Mason Wester

Health Care Database

BSA URGENT CARE

Database Systems
3/30/2024
(CIDM-3350-70)

Introduction:

In the realm of healthcare, data management is pivotal for efficient operations, accurate diagnoses, and patient care. However, healthcare data management often grapples with numerous challenges, ranging from data security breaches to interoperability issues among systems. With the exponential growth of medical data, the need for large database solutions has become more pronounced. Our database system aims to address these challenges by offering superior data integrity, reduced redundancy, and enhanced accessibility, ultimately revolutionizing healthcare data management.

Data Management Problems:

Mismanagement of healthcare data poses significant risks, including compromised patient privacy, increased potential for medical errors, and hindrance to research progress. One of the major challenges is interoperability, where healthcare facilities struggle due to incompatible systems, leading to obstacles in the smooth exchange of patient information. This problem not only impacts the continuity of care but also impedes efforts to establish a comprehensive view of patients' health histories. Redundant data entries further exacerbate inefficiencies, cluttering databases and complicating access to critical information for healthcare professionals. Additionally, cybersecurity threats pose a grave concern, with hackers targeting healthcare databases to exploit sensitive patient information for various nefarious purposes, ranging from financial fraud to identity theft.

Addressing these issues demands an approach that prioritizes data integrity, streamlining processes, and fortifying cybersecurity measures. By fostering interoperability through standardized protocols and robust data exchange mechanisms, healthcare systems can seamlessly communicate and share patient information across diverse platforms. Implementing advanced data management techniques, such as data normalization, can significantly reduce redundancy, optimizing storage space and simplifying data retrieval processes. Moreover, robust cybersecurity protocols, including encryption and access controls, are essential for safeguarding patient data from malicious cyber threats, ensuring confidentiality and trust in healthcare systems. Through concerted efforts to address these challenges, healthcare organizations can enhance efficiency, improve patient care, and safeguard the integrity of sensitive healthcare data.

Motivation:

Our database solution is specifically designed to confront the complex issues surrounding healthcare data management. We prioritize ensuring that healthcare information is consistently accurate and reliable. By using advanced techniques to organize data and prevent duplication, we make sure that storage space is used efficiently and that accessing patient records is quick and easy. Our system is built to allow different healthcare systems to communicate and share information securely, overcoming the barriers that often exist between them.

These features directly tackle the main problems faced in healthcare data management. By focusing on maintaining data accuracy, reducing redundancy, and promoting interoperability, our solution not only makes healthcare organizations more efficient but also improves patient care.

Potential Benefits:

The adoption of our database system within healthcare settings promises a multitude of advantages. By prioritizing the maintenance of data integrity and the reduction of redundancy, healthcare facilities can significantly mitigate the risk of medical errors while ensuring that clinicians have access to accurate patient information when making crucial decisions. This not only enhances the quality of care but also fosters a safer healthcare environment for patients. Furthermore, the enhanced interoperability by our system serves as a vital step for seamless communication among healthcare systems. This interoperability not only streamlines administrative processes but also fosters collaboration between healthcare professionals, ultimately leading to improved patient outcomes through more coordinated and comprehensive care delivery.

In addition to promoting data integrity and interoperability, our database system fortifies cybersecurity measures to safeguard sensitive patient data from unauthorized access and malicious cyber threats. By implementing robust encryption protocols, access controls, and regular security audits, healthcare organizations can bolster patient privacy and instill confidence in the security of their data. This not only aligns with regulatory compliance requirements but also strengthens the trust between patients and healthcare providers, ultimately enhancing the overall reputation and credibility of healthcare organizations. Overall, the adoption of our comprehensive database solution not only enhances operational efficiencies within healthcare settings but also contributes to a safer, more collaborative, and patient-centric healthcare ecosystem.

Potential Users:

Future users of our database system include nurses, administrators, doctors, healthcare providers, medical researchers, insurance companies, and government agencies involved in healthcare regulation and policy. These users stand to benefit significantly from the improved data management capabilities offered by our system.

Business Rules

BSA Urgent Care staff will have the attributes of staff_ID (identifier), name (first-last), address, and phone number.

