Activite3 :

Mapping des associations et de l'héritage en JPA Hibernate Spring Data

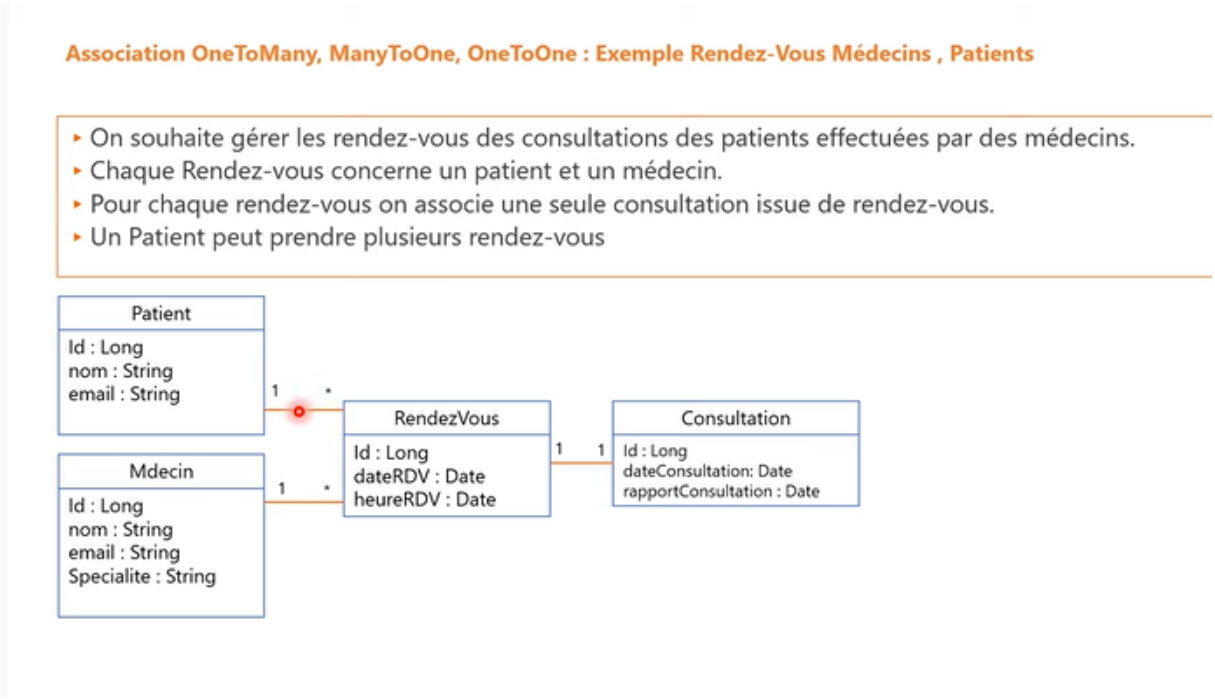
Réaliser par :

