

TD 9

Gestion de comptes bancaires - Gestion d'une entreprise OCL

Rappel de cours

On rappelle les opérations suivantes :

Opérations sur les collections

- `size()` : Integer - retourne le nombre d'éléments de la collection.
- `includes(object : T)` : Boolean - vrai si la collection contient l'objet `object`.
- `excludes(object : T)` : Boolean - vrai si la collection ne contient pas l'objet `object`.
- `isEmpty()` : Boolean - vrai si la collection est vide.
- `notEmpty()` : Boolean - vrai si la collection n'est pas vide.

Opérations sur les éléments d'une collection

- `select(<expression logique>)` - génère une sous-collection ne contenant que les éléments qui satisfont l'expression logique `<expression logique>`.
- `reject(<expression logique>)` - génère une sous-collection ne contenant que les éléments qui ne satisfont pas l'expression logique `<expression logique>`.
- `forall(<expression logique>)` - expression logique vraie si l'expression `<expression logique>` est vraie pour tous les éléments de la collection.
- `exists(<expression logique>)` - expression logique vraie si l'expression `<expression logique>` est vraie pour au moins un élément de la collection.

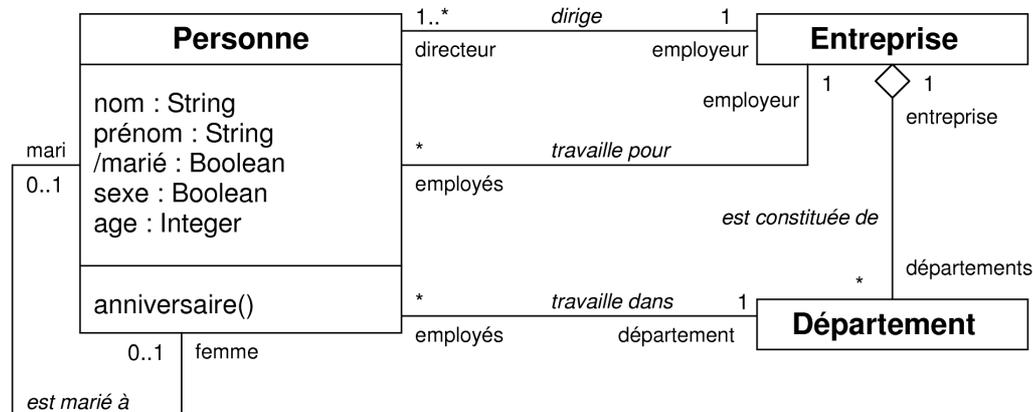
Exercice 1

On considère une personne, définie par son nom, son prénom et son âge qui peut avoir zéro, un ou plusieurs comptes bancaires, chacun défini par un numéro de compte et un solde. On définit de plus une méthode de débit pour le compte bancaire.

1. **Faire un diagramme de classes. On nommera les rôles *propriétaire* et *comptes*.**
2. **Dans le contexte `Personne`, que désignent :**
 - (a) `self.comptes` ?
 - (b) `self.comptes->size()` ?
 - (c) `self.comptes->select(solde>0)` ?
3. **Ecrire les expressions OCL qui permettent d'indiquer les contraintes suivantes :**
 - (a) Le solde du compte doit toujours être positif.
 - (b) Avant de débiter une somme, on doit vérifier qu'il y a assez d'argent sur le compte.
 - (c) Pour avoir un compte il faut avoir au moins 18 ans.
 - (d) Une personne peut avoir au maximum 3 comptes bancaires.

Exercice 2

On considère le diagramme de classes suivant :



Le booléen `sexe` vaut 1 pour une femme et 0 pour un homme.

1. Donner pour les contextes **Personne** et **Entreprise** les significations des différentes associations.
2. Ecrire en langage naturel les contraintes suivantes :
 - (a) context `Entreprise`
`inv : self.employés -> exists(age > 50)`
 - (b) context `Département`
`inv : self.employés -> size() > 6`
 - (c) context `Personne`
`inv : self.marié implies`
 `self.sexe=1 implies`
 `self.nom=self.mari.nom`
 `self.sexe=0 implies`
 `self.nom=self.femme.nom`
3. Ecrire les expressions OCL qui permettent d'indiquer les contraintes suivantes :
 - (a) Une entreprise possède exactement un directeur, et celui-ci est un employé.
 - (b) Les employés travaillant dans l'entreprise sont âgés de 18 à 65 ans.
 - (c) Le jour d'anniversaire de la personne, son âge est augmenté de 1 an.
 - (d) Etre marié c'est avoir au moins une femme ou un mari.
 - (e) Un homme est marié avec exactement une femme et une femme avec exactement un homme.