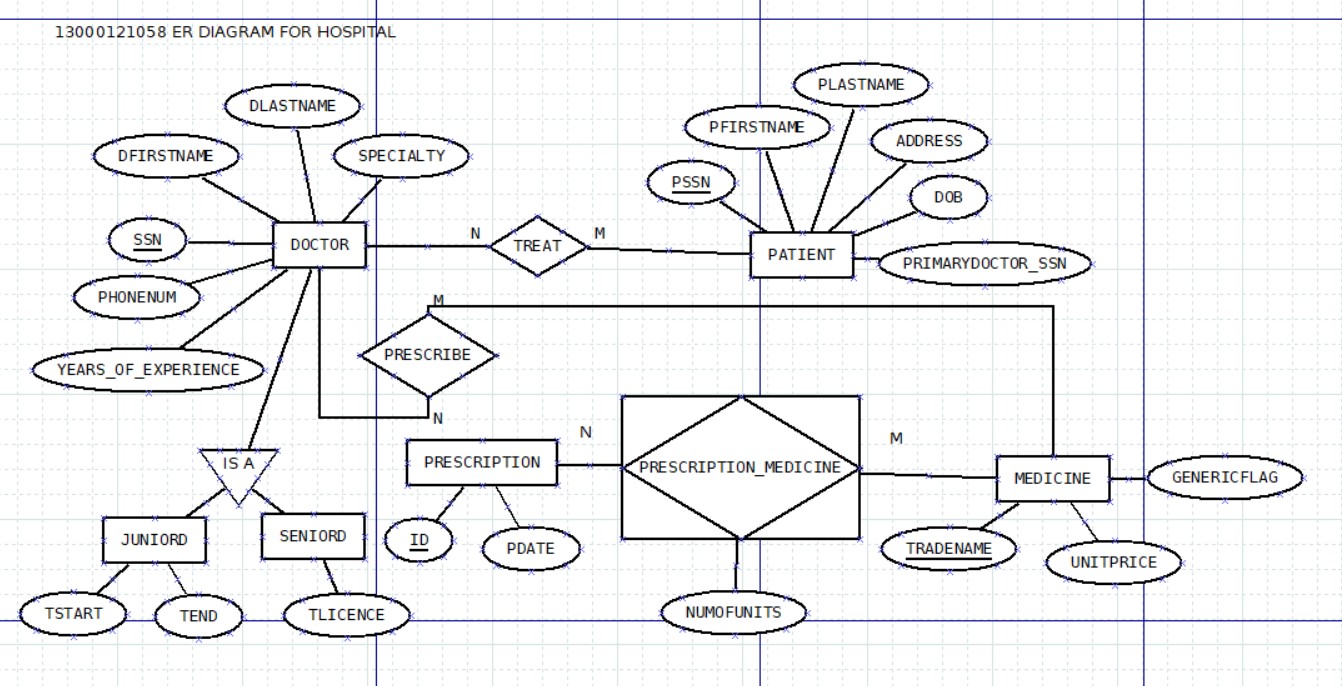
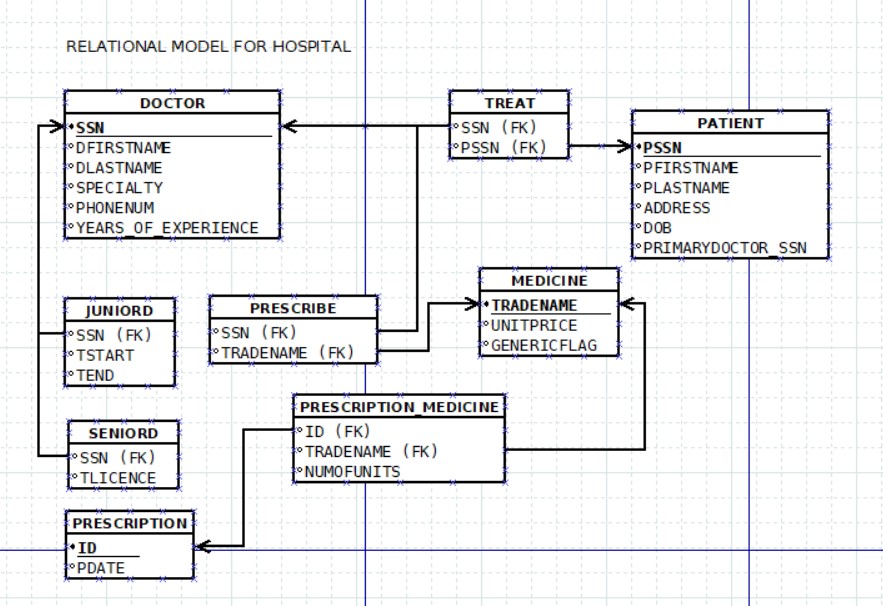
**ASSIGNMENT 7**

I.Design an ER diagram for an application that models a hospital doctors treat patients, prescribe tests, monitor progress etc. Analyse the requirements by identifying the entities, attributes, relationships, keys, constraints etc. Apply extended entity-relationship features to the design. Defend your design with proper assumptions and justifications. Map the ER model into a relational model.





ASSUMPTIONS

1. Doctors can treat multiple patients, and patients can be treated by multiple doctors (many-to-many relationship).

2. Doctors can prescribe multiple tests and medications for patients.

3. Each patient's progress can be monitored through multiple progress reports.

4. Doctors can specialize in different medical fields such as surgery, pediatrics, cardiology, etc.

5. Patients can be categorized as inpatients or outpatients.

6. Tests and medications share common attributes such as ID, name, and price.

7. Each progress report includes details such as description and date.

II.Create tables, populate with data and construct queries (advanced) in SQL to extract information from the hospital doctor’s database.

