HOSPITAL MANAGEMENT SYSTEM

Alexiane Laroye - 5999222011 Wissame Ismael - 5999222008 Flavien Geoffray - 5999222008 Enzo Jehl - 5999222010

01

Describe Project

02

CDM / PDM

03

Queries

04

Triggers

05

Functions & Procedures

06

Indexing

Demonstration at the end (replication)

DESCRIBE PROJECT

01

Hospital Management

The Hospital Management will manage this data :

- **Patient**
- **Doctor**
- **Nurse**
- **Appointment**
- **Medical Record**
- **Medication**
- **Department**

Tables & Relations

The Hospital has several patients, doctors and nurses. The Hospital has also a medication database with the name of the medication and the dosage.

Each patient is assigned with one nurse but a nurse can be assigned to several patients

- **Patients can receive medications from one nurse, and nurses can administer medications to multiple patients.**

Each patient has his own medical record

The hospital has several department. Each doctor is assigned to a department and can cure patients from the department.

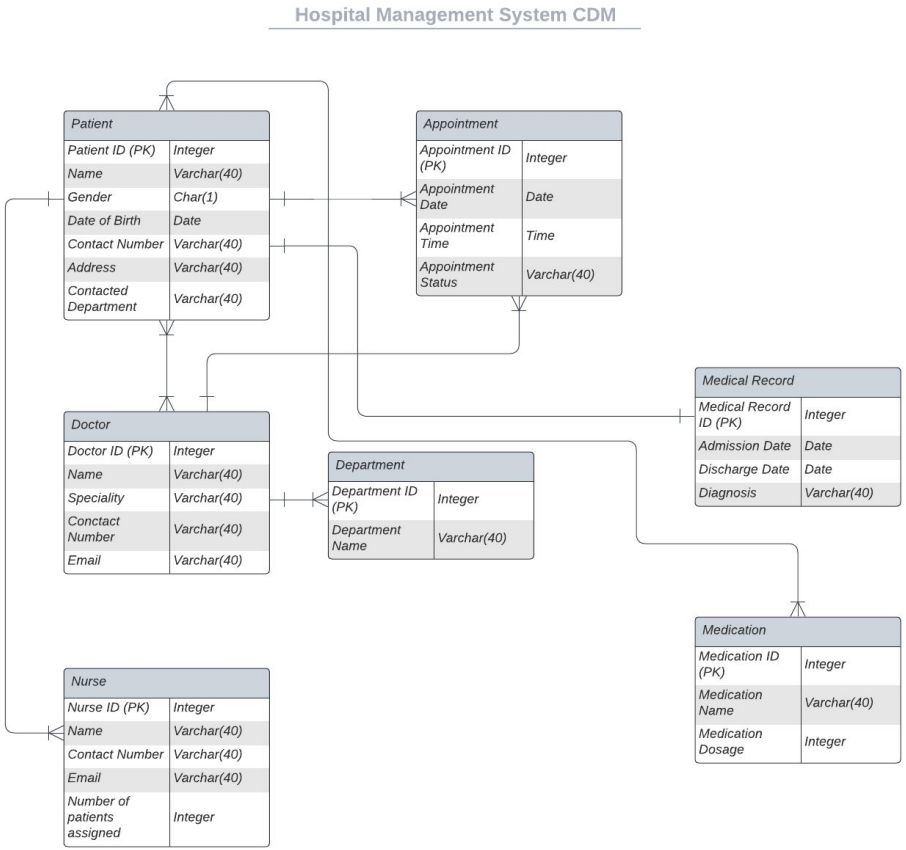
Once the patient is in the database of the hospital it will book one or several appointment with a doctor available from the department contacted.

- **Patients can have appointments with multiple doctors, and doctors can have multiple patients.**
- **Also, Each appointment can be associated with only one patient, but a patient can have multiple appointments.**

