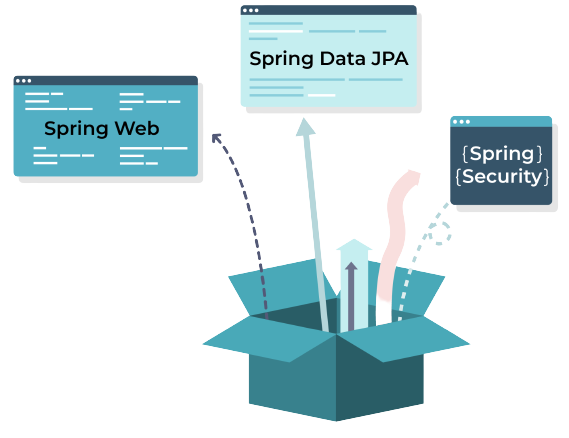
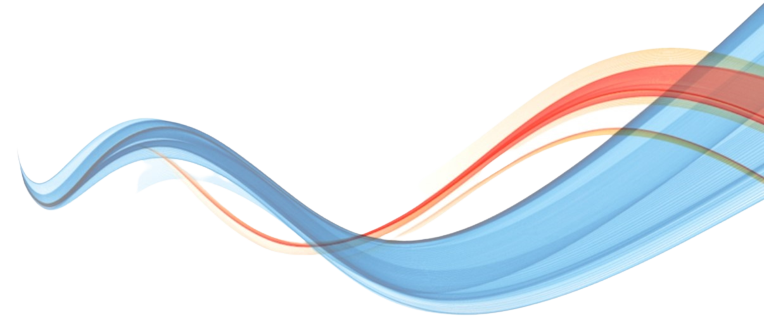
Compte rendu de :

SEANCE 03



**Réalisé Par :** AYOUJJIL Soukayna

**Département  :** Mathématiques et Informatique

**Filière :** II-BDCC

**Professeur :** M. Mohamed YOUSSFI

**Année Universitaire** **: 2021/2022**

Notions plus importantes :

* Pourquoi il faut utiliser des Framework ORM

• Le but d'Hibernate est de libérer le développeur de 95% des tâches de programmation liées à la persistance des données communes.

• Hibernate assure la portabilité de votre application si vous changer de SGBD.

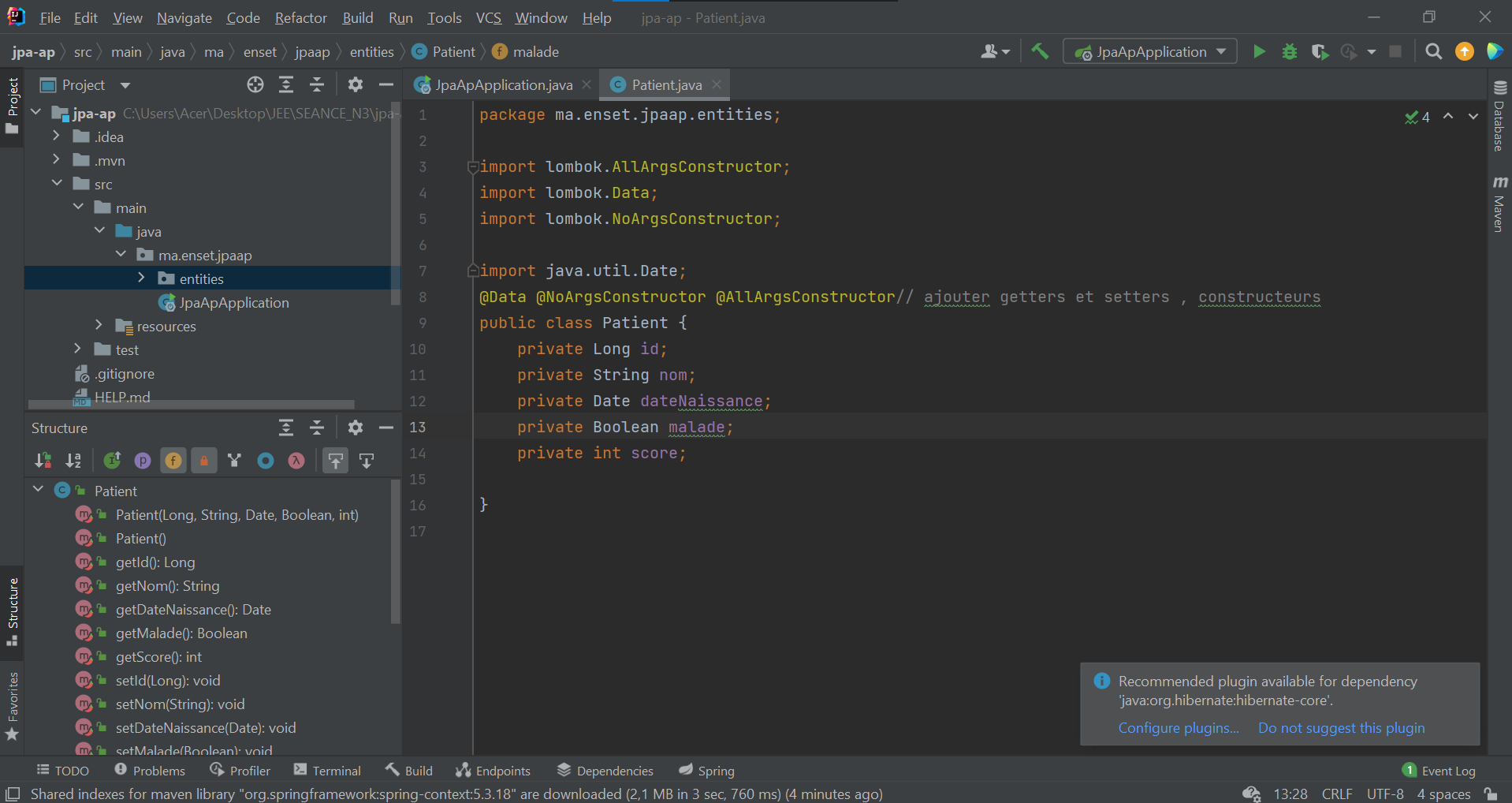
• Hibernate propose au développeur des méthodes d’accès aux bases de données plus efficace ce qui devrait rassurer les développeurs.

* PA est une spécification créée par Sun pour standardiser le mapping Objet relationnel.
* JPA définit un ensemble d’interfaces, de classes abstraites et d’annotations qui permettent la description du mapping objet relationnel
* Il existe plusieurs implémentations de JPA : Hibernate, Toplink, IBatis et EclipseLink
* Spring est Framework qui assure l’inversion de contrôle
* Spring peut s’occuper du code technique comme la configuration de JPA et la gestion des transactions.
* Avec Spring on peut simplifier le code de notre application en lui déléguant des tâches techniques.
* Spring Boot est une version de spring qui permet de simplifier à l’extrême, entre autres :

La gestion des dépendances Maven et Auto Configuration

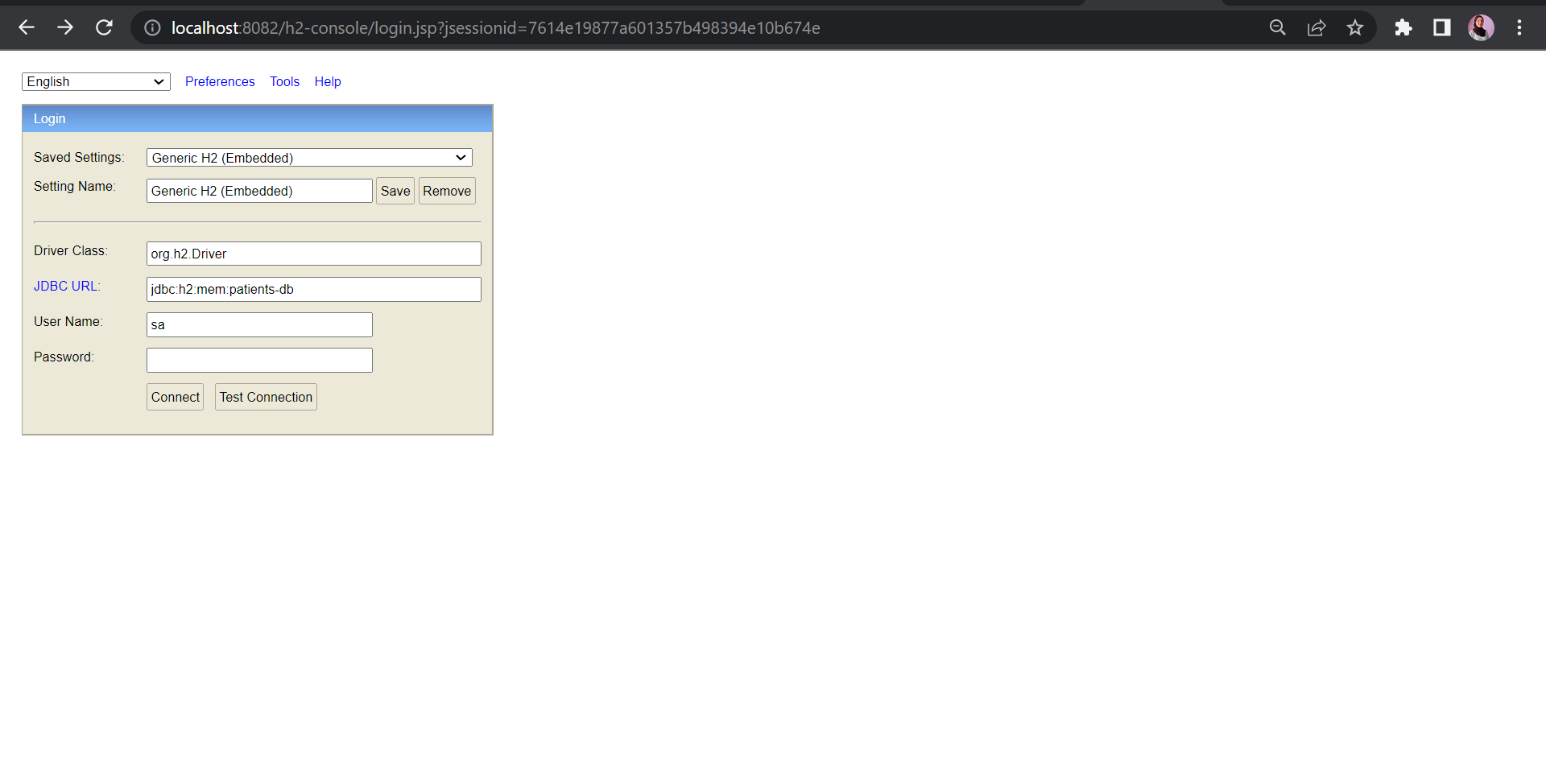
.