CREATE TABLE DOCTOR (

SSN VARCHAR2(10) PRIMARY KEY,

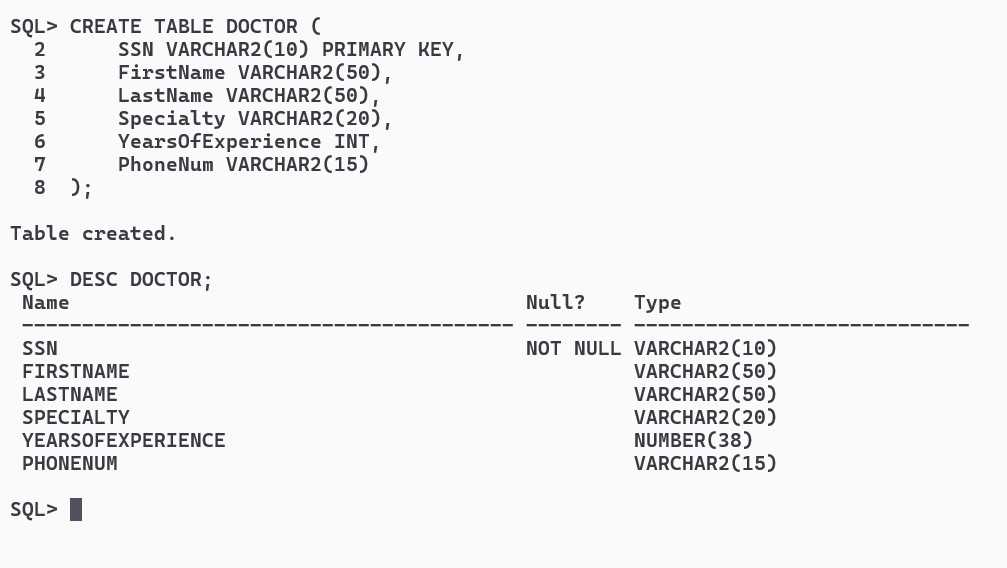
FirstName VARCHAR2(50),

LastName VARCHAR2(50),

Specialty VARCHAR2(20),

YearsOfExperience INT,

PhoneNum VARCHAR2(15));



CREATE TABLE PATIENT (

SSN VARCHAR2(10) PRIMARY KEY,

FirstName VARCHAR2(50),

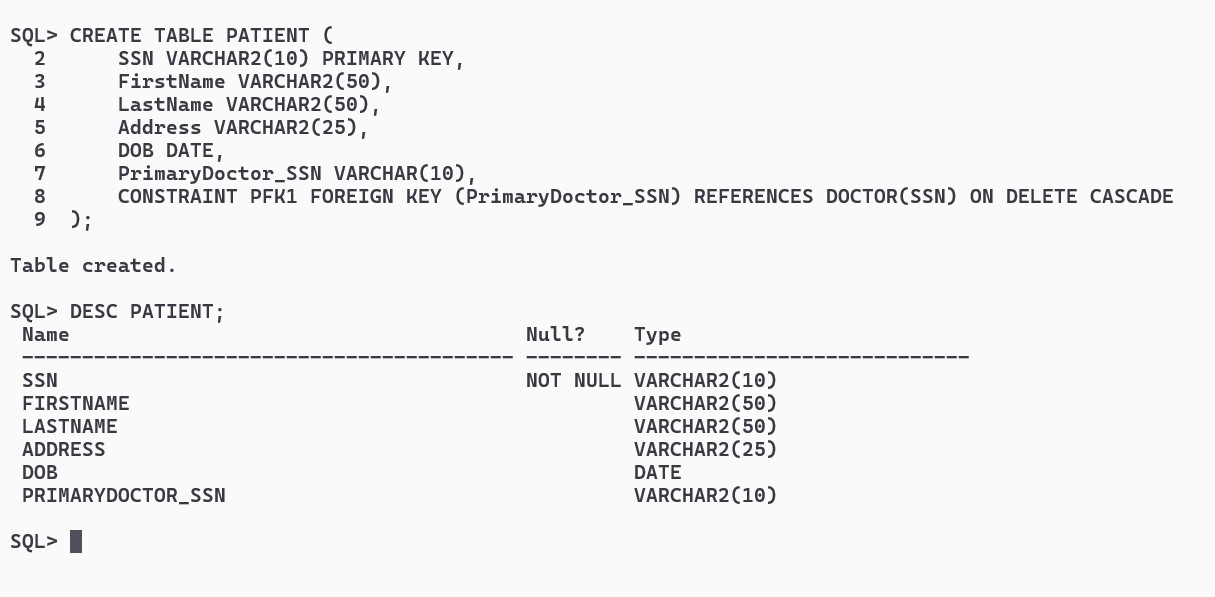
LastName VARCHAR2(50),

Address VARCHAR2(25),

DOB DATE,

PrimaryDoctor\_SSN VARCHAR(10),

CONSTRAINT PFK1 FOREIGN KEY (PrimaryDoctor\_SSN) REFERENCES DOCTOR(SSN) ON DELETE CASCADE);

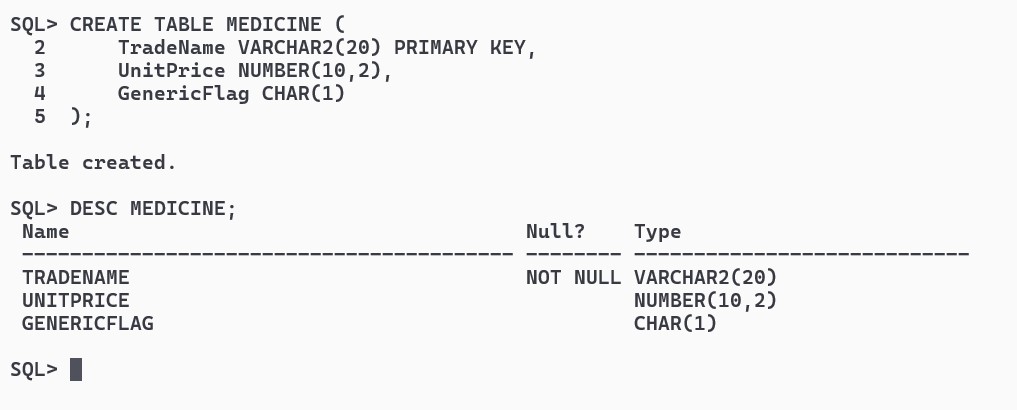


CREATE TABLE MEDICINE (

TradeName VARCHAR2(20) PRIMARY KEY,

UnitPrice NUMBER(10,2),

GenericFlag CHAR(1));



