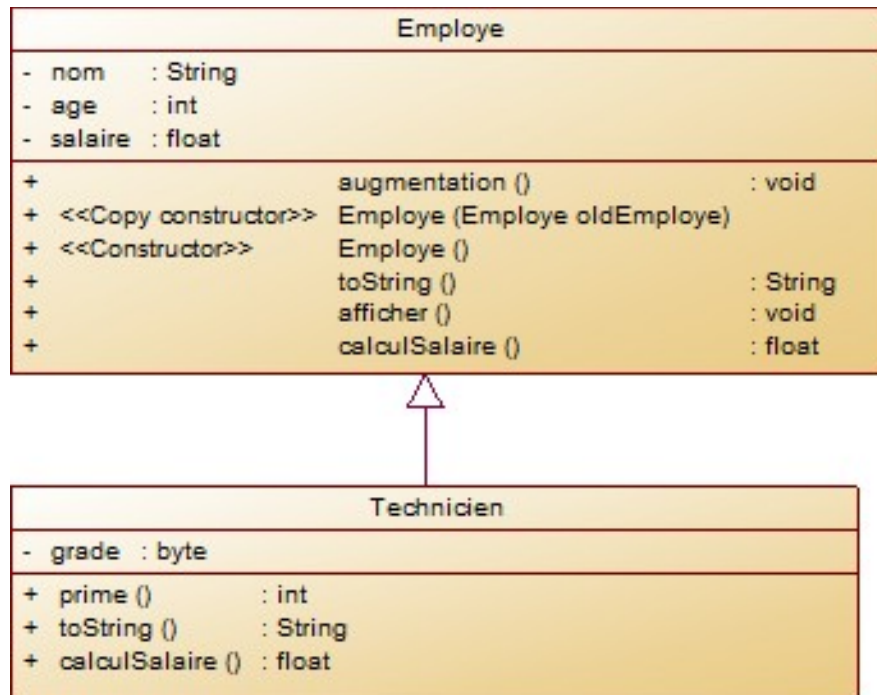
- Augmentation(...) ;
- toString() ;
- afficher() ;
- calculeSalaire() ;

Un **technicien** est décrit en plus par l'attribut : **grade** et la méthode **prime()** et la redéfinition de la méthode toString() et de la méthode calculeSalaire().

- Si grade=1 alors Prime= 100
- Si grade=2 alors Prime= 200
- Si grade=3 alors Prime= 300
- sinon Prime = 0

# UML

Diagramme de classes :



## Question

Dans un projet entreprise avec un package `fr.bge.employe`

**Q1 :** Ecrire la classe Employé.

**Q2 :** Ecrire la classe Technicien.

**Q3 :** Ecrire un programme qui saisie un employé puis un technicien et affiche leurs informations avant et après augmentation de leurs salaires.

Par exemple :

```
public static void main(String[] args) {  
  
    EmployeeC e1=new EmployeeC("Albert",28,4500);  
    EmployeeC e2=new TechnicienC("Bernard",50,8000,(byte)5);  
  
    //TechnicienC e3 =new EmployeeC(); //erreur  
  
    TechnicienC e3 =new TechnicienC("Jacques",25,5000,(byte)4);  
  
    System.out.println("Avant augmentation:");  
    e1.afficher();  
    e2.afficher();  
    e3.afficher();  
  
    e1.augmentation(600);  
    e2.augmentation(500);  
    e3.augmentation(650);  
  
    System.out.println("Après augmentation:");  
    e1.afficher();  
    e2.afficher();  
    e3.afficher();  
  
}
```