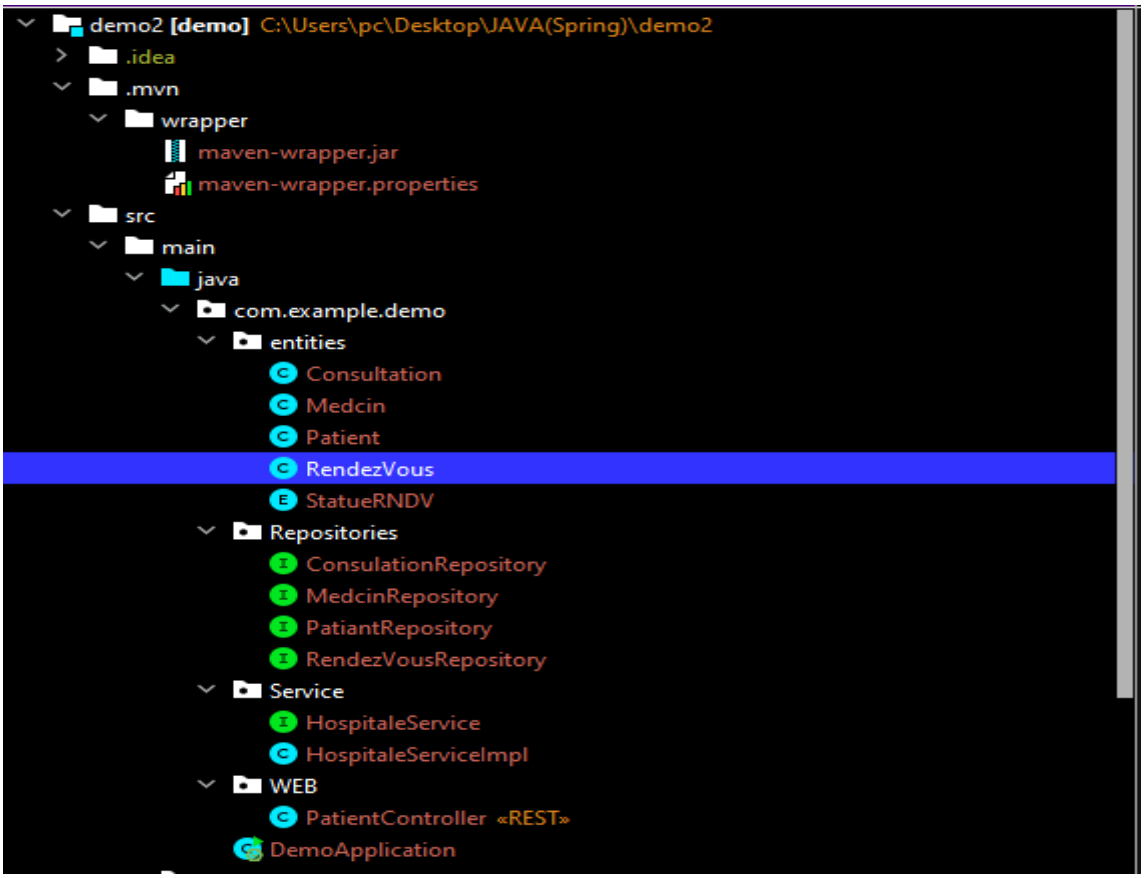
Mohamed amine KHAMMOUR

Professeur :

Mr. Mohamed YOUSSFI

Dans cette activité on souhaite gérer les rendez-vous des consultation des patients





La class patient

```java
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;
import lombok.ToString;

import javax.persistence.*;
import java.util.Collection;
import java.util.Date;
@ToString
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Patient {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    @Column(length = 50)
    private String nom;
    @Temporal(TemporalType.DATE)
    private Date DateNaissnace;
    private boolean malade;
    @OneToMany(mappedBy = "patient")
    private Collection<RendezVous> rendezVous;
    private int score;
```

L’interface patientRepository

```java
package com.example.demo.Repositories;

import com.example.demo.entities.Patient;
import org.springframework.data.jpa.repository.JpaRepository;

public interface PatiantRepository extends JpaRepository<Patient,Long> {
    Patient findByNom(String a);
}
```

La class medcin :

```java
import lombok.NoArgsConstructor;
import lombok.ToString;

import javax.persistence.*;
import java.util.Collection;
@ToString
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Medcin {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String nom;
    private String email;
    private  String specialete;
    @OneToMany(mappedBy = "medcin")
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private Collection<RendezVous> rendezVous;
}
```

L’interface MedcinRepository :

```java
package com.example.demo.Repositories;

import ...

    public interface MedcinRepository extends JpaRepository<Medcin,Long> {
        Medcin findByNom(String nom);
}
```