CREATE TABLE PRESCRIPTION (

Id VARCHAR2(10) PRIMARY KEY,

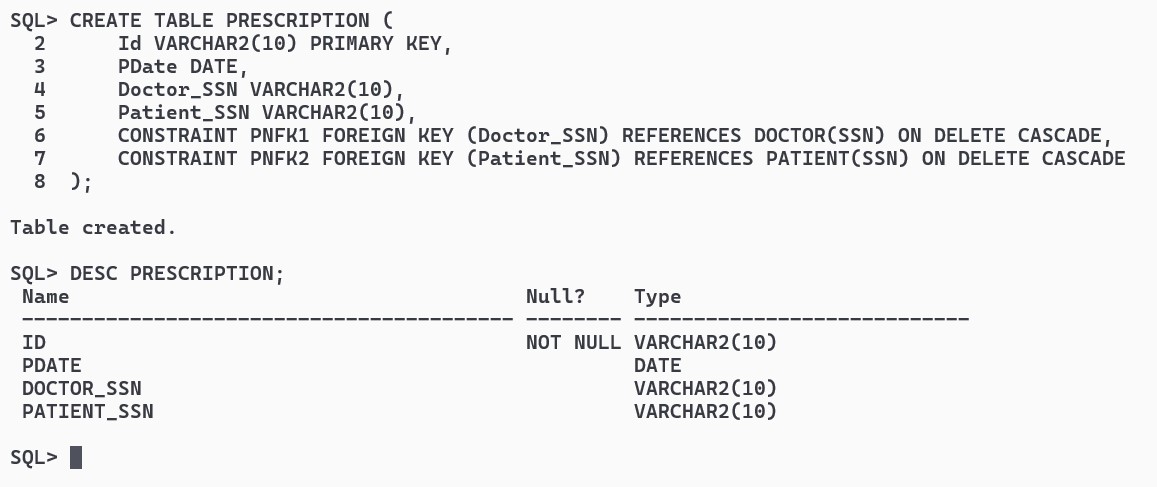
PDate DATE,

Doctor\_SSN VARCHAR2(10),

Patient\_SSN VARCHAR2(10),

CONSTRAINT PNFK1 FOREIGN KEY (Doctor\_SSN) REFERENCES DOCTOR(SSN) ON DELETE CASCADE,

CONSTRAINT PNFK2 FOREIGN KEY (Patient\_SSN) REFERENCES PATIENT(SSN) ON DELETE CASCADE);



CREATE TABLE Prescription\_Medicine (

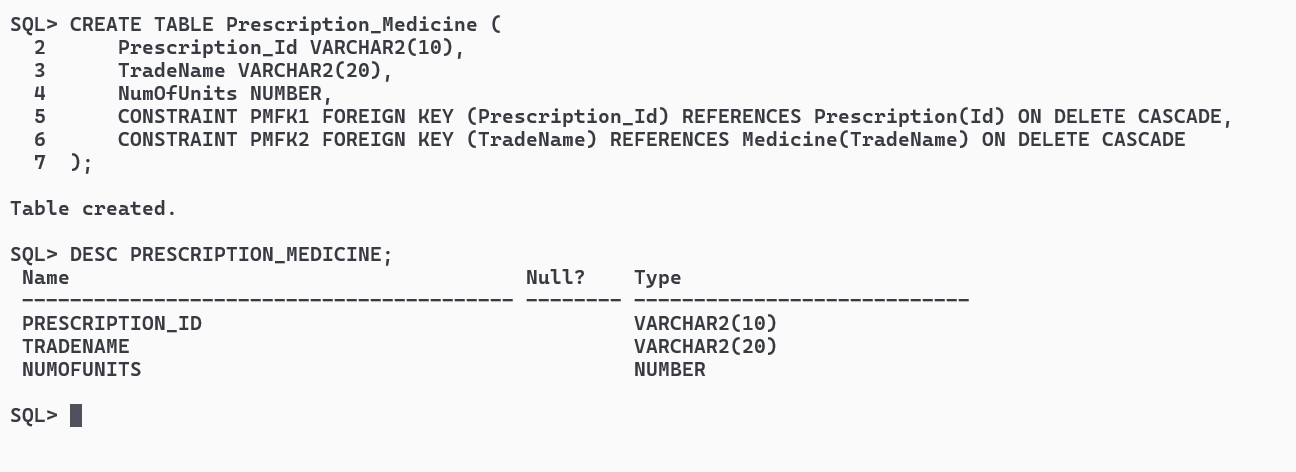
Prescription\_Id VARCHAR2(10),

TradeName VARCHAR2(20),

NumOfUnits NUMBER,

CONSTRAINT PMFK1 FOREIGN KEY (Prescription\_Id) REFERENCES Prescription(Id) ON DELETE CASCADE,

CONSTRAINT PMFK2 FOREIGN KEY (TradeName) REFERENCES Medicine(TradeName) ON DELETE CASCADE);



