Anoop Reddy Yeddula
Siddhartha Challa

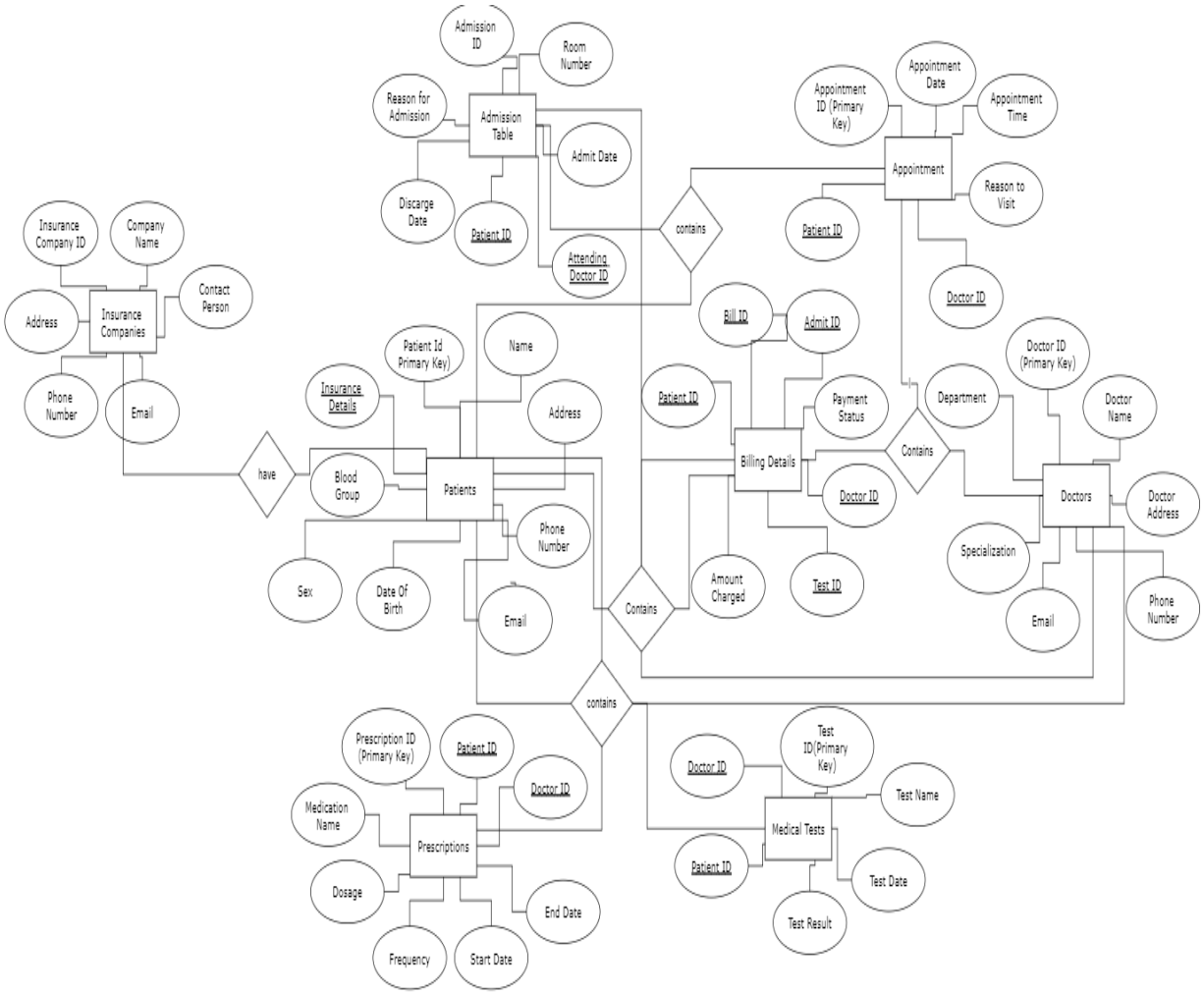
FINAL WRITE-UP

1. Full references:

https://www.kaggle.com/datasets/asjad99/mimiciii

2. ER diagram:

The ER diagram that was previously displayed has been revised. Below is the new ER diagram:



3. CREATE TABLE statements:

i) Patients Table:

```
CREATE TABLE Patients (
  PatientID INT PRIMARY KEY,
  PatientName VARCHAR(50) NOT NULL,
  PatientAddress VARCHAR(100) NOT NULL,
  PatientPhoneNumber VARCHAR(20) NOT NULL,
  PatientEmail VARCHAR(50) NOT NULL,
  DateOfBirth DATE NOT NULL,
  Sex VARCHAR(10) NOT NULL,
  BloodGroup VARCHAR(5) NOT NULL,
  InsuranceDetails VARCHAR(100)
);
```

ii) Doctors Table:

```
CREATE TABLE Doctors (
  DoctorID INT PRIMARY KEY,
  DoctorName VARCHAR(50) NOT NULL,
  DoctorAddress VARCHAR(100) NOT NULL,
  DoctorPhoneNumber VARCHAR(20) NOT NULL,
  DoctorEmail VARCHAR(50) NOT NULL,
  Department VARCHAR(50) NOT NULL,
  Specialization VARCHAR(50) NOT NULL
);
```

iii)Appointments Table:

```
CREATE TABLE Appointments (
  AppointmentID INT PRIMARY KEY,
  PatientID INT NOT NULL,
  DoctorID INT NOT NULL,
  AppointmentDate DATE NOT NULL,
  AppointmentTime TIME NOT NULL,
  ReasonForVisit VARCHAR(200),
  FOREIGN KEY (PatientID) REFERENCES Patients(PatientID),
  FOREIGN KEY (DoctorID) REFERENCES Doctors(DoctorID)
);
```

iv) Medical Tests Table:

```
CREATE TABLE Medical_Tests (
  TestID INT PRIMARY KEY,
```

```
  PatientID INT NOT NULL,
  DoctorID INT NOT NULL,
  TestName VARCHAR(50) NOT NULL,
  TestDate DATE NOT NULL,
  TestResults VARCHAR(200),
  FOREIGN KEY (PatientID) REFERENCES Patients(PatientID),
  FOREIGN KEY (DoctorID) REFERENCES Doctors(DoctorID)
);
```

v) Prescriptions Table:

```
CREATE TABLE Prescriptions (
  PrescriptionID INT PRIMARY KEY,
  PatientID INT NOT NULL,
  DoctorID INT NOT NULL,
  MedicationName VARCHAR(50) NOT NULL,
  Dosage VARCHAR(20) NOT NULL,
  Frequency VARCHAR(20) NOT NULL,
  StartDate DATE NOT NULL,
  EndDate DATE NOT NULL,
  FOREIGN KEY (PatientID) REFERENCES Patients(PatientID),
  FOREIGN KEY (DoctorID) REFERENCES Doctors(DoctorID)
);
```

vi) Admissions:

```
CREATE TABLE Admissions (
  AdmissionID INT PRIMARY KEY,
  PatientID INT NOT NULL,
  RoomNumber INT NOT NULL,
  AdmitDate DATE NOT NULL,
  DischargeDate DATE,
  ReasonForAdmission VARCHAR(200) NOT NULL,
  AttendingDoctorID INT NOT NULL,
  FOREIGN KEY (PatientID) REFERENCES Patients(PatientID),
  FOREIGN KEY (AttendingDoctorID) REFERENCES Doctors(DoctorID)
);
```

vii) Insurance Companies Table:

```
CREATE TABLE InsuranceCompanies (
  InsuranceCompanyID INT PRIMARY KEY,
  InsuranceCompanyName VARCHAR(50) NOT NULL,
  Address VARCHAR(100) NOT NULL,
```

```
  PhoneNumber VARCHAR(20) NOT NULL,
  Email VARCHAR(50),
  ContactPerson VARCHAR(50)
);
```

viii) Billing Table:

```
CREATE TABLE Billing (
  BillID INT PRIMARY KEY,
  PatientID INT NOT NULL,
  AdmitID INT,
  TestID INT,
  DoctorID INT NOT NULL,
  AmountCharged DECIMAL(10, 2) NOT NULL,
  PaymentStatus VARCHAR(20) NOT NULL,
  FOREIGN KEY (PatientID) REFERENCES Patients(PatientID),
  FOREIGN KEY (AdmitID) REFERENCES Admissions(AdmissionID),
  FOREIGN KEY (TestID) REFERENCES Medical_Tests(TestID),
  FOREIGN KEY (DoctorID) REFERENCES Doctors(DoctorID)
);
```

4. Description:

All the datasets of CSV files were downloaded from the website.

A new set of csv files was downloaded, and each dataset had its data cleaned.

To clean up the raw data, here is some sample Python code:

bring in pandas as pd

Open the CSV file by entering df = pd.read csv("filename.csv").