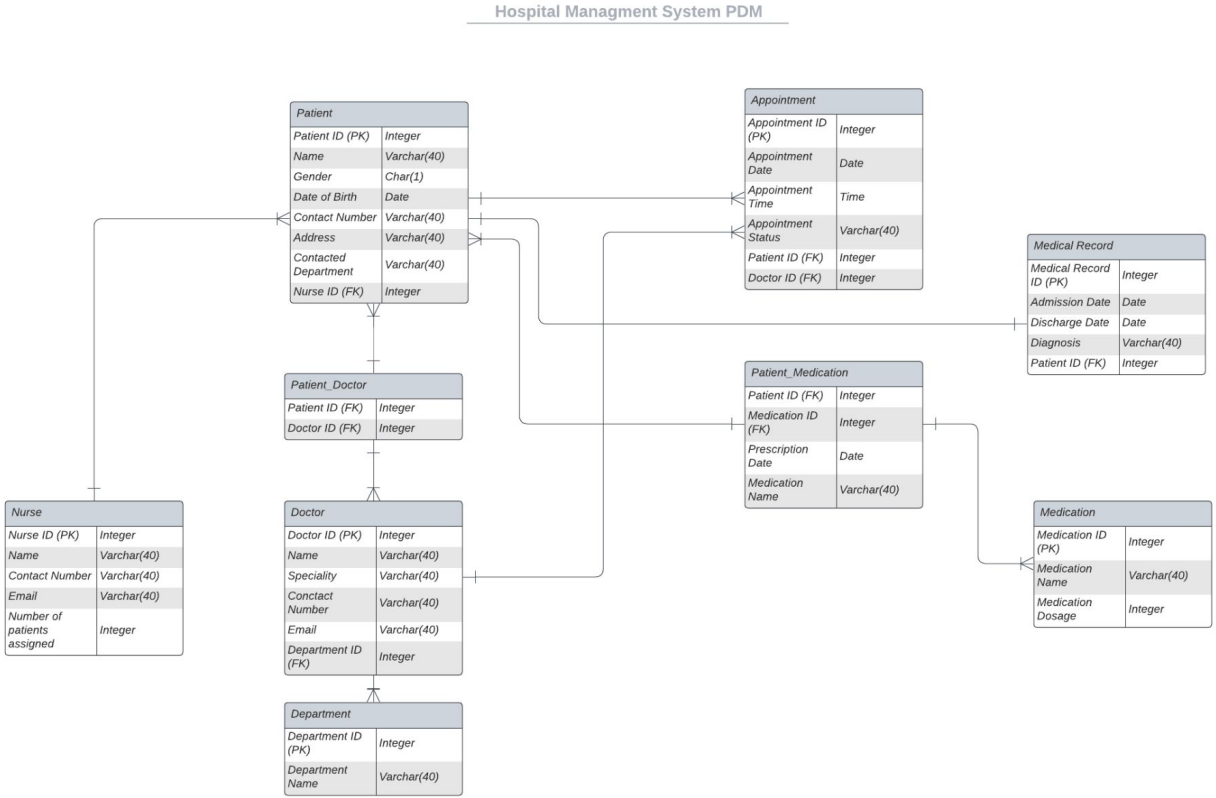
CDM / PDM

02

CDM


Hospital Management System CDM
Patient
Patient ID (PK) Integer
Name Varchar(40)
Gender Char(1)
Date of Birth Date
Contact Number Varchar(40)
Address Varchar(40)
Contacted Department Varchar(40)
Appointment
Appointment ID (PK) Integer
Appointment Date Date
Appointment Time Time
Appointment Status Varchar(40)
Medical Record
Medical Record ID (PK) Integer
Admission Date Date
Discharge Date Date
Diagnosis Varchar(40)
Doctor
Doctor ID (PK) Integer
Name Varchar(40)
Speciality Varchar(40)
Conctact Number Varchar(40)
Email Varchar(40)
Department
Department ID (PK) Integer
Department Name Varchar(40)
Medication
Medication ID (PK) Integer
Medication Name Varchar(40)
Medication Dosage Integer
Nurse
Nurse ID (PK) Integer
Name Varchar(40)
Contact Number Varchar(40)
Email Varchar(40)
Number of patients assigned Integer

PDM


Hospital Managment System PDM
Patient
Patient ID (PK) Integer
Name Varchar(40)
Gender Char(1)
Date of Birth Date
Contact Number Varchar(40)
Address Varchar(40)
Contacted Department Varchar(40)
Nurse ID (FK) Integer
Appointment
Appointment ID (PK) Integer
Appointment Date Date
Appointment Time Time
Appointment Status Varchar(40)
Patient ID (FK) Integer
Doctor ID (FK) Integer
Medical Record
Medical Record ID (PK) Integer
Admission Date Date
Discharge Date Date
Diagnosis Varchar(40)
Patient ID (FK) Integer
Patient_Doctor
Patient ID (FK) Integer
Doctor ID (FK) Integer
Patient_Medication
Patient ID (FK) Integer
Medication ID (FK) Integer
Prescription Date Date
Medication Name Varchar(40)
Nurse
Nurse ID (PK) Integer
Name Varchar(40)
Contact Number Varchar(40)
Email Varchar(40)
Number of patients assigned Integer
Doctor
Doctor ID (PK) Integer
Name Varchar(40)
Speciality Varchar(40)
Conctact Number Varchar(40)
Email Varchar(40)
Department ID (FK) Integer
Medication
Medication ID (PK) Integer
Medication Name Varchar(40)
Medication Dosage Integer
Department
Department ID (PK) Integer
Department Name Varchar(40)

Queries

03

Alexiane	Enzo	Flavien	Wissame
Select patient name, doctor name and appointment date for all appointments	Select patient name, admission date and diagnosis for all medical records	Select all appointments with patient and doctor details	Select all appointments for a specific patient
Select patient name, admission date and diagnosis for patients with a specific diagnosis	Select all appointments per department with the name of the doctor and the patient	Select patient name, appointment date and patient address for all appointments that have been scheduled	Select department name and doctor name for all doctors and their respective departments
Select all medical records with patient name for medical records that have an associated patient	Select the number of patients per department, sorted in descending order of the number of patients	View patients with their appointments and corresponding doctor information	Select patient name, doctor name and appointment date for all appointments associated with a specific patient
Select the total number of scheduled appointments for each patient, sorted in descending order of the number of appointments	Select the total number of scheduled appointments for each doctor, sorted in ascending order of the number of appointments, while displaying their patient's name each time	Select the appointment date, the number of appointments planned for each date, sorted in ascending order of the date	Select the number of patients assigned to each nurse, sorted in descending order of the number of patients
Select patients with at most 3 appointments and display their name, the total number of appointments and the name of the doctor who treated them	Select patients and doctors who have appointments scheduled for a given date, sorted alphabetically by patient name	Select patients who have scheduled appointments with physicians specializing in cardiology, sorted in ascending order by appointment date	Select patients with at most 1 appointment and display their name, the name of the doctor who treated them, and in which department

Alexiane

Select patient name, admission date and diagnosis for patients with a specific diagnosis

Query History | Query

```
1  -- select the name of the patient, the admission date and the diagnosis for a specific diagnosis
2  SELECT patient.name, medical_record.admission_date, medical_record.diagnosis
3  FROM patient
4  JOIN medical_record ON patient.patient_ID = medical_record.patient_ID
5  WHERE medical_record.diagnosis = 'Hypertension';
```

Data Output | Messages | Notifications

	name character varying (40)	admission_date date	diagnosis character varying (40)
1	Flavien Geoffray	2023-06-11	Hypertension

Select patients with at most 3 appointments and display their name, the total number of appointments and the name of the doctor who treated them

```
1  -- select patients with maximum 3 appointments et display their name and the number of appointments and the doctor name
2  SELECT p.name AS patient_name, COUNT(*) AS appointment_count, d.name AS doctor_name
3  FROM patient p
4  INNER JOIN appointment a ON p.patient_ID = a.patient_ID
5  INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
6  GROUP BY p.patient_ID, p.name, d.name
7  HAVING COUNT(*) < 3;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	appointment_count bigint	doctor_name character varying (40)
1	Alexiane Laroye	1	Dr. Michael Davis
2	Wissame Ismael	1	Dr. Doctor
3	Flavien Geoffray	1	Dr. Sarah Adams
4	Enzo Jehl	1	Dr. Jennifer Lee

Enzo

Select the number of patients per department, sorted in descending order of the number of patients

Query History | Query