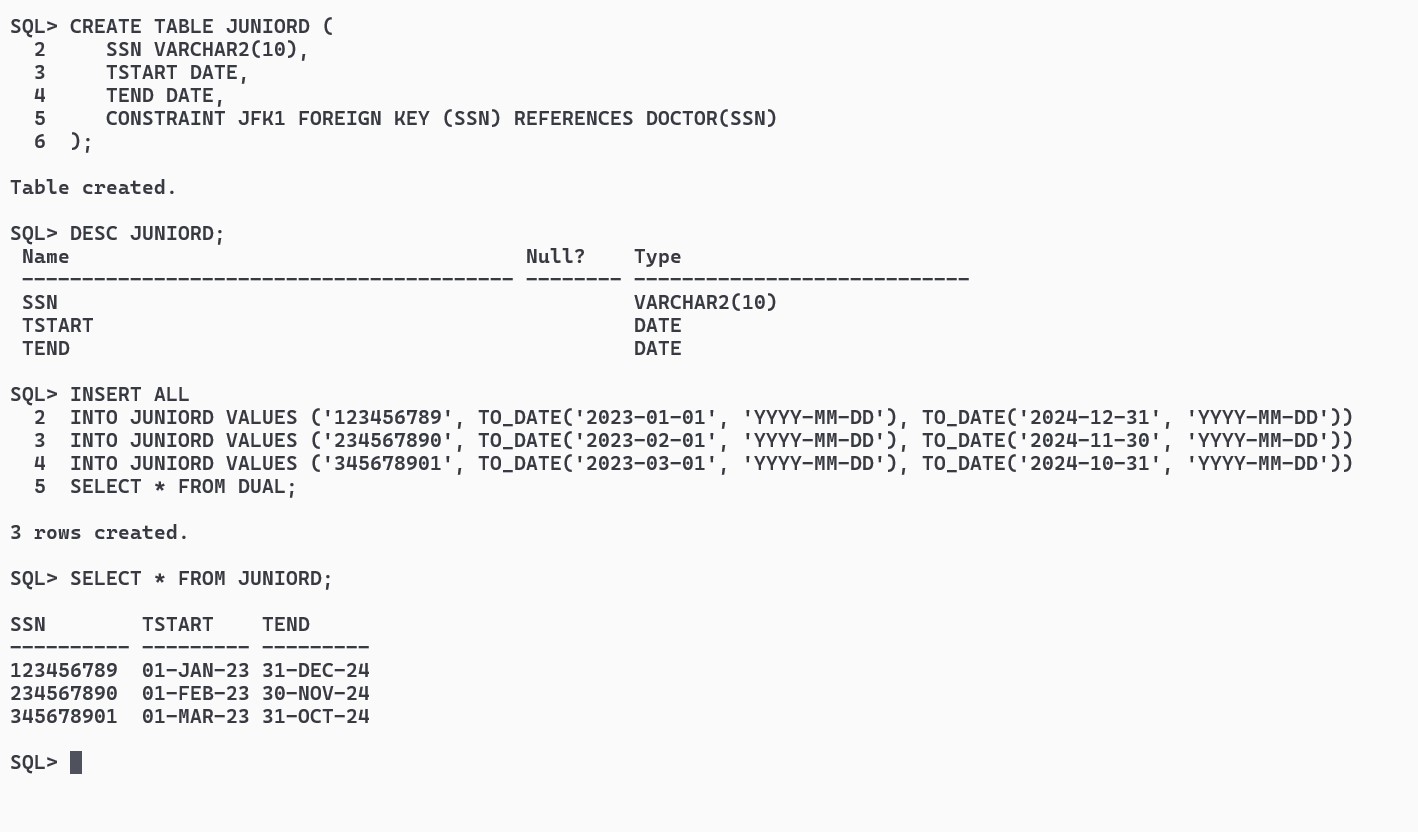
CREATE TABLE JUNIORD (

SSN VARCHAR2(10),

TSTART DATE,

TEND DATE,

CONSTRAINT JFK1 FOREIGN KEY (SSN) REFERENCES DOCTOR(SSN));

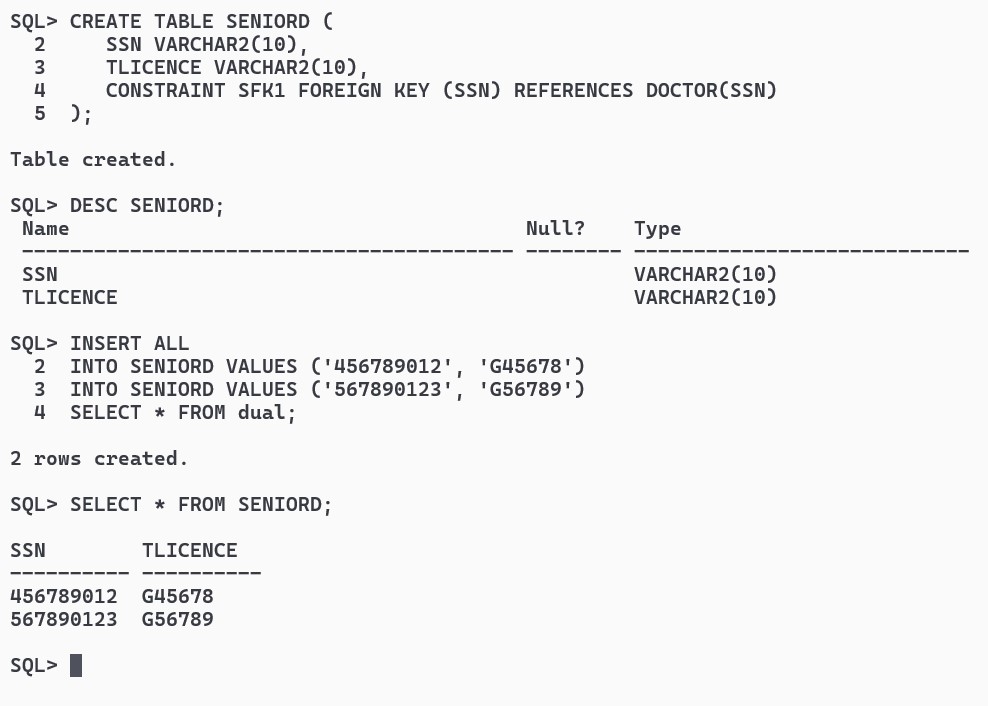


CREATE TABLE SENIORD (

SSN VARCHAR2(10),

TLICENCE VARCHAR2(10),

CONSTRAINT SFK1 FOREIGN KEY (SSN) REFERENCES DOCTOR(SSN));