The staff entity will have the subtypes of Administration, Nurse, or Doctor. Staff can only be one of these subtypes. Staff will only have one manager for all staff.

Each Doctor will need their department, qualifications, and they Doctor ID (doc_ID) (identifier). Nurses will their Nurse Id (nurse_ID) (identifier), department, and hours worked.

Administration will be working with patient registration, and patient payments. When they enter the administration department, they are assigned an admin_NO (identifier). An administrator will only work in one of these subtypes. Registration will include the Patient_ID (identifier), date, and time admitted.

When administration registers a patient at BSA, they will be assigned a room number and their patient_ID.

When Administration is billing, they will have the Billing_Code (identifier). They will also include the Insurance_ID (identifier) and the Admin_NO to keep track of which employees filed the registration of the patients.

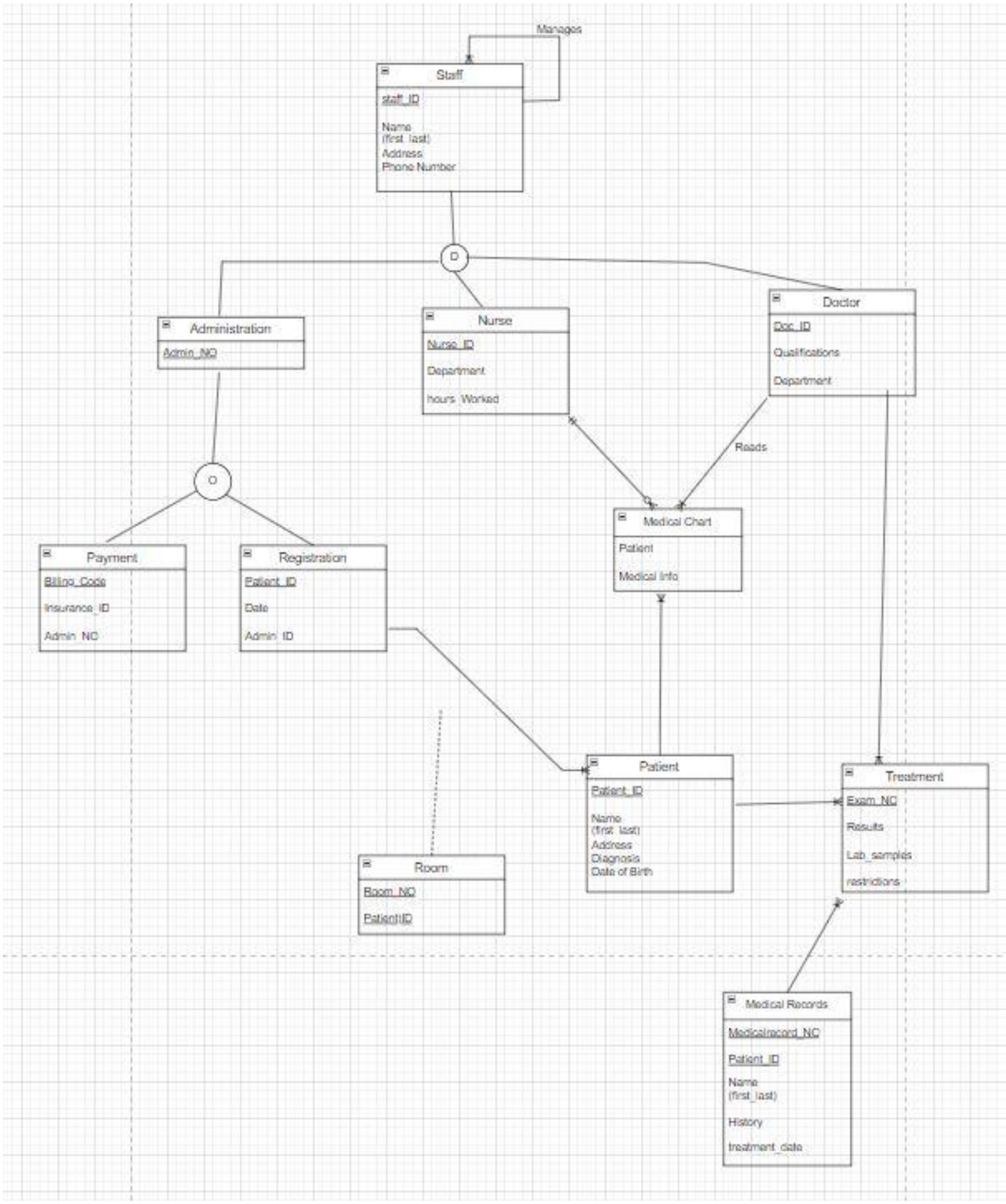
Patients will have an (identifier) of Patient_ID, Name (first_last), date of birth, address, and diagnosis.