Do not leave any blank spaces at the beginning or end of the column names. df.columns equals df.columns.str.strip ()

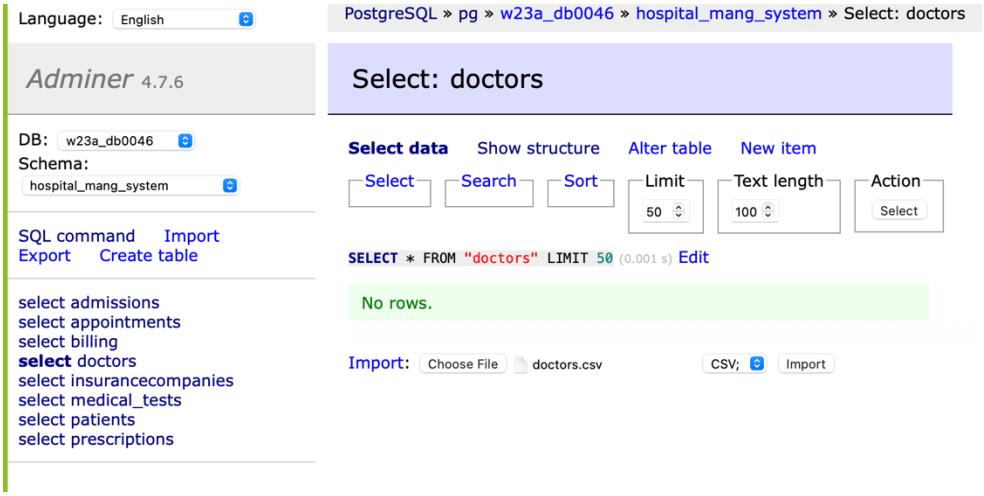
Remove any rows with blank values. df.dropna(inplace=True)

Change the necessary type for the attribute column. "attribute-name" = "attribute name," etc. astype(type)

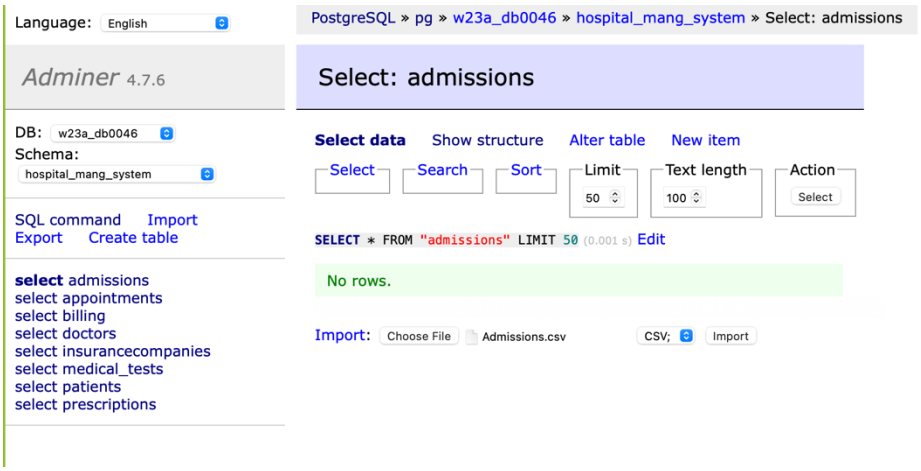
After dropping rows, reset the index. (drop=True, inplace=True) df.reset index

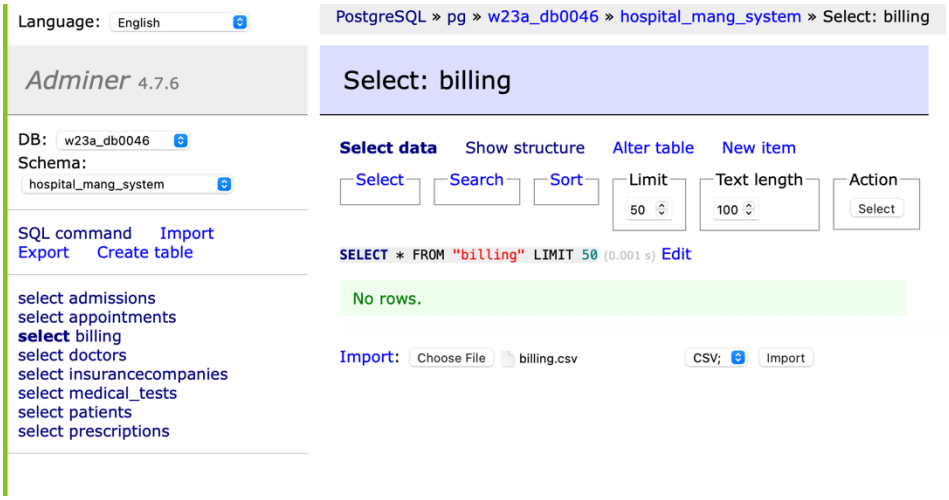
Use the command df.to csv("new filename.csv", index=False) to save the cleaned data to a new CSV file.