Etape 1 : Gestion des entités – Patient

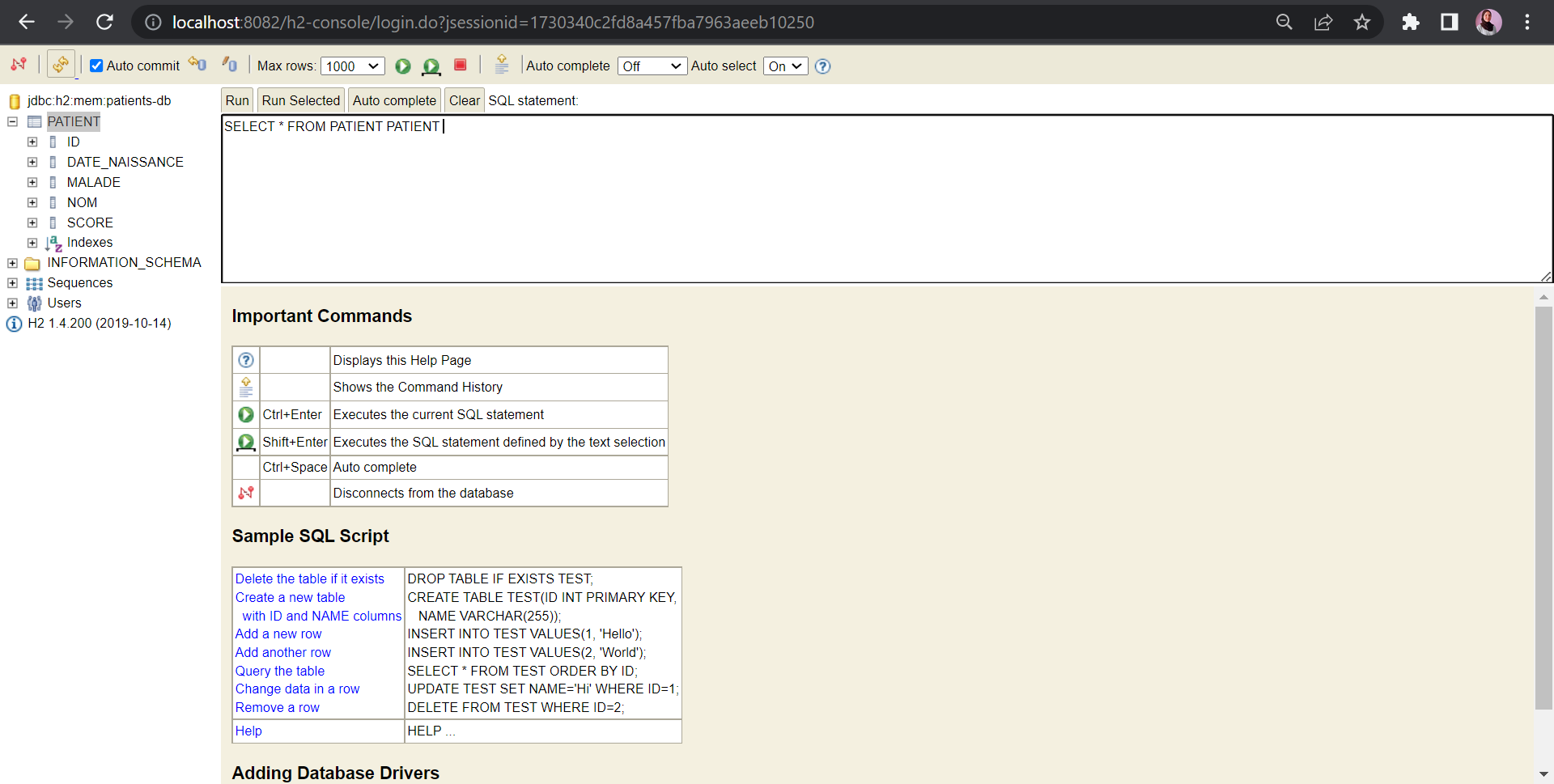


Etape 2 : Démarrer l’application Spring

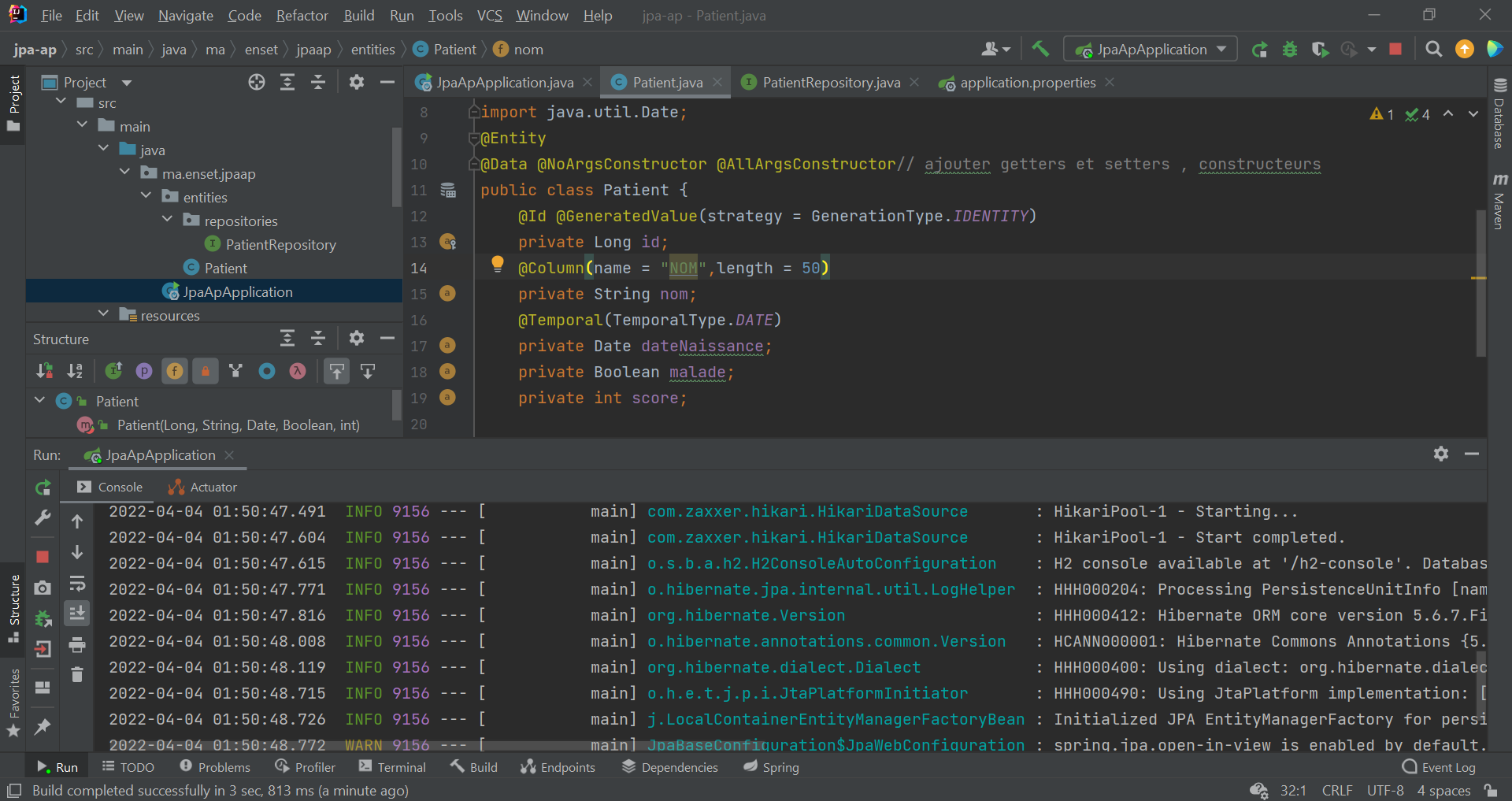
* Se connecter à la base de données



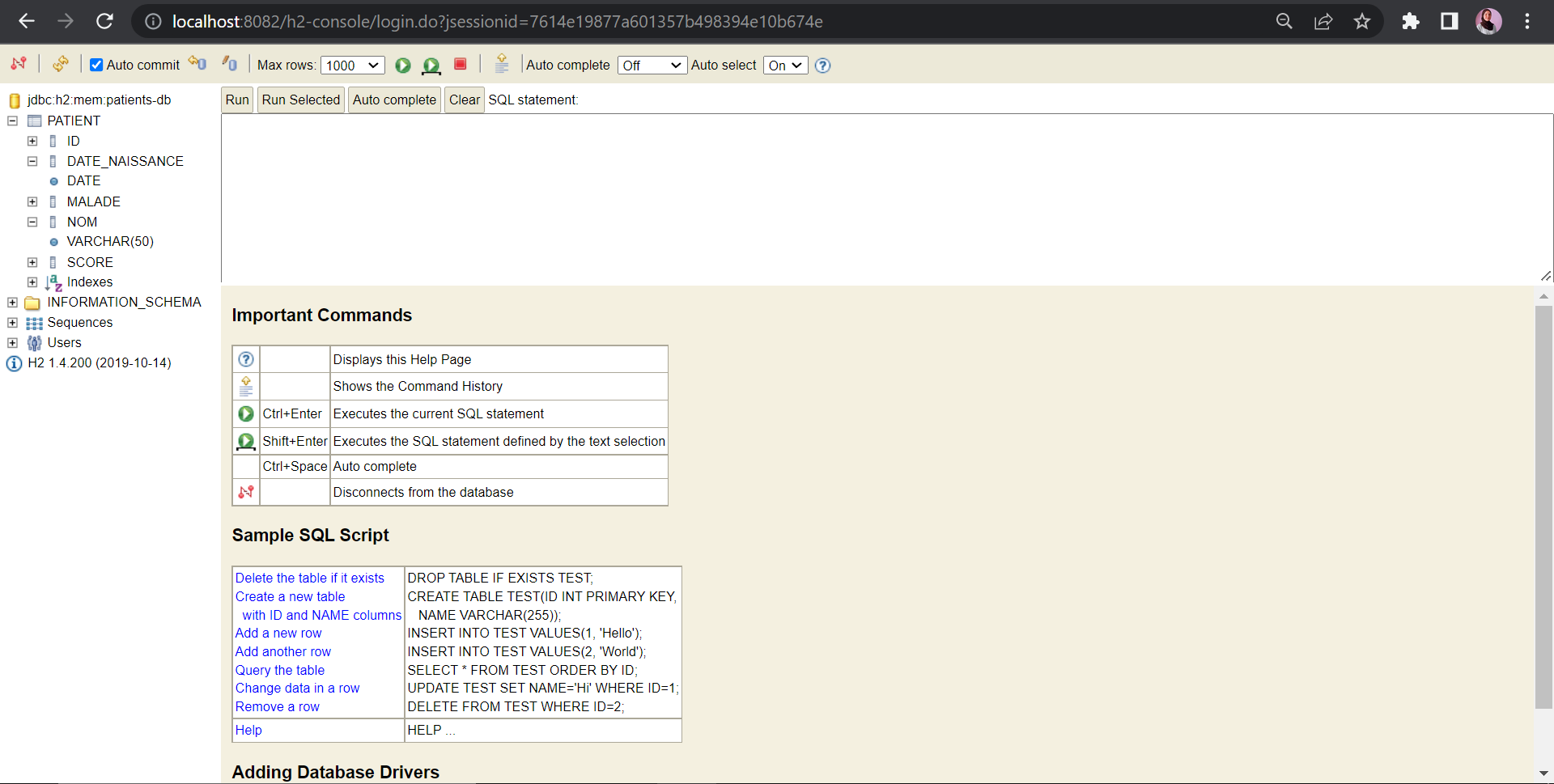
* Consulter tous les patients



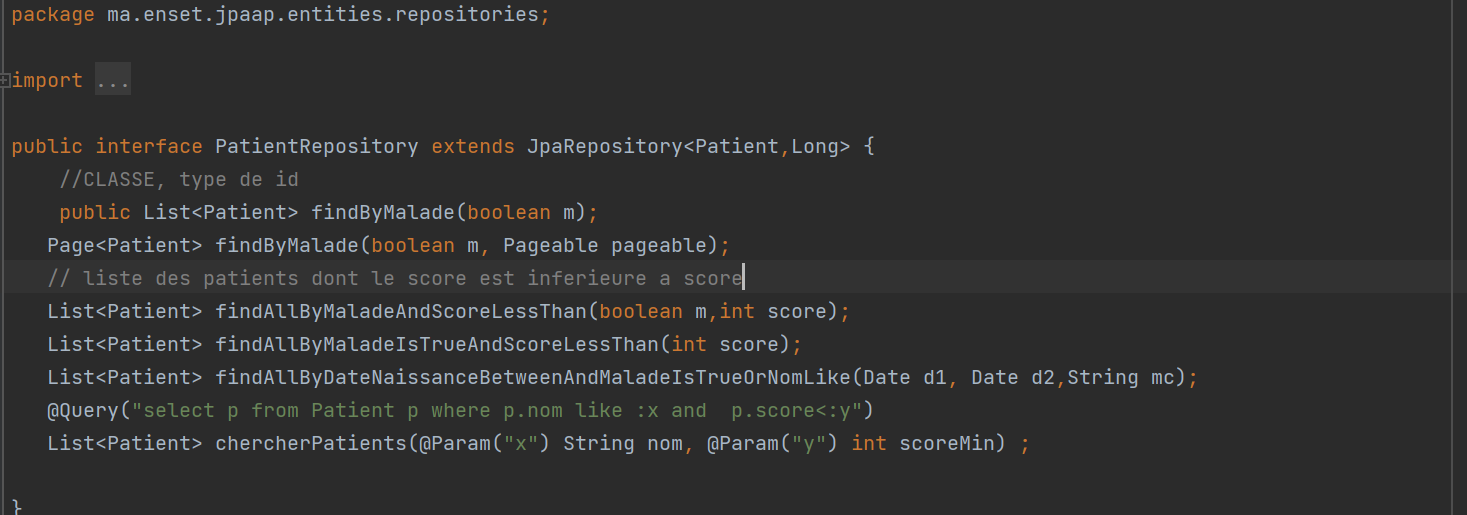
Etape 3 : Associer un champ de la colone à la propriété. + L’annotation @Temporal a la valeur de paramètre unique de type TemporalType. Il peut s'agir de DATE, TIME ou TIMESTAMP, selon le type SQL sous-jacent que nous voulons utiliser pour le mappage.



