Projet Java API : La librairie Java

Archive à rendre sur Netypareo pour vendredi 17h

Vous devez créer une Api avec Java et Spring pour la gestion d’une librairie.

Base de données : H2 en fichiers

**Commencez par modéliser les entités contenues dans cette description et attendez la validation de l’intervenant avant de continuer.**

Notre libraire vend des livres dont nous devons stocker l’ISBN, le titre, le nom et prénom de l’auteur ou son pseudonyme, un prix de ventre hors taxes et son année de parution. Nous avons des livres particuliers : les livres pour enfant où nous devons savoir à partir de quel âge le livre peut être lu. Chaque client est identifié par un numéro de client, son nom, son prénom et son adresse. Les clients ont tous des préférences de lecture parmi : SCIFY, COMEDY, HORROR, TECHNICAL, ROMANCE, THRILLER, FANTASY. Ces préférences sont stockées dans son profil. Le profil contient également le pays du client pour y référer la langue dans laquelle il achète les livres : FR, En, IT, etc.. Toutes les commandes des clients doivent être stockées, une commande concerne un seul client et un ou plusieurs livres mais il ne peut commander qu’un exemplaire de livre par commande. Une commande est passée à une date précise et nous devons pouvoir en changer le statut : en préparation, expédiée, livrée, annulée. La commande comporte le montant total hors taxe et le montant avec la TVA dont le taux change selon le pays du client.

Une fois les entités validées, vous devez toutes les routes qui vont sembles logiques pour gérer notre bibliothèque. **Pour ce faire, avant de coder les services, repo et contrôleurs, vous devez faire valider une liste des routes à votre intervenant.**

Liste des routes

* Author
  + CR et en optionel UD
* Book
  + CRUD, findByAuthor, finfByTitle
* KidBook
  + CRUD, findByAuthor, findByTitle, findByAge
* Profil
  + CR
* Customer
  + CRU
* Order
  + CRU, findByCustomer, findByDate, findBYStatus, Chiffre d’affaires reel HT(uniquement les commandes livrées) , chiffre prévisionnel HT (toutes les commandes)

Ajoutez une exception pour les données non trouvées avec un ControlAdvisor.

Ajoutez une documentation « openapi » à toutes vos routes.

Ajoutez un test unitaire pour tester vos calculs de TVA.

Testez vos routes les plus importantes.

Facultatifs :

* Ajoutez les routes nécessaires pour filtrer les commandes par années, auteur et/ou titres.
* Ajoutez un accès administrateur qui peut voir toutes les commandes et les clients ne peuvent voir que leurs commandes avec une authentification par log/pass en clair.
* Ajoutez actuator à votre application.

Peu importe l’ordre

* Ajoutez des DTO
* Testez toutes vos routes correctement.
* Ajouter d’autres exceptions gérées business.
* Chiffrez le pass en base.
* Authentifiez les utilisateurs avec JWT.