La class RendzVous

```java
import javax.persistence.*;
import java.util.Date;
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class RendezVous {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private Date date;
    @Enumerated(EnumType.STRING)
    private  StatueRNDV annule;
    @ManyToOne
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private Patient patient;
    @ManyToOne

    private  Medcin medcin;
    @OneToOne(mappedBy = "rendezVous")

    private Consultation consultation;
}
```

L’interface RendezVousRipository

```java
package com.example.demo.Repositories;

import com.example.demo.entities.Patient;
import com.example.demo.entities.RendezVous;
import org.springframework.data.jpa.repository.JpaRepository;

import javax.xml.crypto.Data;
import java.sql.Date;

public interface RendezVousRepository extends JpaRepository<RendezVous,Long> {

}
```

La class Consultation :

```java
import lombok.NoArgsConstructor;
import lombok.ToString;

import javax.persistence.*;
import java.util.Date;
@ToString
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Consultation {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private Date dateconsultation;
    private String rapport;
@OneToOne
@JsonProperty(access = JsonProperty.Access.WRITE_ONLY)

    private RendezVous rendezVous;
}
```

L'interface ConsultationRepository :

```java
package com.example.demo.Repositories;

import ...

public interface ConsulationRepository extends JpaRepository<Consultation,Long> {
}
```

La class HospitaleServieImpl :

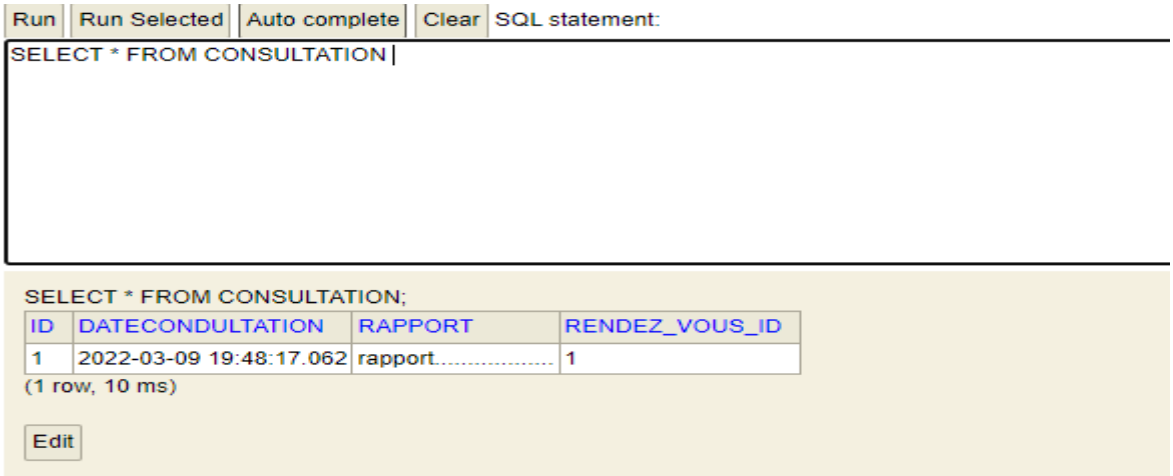
```
package com.example.demo.Service;

import com.example.demo.Repositories.ConsulationRepository;
import com.example.demo.Repositories.MedcinRepository;
import com.example.demo.Repositories.PatiantRepository;
import com.example.demo.Repositories.RendezVousRepository;
import com.example.demo.entities.Consultation;
import com.example.demo.entities.Medcin;
import com.example.demo.entities.Patient;
import com.example.demo.entities.RendezVous;
import org.springframework.stereotype.Service;

import javax.transaction.Transactional;
import java.util.Optional;
import java.util.UUID;

@Service
@Transactional
public class HospitaleServiceImpl implements HospitaleService {
    private PatiantRepository patiantRepository;
    private MedcinRepository medcinRepository;
    private ConsulationRepository consulationRepositor;
    private RendezVousRepository rendezVousRepository;

    public HospitaleServiceImpl(PatiantRepository patiantRepository,
MedcinRepository medcinRepository, ConsulationRepository
consulationRepositor, RendezVousRepository rendezVousRepository) {
        this.patiantRepository = patiantRepository;
        this.medcinRepository = medcinRepository;
        this.consulationRepositor = consulationRepositor;
        this.rendezVousRepository = rendezVousRepository;
    }

    @Override

    public Patient savePtiant(Patient patient) {
        return patiantRepository.save(patient);
    }

    @Override
    public Medcin saveMedcin(Medcin medcin) {
        return medcinRepository.save(medcin);
    }

    @Override
    public RendezVous saveRendezVous(RendezVous rendezVous) {
       // rendezVous.setId(UUID.randomUUID().toString());
        return rendezVousRepository.save(rendezVous);
    }


    @Override
    public Consultation saveConsultation(Consultation consultation) {
        return consulationRepositor.save(consultation);
    }

    @Override
    public Patient recherchparNom(String n) {
        return patiantRepository.findByNom(n);
```

```
    }

    @Override
    public Medcin rechercerParnom(String m) {
        return medcinRepository.findByNom(m);
    }

    @Override
    public RendezVous RecherchRDVbyID(Long l) {
        return rendezVousRepository.findById(l).orElse(null);
    }


}
```