```
1  -- select the number of patients per department, by decreasing order :
2  SELECT department_name, COUNT(*) AS patient_count
3  FROM patient
4  INNER JOIN department ON patient.contacted_department = department.department_name
5  GROUP BY department_name
6  ORDER BY patient_count DESC;
```

Data Output | Messages | Notifications

	department_name character varying (40)	patient_count bigint
1	Cardiology	1
2	Pediatrics	1
3	Orthopedics	1
4	Neurology	1

Select patients and doctors who have appointments scheduled for a given date, sorted alphabetically by patient name

```
1  -- select patients and doctors who have a scheduled appointment for a specific date, by alphabetic order of the patient name
2  SELECT p.name AS patient_name, d.name AS doctor_name
3  FROM patient p
4  INNER JOIN appointment a ON p.patient_ID = a.patient_ID
5  INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
6  WHERE a.appointment_date = '2023-06-10'
7  ORDER BY p.name;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	doctor_name character varying (40)
1	Flavien Geoffray	Dr. Sarah Adams

Flavien

Select patient name, appointment date and patient address for all appointments that have been scheduled

Query History | Query

```
-- select the patient name, the appointment date and the address for all scheduled appointments
SELECT patient.name, appointment.appointment_date, patient.address
FROM appointment
JOIN patient ON appointment.patient_ID = patient.patient_ID
WHERE appointment.appointment_status = 'Scheduled';
```

Data Output | Messages | Notifications

	name character varying (40)	appointment_date date	address character varying (40)
1	Flavien Geoffray	2023-06-10	789 Maple Street
2	Alexiane Laroye	2023-06-11	567 Oakwood Avenue
3	Enzo Jehl	2023-06-12	456 Elmwood Drive
4	Wissame Ismael	2023-06-13	233 Togo Street

Select patients who have scheduled appointments with physicians specializing in cardiology, sorted in ascending order by appointment date

```
-- select patients with scheduled appointments with cardiology specialists, sorted in ascending order of appointment date:
SELECT p.name AS patient_name, a.appointment_date, d.name AS doctor_name
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
WHERE d.speciality = 'Cardiology'
ORDER BY a.appointment_date;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	appointment_date date	doctor_name character varying (40)
1	Flavien Geoffray	2023-06-10	Dr. Sarah Adams

Wissame

Select all appointments for a specific patient

Query History | Query

```sql
-- Select the appointments of a specific patient :
SELECT appointment.appointment_ID, appointment.appointment_date, appointment.patient_ID,
       doctor.doctor_ID, doctor.name, doctor.speciality, doctor.email
FROM appointment
JOIN doctor ON appointment.doctor_ID = doctor.doctor_ID
WHERE appointment.patient_ID = 1;
```

Data Output | Messages | Notifications

	appointment_id integer	appointment_date date	patient_id integer	doctor_id integer	name character varying (40)	speciality character varying (40)	email character varying (40)
1	1	2023-06-10	1	1	Dr. Sarah Adams	Cardiology	sarah.adams@example.com

Select the number of patients assigned to each nurse, sorted in descending order of the number of patients

```sql
-- select the number of patients assigned foreach nurse, by decreasing order
SELECT n.name, COUNT(*) AS patient_count
FROM nurse n
INNER JOIN patient p ON n.nurse_ID = p.nurse_ID
GROUP BY n.name
ORDER BY patient_count DESC;

```

Data Output | Messages | Notifications

	name character varying (40)	patient_count bigint
1	Nurse Emily Thompson	2
2	Nurse Mark Wilson	1
3	Nurse Theo Oliver	1
4	Nurse Lisa Johnson	1

Triggers

04

Alexiane	Enzo	Flavien	Wissame
Trigger for the ID of patient	Trigger for the ID of doctor / Trigger for the ID of department	Trigger for the ID of nurse	Trigger for the ID of medication
Trigger to supr the appointment when a doctor is supr and also the table doctor_patient	Trigger to add the id of a medication to the patient medication when a doctor add a medication name	Trigger to assign a doctor to his departement	Trigger to create the medical record of a patient
Trigger to create the patient_medication	Trigger to assign a patient to the nurse with the less number a patient	Trigger to supr the medical_record, appointment, patient_medication and patient_doctor	Reassigne the nurse when deleted

```sql
--trigger for the ID of department

CREATE OR REPLACE FUNCTION assign_id_functionDep()
RETURNS TRIGGER AS $$
DECLARE
    last_id INT;
BEGIN
    SELECT MAX(department_id) INTO last_id FROM department;
    IF last_id IS NULL THEN
        NEW.department_id := 1;
    ELSE
        NEW.department_id := last_id + 1;
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE TRIGGER assign_id_trigger
BEFORE INSERT ON department
FOR EACH ROW
EXECUTE FUNCTION assign_id_functionDep();
```

```sql
--trigger to add the id of a medication to the patient medication when a doctor add a medication name

CREATE OR REPLACE FUNCTION assign_patient_medication_id()
RETURNS TRIGGER AS $$
DECLARE
    new_med_id INTEGER;
BEGIN
    SELECT medication_id INTO new_med_id FROM medication WHERE medication_name = NEW.patient_medication_name;
    UPDATE patient_medication
        SET medication_id = new_med_id where patient_medication_name = NEW.patient_medication_name;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE TRIGGER assign_patient_medication_id_trigger
AFTER UPDATE ON patient_medication
FOR EACH ROW
When (OLD.patient_medication_name IS DISTINCT FROM NEW.patient_medication_name)
EXECUTE FUNCTION assign_patient_medication_id();
```

Functions & Procedures

05

Alexiane	Enzo	Flavien	Wissame
Display all patient ids from today appointments	Change the appointment status with status	Display doctors info of spec	Display the medical record of a given patient id
Show the name of medication that takes the given patient id	Update the patient_medication table	Display all patients assigned to the given nurse	Show the number of patient of the given nurse
Update the given nurse infos	Show the name of patients who have an appointment with the given doctor	Show all appointments by the give status	Procedure to update the given doctor infos

