Hibernate6 : Nullifier les proxy avec Jackson : Hibernate5Module et Hibernate6Module

Lorsque l’on appelle une url en Spring qui renvoie une entité, qui contient des associations (agrégats) vers d’autres entités.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

Lors de l’appel, si l’on à pas utilisé de D.T.O. on aura l’erreur 500

2023-10-18T08:59:47.478+02:00 ERROR 6020 --- [p-nio-80-exec-8] o.a.c.c.C.[.[.[.[dispatcherServlet] : Servlet.service() for servlet [dispatcherServlet] in context with path [/invoise] threw exception [Request processing failed: org.springframework.http.converter.HttpMessageConversionException: Type definition error: [simple type, class org.hibernate.proxy.pojo.bytebuddy.ByteBuddyInterceptor]] with root cause

com.fasterxml.jackson.databind.exc.InvalidDefinitionException: No serializer found for class org.hibernate.proxy.pojo.bytebuddy.ByteBuddyInterceptor and no properties discovered to create BeanSerializer (to avoid exception, disable SerializationFeature.FAIL\_ON\_EMPTY\_BEANS) (through reference chain: java.util.ArrayList[0]->com.udemy.entity.Invoice["**customer**"]->com.udemy.entity.Customer$HibernateProxy$HBmPEffV["address"]->com.udemy.entity.Address$HibernateProxy$xnPsr7qA["hibernateLazyInitializer"])

Une image contenant texte, capture d’écran, logiciel, ligne

Description générée automatiquement

Cela vient du fait que : Hibernate essaie de déproxifier récursivement via HttpMessageConversion de Jackson des attributs entités d’entité.

Solution :

Mettre dans le pom.xml du projet

Une image contenant texte, Police, capture d’écran

Description générée automatiquement

Puis dans la classe principale

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Relancer l’app, l’appel fonctionnera

Solution 1 FLEXIBLE : L’initialisation des proxy à posteriori

Solution 2 PERFORMANTE : Le fetch à priori - requêtes SQL plus adaptées au niveau des repository – lire les informations qui nous intéressent lors de la requête originale

**Solution 1 : Déproxifier les propriétés des entités qui nous intéressent côté client avec dans la couche service Hibernate.initialize**

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Attention cette solution génère de multiples requêtes secondaires (performance !)

**Solution 2 : réaliser des requêtes adaptées avec les Repository (= améliorer les C.R.U.D JPA) et les entity graphs**