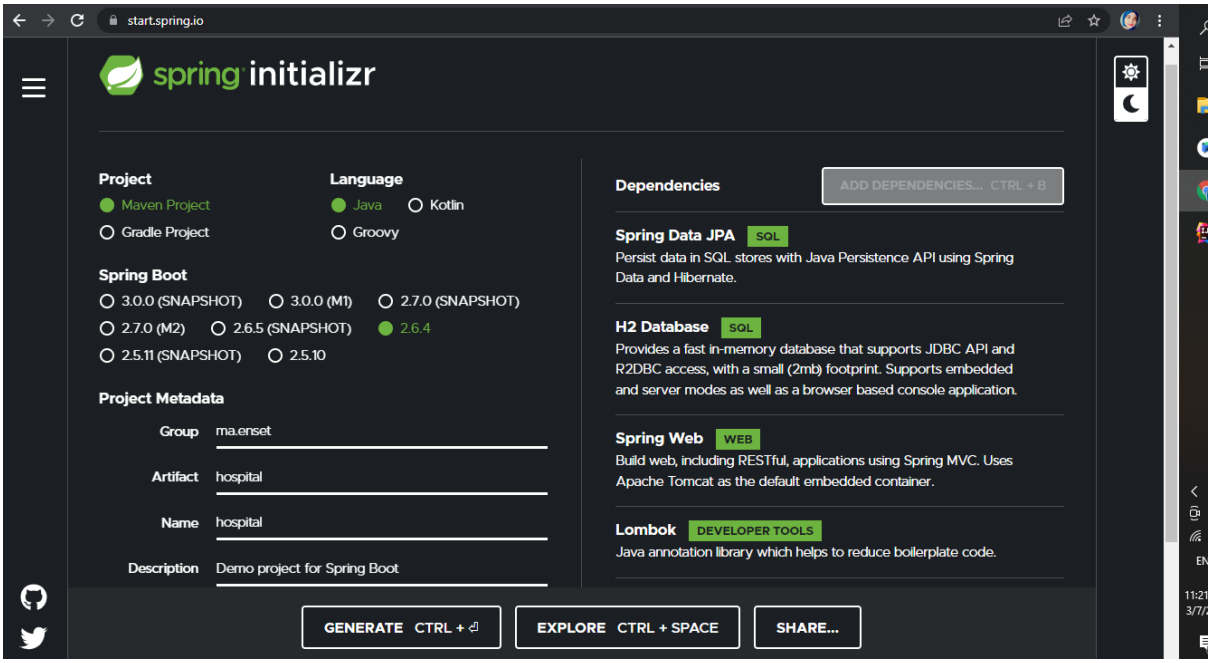
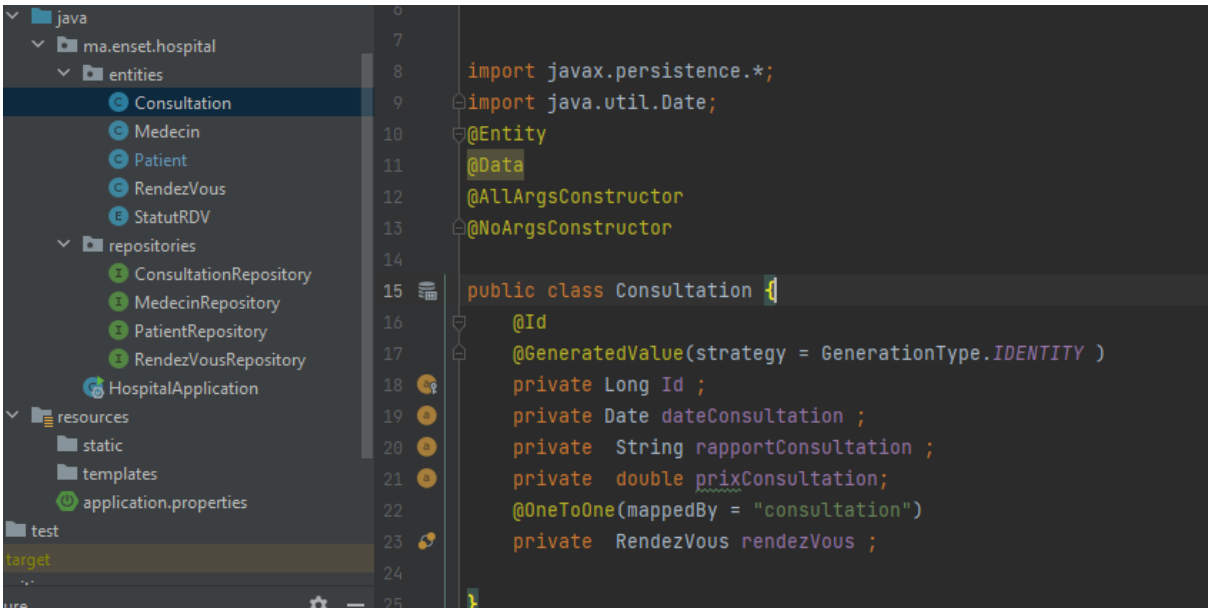
Compte rendu du TP

