

: Hospital Management System using Spring Boot and Spring Data JPA Case Study

Project Structure

src/main/java

```
└── com.example.hospital
    ├── entity
    │   ├── Patient.java
    │   ├── Doctor.java
    │   ├── Appointment.java
    │   └── MedicalRecord.java
    ├── repository
    │   ├── PatientRepository.java
    │   ├── DoctorRepository.java
    │   ├── AppointmentRepository.java
    │   └── MedicalRecordRepository.java
    ├── controller
    │   └── HospitalController.java
    └── HospitalManagementApplication.java
```

Entity Classes

Patient.java

@Entity

@Data

```
@NoArgsConstructor
@AllArgsConstructor
public class Patient {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String name;
    private int age;
    private String gender;
    private String address;

    @OneToMany(mappedBy = "patient", cascade = CascadeType.ALL)
    private List<Appointment> appointments;

    @OneToMany(mappedBy = "patient", cascade = CascadeType.ALL)
    private List<MedicalRecord> medicalRecords;
}
```

Doctor.java

```
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Doctor {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
private Long id;
```

```
private String name;
```

```
private String specialization;
```

```
private String email;
```

```
private String phone;
```

```
@OneToMany(mappedBy = "doctor", cascade = CascadeType.ALL)
```

```
private List<Appointment> appointments;
```

```
}
```

Appointment.java

```
@Entity
```

```
@Data
```

```
@NoArgsConstructor
```

```
@AllArgsConstructor
```

```
public class Appointment {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private LocalDate date;
```

```
    private LocalTime time;
```

```
    private String notes;
```

```
@ManyToOne
```

```
@JoinColumn(name = "patient_id")
```

```
    private Patient patient;
```

```
@ManyToOne
@JoinColumn(name = "doctor_id")
private Doctor doctor;
}
```

MedicalRecord.java

```
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class MedicalRecord {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String diagnosis;
    private String treatment;
    private LocalDate date;

    @ManyToOne
    @JoinColumn(name = "patient_id")
    private Patient patient;
}
```

Repository Interfaces

```
public interface PatientRepository extends JpaRepository<Patient, Long> {}

public interface DoctorRepository extends JpaRepository<Doctor, Long> {}
```

```
public interface AppointmentRepository extends JpaRepository<Appointment, Long> {}  
public interface MedicalRecordRepository extends JpaRepository<MedicalRecord,  
Long> {}
```

application.properties

```
spring.datasource.url=jdbc:mysql://localhost:3306/hospitaldb  
spring.datasource.username=root  
spring.datasource.password=yourpassword  
  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=true  
spring.jpa.properties.hibernate.format_sql=true
```

Sample Controller (Basic)

```
@RestController  
@RequestMapping("/api")  
@RequiredArgsConstructor  
public class HospitalController {  
  
    private final PatientRepository patientRepo;  
    private final DoctorRepository doctorRepo;  
    private final AppointmentRepository appointmentRepo;  
    private final MedicalRecordRepository recordRepo;  
  
    @PostMapping("/patients")  
    public Patient createPatient(@RequestBody Patient patient) {  
        return patientRepo.save(patient);  
    }  
}
```

```
@PostMapping("/doctors")
public Doctor createDoctor(@RequestBody Doctor doctor) {
    return doctorRepo.save(doctor);
}
```

```
@PostMapping("/appointments")
public Appointment createAppointment(@RequestBody Appointment appointment) {
    return appointmentRepo.save(appointment);
}
```

```
@PostMapping("/records")
public MedicalRecord createRecord(@RequestBody MedicalRecord record) {
    return recordRepo.save(record);
}
```

```
@GetMapping("/patients")
public List<Patient> getAllPatients() {
    return patientRepo.findAll();
}
}
```

Sample Postman Requests

Create Patient

POST /api/patients

```
{
  "name": "John Doe",
```

```
"age": 35,  
"gender": "Male",  
"address": "123 Main Street"  
}
```

Create Doctor

POST /api/doctors

```
{  
  "name": "Dr. Smith",  
  "specialization": "Cardiologist",  
  "email": "drsmith@example.com",  
  "phone": "9876543210"  
}
```