

TD OCL MASTER2 ILC

Exercice 1

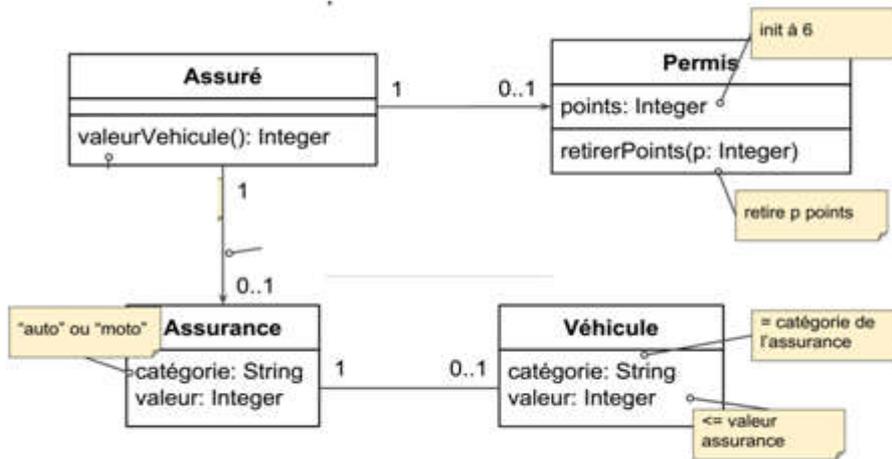
Exprimer les contraintes:

- valeurs est toujours strictement supérieur à 0
- ajoute prend un paramètre toujours supérieur ou égal à 0
- ajoute incrémente le nombre de valeurs et additionne valeur à somme
- moyenne retourne la moyenne de somme sur valeurs

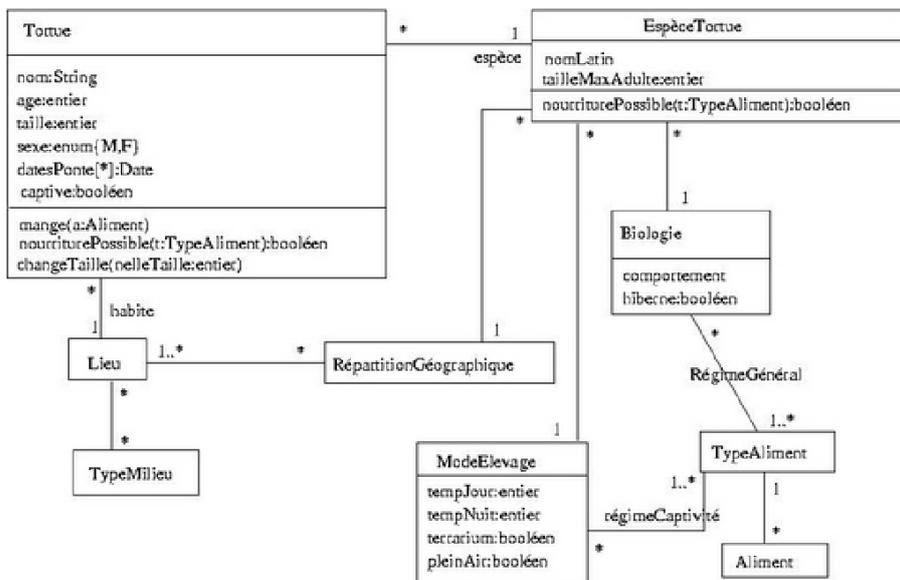
Compteur
valeurs: Integer somme: Integer
ajoute(valeur: Integer) moyenne(): Integer

Exercice 2

Exprimer les contraintes suivantes en OCL



Exercice 3



TD OCL MASTER2 ILC

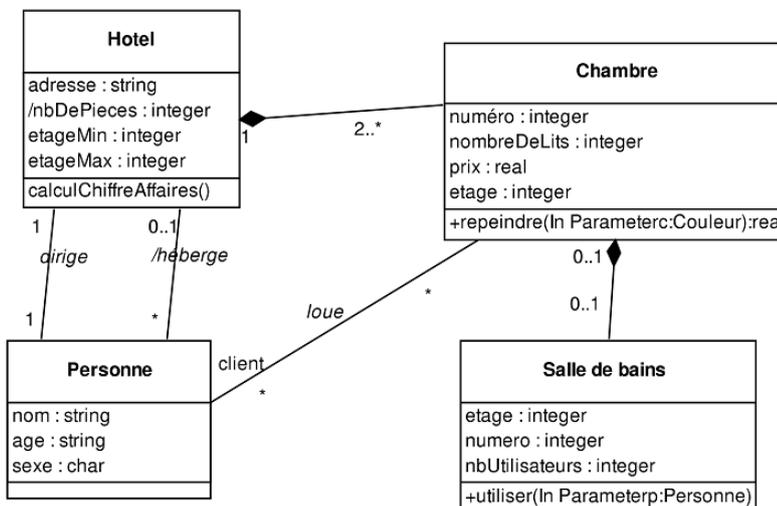
Complétez la modélisation présentée par le diagramme de classe en y ajoutant les contraintes :