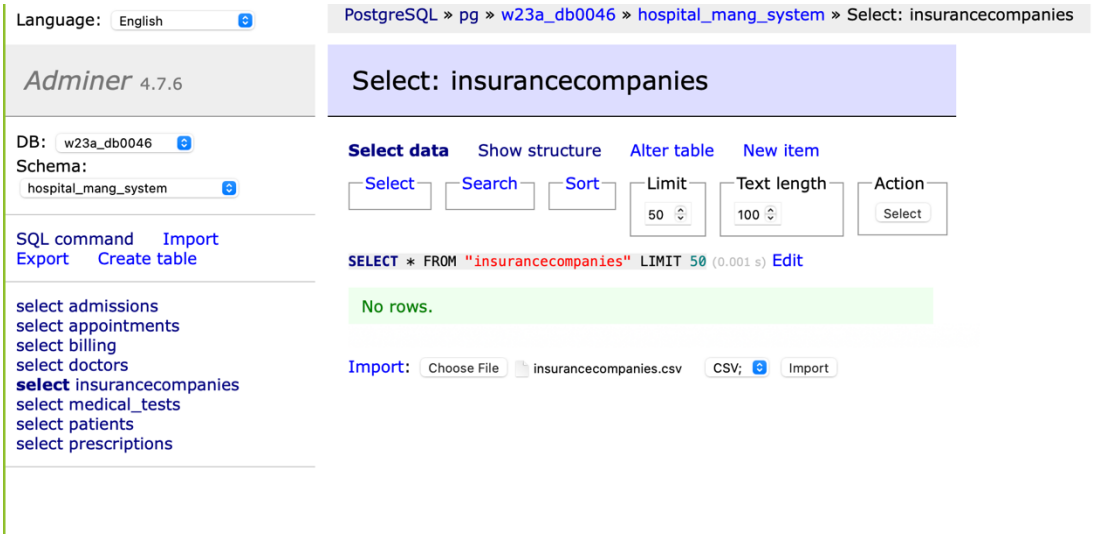
All the tables were created, and the corresponding csv files were imported.

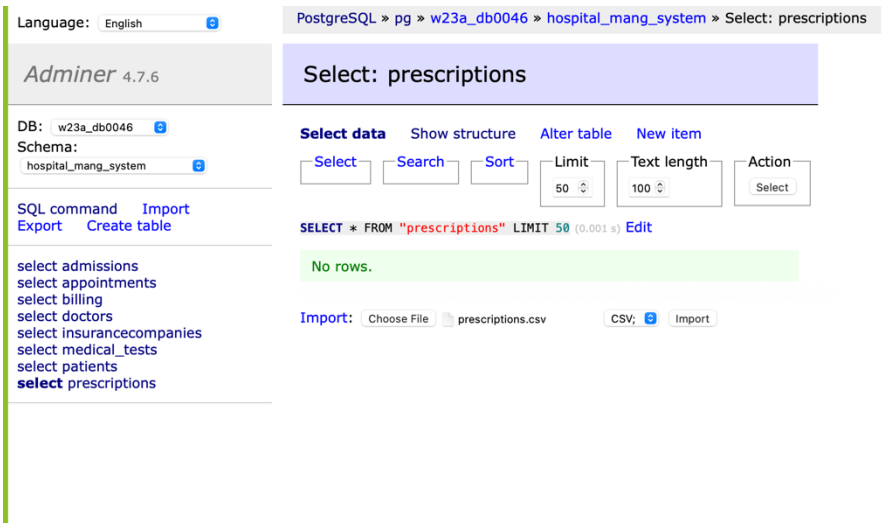


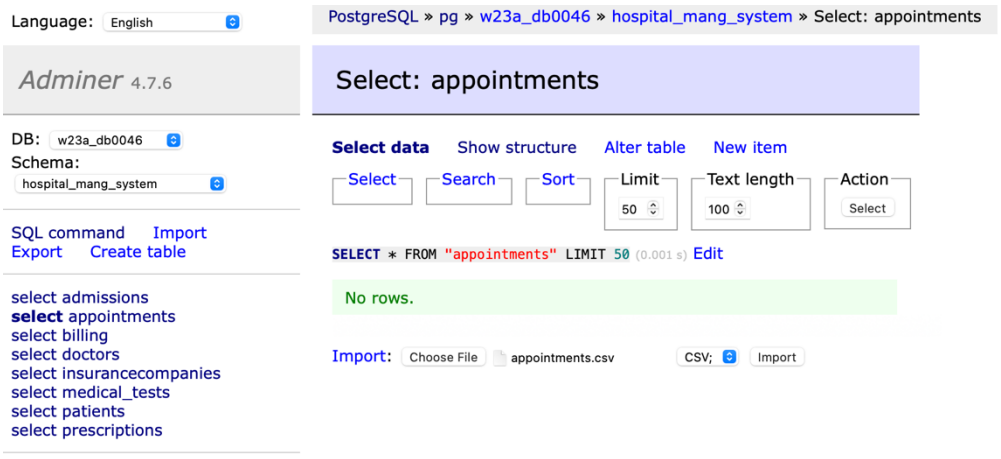

Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: medical_tests
Select: medical_tests
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "medical_tests" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File medical_tests.csv CSV; Import


Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: admissions
Select: admissions
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "admissions" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File Admissions.csv CSV; Import


Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: billing
Select: billing
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "billing" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File billing.csv CSV; Import


Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: insurancecompanies
Select: insurancecompanies
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "insurancecompanies" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File insurancecompanies.csv CSV; Import


Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: prescriptions
Select: prescriptions
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "prescriptions" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File prescriptions.csv CSV; Import


Language: English
Adminer 4.7.6
DB: w23a_db0046
Schema: hospital_mang_system
SQL command Import
Export Create table
select admissions
select appointments
select billing
select doctors
select insurancecompanies
select medical_tests
select patients
select prescriptions
PostgreSQL » pg » w23a_db0046 » hospital_mang_system » Select: appointments
Select: appointments
Select data Show structure Alter table New item
Select Search Sort Limit 50 Text length 100 Action Select
SELECT * FROM "appointments" LIMIT 50 (0.001 s) Edit
No rows.
Import: Choose File appointments.csv CSV; Import

5. Queries:

We have changed queries for 11,12,17 and 20 from the ones we previously submitted. We realized that this was not how we tested ourselves and that we could not complicate a straightforward inquiry by using many notions. Hence, we made the decision to edit the questions and replace them with ones that covered the principles we had learned in class.

1) Which patients have appointments with Dr. Smith this week?

SELECT Patients.patientid, Patients.patientname, Appointments.appointmentdate, Appointments.appointmenttime
FROM Patients
INNER JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.doctorid = 10
AND Appointments.appointmentdate >= CURRENT_DATE - EXTRACT(DOW FROM CURRENT_DATE)::INTEGER
AND Appointments.appointmentdate <= CURRENT_DATE + (6 - EXTRACT(DOW FROM CURRENT_DATE))::INTEGER;

```
SELECT Patients.patientid, Patients.patientname, Appointments.appointmentdate, Appointments.appointmenttime
FROM Patients
INNER JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.doctorid = 10
AND Appointments.appointmentdate >= CURRENT_DATE - EXTRACT(DOW FROM CURRENT_DATE)::INTEGER
AND Appointments.appointmentdate <= CURRENT_DATE + (6 - EXTRACT(DOW FROM CURRENT_DATE))::INTEGER
```

patientid	patientname	appointmentdate	appointmenttime
20	Anthia Hardy-Piggin	2023-03-20	16:00:00

appointments.appointmentdate

1 row (0.002 s) Edit, Explain, Export

2) How many patients have visited the hospital this month?
SELECT COUNT(DISTINCT patientid)
FROM Appointments

WHERE appointmentdate BETWEEN date_trunc('month', CURRENT_DATE) AND CURRENT_DATE;

```
SELECT COUNT(DISTINCT patientid)
FROM Appointments
WHERE appointmentdate BETWEEN date_trunc('month', CURRENT_DATE) AND CURRENT_DATE
```

count
1

1 row (0.003 s) Edit, Explain, Export

3) What is the average age of male patients?
SELECT AVG(EXTRACT(year FROM age(dateofbirth))) AS "Average Age"
FROM Patients
WHERE Sex = 'M';

```
SELECT AVG(EXTRACT(year FROM age(dateofbirth))) AS "Average Age"
FROM Patients
WHERE Sex = 'M'
```

Average Age
NULL

1 row (0.003 s) Edit, Explain, Export

4) Which patients have been admitted to the hospital for more than 3 days?

SELECT d.DoctorName, d.DoctorPhoneNumber, d.Doctoremail
FROM Doctors d
JOIN medical_tests m ON m.Doctorid = d.Doctorid
JOIN Patients p ON p.Patientid = m.Patientid
WHERE m.TestResults = (SELECT MAX(TestResults) FROM Medical_tests)

```
SELECT d.DoctorName, d.DoctorPhoneNumber, d.Doctoremail
FROM Doctors d
JOIN medical_tests m ON m.Doctorid = d.Doctorid
JOIN Patients p ON p.Patientid = m.Patientid
WHERE m.TestResults = (SELECT MAX(TestResults) FROM Medical_tests)
```

doctorname	doctorphonenumber	doctoremail
Dr. Sara Lee	555-3456	sara.lee@email.com
Dr. Mike Adams	555-7890	mike.adams@email.com
Dr. Emily Brown	555-4321	emily.brown@email.com
Dr. Sara Lee	555-3456	sara.lee@email.com
Dr. Bob Johnson	555-9012	bob.johnson@email.com
Dr. John Smith	555-1234	john.smith@email.com
Dr. Emily Brown	555-4321	emily.brown@email.com
Dr. Jane Doe	555-5678	jane.doe@email.com
Dr. John Smith	555-1234	john.smith@email.com
Dr. Sara Lee	555-3456	sara.lee@email.com