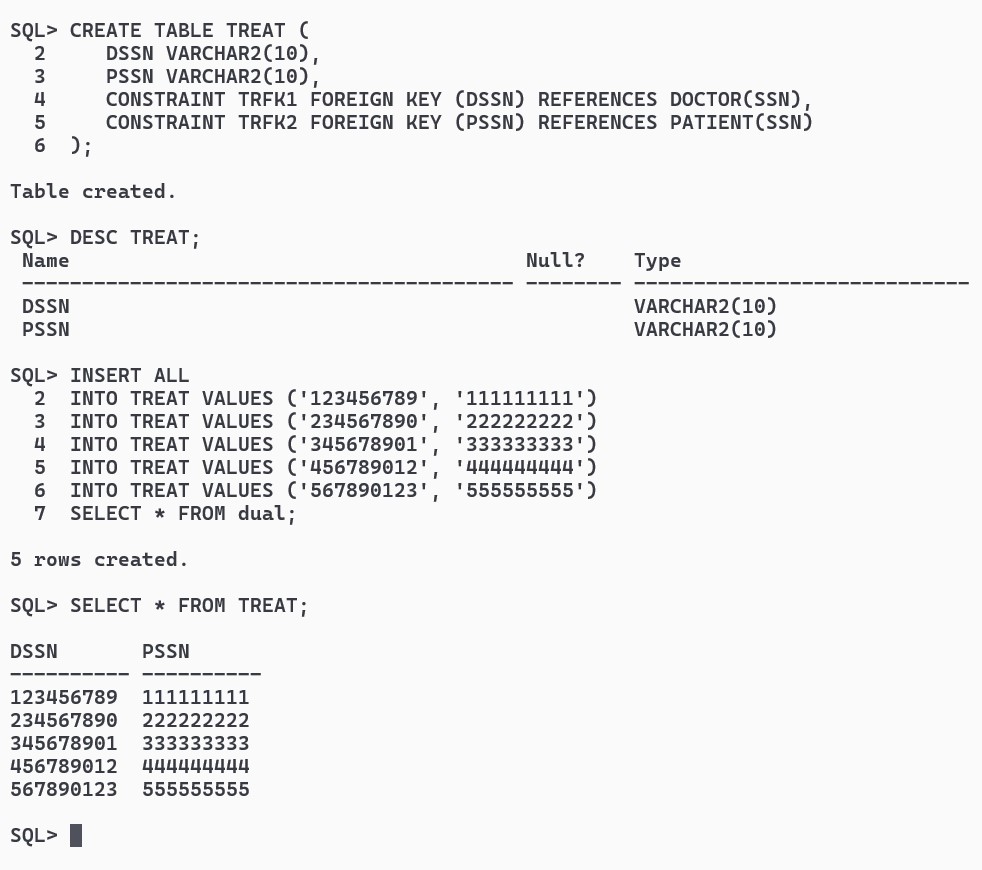
CREATE TABLE TREAT (

DSSN VARCHAR2(10),

PSSN VARCHAR2(10),

CONSTRAINT TRFK1 FOREIGN KEY (DSSN) REFERENCES DOCTOR(SSN),

CONSTRAINT TRFK2 FOREIGN KEY (PSSN) REFERENCES PATIENT(SSN));



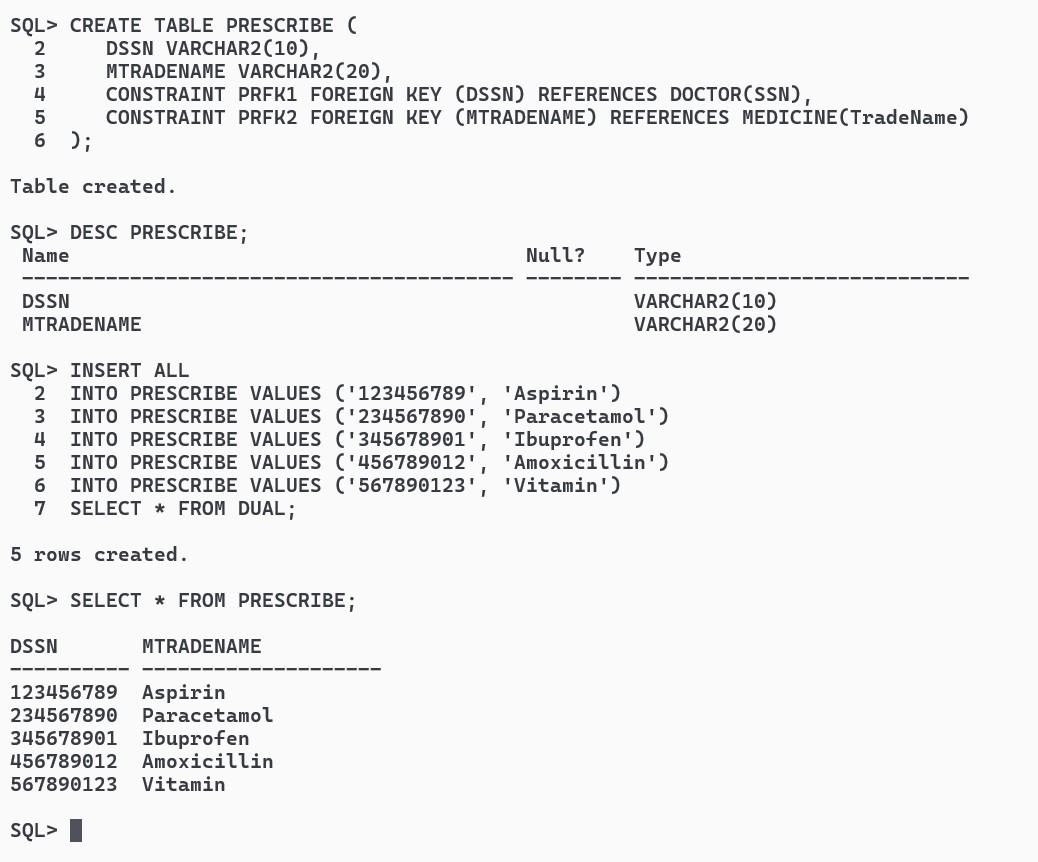
CREATE TABLE PRESCRIBE (

DSSN VARCHAR2(10),

MTRADENAME VARCHAR2(20),

CONSTRAINT PRFK1 FOREIGN KEY (DSSN) REFERENCES DOCTOR(SSN),

CONSTRAINT PRFK2 FOREIGN KEY (MTRADENAME) REFERENCES MEDICINE(TradeName));