"ORM Mapping"

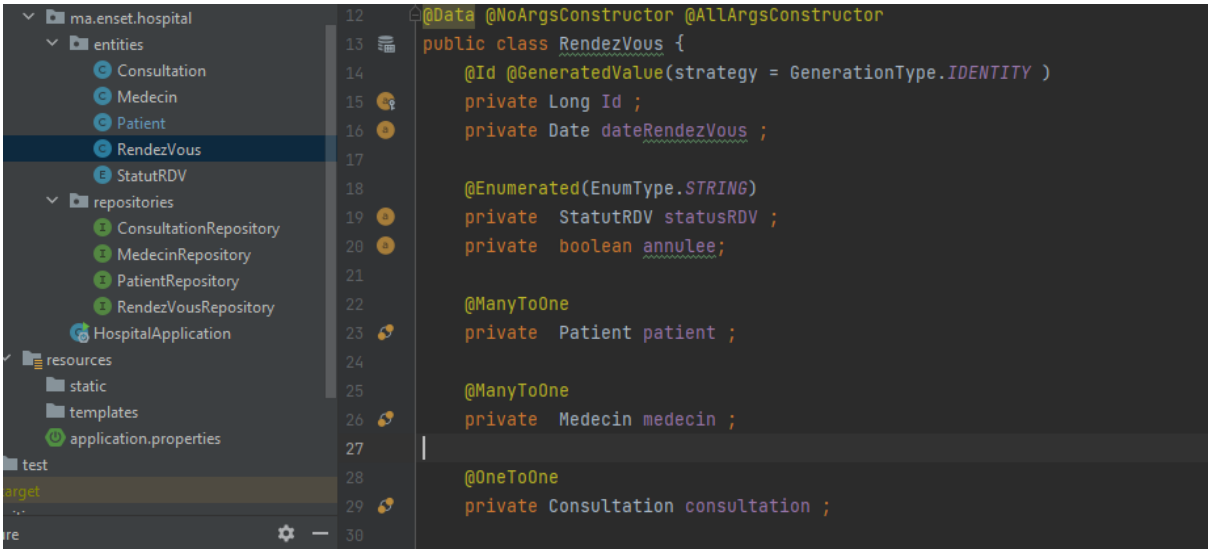
Compte rendu rédigé par Hajar Zarguan,
Étudiante en cycle d'ingénieur: Génie des logiciels et systèmes informatiques distribuées
École Normale Supérieure d'Enseignement Technique .