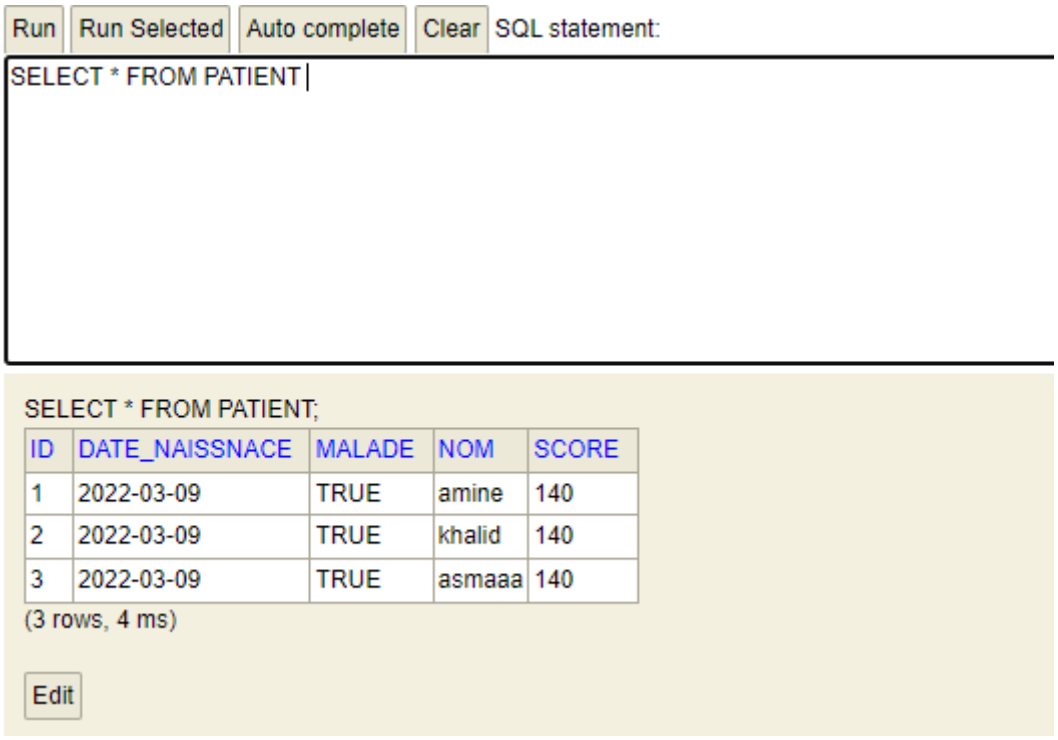
l'interface HospitalService :

```
package com.example.demo.Service;

import com.example.demo.entities.Consultation;
import com.example.demo.entities.Medcin;
import com.example.demo.entities.Patient;
import com.example.demo.entities.RendezVous;

import java.util.Optional;

public interface HospitaleService {
    Patient savePtiant(Patient patient);
    Medcin saveMedcin(Medcin medcin);
    RendezVous saveRendezVous(RendezVous rendezVous);
    Consultation saveConsultation(Consultation consultation);
    Patient recherchparNom(String n);
    Medcin rechercerParnom(String m);
    RendezVous RecherchRDVbyID(Long l);




}
```

