

TP n°7 – Insérez les données Open Food Facts avec JPA

Objectifs

Vous connaissez peut-être l'application **Yuka**, disponible sur smartphone. **Yuka** fournit des informations nutritionnelles sur pratiquement tous les produits alimentaires commercialisés en France. En plus d'informations, elle fournit également un score nutritionnel, de A (excellent) à F (mauvais).

Cette application à succès s'est construite sur une **base de données open source** appelée **Open Food Facts**.

La base de données **Open Food Facts** est une base de données **mondiale**, qu'on peut télécharger sous la forme d'un fichier **CSV**. Le fichier que je vais vous demander de traiter dans le cadre de ce TP est le même que celui sur lequel s'est basé Yuka. Il ne concerne que **les produits alimentaires fabriqués en France**.

Dans ce TP vous allez créer une application qui met en base ce fichier.

Description du contenu du fichier

Le fichier fourni, au format CSV, comporte 13 432 références de produits avec 30 informations associées par produit. Dans ce fichier le caractère séparateur est le | :

- Index 0 : catégorie
- Index 1 : marque
- Index 2 : nom du produit
- Index 3 : score nutritionnel : A (excellent) à F (mauvais)
- Index 4 : liste des ingrédients séparés par des virgules
- Index 5 : énergie pour 100g (en joules)
- Index 6 : quantité de graisse pour 100g
- etc.
- Index 28 : liste des allergènes séparés par des virgules
- Index 29 : liste des additifs séparés par des virgules

Instructions

- Créez un nouveau projet **STS/GitHub/Maven** nommé **traitement-fichier**
- **Nommez les packages comme vous le souhaitez**
- Afin de traiter le fichier voici le **modèle objet préconisé** :

L'idée est d'avoir une classe **Produit** qui va représenter une ligne du fichier.

Quelle est la structure d'un produit ? Un produit reste assez simple dans sa structure.

La classe **Produit** possède :

- une catégorie
- une marque
- un grade nutritionnel (de A à F)
- un tas de données nutritionnelles (sucre, sel, etc..)
- une liste d'ingrédients
- une liste d'allergènes
- une liste d'additifs.

Les attributs de type catégorie, marque, ingrédients, allergènes et additifs peuvent être traités, soient comme des String, soient comme des objets. Il est préférable d'opter pour les objets.

- Créez une classe **Ingredient** avec un attribut id et un attribut libelle
- Créez une classe **Allergene** avec un attribut id et un attribut libelle
- Créez une classe **Categorie** avec un attribut id et un attribut libelle
- Créez une classe **Marque** avec un attribut id et un attribut libelle
- Créez une classe **Produit** avec
 - un attribut de type **Categorie**
 - un attribut de type **Marque**
 - un attribut **scoreNutritionnel**
 - un ensemble d'autres attributs pour stocker les différents valeurs nutritionnelles (énergie, sel, etc...)
 - une liste d'objets **Ingredient**
 - une liste d'objets **Additif**
 - une liste d'objets **Allergene**
- Mettez en place toutes les annotations JPA sur vos entités
- Créez une classe exécutable **IntegrationOpenFoodFacts** qui va mettre en base de données les produits extraits du fichier.
- Pour chaque produit :
 - Commencez par regarder si la marque existe ou pas en base de données
 - Si elle n'existe pas créez la et associez la au produit.
 - Si elle existe récupérez la et associez là au produit
 - Traitez la catégorie de la même manière
 - Puis les ingrédients et les allergènes de la même manière.
- PS : faites attention aux transactions si vous voulez que les commits soient effectués.