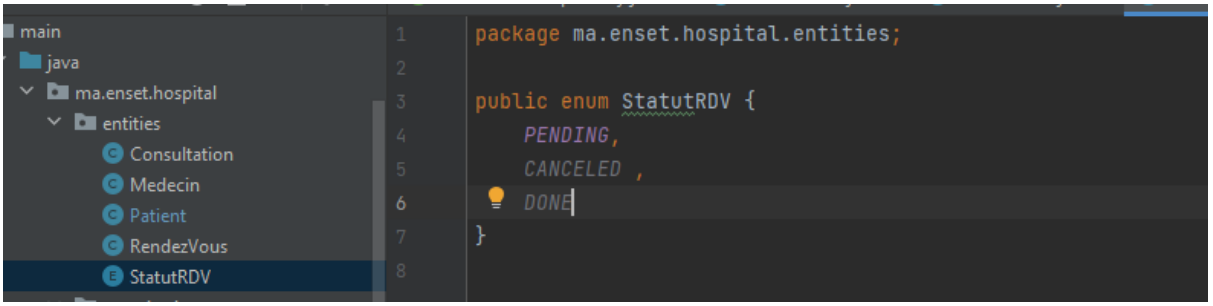

start.spring.io
spring initializr
Project
Maven Project
Gradle Project
Language
Java
Kotlin
Groovy
Spring Boot
3.0.0 (SNAPSHOT)
3.0.0 (M1)
2.7.0 (SNAPSHOT)
2.7.0 (M2)
2.6.5 (SNAPSHOT)
2.6.4
2.5.11 (SNAPSHOT)
2.5.10
Project Metadata
Group ma.enset
Artifact hospital
Name hospital
Description Demo project for Spring Boot
Dependencies
ADD DEPENDENCIES... CTRL + B
Spring Data JPA SQL
Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.
H2 Database SQL
Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.
Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
Lombok DEVELOPER TOOLS
Java annotation library which helps to reduce boilerplate code.
GENERATE CTRL + ⏎
EXPLORE CTRL + SPACE
SHARE...


java
ma.enset.hospital
entities
Consultation
Medecin
Patient
RendezVous
StatutRDV
repositories
ConsultationRepository
MedecinRepository
PatientRepository
RendezVousRepository
HospitalApplication
resources
static
templates
application.properties
test
target
import javax.persistence.*;
import java.util.Date;
@Entity
@Data
@AllArgsConstructor
@NoArgsConstructor
public class Consultation {
@Id
@GeneratedValue(strategy = GenerationType.IDENTITY)
private Long Id ;
private Date dateConsultation ;
private String rapportConsultation ;
private double prixConsultation;
@OneToOne(mappedBy = "consultation")
private RendezVous rendezVous ;
}

