MIGRATION 🡪 Comment utiliser liquibase dans un projet Springboot

1. Dans le pom.xml, on ajoute la dépendance :

<dependency>

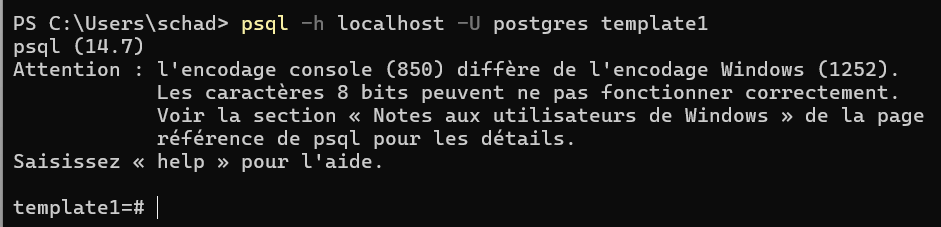
<groupId>org.liquibase</groupId>

<artifactId>liquibase-core</artifactId>

</dependency>

1. Aller dans le powershel et lancer les commandes suivantes pour créer une database :

psql -h localhost -U postgres template1//on cherche à trouver pour l’hôte **localhost**, au nom d’utilisateur **postgres**, sa base de donnée par défaut **template1**

****

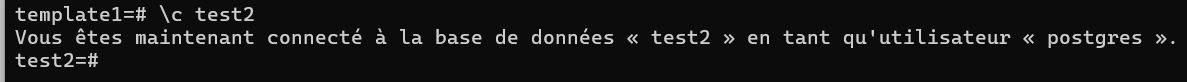
On pourra alors lancer la création d’une nouvelle base de donnes basée sur celle par defaut avec :

CREATE DATABASE test2;