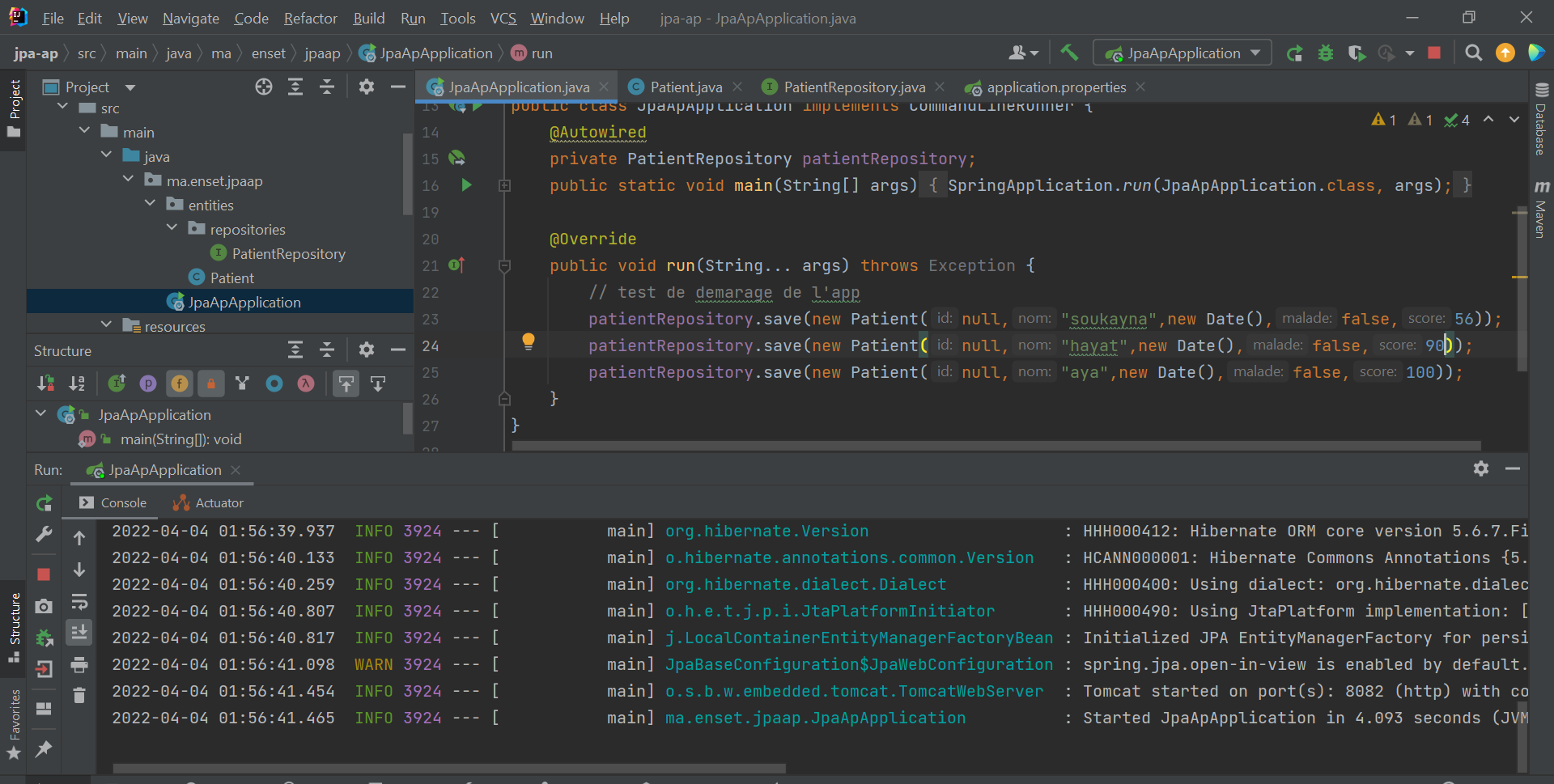
Etape 4 : Démarrer l’application Spring



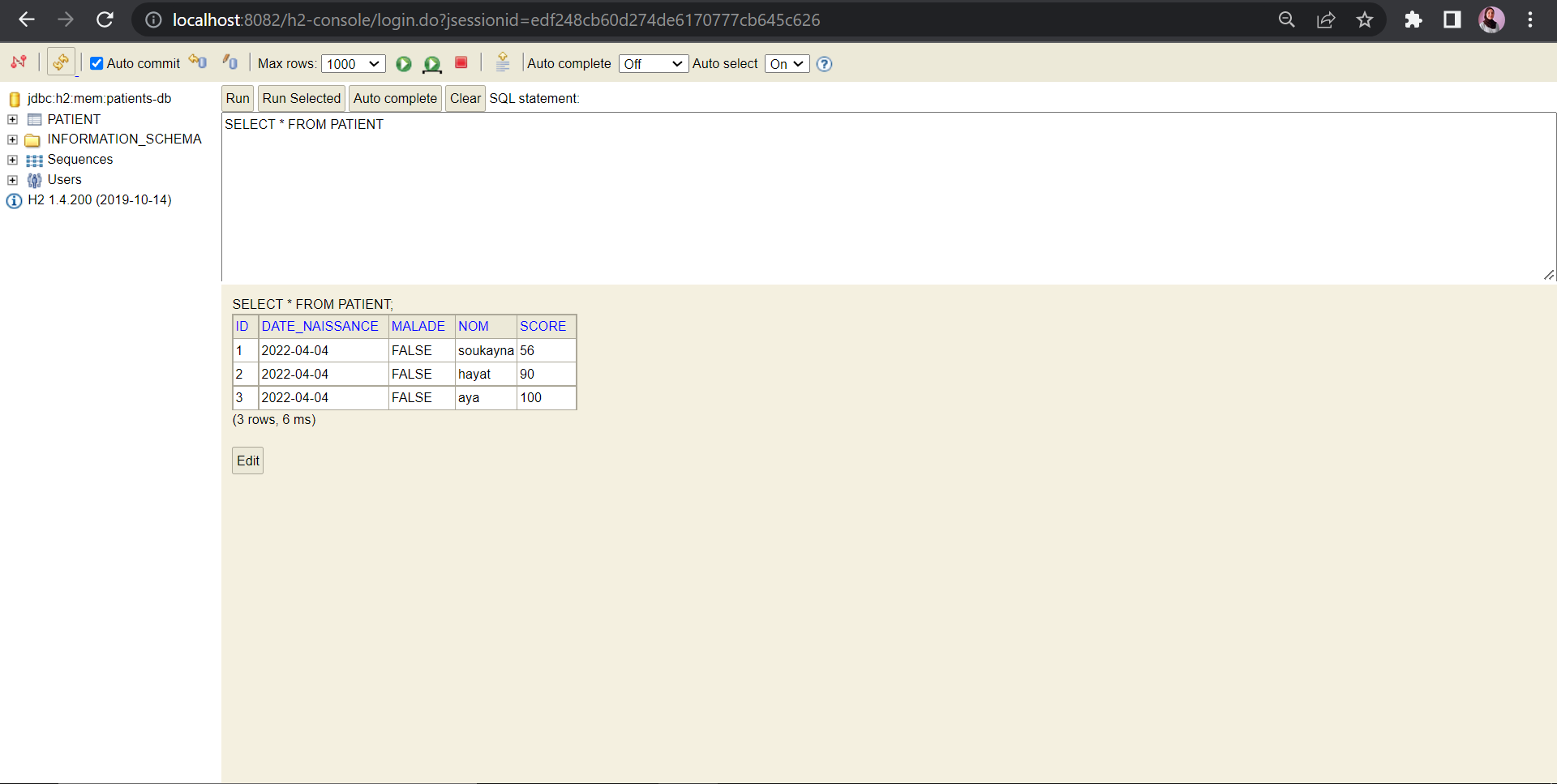
Etape 5 : Gestion des repositories – PatientRepository