```sql
-- display the medical record of a given patient id
CREATE OR REPLACE function showmedicalrecord (id int)
RETURNS SETOF medical_record
AS $$
BEGIN
   return query SELECT * FROM medical_record WHERE patient_id = id;
END;
$$ LANGUAGE plpgsql;
```

```sql
-- procedure to update the given doctor infos
CREATE OR REPLACE PROCEDURE changeinfosdoctor (id int, nemail varchar(40), ncontact_number varchar(40))
AS $$
BEGIN
    IF nemail IS NOT NULL THEN
        UPDATE doctor
        SET email = nemail
        WHERE doctor_id = id;
    END IF;

    IF ncontact_number IS NOT NULL THEN
        UPDATE doctor
        SET contact_number = ncontact_number
        WHERE doctor_id = id;
    END IF;
END;
$$ LANGUAGE plpgsql;
```

Indexing

06

Indexing

```
-- Creation of an index on the medication_name column of the medication table
CREATE INDEX idx_medication_medication_name ON medication(medication_name);

-- Creation of an index on the patient_ID and prescription_date columns of the patient_medication table
CREATE INDEX idx_patient_medication_patient_prescription_date ON patient_medication(patient_ID, prescription_date);
```

Indexing :

- To enhance database performance
- Indexes accelerate search, join, and filtering operations by creating additional data structures that facilitate quick access to matching records
- creating indexes can also impact the performance of insertion, update, and deletion operations, as indexes need to be maintained during these operations.

DEMO

replication

Screenshot

Query History | Query

```
-- Select the appointments of a specific patient :
SELECT appointment.appointment_ID, appointment.appointment_date, appointment.patient_ID,
       doctor.doctor_ID, doctor.name, doctor.speciality, doctor.email
FROM appointment
JOIN doctor ON appointment.doctor_ID = doctor.doctor_ID
WHERE appointment.patient_ID = 1;
```

Data Output | Messages | Notifications

	appointment_id integer	appointment_date date	patient_id integer	doctor_id integer	name character varying (40)	speciality character varying (40)	email character varying (40)
1	1	2023-06-10	1	1	Dr. Sarah Adams	Cardiology	sarah.adams@example.com

Query History | Query

```
-- Select all appointments with patient and doctor details :
SELECT appointment.*, patient.name, patient.contact_number, patient.address,
doctor.name, doctor.speciality, doctor.email FROM appointment
JOIN patient ON appointment.patient_ID = patient.patient_ID
JOIN doctor ON appointment.doctor_ID = doctor.doctor_ID;
```

Data Output | Messages | Notifications

	appointment_id integer	appointment_date date	appointment_time time without time zone	appointment_status character varying (40)	patient_id integer	doctor_id integer	name character varying (40)	contact_number character varying (40)	address character varying (40)	name character va
1	1	2023-06-10	10:00:00	Scheduled	1	1	Flavien Geoffray	1234567890	789 Maple Street	Dr. Sarah Ad
2	2	2023-06-11	15:30:00	Scheduled	2	2	Alexiane Laroye	9876543210	567 Oakwood Avenue	Dr. Michael
3	3	2023-06-12	14:00:00	Scheduled	3	3	Enzo Jehl	5555555555	456 Elmwood Drive	Dr. Jennifer
4	4	2023-06-13	16:00:00	Scheduled	4	4	Wissame Ismael	1516484622	233 Togo Street	Dr. Doctor

Query History | Query

```
-- select the patient name, the admission date and the diagnosis for all medical records
SELECT patient.name, medical_record.admission_date, medical_record.diagnosis FROM medical_record
JOIN patient ON medical_record.patient_ID = patient.patient_ID;
```

Data Output | Messages | Notifications

	name character varying (40)	admission_date date	diagnosis character varying (40)
1	Flavien Geoffray	2023-06-11	Hypertension
2	Alexiane Laroye	2023-06-11	Common cold
3	Enzo Jehl	2023-06-11	Fractured ankle
4	Wissame Ismael	2023-06-11	Parkinson

Query History | Query

```
-- select the patient name, the doctor name and the appointment date for all appointments.
SELECT patient.name, doctor.name, appointment.appointment_date FROM appointment
JOIN patient ON appointment.patient_ID = patient.patient_ID
JOIN doctor ON appointment.doctor_ID = doctor.doctor_ID;
```

Data Output | Messages | Notifications

	name character varying (40)	name character varying (40)	appointment_date date
1	Flavien Geoffray	Dr. Sarah Adams	2023-06-10
2	Alexiane Laroye	Dr. Michael Davis	2023-06-11
3	Enzo Jehl	Dr. Jennifer Lee	2023-06-12
4	Wissame Ismael	Dr. Doctor	2023-06-13

Screenshot

Query History | Query

```
1 -- select the department name and the doctor name for all doctors and their department
2 SELECT doctor.name, department.department_name FROM doctor
3 JOIN department ON doctor.speciality = department.department_name;
```

Data Output | Messages | Notifications

	name character varying (40)	department_name character varying (40)
1	Dr. Sarah Adams	Cardiology
2	Dr. Michael Davis	Pediatrics
3	Dr. Jennifer Lee	Orthopedics
4	Dr. Doctor	Neurology

Query History | Query

```
1 -- select the patient name, the appointment date and the address for all scheduled appointments
2 SELECT patient.name, appointment.appointment_date, patient.address
3 FROM appointment
4 JOIN patient ON appointment.patient_ID = patient.patient_ID
5 WHERE appointment.appointment_status = 'Scheduled';
```

Data Output | Messages | Notifications

	name character varying (40)	appointment_date date	address character varying (40)
1	Flavien Geoffray	2023-06-10	789 Maple Street
2	Alexiane Laroye	2023-06-11	567 Oakwood Avenue
3	Enzo Jehl	2023-06-12	456 Elmwood Drive
4	Wissame Ismael	2023-06-13	233 Togo Street

Query History | Query