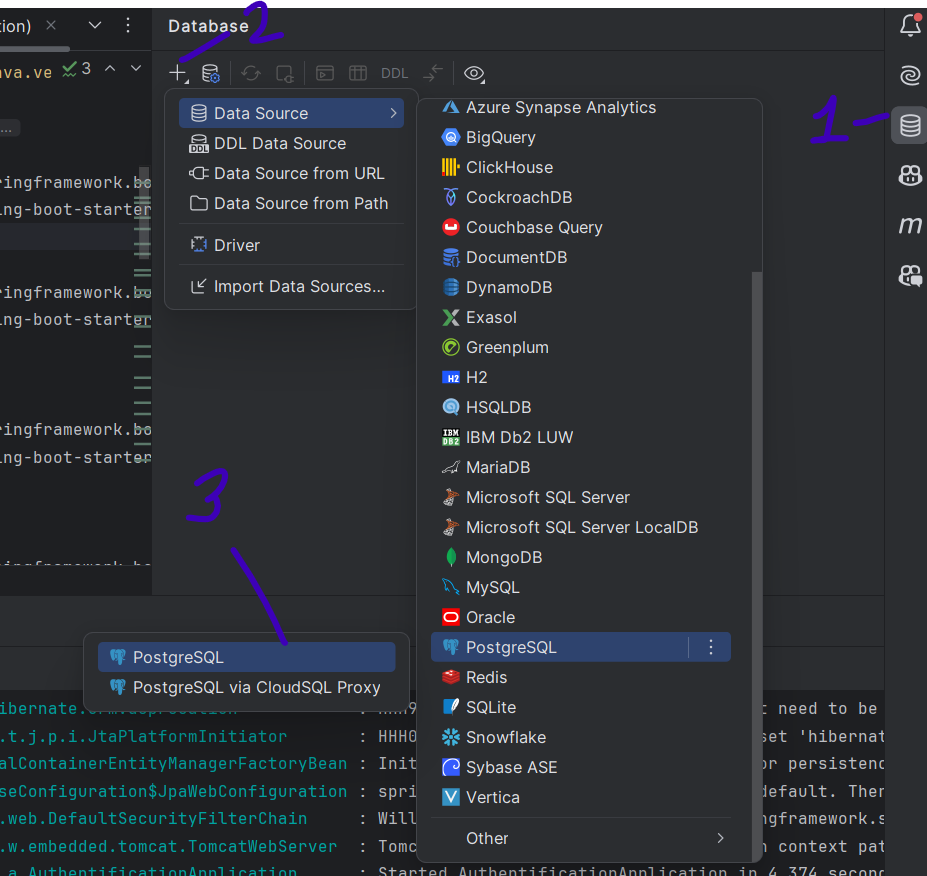
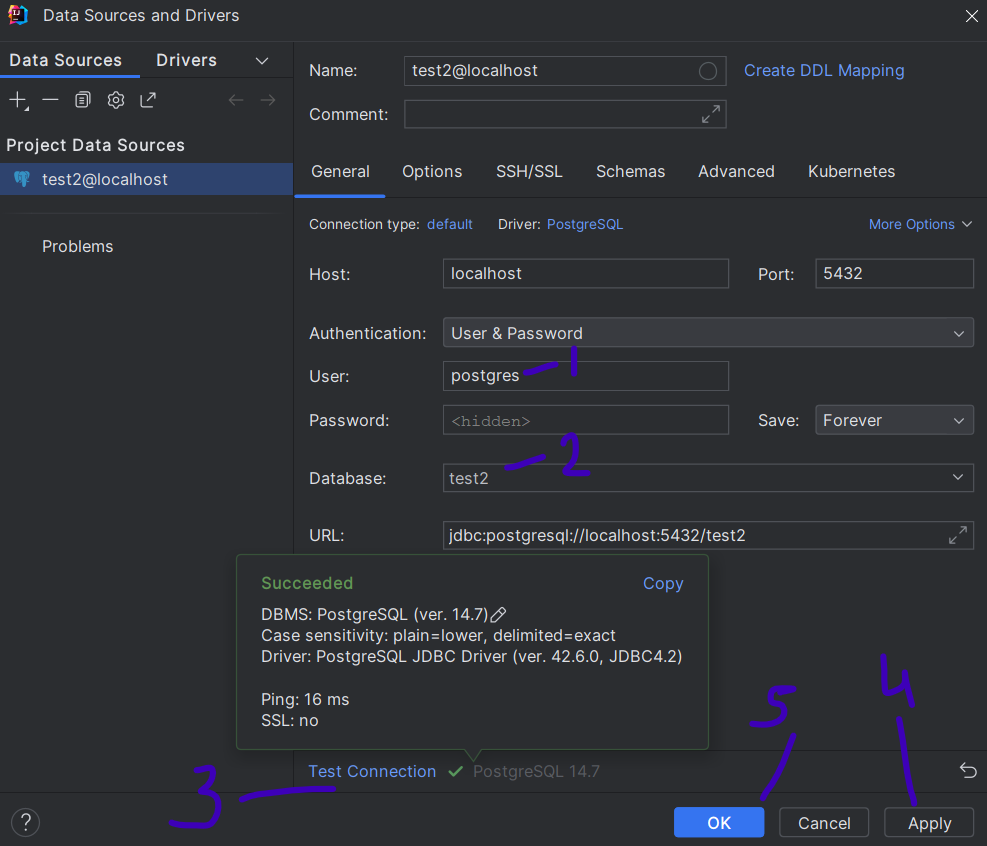


Et on finit en se connectant a la base de donnes :

\c test2



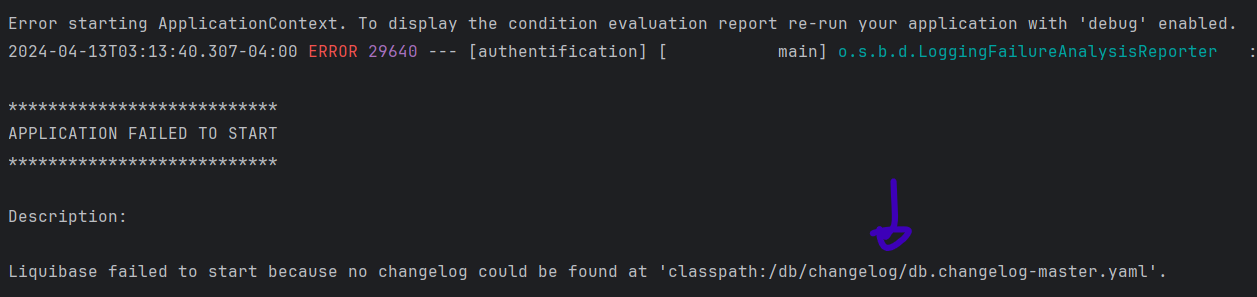
1. Maintenant, on doit connecter le projet a la base de donne dans inteliji, avec :