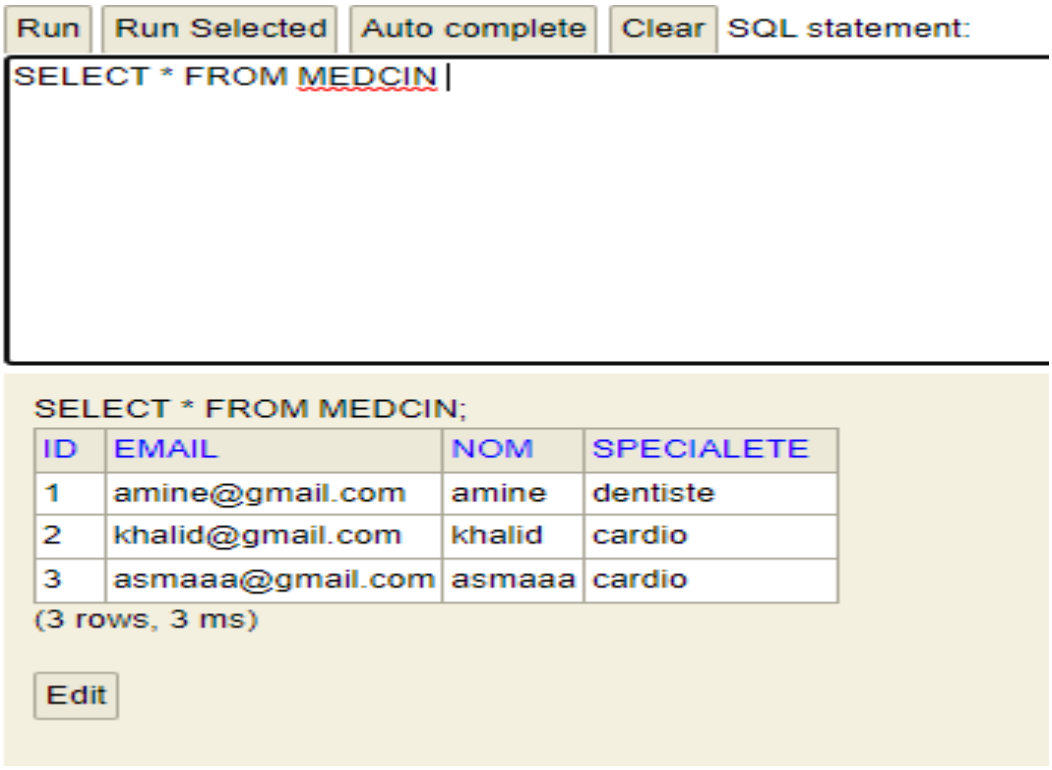
La table Consultation :