```
1 -- select the name of the patient, the admission date and the diagnosis for a specific diagnosis
2 SELECT patient.name, medical_record.admission_date, medical_record.diagnosis
3 FROM patient
4 JOIN medical_record ON patient.patient_ID = medical_record.patient_ID
5 WHERE medical_record.diagnosis = 'Hypertension';
```

Data Output | Messages | Notifications

	name character varying (40)	admission_date date	diagnosis character varying (40)
1	Flavien Geoffray	2023-06-11	Hypertension

Query History | Query

```
1 -- select all appointments per department with the name of the doctor and the patient
2 SELECT d.department_name, a.appointment_ID, a.appointment_date, a.appointment_time, doc.name AS doctor_name, p.name AS patient_name
3 FROM department d
4 JOIN doctor doc ON d.department_ID = doc.department_ID
5 JOIN appointment a ON doc.doctor_ID = a.doctor_ID
6 JOIN patient p ON a.patient_ID = p.patient_ID;
```

Data Output | Messages | Notifications

	department_name character varying (40)	appointment_id integer	appointment_date date	appointment_time time without time zone	doctor_name character varying (40)	patient_name character varying (40)
1	Cardiology	1	2023-06-10	10:00:00	Dr. Sarah Adams	Flavien Geoffray
2	Pediatrics	2	2023-06-11	15:30:00	Dr. Michael Davis	Alexiane Laroye
3	Orthopedics	3	2023-06-12	14:00:00	Dr. Jennifer Lee	Enzo Jehl
4	Neurology	4	2023-06-13	16:00:00	Dr. Doctor	Wissame Ismael

Screenshot

Query History | Query

```sql
-- select the medical record with a patient name
SELECT medical_record.*, patient.name AS patient_name
FROM medical_record
JOIN patient ON medical_record.patient_ID = patient.patient_ID
where patient.name = 'Alexiane Laroye';
```

Data Output | Messages | Notifications

	medical_record_id integer	admission_date date	discharge_date date	diagnosis character varying (40)	patient_id integer	patient_name character varying (40)
1	2	2023-06-11	2023-06-16	Common cold	2	Alexiane Laroye

Query History | Query

```sql
-- display all patients with their appointmens and the doctor data :
SELECT p.patient_ID, p.name AS patient_name, p.gender, p.date_of_birth, p.contact_number,
p.address, a.appointment_ID, a.appointment_date, a.appointment_time, a.appointment_status,
d.doctor_ID, d.name AS doctor_name, d.speciality, d.contact_number AS doctor_contact
FROM patient p
JOIN appointment a ON p.patient_ID = a.patient_ID
JOIN doctor d ON a.doctor_ID = d.doctor_ID;
```

Data Output | Messages | Notifications

	patient_id integer	patient_name character varying (40)	gender character	date_of_birth date	contact_number character varying (40)	address character varying (40)	appointment_id integer	appointment_date date	appointment_time time without time zone	appointment_status character varying (40)
1	1	Flavien Geoffray	M	2003-10-03	1234567890	789 Maple Street	1	2023-06-10	10:00:00	Scheduled
2	2	Alexiane Laroye	F	2003-04-06	9876543210	567 Oakwood Avenue	2	2023-06-11	15:30:00	Scheduled
3	3	Enzo Jehl	M	2003-10-10	5555555555	456 Elmwood Drive	3	2023-06-12	14:00:00	Scheduled
4	4	Wissame Ismael	M	2004-08-29	1516484622	233 Togo Street	4	2023-06-13	16:00:00	Scheduled

```sql
-- select the number of patients assigned foreach nurse, by decreasing order
SELECT n.name, COUNT(*) AS patient_count
FROM nurse n
INNER JOIN patient p ON n.nurse_ID = p.nurse_ID
GROUP BY n.name
ORDER BY patient_count DESC;
```

Data Output | Messages | Notifications

	name character varying (40)	patient_count bigint
1	Nurse Emily Thompson	2
2	Nurse Mark Wilson	1
3	Nurse Theo Oliver	1
4	Nurse Lisa Johnson	1

Query History | Query

```sql
-- select the number of patients per department, by decreasing order :
SELECT department_name, COUNT(*) AS patient_count
FROM patient
INNER JOIN department ON patient.contacted_department = department.department_name
GROUP BY department_name
ORDER BY patient_count DESC;
```

Data Output | Messages | Notifications

	department_name character varying (40)	patient_count bigint
1	Cardiology	1
2	Pediatrics	1
3	Orthopedics	1
4	Neurology	1

Screenshot

```sql
-- select the date, the number of appointments foreach date, by decreasing order
SELECT appointment_date, COUNT(*) AS appointment_count
FROM appointment
GROUP BY appointment_date
ORDER BY appointment_date;
```

Data Output | Messages | Notifications

	appointment_date date	appointment_count bigint
1	2023-06-10	1
2	2023-06-11	1
3	2023-06-12	1
4	2023-06-13	1

```sql
-- select the number of appointments foreach doctors by decreasing order. display the name of the patient
SELECT d.name AS doctor_name, p.name AS patient_name, COUNT(a.appointment_ID) AS total_appointments
FROM doctor d
JOIN appointment a ON d.doctor_ID = a.doctor_ID
JOIN patient p ON a.patient_ID = p.patient_ID
GROUP BY d.doctor_ID, d.name, p.patient_ID, p.name
ORDER BY total_appointments ASC;
```

Data Output | Messages | Notifications

	doctor_name character varying (40)	patient_name character varying (40)	total_appointments bigint
1	Dr. Sarah Adams	Flavien Geoffray	1
2	Dr. Michael Davis	Alexiane Laroye	1
3	Dr. Jennifer Lee	Enzo Jehl	1
4	Dr. Doctor	Wissame Ismael	1