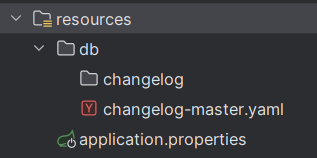
Avec un fichier.properties :

spring.application.name=authentification  
server.port=8181  
  
spring.datasource.driver-class-name=org.postgresql.Driver  
spring.jpa.database-platform=org.hibernate.dialect.PostgreSQLDialect  
spring.jpa.properties.hibernate.temp.use\_jdbc\_metadata\_defaults = false  
spring.jpa.properties.hibernate.format\_sql=true  
spring.datasource.url=jdbc:postgresql://localhost:5432/test2  
spring.datasource.username=postgres  
spring.datasource.password=''  
  
inf1013.jwt.secret="CECI EST LE MOT SECRET"  
inf1013.jwt.token-prefix=Bearer   
inf1013.jwt.token-header=authorization  
inf1013.jwt.token-issuer=PLUTON\_AUTH  
inf1013.jwt.token-expiration=2

1. Si on essayait maintenant de lancer notre projet, il planterait, car Liquibase a nécessairement besoin d’un fichier changelog-master.yaml, et par défaut, on vx que ce fichier soit dans le ressources/db/



Alors, on crée l’architecture suivante :



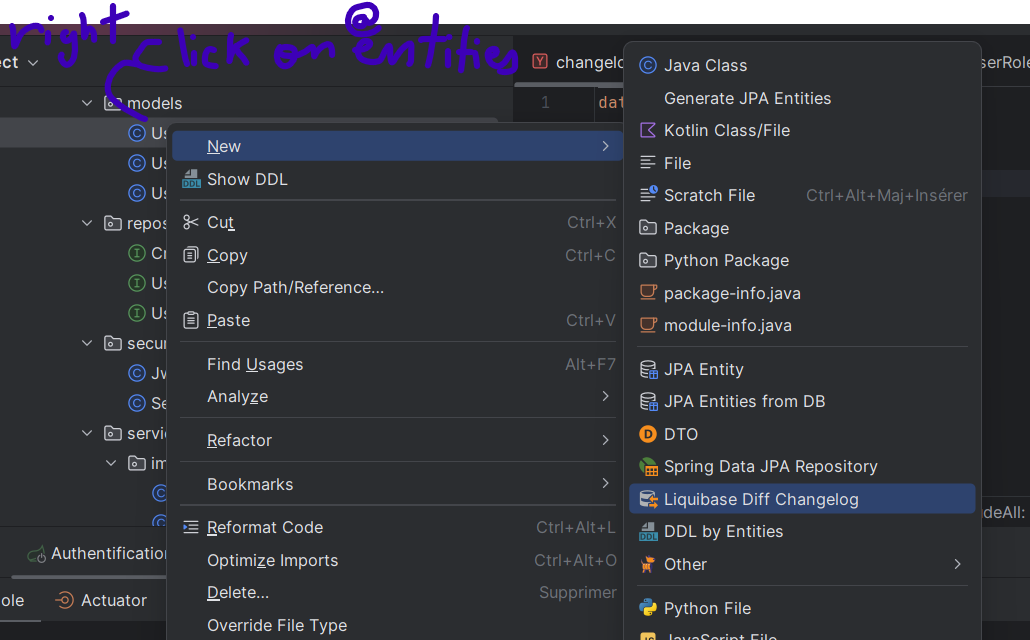
Et on ajuste le fichier.properties :