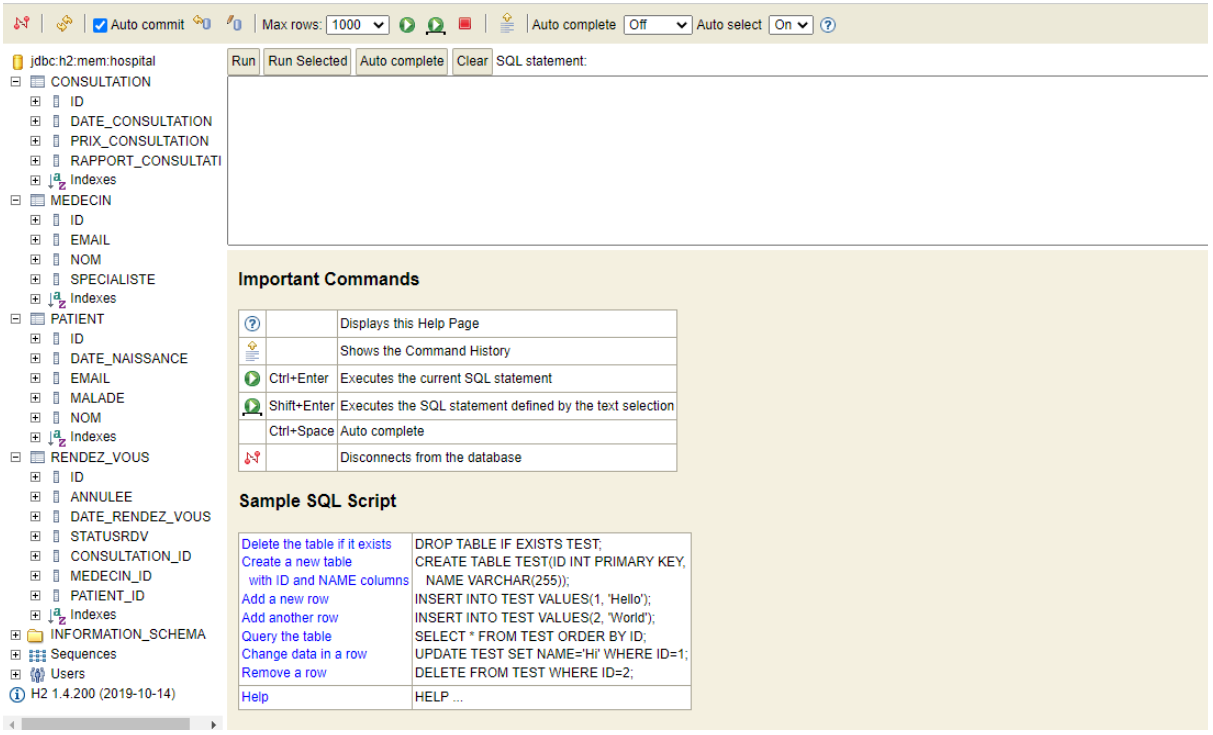

ma.enset.hospital
entities
Consultation
Medecin
Patient
RendezVous
StatutRDV
repositories
ConsultationRepository
MedecinRepository
PatientRepository
RendezVousRepository
HospitalApplication
resources
static
templates
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Medecin {
@Id @GeneratedValue(strategy = GenerationType.IDENTITY)
private Long Id ;
private String nom;
private String email ;
private String specialiste ;
@OneToMany(mappedBy = "medecin")
private Collection <RendezVous> rendezVous;
}


ma.enset.hospital
entities
Consultation
Medecin
Patient
RendezVous
StatutRDV
repositories
ConsultationRepository
MedecinRepository
PatientRepository
RendezVousRepository
HospitalApplication
resources
static
templates
application.properties
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Patient {
@Id @GeneratedValue(strategy = GenerationType.IDENTITY)
private Long id ;
private String nom ;
private Boolean malade;
//@Temporal(TemporalType.DATE)
private Date dateNaissance ;
private String email ;
@OneToMany(mappedBy = "patient",fetch = FetchType.LAZY)
private Collection<RendezVous> rendezVous ;
}


ma.enset.hospital
entities
Consultation
Medecin
Patient
RendezVous
StatutRDV
repositories
ConsultationRepository
MedecinRepository
PatientRepository
RendezVousRepository
HospitalApplication
resources
static
templates
application.properties
test
@Data @NoArgsConstructor @AllArgsConstructor
public class RendezVous {
@Id @GeneratedValue(strategy = GenerationType.IDENTITY)
private Long Id ;
private Date dateRendezVous ;
@Enumerated(EnumType.STRING)
private StatutRDV statusRDV ;
private boolean annulee;
@ManyToOne
private Patient patient ;
@ManyToOne
private Medecin medecin ;
@OneToOne
private Consultation consultation ;


main
java
ma.enset.hospital
entities
Consultation
Medecin
Patient
RendezVous
StatutRDV
package ma.enset.hospital.entities;
public enum StatutRDV {
PENDING,
CANCELED ,
DONE
}


Patient
RendezVous
StatutRDV
repositories
ConsultationRepository
MedecinRepository
PatientRepository
RendezVousRepository
HospitalApplication
resources
static
templates
application.properties
import ...
public interface MedecinRepository extends JpaRepository<Medecin, Long> {
Medecin findByNom(String name);
}


Auto commit
Max rows: 1000
Auto complete Off
Auto select On
jdbc:h2:mem:hospital
CONSULTATION
ID
DATE_CONSULTATION
PRIX_CONSULTATION
RAPPORT_CONSULTATI
Indexes
MEDECIN
ID
EMAIL
NOM
SPECIALISTE
Indexes
PATIENT
ID
DATE_NAISSANCE
EMAIL
MALADE
NOM
Indexes
RENDEZ_VOUS
ID
ANNULEE
DATE_RENDEZ_VOUS
STATUSRDV
CONSULTATION_ID
MEDECIN_ID
PATIENT_ID
Indexes
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
Important Commands
Displays this Help Page
Shows the Command History
Ctrl+Enter
Executes the current SQL statement
Shift+Enter
Executes the SQL statement defined by the text selection
Ctrl+Space
Auto complete
Disconnects from the database
Sample SQL Script
Delete the table if it exists
DROP TABLE IF EXISTS TEST;
Create a new table with ID and NAME columns
CREATE TABLE TEST(ID INT PRIMARY KEY, NAME VARCHAR(255));
Add a new row
INSERT INTO TEST VALUES(1, 'Hello');
Add another row
INSERT INTO TEST VALUES(2, 'World');
Query the table
SELECT * FROM TEST ORDER BY ID;
Change data in a row
UPDATE TEST SET NAME='Hi' WHERE ID=1;
Remove a row
DELETE FROM TEST WHERE ID=2;
Help
HELP ...