Nurses will be caring for multiple patients, but each patient will only have one nurse. When caring for the patients, the nurses must enter the medical information and the patient ID to maintain proper care and follow HIPPA guidelines.

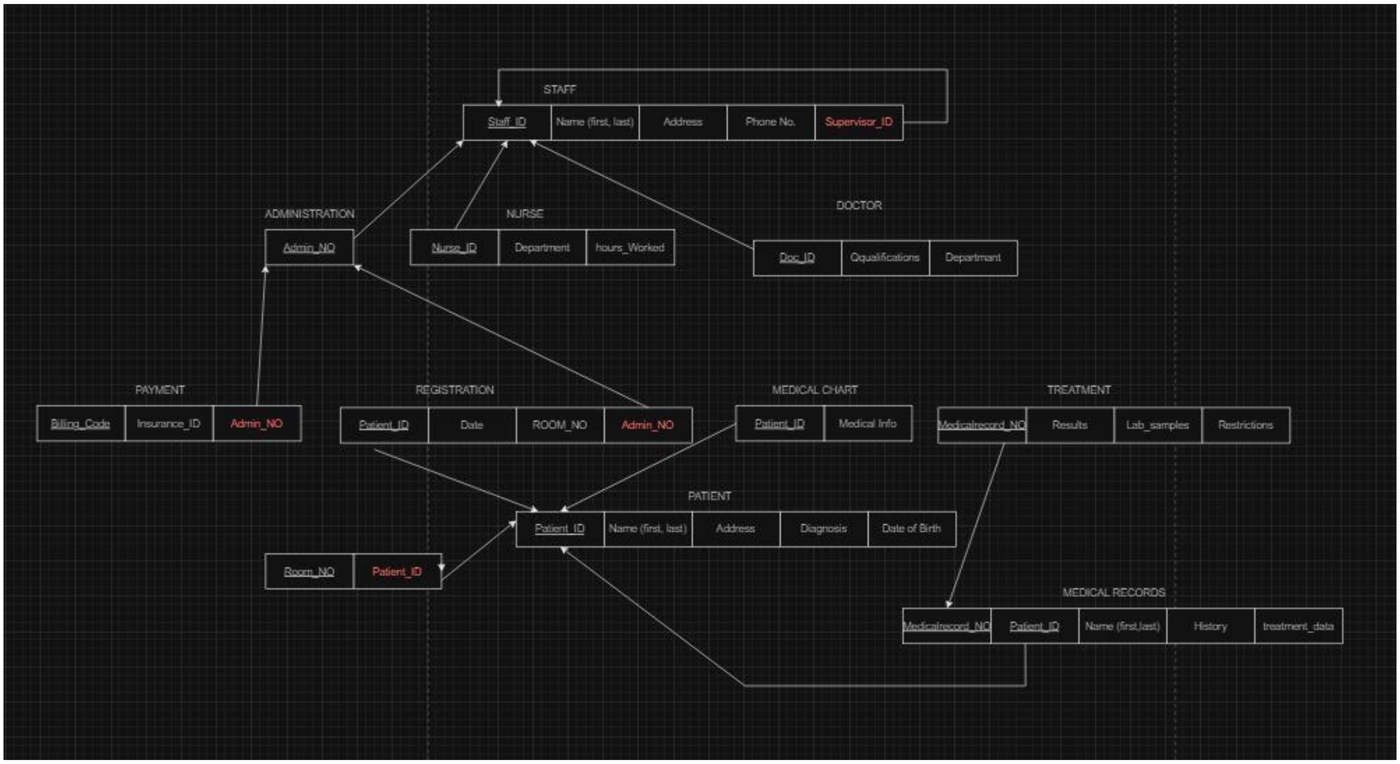
Doctors will also be caring for multiple patients while patients will only be assigned one Doctor. Doctors will enter the patient’s exam_NO, results, lab_samples, and will update the patient restrictions.