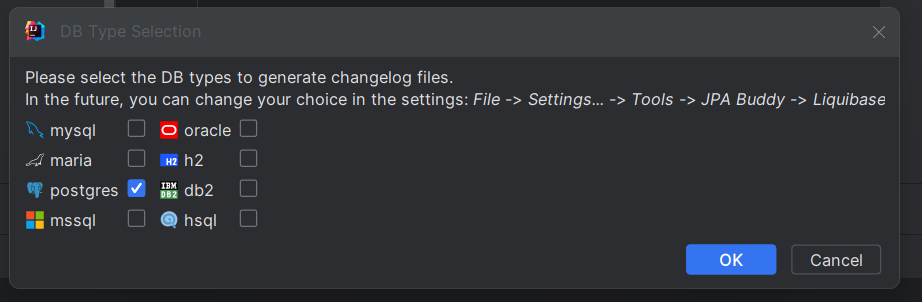
#ajout de liquibade  
spring.liquibase.enabled=true  
spring.liquibase.change-log=classpath:db/changelog-master.yaml

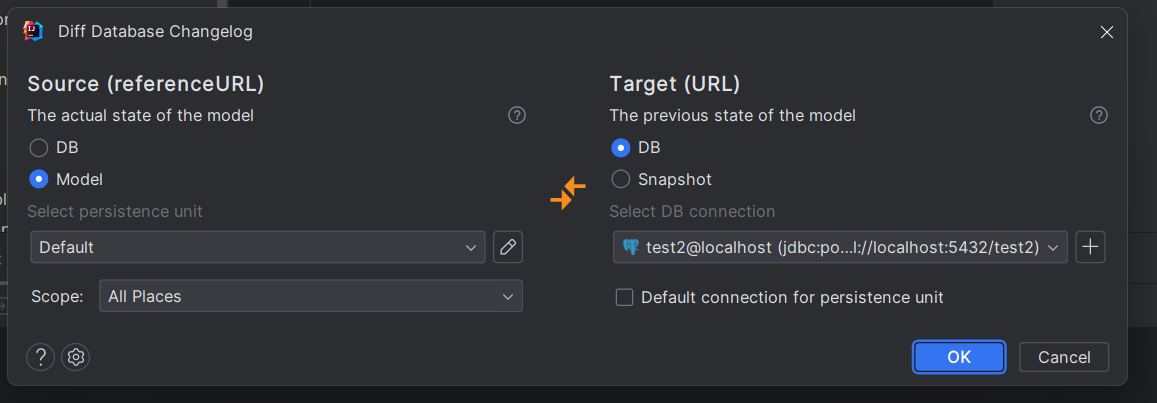
Puis le fichier changelog-master.yaml

databaseChangeLog:  
 - includeAll:  
 path: ./changelog/  
 relativeToChangelogFile: true

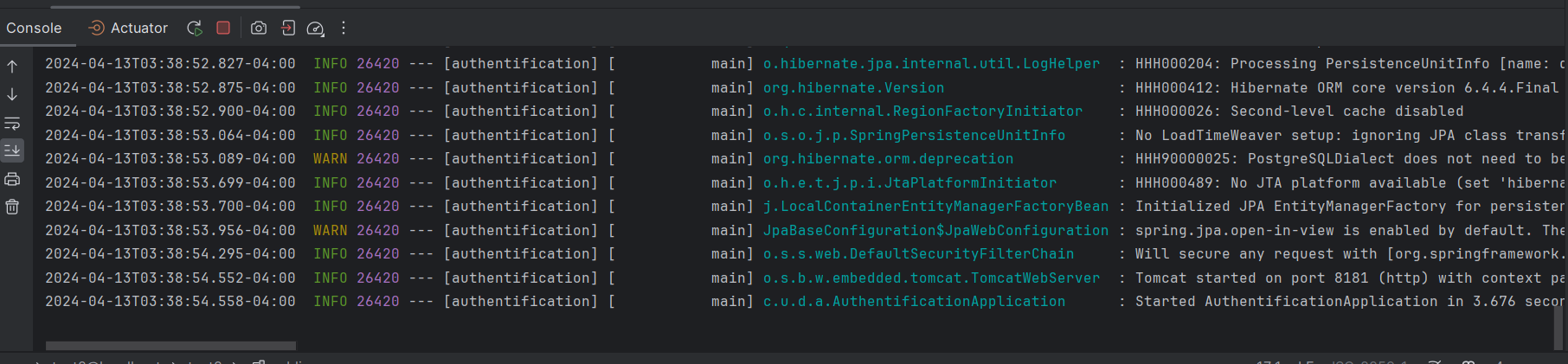
1. On px alors générer le fichier changelog avec l’extension : JPA buddy