```sql
-- select the number of appointments foreach patient, by decreasing order
SELECT p.patient_ID, p.name, COUNT(*) AS appointment_count
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
GROUP BY p.patient_ID, p.name
ORDER BY appointment_count DESC;
```

Data Output | Messages | Notifications

	patient_id [PK] integer	name character varying (40)	appointment_count bigint
1	3	Enzo Jehl	1
2	4	Wissame Ismael	1
3	2	Alexiane Laroye	1
4	1	Flavien Geoffray	1

```sql
-- select patients with maximum 3 appointments et display their name and the number of appointments and the doctor name
SELECT p.name AS patient_name, COUNT(*) AS appointment_count, d.name AS doctor_name
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
GROUP BY p.patient_ID, p.name, d.name
HAVING COUNT(*) < 3;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	appointment_count bigint	doctor_name character varying (40)
1	Alexiane Laroye	1	Dr. Michael Davis
2	Wissame Ismael	1	Dr. Doctor
3	Flavien Geoffray	1	Dr. Sarah Adams
4	Enzo Jehl	1	Dr. Jennifer Lee

Screenshot

```sql
-- Select patients with at most 1 appointment and display their name, the name of the doctor who treated them, and in which department:
SELECT p.name AS patient_name, d.name AS doctor_name, dep.department_name
FROM patient p
LEFT JOIN appointment a ON p.patient_ID = a.patient_ID
LEFT JOIN doctor d ON a.doctor_ID = d.doctor_ID
LEFT JOIN department dep ON d.department_ID = dep.department_ID
GROUP BY p.patient_ID, p.name, d.name, dep.department_name
HAVING COUNT(a.appointment_ID) <= 1;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	doctor_name character varying (40)	department_name character varying (40)
1	Rocky Balboa	[null]	[null]
2	Wissame Ismael	Dr. Doctor	Neurology
3	Alexiane Laroye	Dr. Michael Davis	Pediatrics
4	Enzo Jehl	Dr. Jennifer Lee	Orthopedics
5	Flavien Geoffray	Dr. Sarah Adams	Cardiology

```sql
-- select patients and doctors who have a scheduled appointment for a specific date, by alphabetic order of the patient name
SELECT p.name AS patient_name, d.name AS doctor_name
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
WHERE a.appointment_date = '2023-06-10'
ORDER BY p.name;
```

Data Output | Messages | Notifications

	patient_name character varying (40)	doctor_name character varying (40)
1	Flavien Geoffray	Dr. Sarah Adams

```sql
-- select patients with scheduled appointments with cardiology specialists, sorted in ascending order of appointment date:
SELECT p.name AS patient_name, a.appointment_date, d.name AS doctor_name
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
WHERE d.speciality = 'Cardiology'
ORDER BY a.appointment_date;
```

Data Output | Messages | Notifications

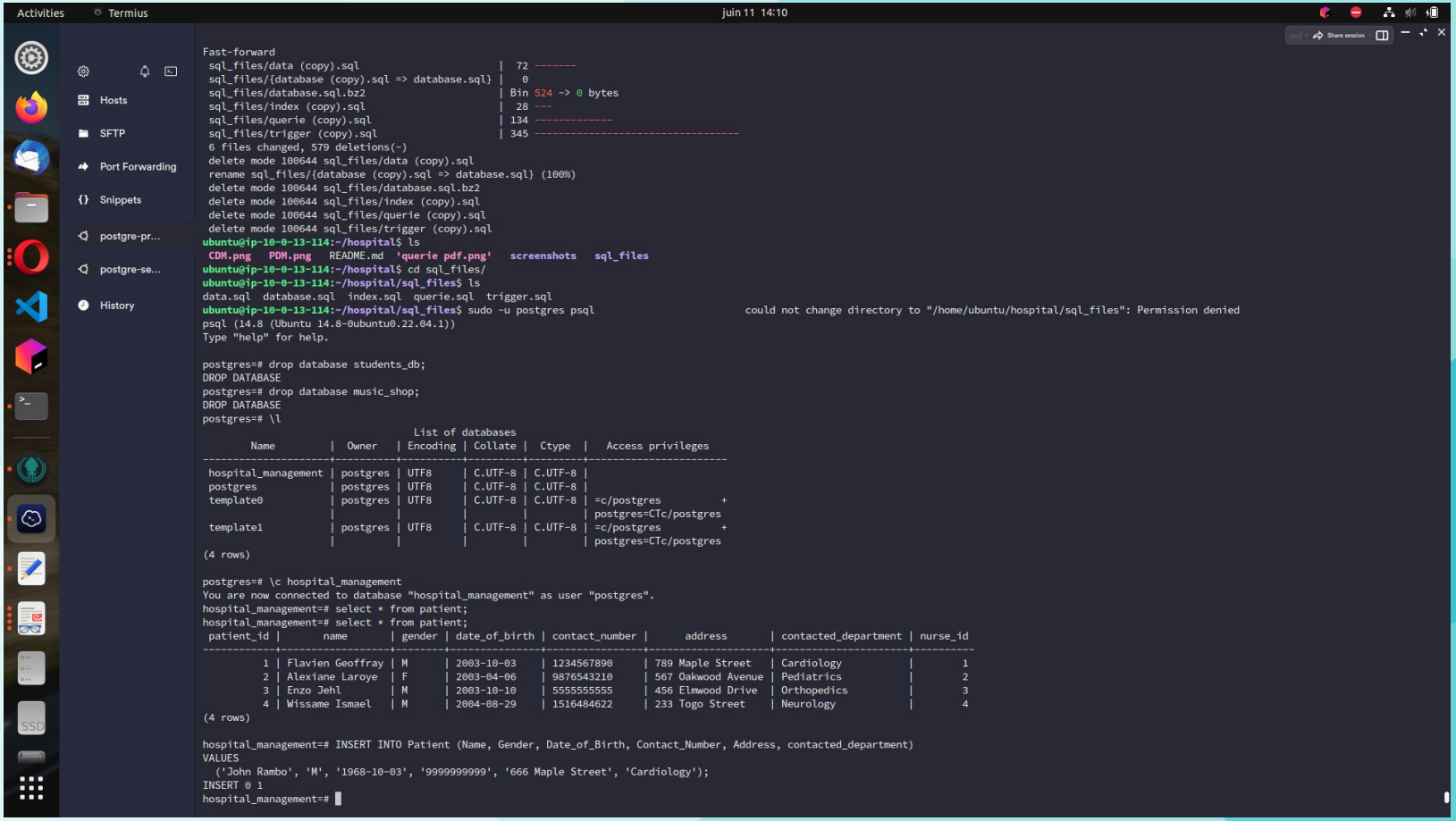
	patient_name character varying (40)	appointment_date date	doctor_name character varying (40)
1	Flavien Geoffray	2023-06-10	Dr. Sarah Adams

