

## Modélisation et programmation par objets

### Spécialisation/généralisation et héritage

Dans ce sujet, nous élaborons autour des notions d'additif alimentaire. Un additif alimentaire est un produit ajouté à une préparation afin d'en réduire l'oxydation ou d'en améliorer la conservation, l'aspect, la texture ou le goût. Une énumération décrivant les différentes fonctions possibles pour un additif vous est donnée ci-dessous.

**public enum TypeFonction {antioxydant, conservation, aspect, gout, texture}**

Q0. Représentez en UML et écrivez en Java le code d'une classe pour représenter les additifs. Un additif est décrit par un nom, un code (chaîne de caractères), une fonction et s'il est autorisé en France.

Q1. Deux sous-classes d'additifs sont considérées : les additifs naturels et les additifs de synthèse.

Les *additifs naturels* sont des additifs décrits en plus par le produit naturel d'*origine*, c'est-à-dire dont ils sont extraits (chaîne de caractères) et par le *type* de cette origine, qui prend sa valeur dans l'énumération qui suit : **public enum TypeOrigine {mineral, vegetal, animal, sousProduitAnimal}**. Les sous-produits animaux correspondent par exemple aux œufs et aux laitages.

Les additifs de synthèse sont des additifs créés par un procédé de synthèse chimique. Ils sont décrits par la *stratégie* de synthèse (chaîne de caractères).

Représentez en UML puis écrivez en Java les sous-classes **AdditifNaturel** et **AdditifDeSynthese** :

- Leur entête
- Leur(s) attribut(s)
- Leurs constructeurs, en particulier les constructeurs avec paramètres permettant d'initialiser tous leurs attributs.

Q2. Écrivez dans les trois classes une méthode de signature **public void saisie(Scanner clavier)** qui permette de saisir des valeurs pour tous les attributs d'un objet.

Q3. Écrivez dans les trois classes une méthode de signature **public TypeRegime regime()** qui retourne le régime alimentaire le plus restrictif autorisant l'additif.

Vous utiliserez l'énumération : **public enum TypeRegime { general, vegetarien, vegetalien }.**

Cette méthode se comporte différemment dans les deux sous-classes, comme décrit ci-dessous :

<i>Additif naturel</i>		<i>Additif de synthèse</i>	
Type d'origine	Régime à retourner		Régime à retourner
<b>animal</b>	<b>general</b>	Dans tous les cas	<b>general</b>
<b>sousProduitAnimal</b>	<b>vegetarien</b>		
<b>vegetal</b>	<b>vegetalien</b>		
<b>mineral</b>	<b>vegetalien</b>		

Après l'avoir écrite, indiquez si vous avez dû modifier quelque chose dans l'entête de la classe **Additif** (expliquer pourquoi cela a été ou non nécessaire). Il y a plusieurs solutions, expliquez simplement votre choix.

Q4. Écrivez une méthode **main** dans laquelle vous testerez vos classes.