When a patient is treated by the Doctor, their medical records will be updated. They will include the Medicalrecord_NO (identifier), patient_ID, patient name, history, and treatment_Date.

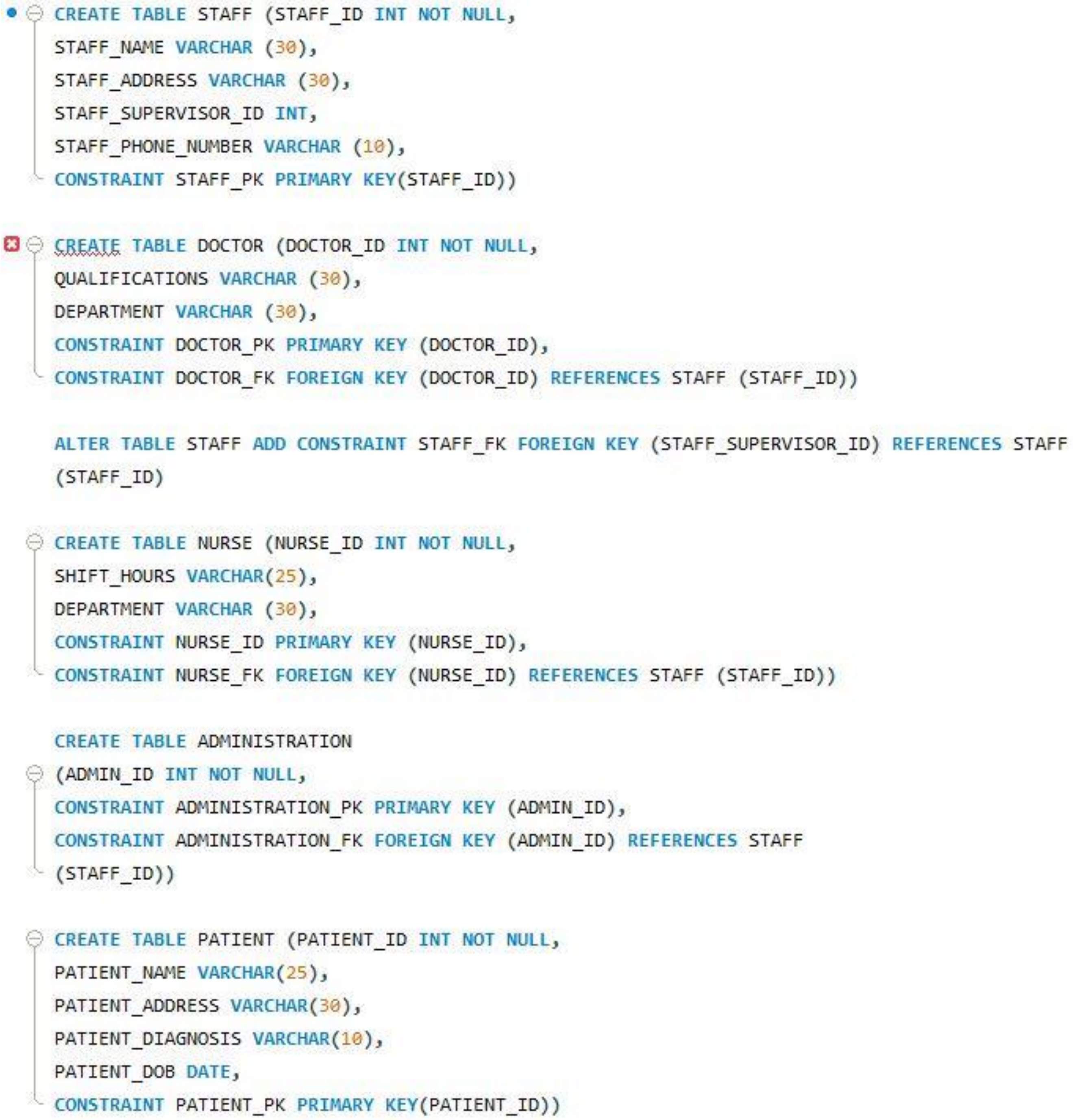
Entity Relationship Diagram



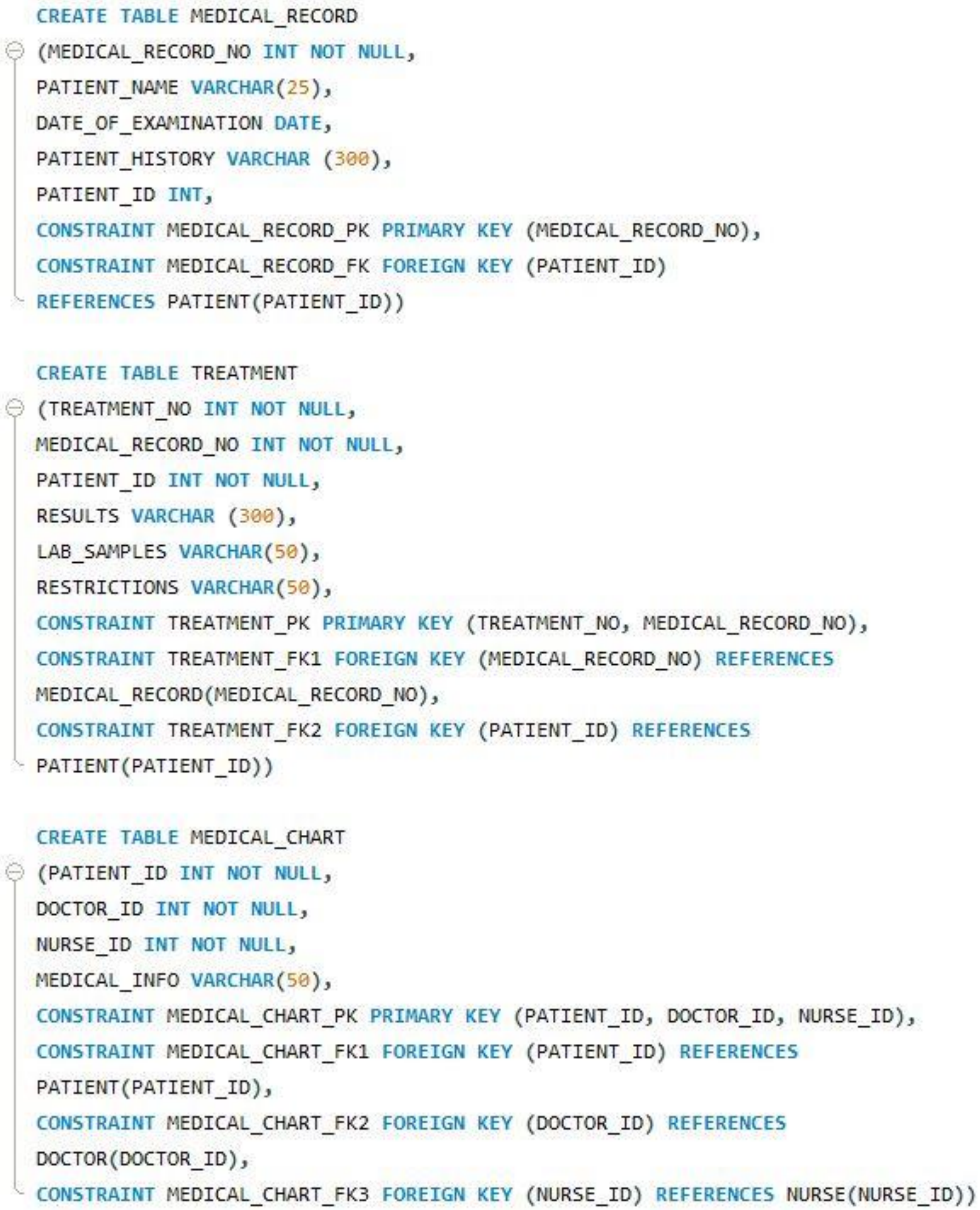
3RD Nominal Form



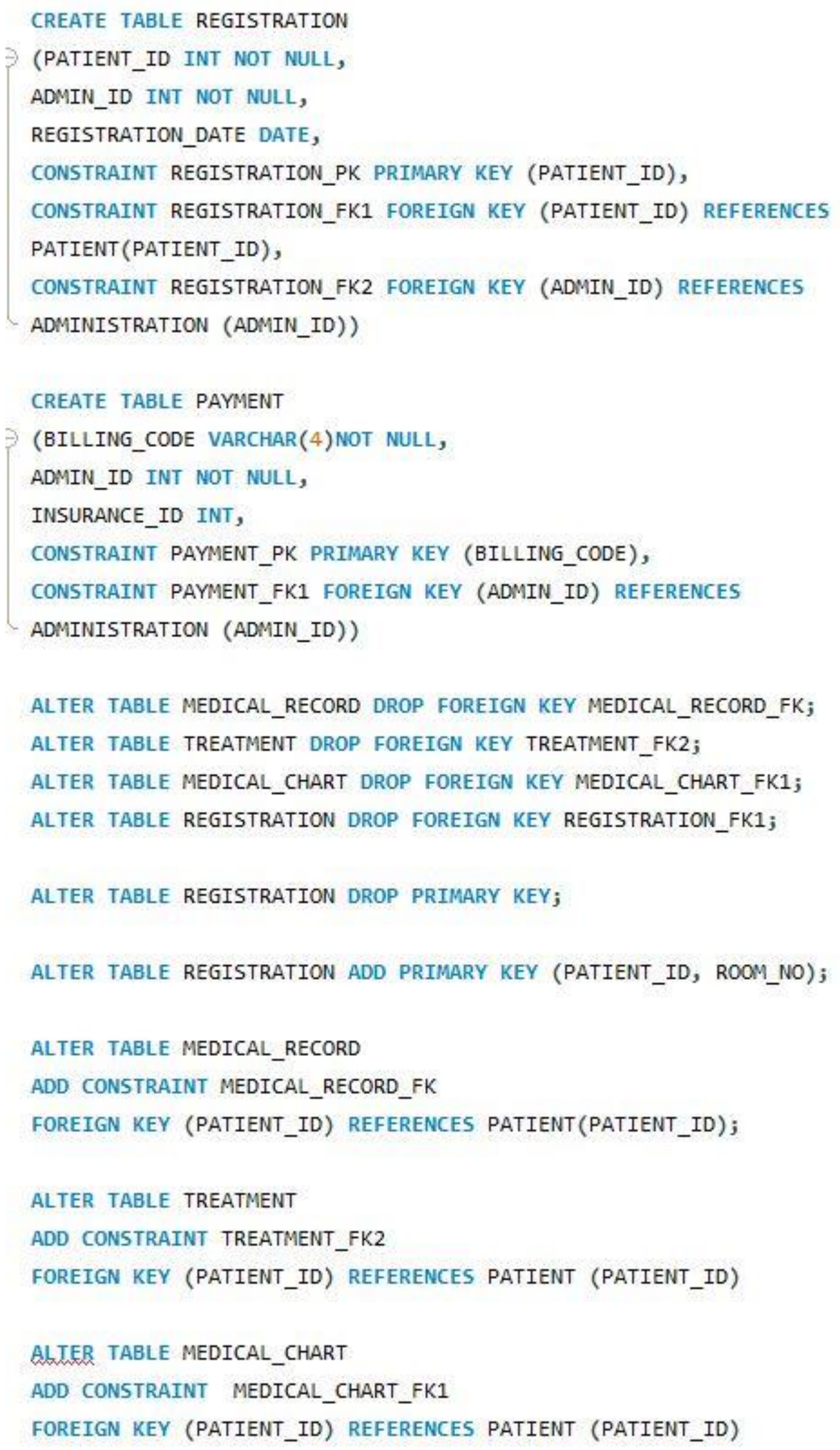
SQL Tables



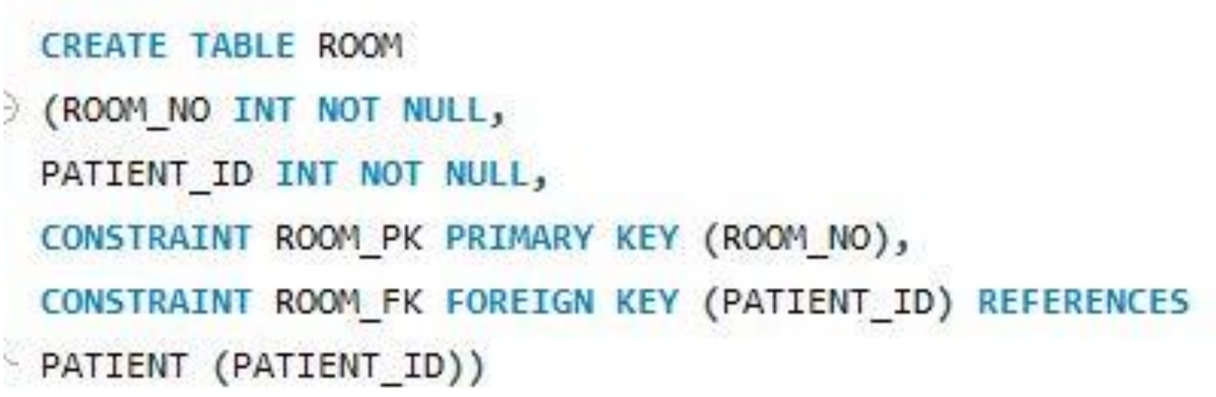
```