INSERT ALL

INTO DOCTOR VALUES ('123456789', 'John', 'Smith', 'Cardiology', 15, '123-456-7890')

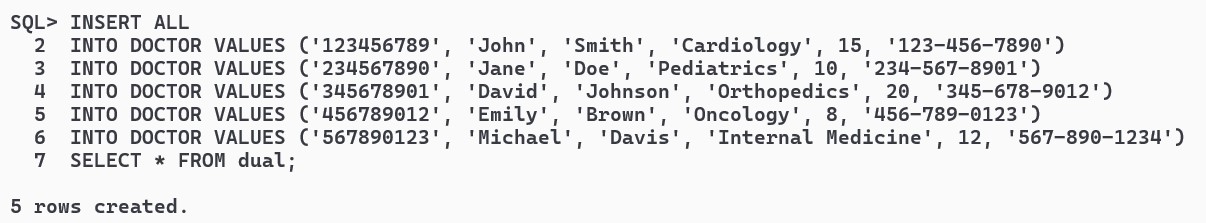
INTO DOCTOR VALUES ('234567890', 'Jane', 'Doe', 'Pediatrics', 10, '234-567-8901')

INTO DOCTOR VALUES ('345678901', 'David', 'Johnson', 'Orthopedics', 20, '345-678-9012')

INTO DOCTOR VALUES ('456789012', 'Emily', 'Brown', 'Oncology', 8, '456-789-0123')

INTO DOCTOR VALUES ('567890123', 'Michael', 'Davis', 'Internal Medicine', 12, '567-890-1234')

SELECT \* FROM dual;



INSERT ALL

INTO PATIENT VALUES ('111111111', 'Alice', 'Johnson', '123 Main St', TO\_DATE('1990-05-15', 'YYYY-MM-DD'), '123456789')

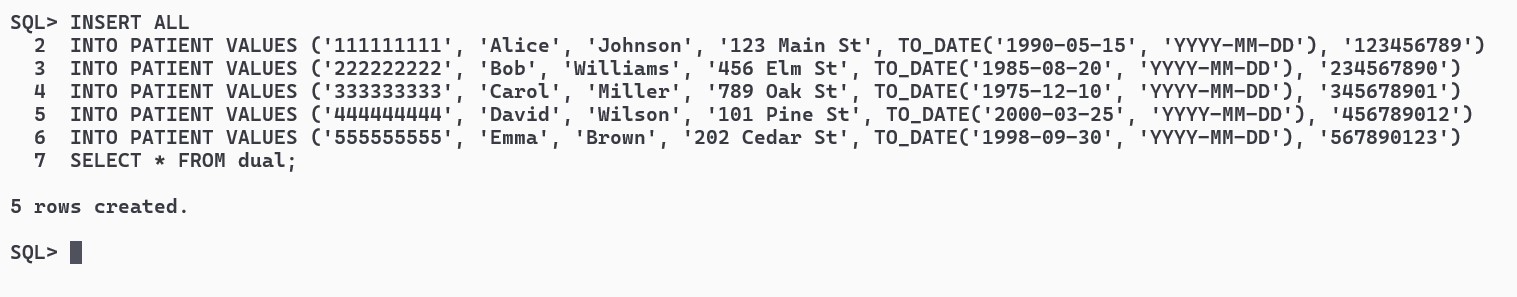
INTO PATIENT VALUES ('222222222', 'Bob', 'Williams', '456 Elm St', TO\_DATE('1985-08-20', 'YYYY-MM-DD'), '234567890')

INTO PATIENT VALUES ('333333333', 'Carol', 'Miller', '789 Oak St', TO\_DATE('1975-12-10', 'YYYY-MM-DD'), '345678901')

INTO PATIENT VALUES ('444444444', 'David', 'Wilson', '101 Pine St', TO\_DATE('2000-03-25', 'YYYY-MM-DD'), '456789012')

INTO PATIENT VALUES ('555555555', 'Emma', 'Brown', '202 Cedar St', TO\_DATE('1998-09-30', 'YYYY-MM-DD'), '567890123')

SELECT \* FROM dual;



INSERT ALL

INTO MEDICINE VALUES ('Aspirin', 10.00, 'Y')

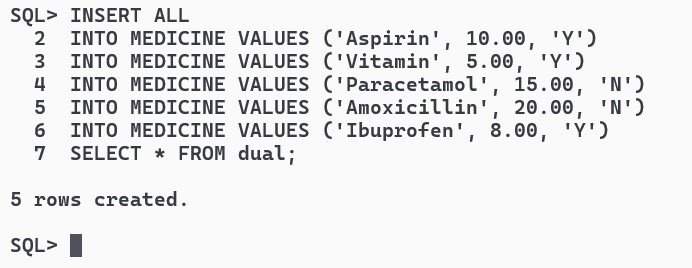
INTO MEDICINE VALUES ('Vitamin', 5.00, 'Y')

INTO MEDICINE VALUES ('Paracetamol', 15.00, 'N')

INTO MEDICINE VALUES ('Amoxicillin', 20.00, 'N')

INTO MEDICINE VALUES ('Ibuprofen', 8.00, 'Y')

SELECT \* FROM dual;



INSERT ALL

INTO PRESCRIPTION VALUES ('00001', TO\_DATE('2024-03-19', 'YYYY-MM-DD'), '123456789', '111111111')