1. La taille d'une tortue est comprise entre 0 et la taille maximale admise pour son espèce ;
2. Une tortue mâle ne peut avoir de dates de ponte ;
3. Une tortue habite l'un des lieux de la répartition géographique de son espèce ;
4. La température de jour est supérieure à la température de nuit
5. On ne peut changer la taille d'une tortue que pour l'augmenter ; précisez les pré- et post-conditions de l'opération ;

Exercice 4

Le directeur d'une chaîne d'hôtels vous demande de concevoir une application de gestion de ses hôtels. Un hôtel est constitué d'un certain nombre de chambres. Un responsable de l'hôtel gère la location des chambres. Chaque chambre se loue à un prix donné. L'accès aux salles de bains est compris dans le prix de la location d'une chambre. Certaines chambres comportent une salle de bains, mais pas toutes. Les hôtes de chambres sans salle de bain peuvent utiliser une salle de bains sur le palier. Ces dernières peuvent être utilisées par plusieurs hôtes. Les pièces de l'hôtel qui ne sont ni des chambres ni des salles de bain (hall d'accueil, cuisine...) ne font pas partie de l'étude (hors sujet). Des personnes peuvent louer une ou plusieurs chambres d'hôtel an d'y résider. En d'autres termes : l'hôtel héberge un certain nombre de personnes, ses hôtes (il s'agit des personnes qui louent au moins une chambre de l'hôtel).

Le diagramme de classes suivant modélise ce problème :



Question : Donnez une formulation en langage naturel pour chacune des contraintes OCL suivantes :

1. context Chambre inv :

self.etage <>13

context SalleDeBains inv :

self.etage <>13

TD OCL MASTER2 ILC

2. context Chambre inv :

```
client->size <= nombreDeLits or  
(client->size = nombreDeLits +1 and  
client->exists(p:Personne | p.age < 4))
```

3. context Hotel inv :

```
self.chambre->forall (c : Chambre |  
c.etage <= self.etageMax and c.etage >= self.etageMin)
```

4. context Chambre::repeindre(c:Couleur)

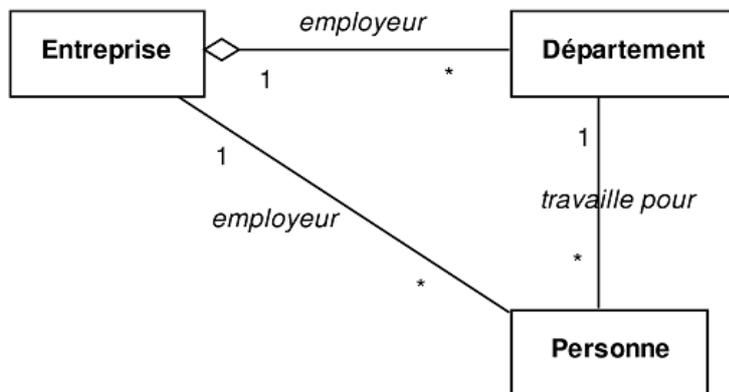
```
pre : client->isEmpty  
post : prix = prix@pre * 1.1
```

6. context Hotel::calculerLoyer() : integer

```
post : result = self.chambre->select(client->notEmpty).prix->sum
```

Exercice 5

Considérant le diagramme de classe suivant :



1. Donnez une expression OCL qui permette d'indiquer que la personne qui travaille dans le département est la même que celle qui est employée par l'entreprise.
2. Donner une expression OCL qui permette d'indiquer qu'une personne travaillant pour une entreprise doit être âgée de 18 ans et plus. On suppose que la classe Personne a un attribut âge.
3. Modifier, graphiquement, le diagramme des classes précédent pour prendre en compte la contrainte suivante :
context personne inv :
(self.departement -> isEmpty)
xor
(self.Entreprise -> isEmpty)

TD OCL MASTER2 ILC

4. Ajouter la contrainte indiquant que deux personnes ne doivent pas avoir le même nom.
5. Aucune personne n'est âgée de plus de 130 ans. Le jour d'anniversaire de la personne, son âge est augmenté de 1 an. Ajouter l'expression OCL permettant de représenter cette précondition et cette post-condition associées à l'opération anniversaire de Personne.
6. Les personnes qui travaillent dans l'entreprise sont âgées de 18 à 65 ans. Donner l'expression OCL correspondante.