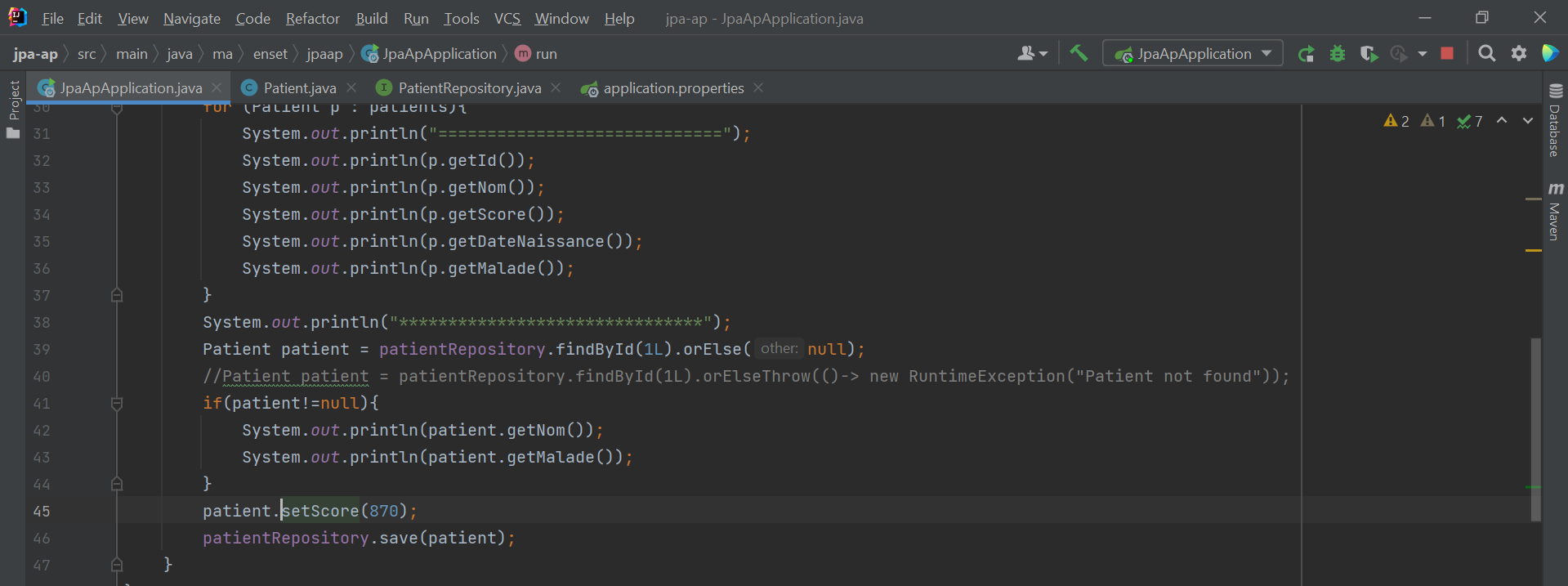
Etape 6 : Création des patients



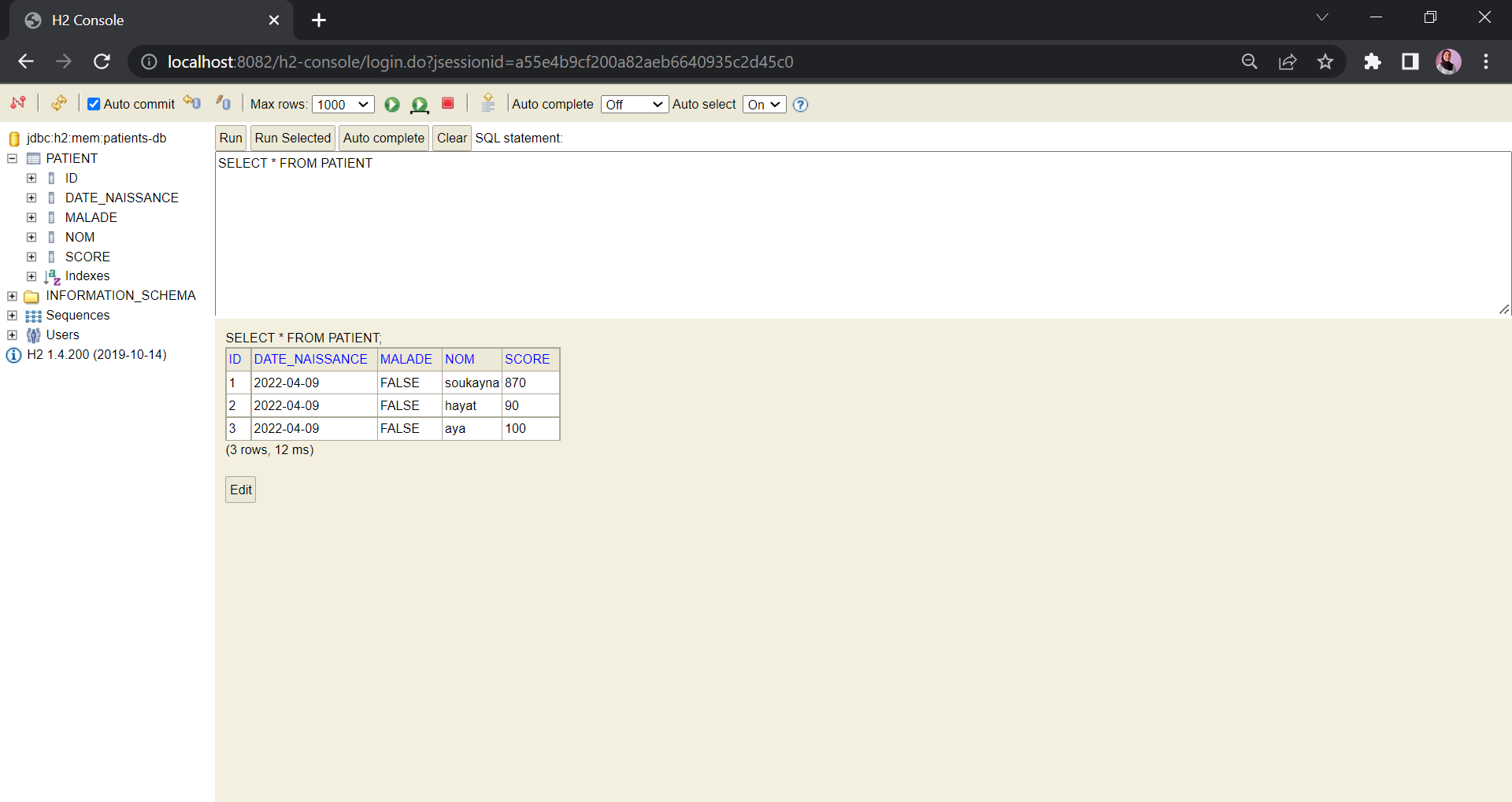
Etape 7 : Resultats d’execution – consulter tous les patients



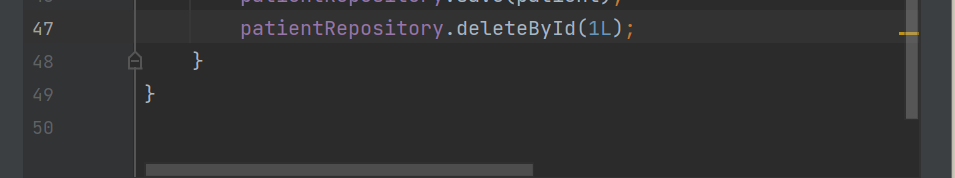
Etape 7 : Rechercher sur un patient par son id, afficher ses informations et modifier son score



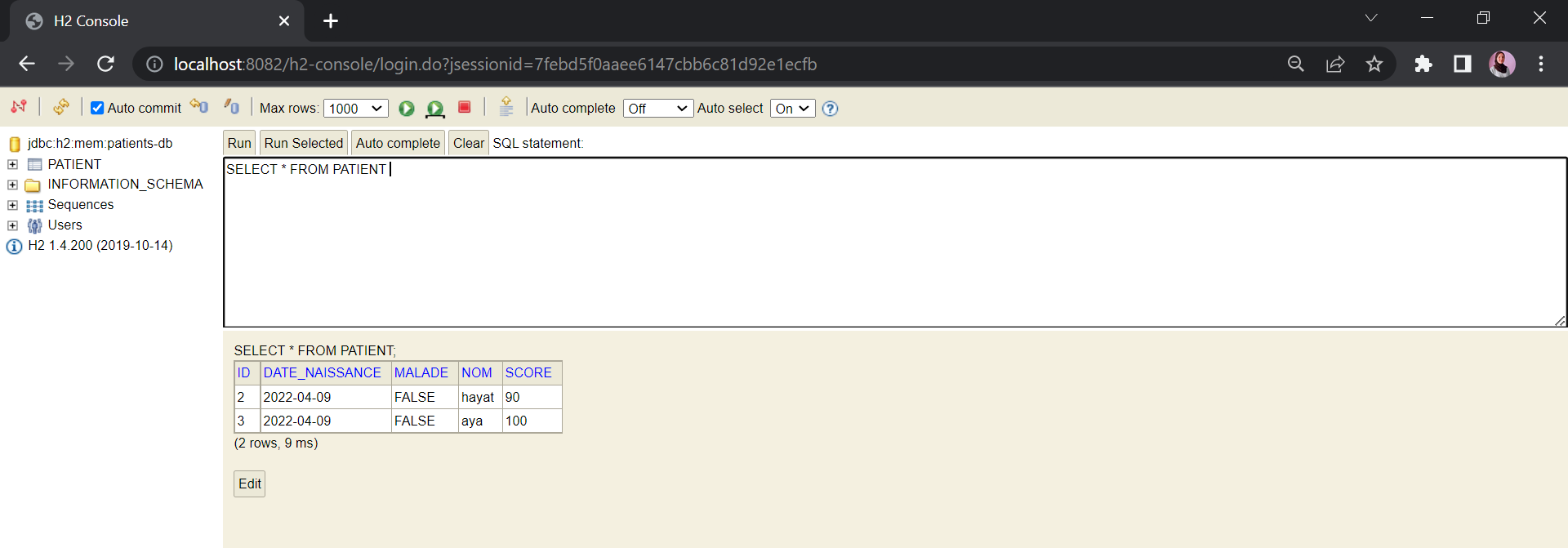
Etape 8 : Résultats des modifications apres demarage de l’application



