

## ING 1

### TD sur l'héritage

#### Écriture de classe – Héritage

Un éleveur de volaille reçoit d'un fournisseur de jeunes canards et de jeunes poulets qu'il élève jusqu'à ce qu'ils aient la taille nécessaire à leur commercialisation. Une volaille est caractérisée par son poids et un numéro d'identification reporté sur une bague qu'elle porte à sa patte. Les volailles arrivent à l'élevage à l'âge de trois semaines.

Elles sont baguées et enregistrées dans le système informatique. Il y a deux sortes de volailles : des canards et des poulets. Le prix du canard et celui du poulet sont deux prix différents, exprimés en euros par kilo.

En revanche, le prix est le même pour tous les individus de la même espèce. Ce prix varie chaque jour. Le poids auquel on abat les bêtes est différents pour les canards et les poulets, mais c'est le même pour tous les poulets (respectivement, tous les canards).

Donnez la modélisation UML de la classe des volailles avec deux sous-classes des poulets et des canards. Il faut pouvoir enregistrer les prix du jours, les poids d'abatage, le poids d'une volaille donnée.

Écrivez une classe permettant de représenter l'ensemble des animaux de l'élevage au moyen d'un tableau. Des méthodes doivent permettre de trier les animaux à abattre et d'évaluer le prix obtenu pour ces animaux. Il faut également pouvoir enregistrer les jeunes animaux qui arrivent.

## Écriture de classe – Modélisation

### Transport de marchandises

On souhaite modéliser en java le calcul de coûts de transport de marchandises. Les marchandises transportées seront des instances de la classe Marchandise dont le source java est donné ci-dessous :

```
public class Marchandise {
    private int poids ;
    private int volume ;
    public Marchandise (int poids, int volume) {
        this.poids = poids ;
        this.volume = volume ;
    }
    public int getPoids () { // retourne le poids en kg
        return poids ;
    }
    public int getVolume () { // retourne le volume en dm3
        return volume ;
    }
}
```

Donnez la modélisation UML de cette classe.

Les marchandises sont transportées sous la forme de cargaisons. Les seules fonctionnalités publiques des cargaisons sont :

- ajouter qui permet d'ajouter une marchandise dans cette cargaison si cela est encore possible ;
- getCoût() qui retourne, sous la forme d'un nombre entier d'euros, le coût total du transport de cette cargaison.

Une cargaison est par ailleurs également caractérisée par la distance sur laquelle elle est transportée. Ce renseignement est communiqué à la construction de la cargaison sous la forme d'un nombre entier de kilomètres.

On précise qu'une cargaison ne peut réunir qu'un nombre limité de marchandises qui dépend d'un encombrement total de ces marchandises à ne pas dépasser. Cet encombrement est soit le poids total, soit le volume total des marchandises, selon le type de transport utilisé. Ce dernier influe aussi sur le calcul du coût de transport de la cargaison qui, de la même façon, dépend de l'encombrement des marchandises de la cargaison. On distingue donc plusieurs types de cargaisons selon le moyen de transport utilisé. On peut cependant trouver une certain nombre de caractéristiques communes à toutes les cargaisons que vous devrez identifier.

Les différents types de cargaison et leurs caractéristiques sont donnés par le tableau suivant :

type	encombrement	coût	limite
Fluviale	poids	distance × encombrement	encombrement ≤ 300000
Routiere	poids	4 × distance × encombrement	encombrement ≤ 38000
Aerienne	volume	10 × distance × encombrement	encombrement ≤ 80000
AerienneUrgente	volume	2 × le coût d'une cargaison Aerienne	encombrement ≤ 80000