```sql
-- select patients with maximum 3 appointments et display their name and the number of appointments and the doctor name
SELECT p.name AS patient_name, COUNT(*) AS appointment_count, d.name AS doctor_name
FROM patient p
INNER JOIN appointment a ON p.patient_ID = a.patient_ID
INNER JOIN doctor d ON a.doctor_ID = d.doctor_ID
GROUP BY p.patient_ID, p.name, d.name
HAVING COUNT(*) < 3;
```

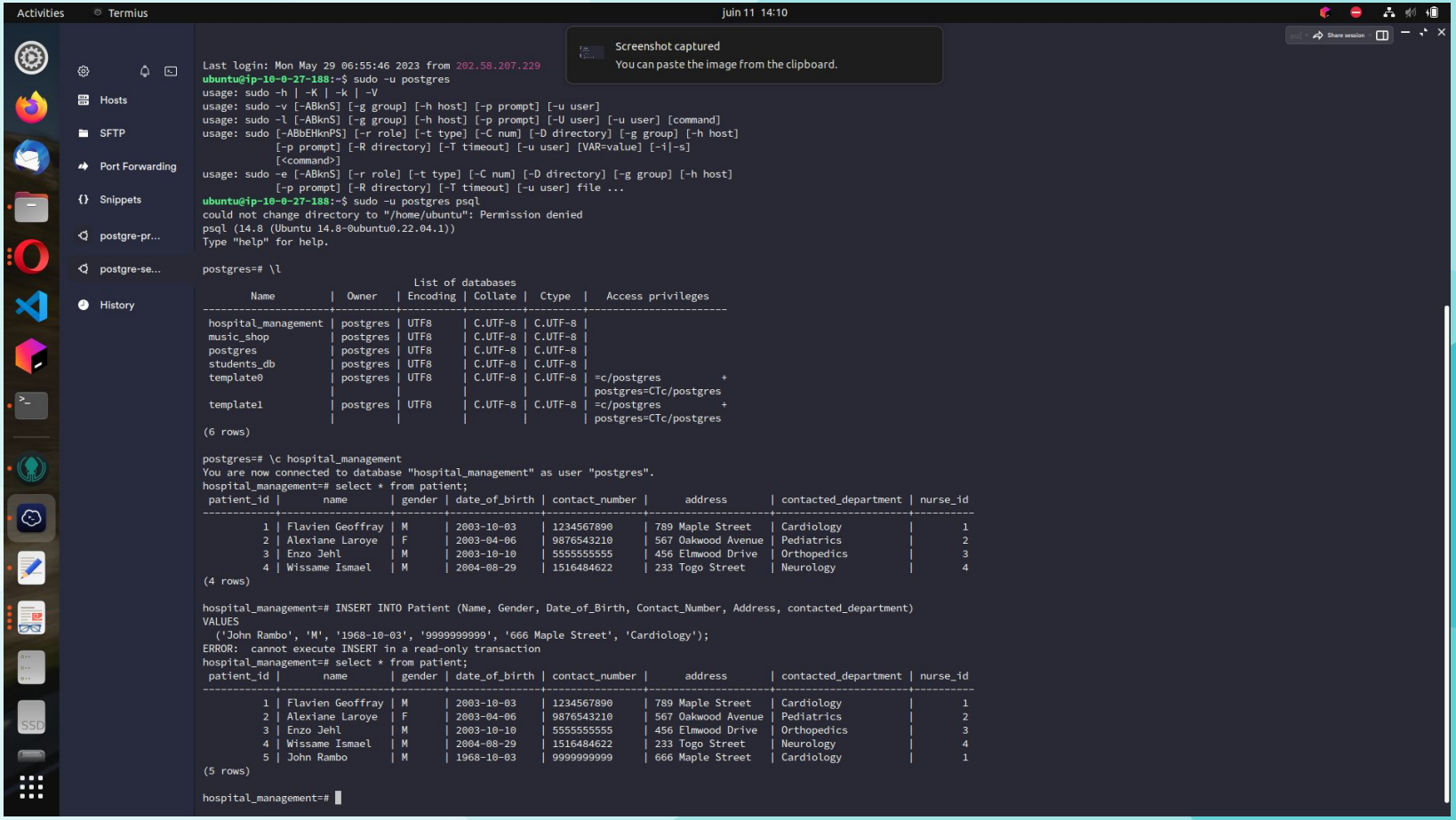
Data Output | Messages | Notifications

	patient_name character varying (40)	appointment_count bigint	doctor_name character varying (40)
1	Alexiane Laroye	1	Dr. Michael Davis
2	Wissame Ismael	1	Dr. Doctor
3	Flavien Geoffray	1	Dr. Sarah Adams
4	Enzo Jehl	1	Dr. Jennifer Lee

Screenshot


Activities
Termius
juin 11 14:10
Hosts
SFTP
Port Forwarding
Snippets
postgre-pr...
postgre-se...
History
Fast-forward
sql_files/data (copy).sql | 72 -------
sql_files/{database (copy).sql => database.sql} | 0
sql_files/database.sql.bz2 | Bin 524 -> 0 bytes
sql_files/index (copy).sql | 28 ---
sql_files/querie (copy).sql | 134 ----------
sql_files/trigger (copy).sql | 345 ----------
6 files changed, 579 deletions(-)
delete mode 100644 sql_files/data (copy).sql
rename sql_files/{database (copy).sql => database.sql} (100%)
delete mode 100644 sql_files/database.sql.bz2
delete mode 100644 sql_files/index (copy).sql
delete mode 100644 sql_files/querie (copy).sql
delete mode 100644 sql_files/trigger (copy).sql
ubuntu@ip-10-0-13-114:~/hospital$ ls
CDM.png PDM.png README.md 'querie pdf.png' screenshots sql_files
ubuntu@ip-10-0-13-114:~/hospital$ cd sql_files/
ubuntu@ip-10-0-13-114:~/hospital/sql_files$ ls
data.sql database.sql index.sql querie.sql trigger.sql
ubuntu@ip-10-0-13-114:~/hospital/sql_files$ sudo -u postgres psql could not change directory to "/home/ubuntu/hospital/sql_files": Permission denied
psql (14.8 (Ubuntu 14.8-0ubuntu0.22.04.1))
Type "help" for help.
postgres=# drop database students_db;
DROP DATABASE
postgres=# drop database music_shop;
DROP DATABASE
postgres=# \l
List of databases
Name | Owner | Encoding | Collate | Ctype | Access privileges
hospital_management | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
postgres | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
template0 | postgres | UTF8 | C.UTF-8 | C.UTF-8 | =c/postgres + postgres=CTc/postgres
template1 | postgres | UTF8 | C.UTF-8 | C.UTF-8 | =c/postgres + postgres=CTc/postgres
(4 rows)
postgres=# \c hospital_management
You are now connected to database "hospital_management" as user "postgres".
hospital_management=# select * from patient;
hospital_management=# select * from patient;
patient_id | name | gender | date_of_birth | contact_number | address | contacted_department | nurse_id
1 | Flavien Geoffray | M | 2003-10-03 | 1234567890 | 789 Maple Street | Cardiology | 1
2 | Alexiane Laroye | F | 2003-04-06 | 9876543210 | 567 Oakwood Avenue | Pediatrics | 2
3 | Enzo Jehl | M | 2003-10-10 | 5555555555 | 456 Elmwood Drive | Orthopedics | 3
4 | Wissame Ismael | M | 2004-08-29 | 1516484622 | 233 Togo Street | Neurology | 4
(4 rows)
hospital_management=# INSERT INTO Patient (Name, Gender, Date_of_Birth, Contact_Number, Address, contacted_department)