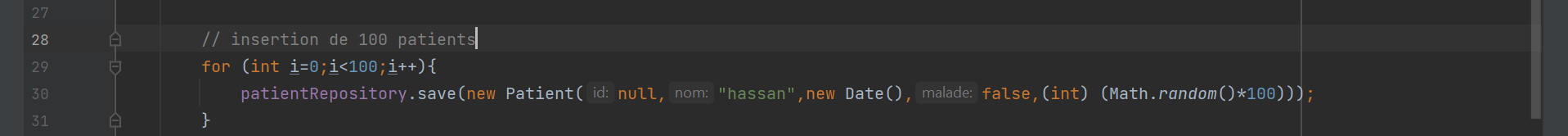
Etape 9 : Supprimer un patient



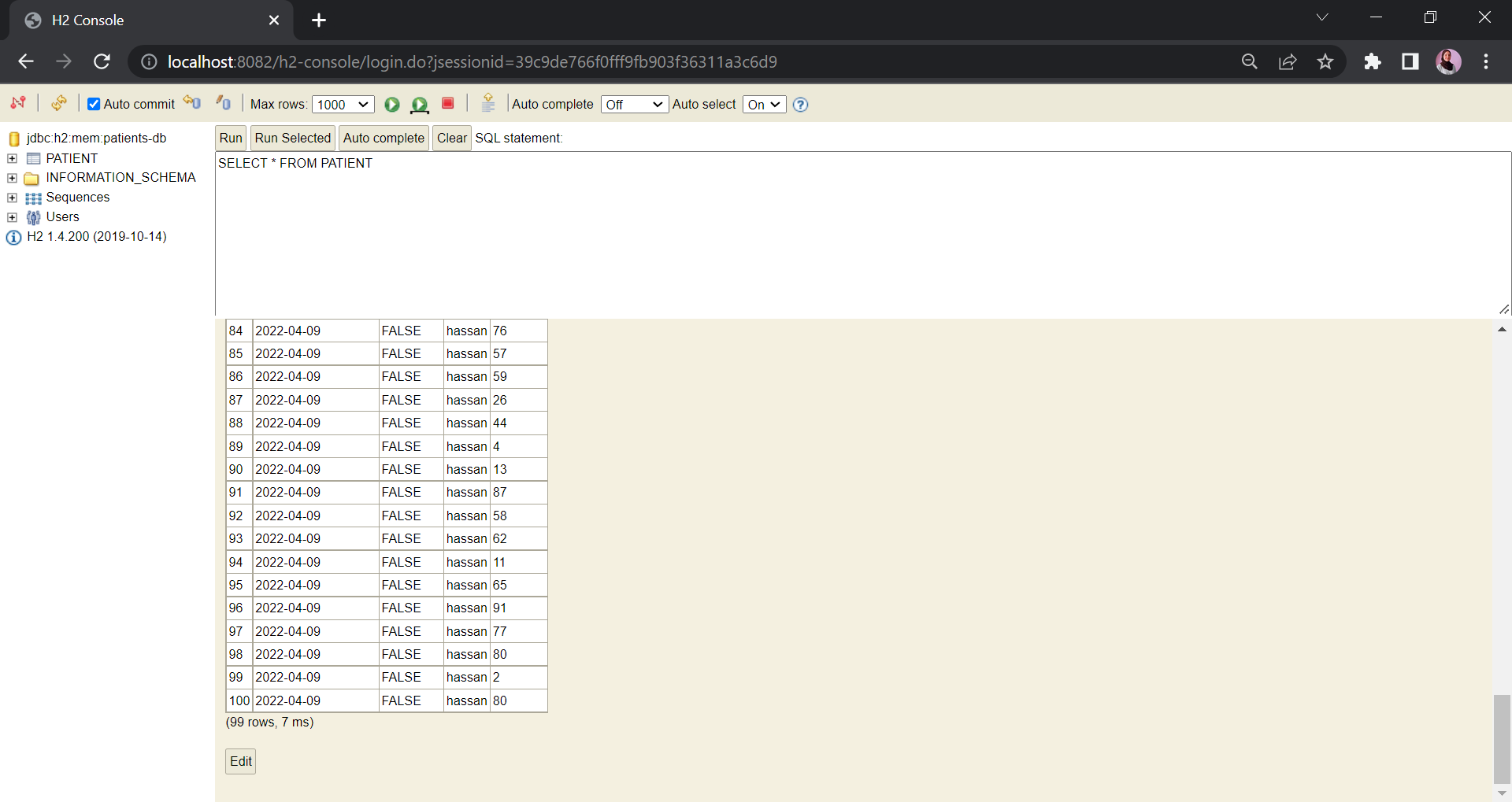
Etape 10 : Résultats coté base de données



