

Objectif

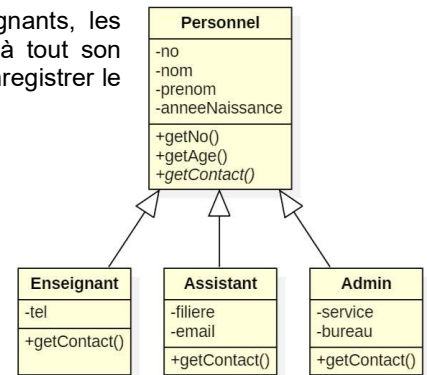
- Extension de fonctionnalité sur des classes existantes
- Type Abstrait de Données (TAD)
- Création de classe « **collection** ».

Énoncé du problème – votre travail :

La HEG possède la liste de tout le personnel dans 3 « groupes » : les enseignants, les assistants, et le personnel administratif. Elle a décidé d'effectuer le TestCovid à tout son personnel. On vous demande de développer une petite application permettant d'enregistrer le résultat des tests, puis de contacter tout le personnel testé positif.

Projet TestCovid : code fourni – ne pas modifier ces classes !

- Dans le package `domaine`, les classes (*ci-contre*) déjà utilisées dans toutes les autres applications à la HEG (*donc ne pas les modifier !*).
- Dans le package `dao`, une méthode `Bdd.getLstPersonnel()` qui retourne la liste de tout le personnel de la HEG.
- Dans le package `outils`, une méthode `Test.resultat(noPersonne)` qui retourne un boolean indiquant si cette personne a été testée positive.

**Travail demandé** (dans le package `metier`)

- La classe **Main** contient le chablon de l'application à compléter (*vous pouvez modifier la signature des méthodes si nécessaire*).
- Compléter la boucle de la méthode `testerToutLePersonnel()` afin d'effectuer les traitements suivants :
 - tester la personne ==> appeler `outils.Test.resultat()`
 - conserver le résultat du test de la personne avec (et non pas dans !) la classe `Personnel` !
 - ajouter ces infos dans `lst`, par catégorie (*selon l'âge de la personne*) => `categorie = age < 25 ? 0 : age >= 50 ? 2 : 1;`
- Les personnes seront répertoriées dans 3 catégories selon leur âge : catégorie 0 en dessous de 25 ans, catégorie 1 de 25 à 49 ans, et catégorie 2 à partir de 50 ans.
 Instruction java : `categorie = age < 25 ? 0 : age >= 50 ? 2 : 1;`
- Pour des questions de performance (*en vue de traitements ultérieurs*), les personnes seront stockées dans 3 listes différentes selon leur catégorie d'âge. Effectivement, chaque traitement ne portera généralement que sur une seule catégorie de personne, donc on ne souhaite pas « mélanger » toutes les personnes dans la même liste !
- La méthode `afficherTauxParCategorie()` doit afficher le pourcentage de personnel testé positif de chaque catégorie. Pour des questions de performance, le pourcentage doit pouvoir être retrouvé(calculé) **SANS PARCOURIR** la liste des personnes testées ! Comme il doit y avoir une liste de personnes par catégorie (*voir la méthode `testerToutLePersonnel`*), le taux est calculé simplement en divisant le nombre de positifs par le nombre de personnes dans la liste.
- Dans la méthode `contacterPersonnelPositif()`, effectuer une boucle qui récupère toutes les personnes d'une certaine catégorie (*reçue en paramètre*), en appelant la méthode `lst.get(categorie)`
 - vérifier si cette personne est positive (*cette information a du être conservée avec (et non pas dans !) Personnel*)
 - si oui, afficher la personne ainsi que ses données de contact ==> utiliser `getContact()`