Gestion des avis (Partie 07)

# TP02 du module 05 – Spring Data REST

|  |
| --- |
| Ce TP sera un TP fil rouge du cours   * Manipulation de requêtes spécifiques |

|  |
| --- |
| **Durée estimée** |
| 30 à 45 minutes |

Contexte

En vous appuyant sur l’ensemble des concepts abordés dans le cours, mettez en place une application Spring Boot permettant la gestion des avis d’une cave à vins.

L’application de cave à vin sera une application d’API Web sécurisée à la fin des itérations.

L’application de gestion des avis sera aussi une application d’API Web :

* Elle doit permettre de gérer les avis des clients sur le vin commandé.
  + La date de l’avis
  + L’identifiant de la bouteille commandée, ainsi que de la Région et couleur
  + La quantité commandée par le client
  + Une note (entre 0 et 5), un commentaire et le pseudo du client (permettre de protéger les informations du client RGPD)
* Le but étant de faire des statistiques pour le propriétaire du Vignoble

Dans les itérations de ce module ; nous réaliserons uniquement du back de l’application. En particulier, la couche DAL et les BO pour le moment.

Cette application se fera avec Spring Boot.

Déclarer la structure des données en Anglais

Dans les itérations de ce module, nous allons développer l’API Web.

Enoncé

# Réalisation :

Les niveaux 1 sont obligatoires,

Les niveaux 2 sont plus poussés et optionnels

# Niveau 1 : Appel des requêtes spécifiques avec paramètres simples

* Tester les 2 méthodes de recherche sur la note des Avis
  + Aucun code n’est à ajouter car nous avons déjà déclaré les méthodes dans le module 03

# Niveau 1 : Appel des requêtes spécifiques sur des associations

* Dans AvisRepository, tester la méthode de findByClientPseudo
* Dans AvisRepository, tester la méthode de findByClientQuantiteCommandeeGreaterThan

# Niveau 2 : Clef composite et requête spécifique

* Une clef composite est un objet complexe
  + Il faudrait passer un JSON dans l’URL
  + Ce n’est pas possible
  + Il n’est donc pas envisageable de travailler sur des clefs de ce type
* Il faut donc repenser son projet :
  1. Soit ; mettre en place un @RestController et un @Service pour manipuler la méthode du Repository en passant les données de la clef composite en plusieurs paramètres dans l’URL
  2. Soit ; repenser les BO pour manipuler des clefs primaires simples
* Pour manipuler la méthode findByBouteille : il faut travailler sur l’association Avis-Bouteille
  + Le Document Bouteille a une clef composite
  + Demandez-vous qu’elle solution est viable pour manipuler cette méthode entre les 2 choix
  + Et mettre en place votre solution.

# Niveau 2 : LocalDateTime et requête spécifique

* Pour utiliser la méthode findByDateBetween ; il faut pouvoir passer des LocalDateTime par l’URL
* Il s’agit d’objet, mais il y a une solution en utilisant l’annotation @DateTimeFormat
* Mettre en place la solution et la tester avec l’URL :

http://localhost:8080/avis/search/findByDateBetween?deb=2023-07-13T00:00:00.000Z&fin=2023-07-31T00:00:00.000Z