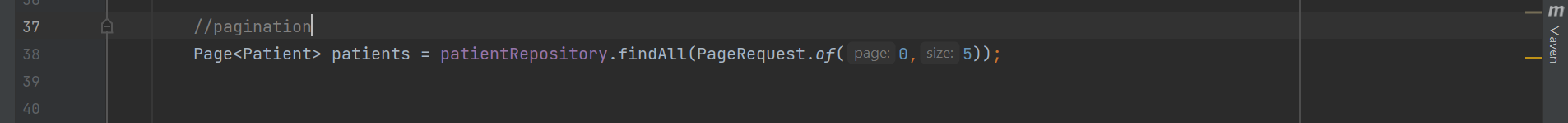
Etape 11 : Insérer 100 patients

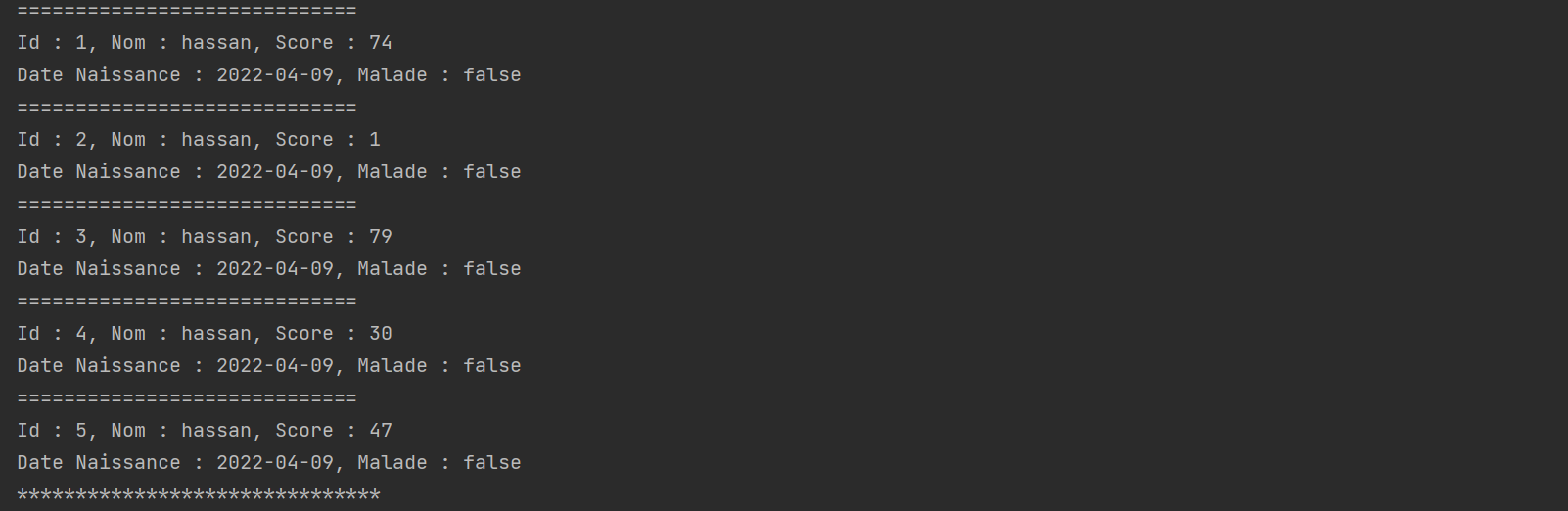


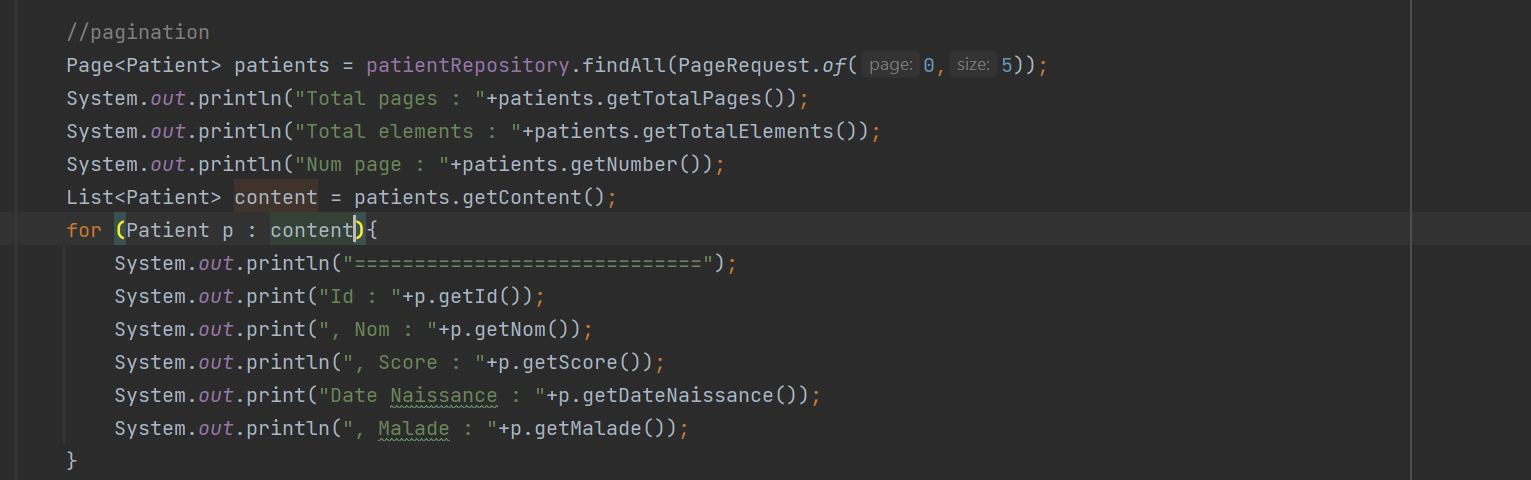
Etape 12 : Résultats

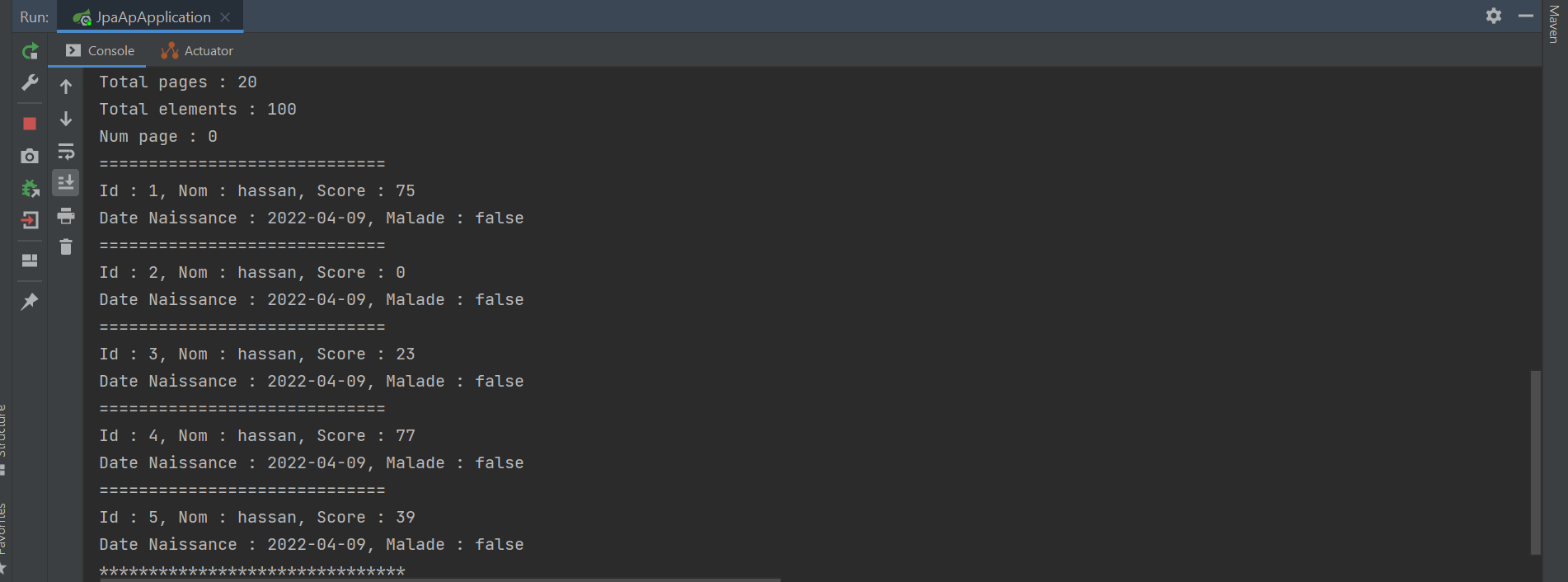


Etape 13 : Pagination

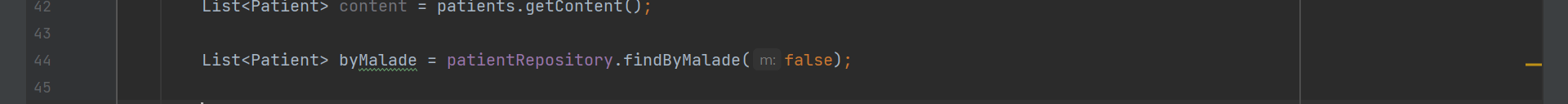


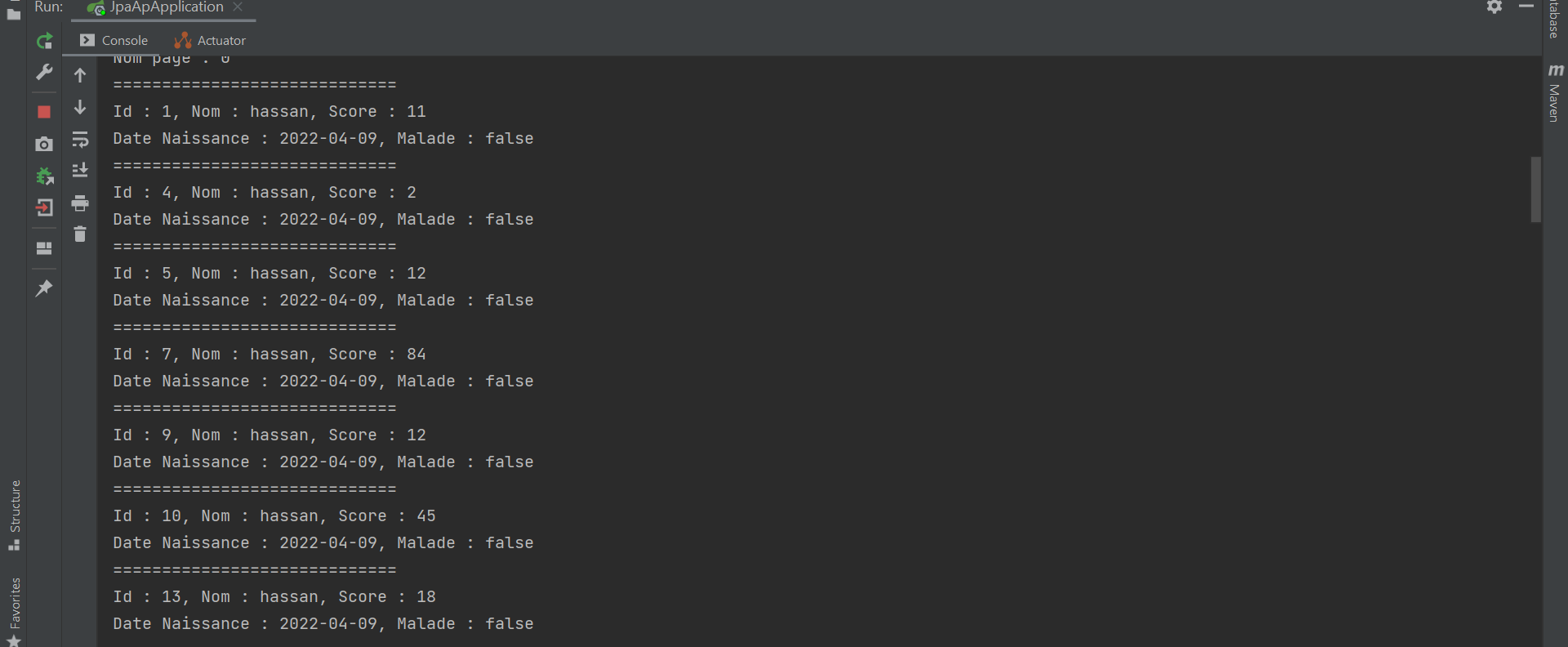