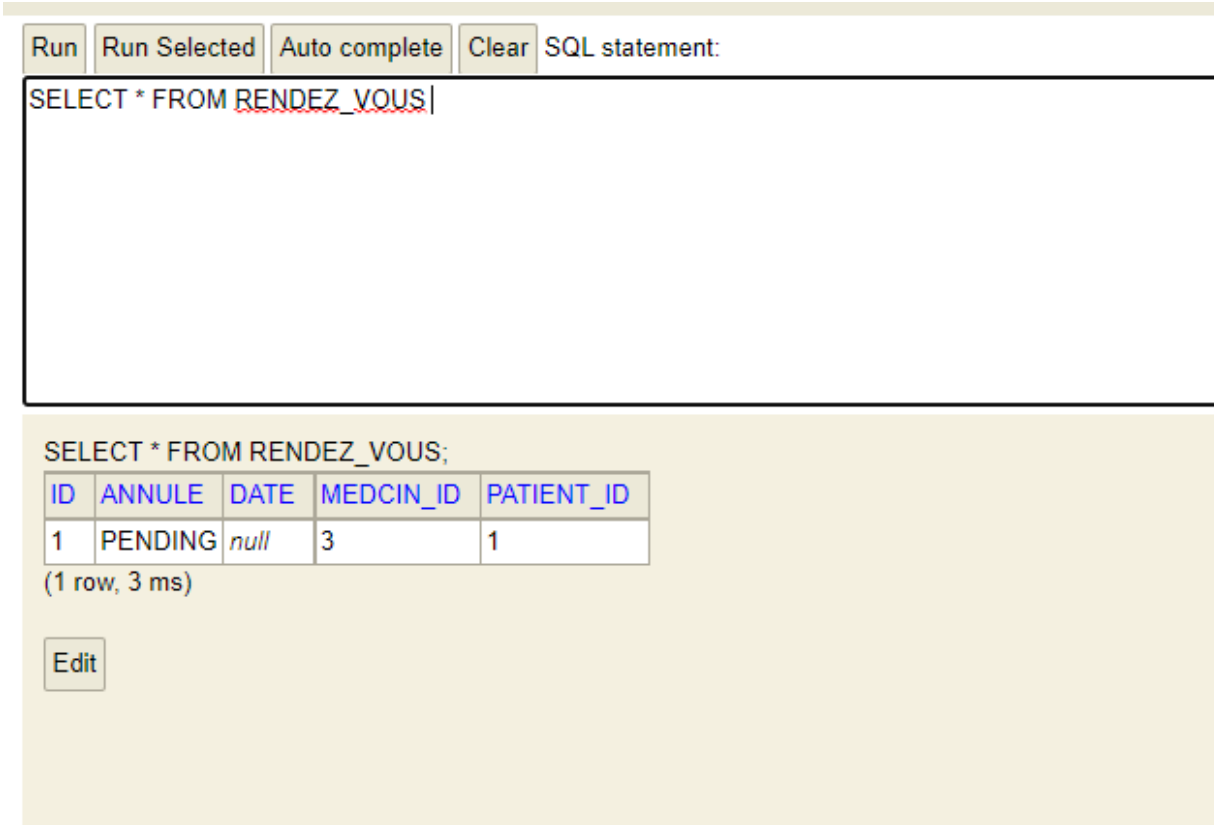
La table patiente :



La table Medcin :



La table Rendez vous :



La partie web :


```
// 20220309194823
// http://localhost:8086/patients

[
  {
    "id": 1,
    "nom": "amine",
    "malade": true,
    "rendezVous": [
      {
        "id": 1,
        "date": null,
        "annule": "PENDING",
        "medcin": {
          "id": 3,
          "nom": "asmaaa",
          "email": "asmaaa@gmail.com",
          "specialete": "cardio"
        },
        "consultation": {
          "id": 1,
          "dateCondultation": "2022-03-09T18:48:17.062+00:00",
          "rapport": "rapport..................."
        }
      }
    ],
    "score": 140,
    "dateNaissnace": "2022-03-09"
  },
```


La class PatientController :

```
package com.example.demo.WEB;

import com.example.demo.Repositories.PatiantRepository;
import com.example.demo.entities.Patient;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;
```

```java
import java.util.List;

@RestController
public class PatientController {
    @Autowired
    private PatiantRepository patiantRepository;
    @GetMapping("/patients")
    public List<Patient> patientliste(){
        return patiantRepository.findAll();
    }
}
```