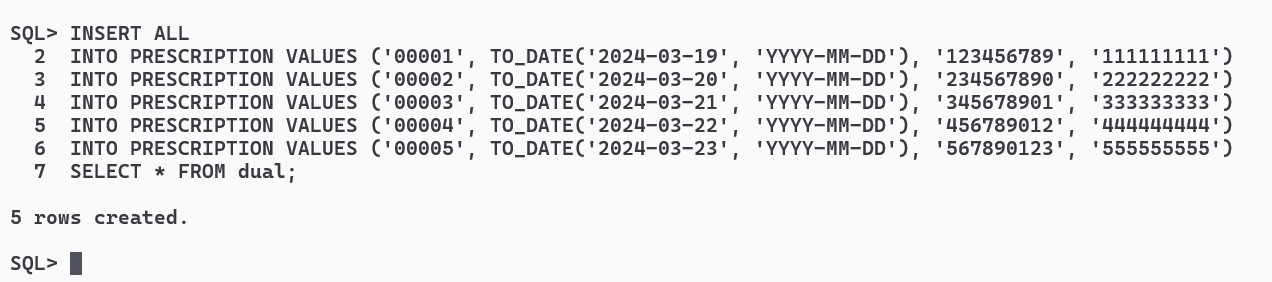
INTO PRESCRIPTION VALUES ('00002', TO\_DATE('2024-03-20', 'YYYY-MM-DD'), '234567890', '222222222')

INTO PRESCRIPTION VALUES ('00003', TO\_DATE('2024-03-21', 'YYYY-MM-DD'), '345678901', '333333333')

INTO PRESCRIPTION VALUES ('00004', TO\_DATE('2024-03-22', 'YYYY-MM-DD'), '456789012', '444444444')

INTO PRESCRIPTION VALUES ('00005', TO\_DATE('2024-03-23', 'YYYY-MM-DD'), '567890123', '555555555')

SELECT \* FROM dual;



INSERT ALL

INTO Prescription\_Medicine VALUES ('00001', 'Aspirin', 2)

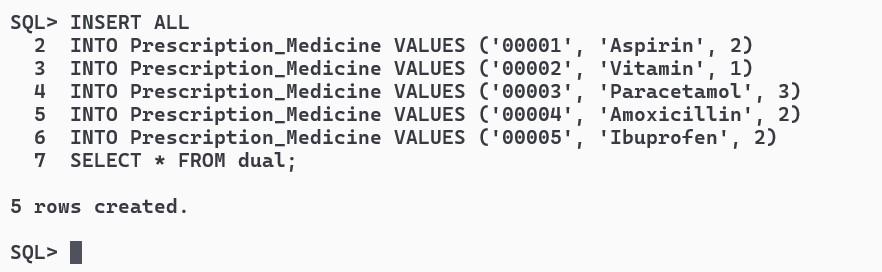
INTO Prescription\_Medicine VALUES ('00002', 'Vitamin', 1)

INTO Prescription\_Medicine VALUES ('00003', 'Paracetamol', 3)

INTO Prescription\_Medicine VALUES ('00004', 'Amoxicillin', 2)

INTO Prescription\_Medicine VALUES ('00005', 'Ibuprofen', 2)

SELECT \* FROM dual;



INSERT ALL

INTO JUNIORD VALUES ('123456789', TO\_DATE('2023-01-01', 'YYYY-MM-DD'), TO\_DATE('2024-12-31', 'YYYY-MM-DD'))

INTO JUNIORD VALUES ('234567890', TO\_DATE('2023-02-01', 'YYYY-MM-DD'), TO\_DATE('2024-11-30', 'YYYY-MM-DD'))

INTO JUNIORD VALUES ('345678901', TO\_DATE('2023-03-01', 'YYYY-MM-DD'), TO\_DATE('2024-10-31', 'YYYY-MM-DD'))

SELECT \* FROM DUAL;

INSERT ALL

INTO SENIORD VALUES ('456789012', 'G45678')

INTO SENIORD VALUES ('567890123', 'G56789')

SELECT \* FROM dual;

INSERT ALL

INTO TREAT VALUES ('123456789', '111111111')

INTO TREAT VALUES ('234567890', '222222222')

INTO TREAT VALUES ('345678901', '333333333')

INTO TREAT VALUES ('456789012', '444444444')

INTO TREAT VALUES ('567890123', '555555555')

SELECT \* FROM DUAL;

INSERT ALL

INTO PRESCRIBE VALUES ('123456789', 'Aspirin')

INTO PRESCRIBE VALUES ('234567890', 'Paracetamol')

INTO PRESCRIBE VALUES ('345678901', 'Ibuprofen')

INTO PRESCRIBE VALUES ('456789012', 'Amoxicillin')

INTO PRESCRIBE VALUES ('567890123', 'Vitamin')

SELECT \* FROM DUAL;

III.Consider the following relations run the following SQL queries :

Doctor(SSN, FirstName, LastName, Specialty, YearsOfExperience, PhoneNum)

Patient(SSN, FirstName, LastName, Address, DOB, PrimaryDoctor\_SSN)

Medicine(TradeName, UnitPrice, GenericFlag)

Prescription(Id, Date, Doctor\_SSN, Patient\_SSN)

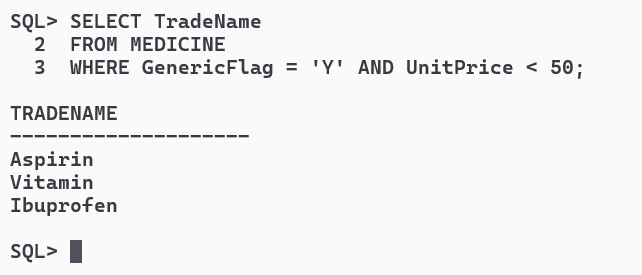
Prescription\_Medicine(Prescription Id, TradeName, NumOfUnits)

1.List the trade name of generic medicine with unit price less than $50.

SELECT TradeName

FROM MEDICINE

WHERE GenericFlag = 'Y' AND UnitPrice < 50;



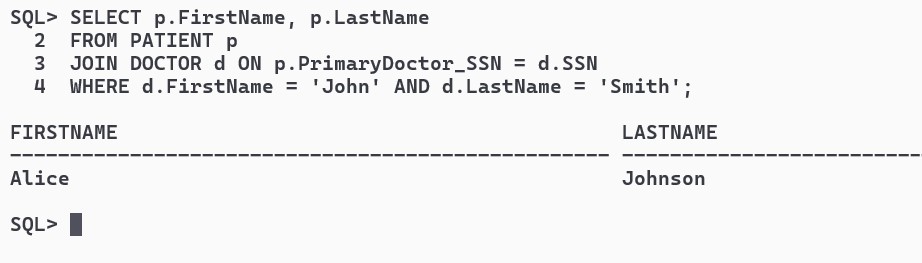
2.List the first and last name of patients whose primary doctor named ʻJohn Smithʼ.