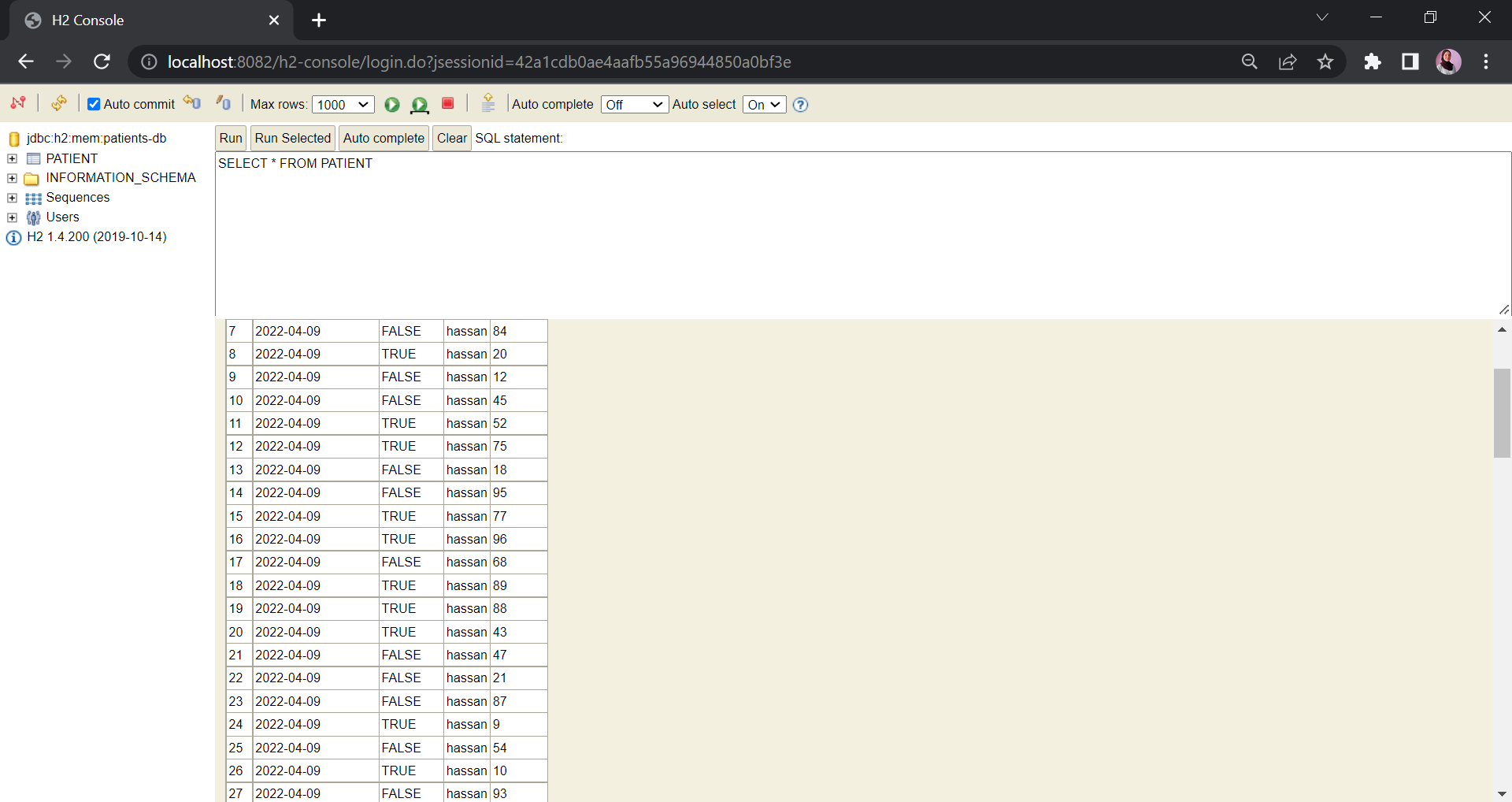


Etape 14 : Rechercher listes des patients qui ne sont pas malade

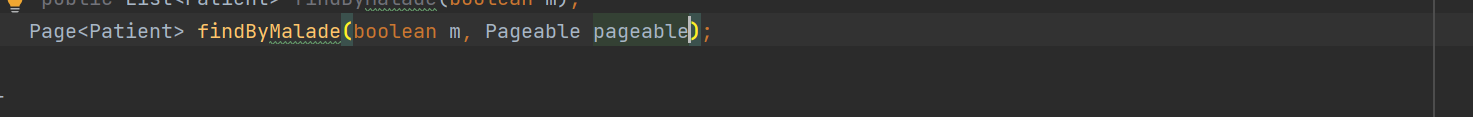


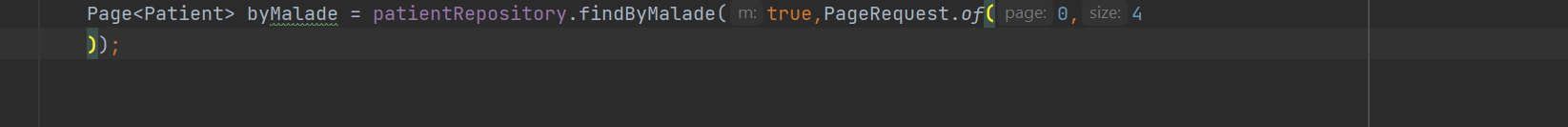


Etape 15 : Resultats



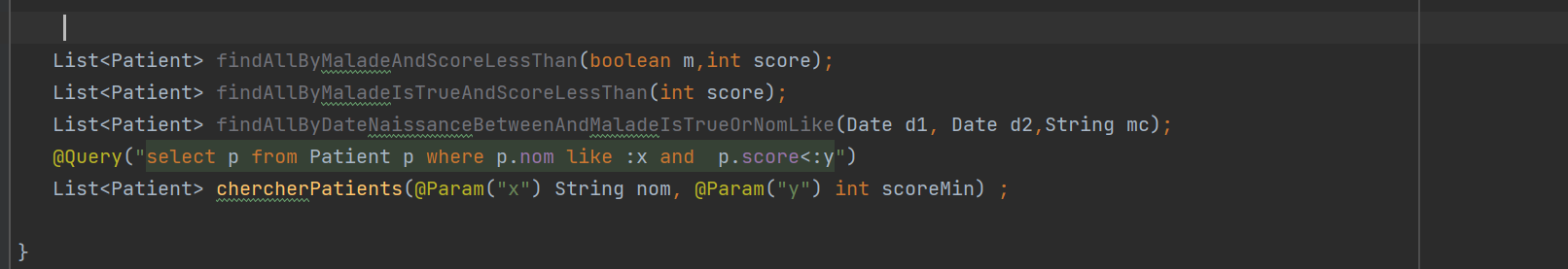
Etape 16 : Recherche + Pagination

