# R21\_Exercice3\_Dev\_Programme.py

Dans le fichier **R21\_Exercice3\_Dev\_Programme\_DEPART.py** vous avez les classes pour les différents employés d’une compagnie

Maintenant que nous sommes plus expérimentés, nous allons utiliser les notions de classes et méthodes abstraites.

**Pour la classe Employe**, on veut que :

* L’attribut **salaire** devient privé,
* Donner accès en lecture et en écriture à cet attribut
* Quand on change le salaire, faites une validation, pour accepter uniquement un nouveau salaire supérieur à l’ancien salaire. Indiquez que le salaire a changé quand c’est le cas. Sinon soulevez une erreur (raise ValueError) avec un message comme quoi "Le nouveau salaire doit être supérieur à l'ancien".
* Nous allons modifier la classe Employe pour qu’elle devienne Abstraite (n’oubliez pas de faire les imports). Ajoutez la méthode abstraite logger() qui devra être redéfinie dans les sous-classes qui implémenteront la classe Employe

**Pour la classe Programmeur**, on veut que :

* Changer le nom de la méthode coder() pour logger(). Ce sera l’implémentation de la méthode abstraite logger() de Employe.

**Pour la classe Designer**, on veut que :

* Changer le nom de la méthode dessiner() pour logger(). Ce sera l’implémentation de la méthode abstraite logger() de Employe.

**Pour la classe Tech\_Reseau**, on veut que :

* Changer le nom de la méthode intervenir() pour logger(). Ce sera l’implémentation de la méthode abstraite logger() de Employe.

On veut que le script puisse être exécuter sans changement de comportement après avoir fait nos modifications