SELECT p.FirstName, p.LastName

FROM PATIENT p

JOIN DOCTOR d ON p.PrimaryDoctor\_SSN = d.SSN

WHERE d.FirstName = 'John' AND d.LastName = 'Smith';

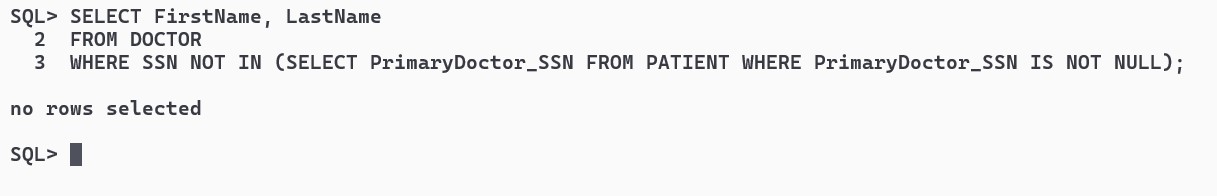


3.List the first and last name of doctors who are not primary doctors to any patient.

SELECT FirstName, LastName

FROM DOCTOR

WHERE SSN NOT IN (SELECT PrimaryDoctor\_SSN FROM PATIENT WHERE PrimaryDoctor\_SSN IS NOT NULL);



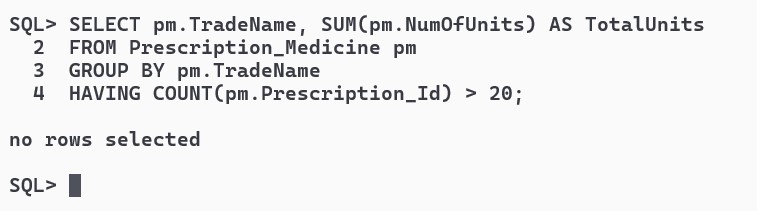
4.For medicines written in more than 20 prescriptions, report the trade name and the total number of units prescribed.

SELECT pm.TradeName, SUM(pm.NumOfUnits) AS TotalUnits

FROM Prescription\_Medicine pm

GROUP BY pm.TradeName

HAVING COUNT(pm.Prescription\_Id) > 20;



5.List the SSN of patients who have ʻAspirinʼ and ʻVitaminʼ trade names in one prescription.

SELECT p.SSN

FROM PATIENT p

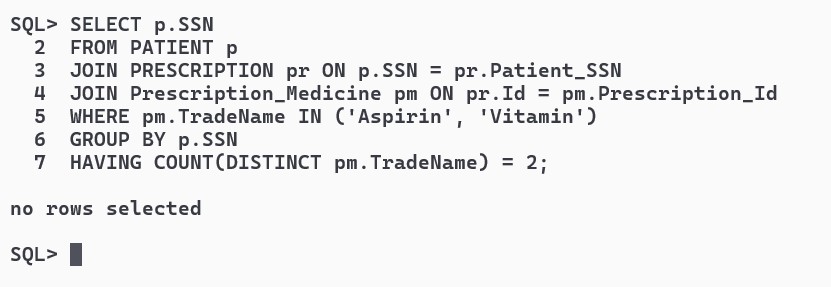
JOIN PRESCRIPTION pr ON p.SSN = pr.Patient\_SSN

JOIN Prescription\_Medicine pm ON pr.Id = pm.Prescription\_Id

WHERE pm.TradeName IN ('Aspirin', 'Vitamin')

GROUP BY p.SSN

HAVING COUNT(DISTINCT pm.TradeName) = 2;



6.List the SNN of distinct patients who have ʻAspirinʼ prescribed to them by doctor named ʻJohn Smithʼ.

SELECT DISTINCT p.SSN

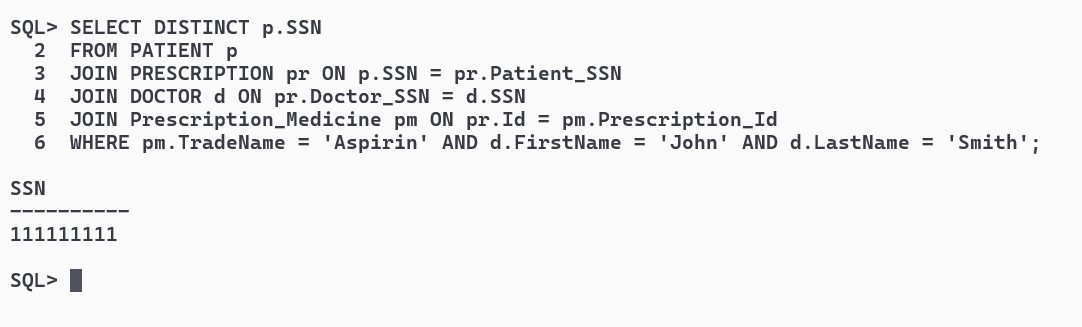
FROM PATIENT p

JOIN PRESCRIPTION pr ON p.SSN = pr.Patient\_SSN

JOIN DOCTOR d ON pr.Doctor\_SSN = d.SSN

JOIN Prescription\_Medicine pm ON pr.Id = pm.Prescription\_Id

WHERE pm.TradeName = 'Aspirin' AND d.FirstName = 'John' AND d.LastName = 'Smith';



7.List the first and last name of patients who have no prescriptions written by doctors other than their primary doctors.

SELECT p.FirstName, p.LastName FROM PATIENT p WHERE NOT EXISTS (

SELECT \* FROM PRESCRIPTION pr JOIN DOCTOR d ON pr.Doctor\_SSN = d.SSN

WHERE pr.Patient\_SSN = p.SSN AND pr.Doctor\_SSN <> p.PrimaryDoctor\_SSN);