10 rows (0.004 s) Edit, Explain, Export

5) Which patients have not visited the hospital in the last 6 months?
SELECT Patients.patientid, Patients.patientname
FROM Patients
LEFT JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.appointmentdate < (CURRENT_DATE - INTERVAL '6 months')
OR Appointments.appointmentdate IS NULL;

```
SELECT Patients.patientid, Patients.patientname
FROM Patients
LEFT JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.appointmentdate < (CURRENT_DATE - INTERVAL '6 months')
OR Appointments.appointmentdate IS NULL
```

patientid	patientname
4	Sara Lee
9	Chris Evans

2 rows (0.002 s) Edit, Explain, Export

6) What are the top 5 most prescribed medications?

SELECT Prescriptions.medicationname, COUNT(Prescriptions.medicationname) AS "Prescriptions Count"
FROM Prescriptions
GROUP BY Prescriptions.medicationname
ORDER BY COUNT(Prescriptions.medicationname) DESC
LIMIT 5;

```
SELECT Prescriptions.medicationname, COUNT(Prescriptions.medicationname) AS "Prescriptions Count"
FROM Prescriptions
GROUP BY Prescriptions.medicationname
ORDER BY COUNT(Prescriptions.medicationname) DESC
LIMIT 5
```

medicationname	Prescriptions Count
Albuterol	1
Warfarin	1
Omeprazole	1
Amlodipine	1
Doxycycline	1

5 rows (0.002 s) Edit, Explain, Export

7) How many patients have a blood group of AB+?

SELECT COUNT(patientid)
FROM Patients
WHERE bloodgroup = 'AB+'

```
SELECT COUNT(patientid)
FROM Patients
WHERE bloodgroup = 'AB+'
```

count
3

1 row (0.002 s) Edit, Explain, Export

8) What is the total amount charged for medical tests performed this year?

SELECT SUM(Billing.amountcharged) AS "Total Amount Charged"
FROM Billing
INNER JOIN medical_tests ON Billing.testid = medical_tests.testid
WHERE EXTRACT(YEAR FROM medical_tests.testdate) = EXTRACT(YEAR FROM CURRENT_DATE);

```
SELECT SUM(Billing.amountcharged) AS "Total Amount Charged"
FROM Billing
INNER JOIN medical_tests ON Billing.testid = medical_tests.testid
WHERE EXTRACT(YEAR FROM medical_tests.testdate) = EXTRACT(YEAR FROM CURRENT_DATE)
```

Total Amount Charged
NULL

1 row (0.004 s) Edit, Explain, Export

9) Which patients have undergone an MRI test?

SELECT Patients.patientid, Patients.patientname, Medical_Tests.testdate
FROM Patients
INNER JOIN Medical_Tests
ON Patients.patientid = Medical_Tests.patientid
WHERE Medical_Tests.testname = 'MRI';

```
SELECT Patients.patientid, Patients.patientname, Medical_Tests.testdate
FROM Patients
INNER JOIN Medical_Tests
ON Patients.patientid = Medical_Tests.patientid
WHERE Medical_Tests.testname = 'MRI'
```

patientid	patientname	testdate
3	Bob Johnson	2022-03-15
10	Linda Rodriguez	2022-10-20
17	Jarrod Laneham	2023-05-25

3 rows (0.002 s) Edit, Explain, Export

10) How many patients have a medical condition that requires regular follow-up appointments?

SELECT COUNT(DISTINCT Patients.patientid)
FROM Patients
INNER JOIN Appointments ON Patients.patientid = Appointments.patientid
INNER JOIN Prescriptions ON Patients.patientid = Prescriptions.patientid
WHERE Prescriptions.Frequency IN ('Monthly', 'Weekly', 'Daily');

```
SELECT COUNT(DISTINCT Patients.patientid)
FROM Patients
INNER JOIN Appointments ON Patients.patientid = Appointments.patientid
INNER JOIN Prescriptions ON Patients.patientid = Prescriptions.patientid
WHERE Prescriptions.Frequency IN ('Monthly', 'Weekly', 'Daily')
```

count
0

1 row (0.004 s) Edit, Explain, Export

11) Retrieve the names of all the patients who have had a medical test conducted by a doctor in the Oncology department and have been prescribed medication with a start date within the last month.

