TD 2- Introduction à la Programmation Orientée Objet

TD d’application POO : package et agrégation

# Bienvenue dans le monde merveilleux de la POO !

## Introduction

Les choses sérieuses commencent… Pour mettre en application le cours, je vous propose un TD qui va vous accompagner à la mise en pratique de ce que l’on a vu en cours précédemment. Pour cela, nous allons nous intéresser au robot de cuisine.

Nous allons, au fur et à mesure des cours, venir compléter ce TD et appliquer les principes de la POO.

## Contexte

Pour rappel, On veut réaliser un programme de gestion des recettes de cuisine, qui sera installé sur  
des appareils électroménagers pour leur permettre de faire la cuisine de façon autonome. Celui-ci aura la possibilité d’enregistrer des plats et, surtout, les ingrédients qui composent les plats. Nous allons nous intéresser tout d’abord aux ingrédients. Pour chaque ingrédient, nous allons nous intéresser à leur nom, la quantité nécessaire ainsi que leur unité. L'unité peut être une unité de poids (gramme, kg, etc), de volume (litre, ml, cl) ou simplement une cardinalité.

Maintenant que notre classe ingrédient est créé et que l’on a pu tester la création d’objet grâce à notre main. Nous allons l’améliorer pour créer cette fois-ci une recette.

# Prérequis

* Connaitre les bases de JAVA
* Avoir réalisé le TD 1
* Savoir poser des questions (n’est-ce pas ?)

# Ressources utiles

* La procédure de création d’un projet
* La procédure d’import de JDBC
* Le 1er et le 2e cours sur la POO

Durant le TD, n’hésitez pas à nous solliciter afin de valider certaines étapes si vous avez un doute !

# Début du TD

## Etape 1 : Structuration de nos fichier source

Actuellement, dans notre projet, nous avons nos deux/trois fichiers qui sont Ingredient.java / Main.java / MainBonus.java qui se trouve dans le package par défaut.

Or, dans le cours, il est dit qu’il est fortement conseillé de structurer notre projet en package.

Nous allons donc créer deux packages :

* Un package qui va contenir nos fichier source en lien avec les recettes
  + On nommera ce package « recette »
* Un package qui servira à l’exécution du programme.
  + On nommera ce package « application »

Travail à faire : Créez les deux packages et ranger les fichier source au bonne endroit. Re-testez votre application pour savoir si tout fonctionne

Maintenant que l’on a restructuré notre projet et vérifié que tout fonctionne normalement, regardons ce qui a été rajouté dans nos fichiers.

Concernant le fichier Ingredient.java :

* recette qui correspond au nom du package où se trouve le fichier

Concernant le(s) ficher(s) Main et MainBonus :

* application qui correspond au nom du package où se trouve le fichier
* import recette.ingredient qui permet d’utiliser la classe Ingredient dans notre application

## Etape 2 : l’agrégation

Notre objectif est de créer des recettes à base d’ingrédient. Ce qui veut dire qu’une recette contiens plusieurs ingrédient. On veut également le nom de la recette.

De plus, pour interagir avec notre recette nous avons répertorié les méthodes suivantes à réaliser :

* La méthode ajouterIngredient
  + Elle prend un paramètre qui est l’ingrédient à ajouter.
  + Elle a pour but d’enregistrer l’ingrédient dans la recette
* La méthode getNbIngredients
  + Elle ne prend aucun paramètre
  + Elle retourne simplement le nombre d’ingrédient
* La méthode changeQuantiteIngredient
  + Elle prend deux paramètres
    - La position de l’ingrédient dans votre recette
    - La nouvelle quantité
  + Ne retourne aucun élément
* La méthode afficheIngredient
  + Elle prend un paramètre
    - La position de l’ingrédient dans votre recette que vous souhaitez afficher
  + Elle renvoie une chaine de caractère représentant l’ingrédient
* La méthode afficherRecette :
  + Elle ne prend aucun paramètre
  + Elle retourne le nom de la recette ainsi que tous les ingrédients qui la compose
* Un constructeur
  + Prenant en paramètre le nom de la recette
* Un getter/setter du nom de la recette

Avant de commencer, nous allons réaliser le schéma UML de nos classes.

Travail à faire : réaliser le schéma UML de notre classe ingrédient et de notre classe recette

Notre Classe Recette aura deux attributs :

* Un arraylist pour les ingrédients
* Un string pour le nom de la recette

Travail à faire : réalisez la classe recette

## Etape finale : l’application

* Notre nouvelle classe est prête pour un essai. Nous allons, dans notre classe « Main » venir compléter la fonction « main() » pour créer et essayer notre application :
* Voici le déroulé souhaiter de notre programme de test :
  + Créez une recette nommé « crêpes »
    - Trouvez une recette de crêpe sur internet en parallèle.
  + Créez les ingrédients de votre recette
    - Demander à l’utilisateur de saisir le nom de l’ingrédient
    - Demander à l’utilisateur de saisir la quantité de l’ingrédient
    - Demander à l’utilisateur de saisir l’unité de l’ingrédient
    - Instancier un objet de type « Ingredient » dans une variable nommée « monIngredient » grâce à votre constructeur et les valeurs saisi par l’utilisateur.
  + Ajoutez les ingrédients dans votre recette
  + Affichez le nombre d’ingrédient dans votre recette
  + Affichez ensuite le premier ingrédient de votre recette
  + Affichez votre recette complète
  + Vous voulez changer la quantité de vos ingrédients pour qu’elles correspondent à personnes
  + Affichez de nouveau votre recette et vérifiez que les quantités ont bien changés.

## Etape complémentaire

Pour ceux qui ont terminé ou ceux qu’ils souhaitent s’entrainer à la maison

* Créer un nouveau fichier nommé « MainRecetteBonus » afin de ne pas se mélanger avec le 1er
* A l’intérieur de ce nouveau fichier, créez un nouveau main()
* L’objectif ici est de créer une petite application permettant à un utilisateur de venir créer plusieurs recettes que l’on viendra enregistrer dans une ArrayList.
* Cette application aura donc un menu avec :
  + La possibilité de créer une nouvelle recette et de le placer dans notre ArrayList
  + La possibilité d’afficher l’ensemble des recettes présent dans notre ArrayList
  + La possibilité d’afficher une recette de notre liste à la position donné.
  + La possibilité de supprimer une recette de notre ArrayList
  + La possibilité de modifier la quantité d’un ingrédient à la position donné d’une recette.