VALUES
('John Rambo', 'M', '1968-10-03', '9999999999', '666 Maple Street', 'Cardiology');
INSERT 0 1
hospital_management=#

Screenshot


Activities
Termius
juin 11 14:10
Screenshot captured
You can paste the image from the clipboard.
Hosts
SFTP
Port Forwarding
Snippets
postgre-pr...
postgre-se...
History
Last login: Mon May 29 06:55:46 2023 from 202.58.207.229
ubuntu@ip-10-0-27-188:~$ sudo -u postgres
usage: sudo -h | -K | -k | -V
usage: sudo -v [-ABknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-ABknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-ABbEHknPS] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host]
[-p prompt] [-R directory] [-T timeout] [-u user] [VAR=value] [-i|-s]
[<command>]
usage: sudo -e [-ABknS] [-r role] [-t type] [-C num] [-D directory] [-g group] [-h host]
[-p prompt] [-R directory] [-T timeout] [-u user] file ...
ubuntu@ip-10-0-27-188:~$ sudo -u postgres psql
could not change directory to "/home/ubuntu": Permission denied
psql (14.8 (Ubuntu 14.8-0ubuntu0.22.04.1))
Type "help" for help.
postgres=# \l
List of databases
Name | Owner | Encoding | Collate | Ctype | Access privileges
hospital_management | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
music_shop | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
postgres | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
students_db | postgres | UTF8 | C.UTF-8 | C.UTF-8 |
template0 | postgres | UTF8 | C.UTF-8 | C.UTF-8 | =c/postgres + postgres=CTc/postgres
template1 | postgres | UTF8 | C.UTF-8 | C.UTF-8 | =c/postgres + postgres=CTc/postgres
(6 rows)
postgres=# \c hospital_management
You are now connected to database "hospital_management" as user "postgres".
hospital_management=# select * from patient;
patient_id | name | gender | date_of_birth | contact_number | address | contacted_department | nurse_id
1 | Flavien Geoffray | M | 2003-10-03 | 1234567890 | 789 Maple Street | Cardiology | 1
2 | Alexiane Laroye | F | 2003-04-06 | 9876543210 | 567 Oakwood Avenue | Pediatrics | 2
3 | Enzo Jehl | M | 2003-10-10 | 5555555555 | 456 Elmwood Drive | Orthopedics | 3
4 | Wissame Ismael | M | 2004-08-29 | 1516484622 | 233 Togo Street | Neurology | 4
(4 rows)
hospital_management=# INSERT INTO Patient (Name, Gender, Date_of_Birth, Contact_Number, Address, contacted_department)
VALUES
('John Rambo', 'M', '1968-10-03', '9999999999', '666 Maple Street', 'Cardiology');
ERROR: cannot execute INSERT in a read-only transaction
hospital_management=# select * from patient;
patient_id | name | gender | date_of_birth | contact_number | address | contacted_department | nurse_id
1 | Flavien Geoffray | M | 2003-10-03 | 1234567890 | 789 Maple Street | Cardiology | 1
2 | Alexiane Laroye | F | 2003-04-06 | 9876543210 | 567 Oakwood Avenue | Pediatrics | 2
3 | Enzo Jehl | M | 2003-10-10 | 5555555555 | 456 Elmwood Drive | Orthopedics | 3
4 | Wissame Ismael | M | 2004-08-29 | 1516484622 | 233 Togo Street | Neurology | 4
5 | John Rambo | M | 1968-10-03 | 9999999999 | 666 Maple Street | Cardiology | 1
(5 rows)
hospital_management=#

THANKS!

Do you have any questions?