SELECT p.Patientname
FROM Patients p
JOIN Medical_tests m ON m.Patientid = p.Patientid
JOIN Prescriptions pr ON pr.Patientid = p.Patientid
JOIN Doctors d ON d.Doctorid = m.Doctorid
WHERE d.department='Cardiology'

```
SELECT p.Patientname
FROM Patients p
JOIN Medical_tests m ON m.Patientid = p.Patientid
JOIN Prescriptions pr ON pr.Patientid = p.Patientid
JOIN Doctors d ON d.Doctorid = m.Doctorid
WHERE d.department='Cardiology'
```

patientname
Bob Johnson
David Kim
Karen Johnson
Hermy Plet

4 rows (0.004 s) Edit, Explain, Export

12) Find all appointments that have been scheduled for a specific date:

SELECT Appointments.AppointmentID, Patients.PatientName, Doctors.DoctorName, Appointments.AppointmentTime, Appointments.ReasonForVisit
FROM Appointments
INNER JOIN Patients ON Appointments.PatientID = Patients.PatientID
INNER JOIN Doctors ON Appointments.DoctorID = Doctors.DoctorID
WHERE Appointments.AppointmentDate = '2023-03-20'

```
SELECT Appointments.AppointmentID, Patients.PatientName, Doctors.DoctorName, Appointments.AppointmentTime, Appointments.ReasonForVisit
FROM Appointments
INNER JOIN Patients ON Appointments.PatientID = Patients.PatientID
INNER JOIN Doctors ON Appointments.DoctorID = Doctors.DoctorID
WHERE Appointments.AppointmentDate = '2023-03-20'
```

appointmentid	patientname	doctorname	appointmenttime	reasonforvisit
20	Anthia Hardy-Piggin	Dr. Linda Rodriguez	16:00:00	Joint pain

1 row (0.002 s) Edit, Explain, Export

13) What are the names of the doctors who have treated the most patients in the last quarter?

SELECT Doctors.doctorname, COUNT(DISTINCT Appointments.patientid) AS "Patients Treated"
FROM Doctors
INNER JOIN Appointments ON Doctors.doctorid = Appointments.doctorid
WHERE Appointments.appointmentdate BETWEEN date_trunc('quarter', current_date - interval '3 month') AND current_date
GROUP BY Doctors.doctorname
ORDER BY COUNT(DISTINCT Appointments.patientid) DESC
LIMIT 5;

```
SELECT Doctors.doctorname, COUNT(DISTINCT Appointments.patientid) AS "Patients Treated"
FROM Doctors
INNER JOIN Appointments ON Doctors.doctorid = Appointments.doctorid
WHERE Appointments.appointmentdate BETWEEN date_trunc('quarter', current_date - interval '3 month') AND current_date
GROUP BY Doctors.doctorname
ORDER BY COUNT(DISTINCT Appointments.patientid) DESC
LIMIT 5
```

doctorname	Patients Treated
Dr. Linda Rodriguez	1

1 row (0.004 s) Edit, Explain, Export

14) What is the average age of patients who have undergone surgery in the last year?

SELECT AVG(EXTRACT(year FROM age(Patients.dateofbirth, CURRENT_DATE))) AS "Average Age"
FROM Patients
INNER JOIN admissions
ON Patients.patientid = admissions.patientid
WHERE EXTRACT(year FROM admissions.admitdate) = EXTRACT(year FROM CURRENT_DATE) - 1;

```
SELECT AVG(EXTRACT(year FROM age(Patients.dateofbirth, CURRENT_DATE))) AS "Average Age"
FROM Patients
INNER JOIN admissions
ON Patients.patientid = admissions.patientid
WHERE EXTRACT(year FROM admissions.admitdate) = EXTRACT(year FROM CURRENT_DATE) - 1
```

Average Age
-37.333333333333336

1 row (0.005 s) Edit, Explain, Export

15) Which patients have visited the hospital more than 5 times in the last 6 months?

```
SELECT Patients.patientid, Patients.patientname, COUNT(Appointments.patientid) AS "Visit Count"
FROM Patients
INNER JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.appointmentdate BETWEEN CURRENT_DATE - INTERVAL '6 months' AND CURRENT_DATE
GROUP BY Patients.patientid, Patients.patientname
HAVING COUNT(Appointments.patientid) > 5;
```

```
SELECT Patients.patientid, Patients.patientname, COUNT(Appointments.patientid) AS "Visit Count"
FROM Patients
INNER JOIN Appointments
ON Patients.patientid = Appointments.patientid
WHERE Appointments.appointmentdate BETWEEN CURRENT_DATE - INTERVAL '6 months' AND CURRENT_DATE
GROUP BY Patients.patientid, Patients.patientname
HAVING COUNT(Appointments.patientid) > 5
```

No rows.