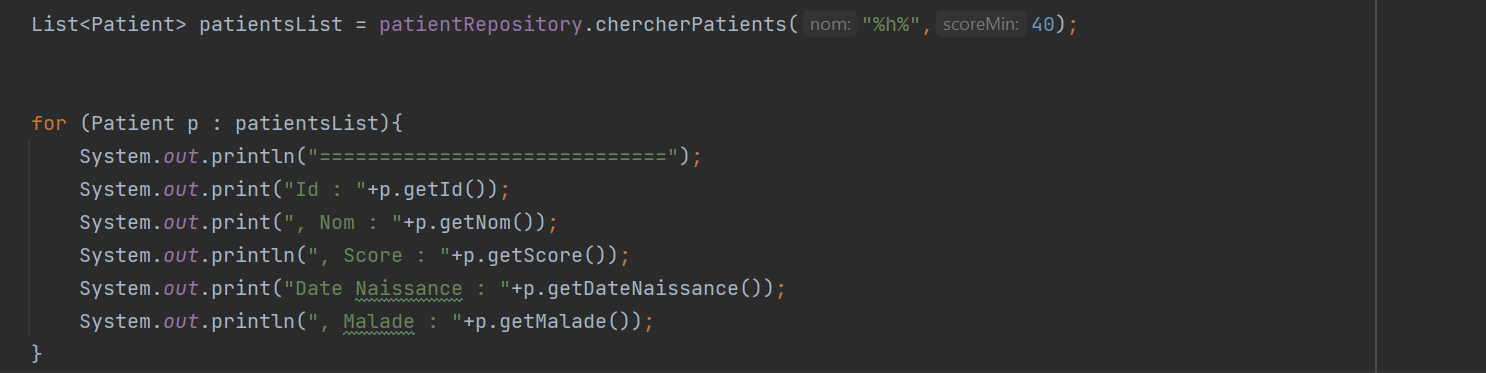


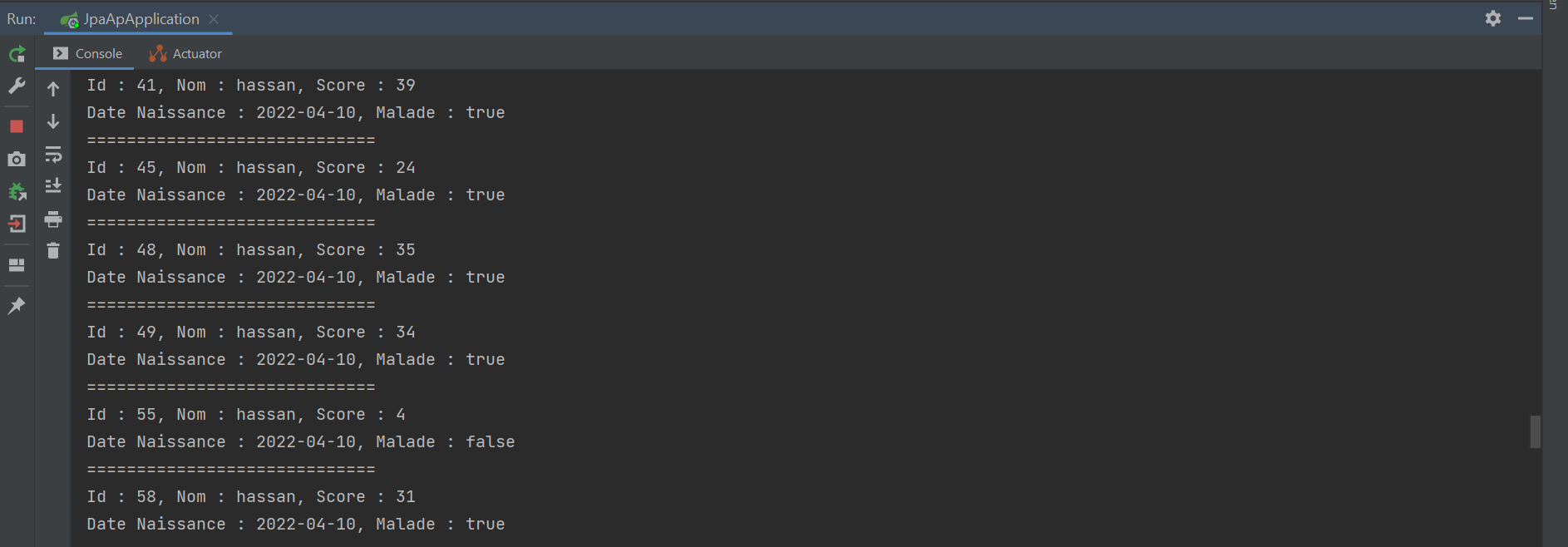




Etape 17 : Annotation @Query

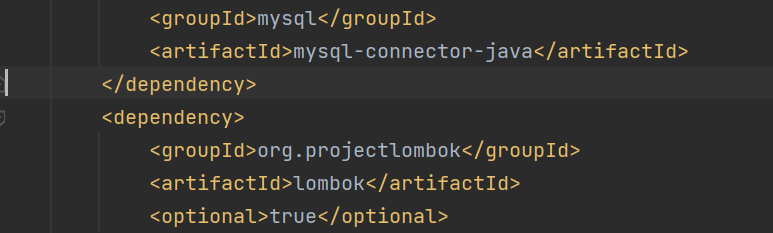






Etape 18 : Basculement vers la base de données MySQL

* Ajouter les dépendances



* application.properties



* Résultats du démarrage de l’application spring