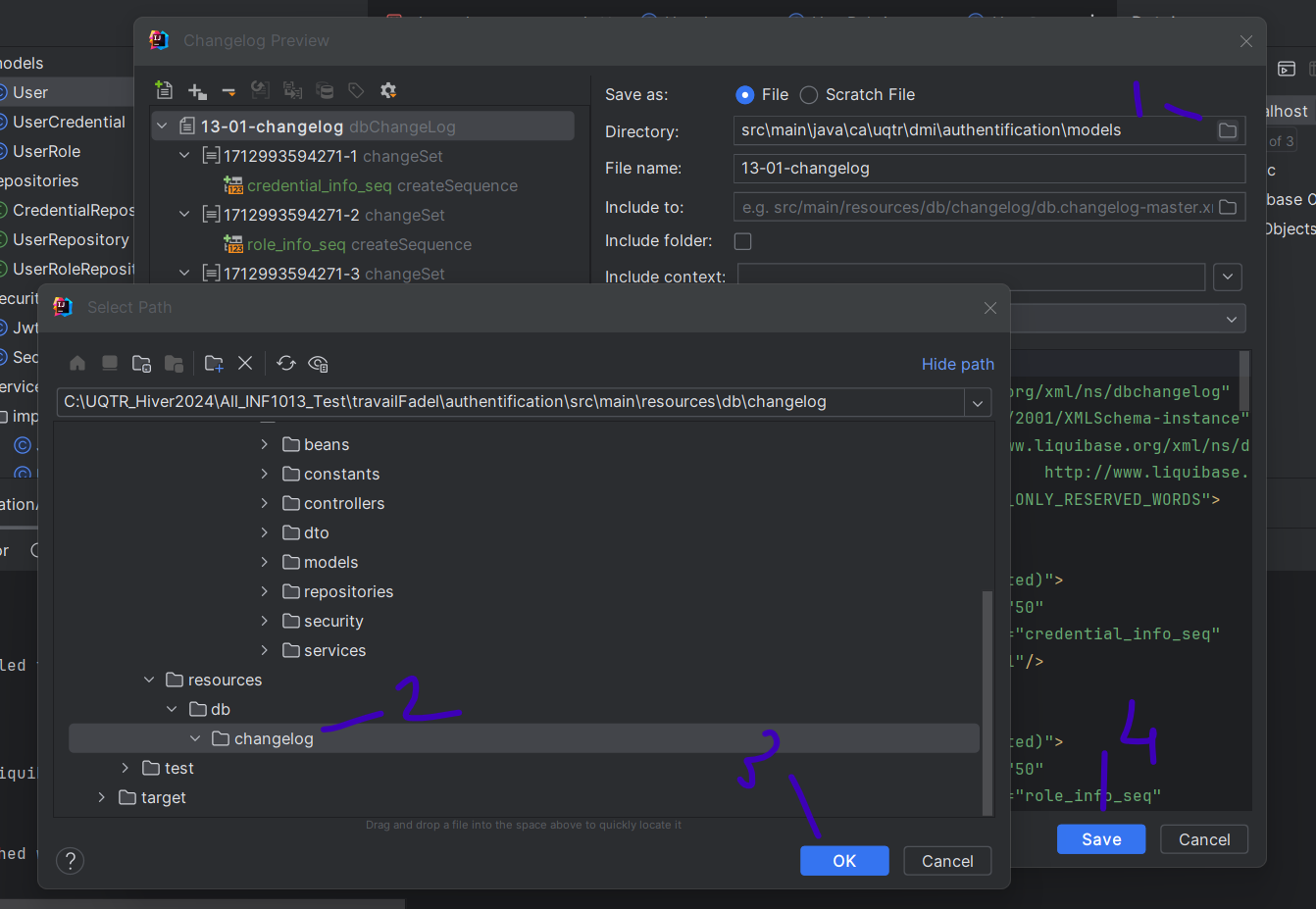






**Notons que cela ne génère pas les entités de la bd, directement. Pour les avoir, il faut utiliser DBL by entity(pas nécessaire, a moins de vouloir visionner les entités)**

**Et ca marche   
**



1. Il manque a setter de manière locale avant de déployer…