(0.004 s) Edit, Explain, Export

16) What is the most common reason for hospitalization among patients over 60 years of age?

```
SELECT appointments.reasonforvisit, COUNT(appointments.reasonforvisit) AS "Count"
FROM appointments
INNER JOIN Patients
ON appointments.patientid = Patients.patientid
WHERE EXTRACT(year FROM age(CURRENT_DATE, Patients.dateofbirth)) > 60
GROUP BY appointments.reasonforvisit
ORDER BY COUNT(appointments.reasonforvisit) DESC
LIMIT 1;
```

```
SELECT appointments.reasonforvisit, COUNT(appointments.reasonforvisit) AS "Count"
FROM appointments
INNER JOIN Patients
ON appointments.patientid = Patients.patientid
WHERE EXTRACT(year FROM age(CURRENT_DATE, Patients.dateofbirth)) > 60
GROUP BY appointments.reasonforvisit
ORDER BY COUNT(appointments.reasonforvisit) DESC
LIMIT 1
```

reasonforvisit	Count
Eye exam	1

1 row (0.003 s) Edit, Explain, Export

17) How many patients have a slight arrhythmia was detected medical condition?

SELECT COUNT(DISTINCT Patients.patientid)
FROM Patients
INNER JOIN Medical_tests
ON Patients.patientid = Medical_tests.patientid
WHERE Medical_tests.testresults = 'A slight arrhythmia was detected.';

```
SELECT COUNT(DISTINCT Patients.patientid)
FROM Patients
INNER JOIN Medical_tests
ON Patients.patientid = Medical_tests.patientid
WHERE Medical_tests.testresults = 'A slight arrhythmia was detected.'
```

count
2

1 row (0.003 s) Edit, Explain, Export

18) Which patients have received the same prescription for more than 6 months?

SELECT Patients.patientid, Patients.patientname, Prescriptions.medicationname,
Prescriptions.frequency
FROM Patients
INNER JOIN Prescriptions
ON Patients.patientid = Prescriptions.patientid
WHERE Prescriptions.frequency = 'Monthly'
AND EXTRACT(month FROM age(current_date, Prescriptions.startdate)) >= 6;

```
SELECT Patients.patientid, Patients.patientname, Prescriptions.medicationname, Prescriptions.frequency
FROM Patients
INNER JOIN Prescriptions
ON Patients.patientid = Prescriptions.patientid
WHERE Prescriptions.frequency = 'Monthly'
AND EXTRACT(month FROM age(current_date, Prescriptions.startdate)) >= 6
```

No rows.

(0.003 s) Edit, Explain, Export

19) What is the average waiting time for patients to see a doctor?

SELECT AVG(EXTRACT(EPOCH FROM (Appointments.appointmentdate - Appointments.appointmenttime)) / 60) AS "Average Wait Time"
FROM Appointments
WHERE Appointments.appointmentdate IS NOT NULL;

```
SELECT AVG(EXTRACT(EPOCH FROM (Appointments.appointmentdate - Appointments.appointmenttime)) / 60) AS "Average Wait Time"
FROM Appointments
WHERE Appointments.appointmentdate IS NOT NULL
```

Average Wait Time
27953694.75

1 row (0.003 s) Edit, Explain, Export

20) Get the name and contact information of all the doctors who have prescribed medication to patients with blood group "AB+".

SELECT Doctors.Doctorname, Doctors.DoctorAddress, Doctors.DoctorPhoneNumber, Doctors.DoctorEmail
FROM Doctors
INNER JOIN Prescriptions ON Doctors.Doctorid = Prescriptions.Doctorid
INNER JOIN Patients ON Prescriptions.Patientid = Patients.Patientid
WHERE Patients.BloodGroup = 'AB+'

```
SELECT Doctors.Doctorname, Doctors.DoctorAddress, Doctors.DoctorPhoneNumber, Doctors.DoctorEmail
FROM Doctors
INNER JOIN Prescriptions ON Doctors.Doctorid = Prescriptions.Doctorid
INNER JOIN Patients ON Prescriptions.Patientid = Patients.Patientid
WHERE Patients.BloodGroup = 'AB+'
```

doctorname	doctoraddress	doctorphonenumber	doctoremail
Dr. Mark Thompson	91011 Maplewood Rd, Anytown, USA	555-2109	mark.thompson@email.com
Dr. Emily Brown	987 Cedar Rd, Anytown, USA	555-4321	emily.brown@email.com

2 rows (0.002 s) Edit, Explain, Export