Hospital Appointment System

🎯 **Scenario:**

Patients can book appointments with doctors, and admins manage doctor schedules.

🧩 **Entities:**

- **Patient**
- **Doctor**
- **Appointment**
- **Prescription**

🛣️ **Routes:**

- **/patients → register, login, view profile**
- **/doctors → add, update, get doctor info**
- **/appointments → book, cancel, list all**
- **/prescriptions → doctor issues prescription**

🧠 **Controllers:**

- **patientController.js**
- **doctorController.js**
- **appointmentController.js**
- **prescriptionController.js**

Hospital Management System — Controller Functionalities

👤 **1. patientController.js**

Handles all operations related to **patients** — registration, profile management, and appointment viewing.

Functions:

1. **getAllPatients**

- Return a list of all patients.
- Used by admins or doctors to view registered patients.

2. **addPatient**
    - Register a new patient (name, age, gender, contact info, medical history, etc.).
    - Generate a unique patient ID.
3. **getPatientById**
    - Fetch detailed info of a specific patient using ID.
    - Include basic details + appointment/prescription history (optional).
4. **updatePatientInfo**
    - Edit patient details such as address, contact, or medical history.
5. **deletePatient**
    - Remove patient record from the system (admin use only).
6. **getPatientAppointments**
    - Show all appointments booked by a specific patient (past and upcoming).

🩺 2. doctorController.js

Manages doctors, their schedules, and assigned appointments.

Functions:

1. **getAllDoctors**
    - Retrieve list of all doctors with their specialties and availability.
2. **addDoctor**
    - Register a new doctor (name, department, specialization, timings, etc.).
3. **getDoctorById**
    - View profile of a specific doctor along with assigned patients or appointments.

4. **updateDoctorInfo**
    - Update doctor details (availability, specialization, etc.).
5. **deleteDoctor**
    - Remove a doctor's record (used when doctor leaves hospital).
6. **getDoctorAppointments**
    - List all appointments scheduled for a doctor.
    - Can include filters like today's appointments, completed, or upcoming.
7. **setAvailability**
    - Mark doctor as available/unavailable for specific dates or times.

🗓️ 3. appointmentController.js

Handles the booking, cancellation, and tracking of appointments between patients and doctors.

Functions:

1. **getAllAppointments**
    - Retrieve all appointments (for admin or analytics use).
2. **bookAppointment**
    - Create a new appointment between a patient and doctor.
    - Check doctor availability and assign a time slot.
3. **getAppointmentById**
    - Fetch details of a single appointment (doctor, patient, time, status).
4. **getAppointmentsByPatient**
    - Retrieve all appointments booked by a specific patient.
5. **getAppointmentsByDoctor**
    - Retrieve all appointments assigned to a specific doctor.
6. **updateAppointmentStatus**

- Change appointment status (e.g., “Scheduled”, “Completed”, “Cancelled”, “No-show”).

7. **cancelAppointment**
    - Allow patient or admin to cancel appointment before it starts.
8. **rescheduleAppointment**
    - Change appointment date/time (with availability check).

💊 4. prescriptionController.js

Manages doctor-issued prescriptions after appointments.

Functions:

1. **getAllPrescriptions**
    - Retrieve all prescriptions (for admin overview or pharmacy use).
2. **addPrescription**
    - Doctor adds a new prescription for a patient (after appointment completion).
    - Include medicine name, dosage, duration, notes, etc.
3. **getPrescriptionById**
    - Fetch a specific prescription by ID.
4. **getPrescriptionsByPatient**
    - Show all prescriptions of a specific patient (with doctor info).
5. **getPrescriptionsByDoctor**
    - List all prescriptions issued by a specific doctor.
6. **updatePrescription**
    - Edit prescription details (allowed only to the doctor who issued it).
7. **deletePrescription**
    - Remove a prescription (admin-only or doctor correction).

💡 **Example Workflow (How Controllers Connect)**

1. **Patient registers** → patientController.addPatient
2. **Doctor registers** → doctorController.addDoctor
3. **Patient books appointment** → appointmentController.bookAppointment
4. **Doctor checks their appointments** → doctorController.getDoctorAppointments
5. **Doctor issues prescription** → prescriptionController.addPrescription
6. **Patient views prescription history** → prescriptionController.getPrescriptionsByPatient