CREATE TABLE STAFF (STAFF_ID INT NOT NULL,
STAFF_NAME VARCHAR (30),
STAFF_ADDRESS VARCHAR (30),
STAFF_SUPERVISOR_ID INT,
STAFF_PHONE_NUMBER VARCHAR (10),
CONSTRAINT STAFF_PK PRIMARY KEY(STAFF_ID))

CREATE TABLE DOCTOR (DOCTOR_ID INT NOT NULL,
QUALIFICATIONS VARCHAR (30),
DEPARTMENT VARCHAR (30),
CONSTRAINT DOCTOR_PK PRIMARY KEY (DOCTOR_ID),
CONSTRAINT DOCTOR_FK FOREIGN KEY (DOCTOR_ID) REFERENCES STAFF (STAFF_ID))

ALTER TABLE STAFF ADD CONSTRAINT STAFF_FK FOREIGN KEY (STAFF_SUPERVISOR_ID) REFERENCES STAFF
(STAFF_ID)

CREATE TABLE NURSE (NURSE_ID INT NOT NULL,
SHIFT_HOURS VARCHAR(25),
DEPARTMENT VARCHAR (30),
CONSTRAINT NURSE_ID PRIMARY KEY (NURSE_ID),
CONSTRAINT NURSE_FK FOREIGN KEY (NURSE_ID) REFERENCES STAFF (STAFF_ID))

CREATE TABLE ADMINISTRATION
(ADMIN_ID INT NOT NULL,
CONSTRAINT ADMINISTRATION_PK PRIMARY KEY (ADMIN_ID),
CONSTRAINT ADMINISTRATION_FK FOREIGN KEY (ADMIN_ID) REFERENCES STAFF
(STAFF_ID))

CREATE TABLE PATIENT (PATIENT_ID INT NOT NULL,
PATIENT_NAME VARCHAR(25),
PATIENT_ADDRESS VARCHAR(30),
PATIENT_DIAGNOSIS VARCHAR(10),
PATIENT_DOB DATE,
CONSTRAINT PATIENT_PK PRIMARY KEY(PATIENT_ID))
```



```sql
CREATE TABLE MEDICAL_RECORD
(MEDICAL_RECORD_NO INT NOT NULL,
PATIENT_NAME VARCHAR(25),
DATE_OF_EXAMINATION DATE,
PATIENT_HISTORY VARCHAR (300),
PATIENT_ID INT,
CONSTRAINT MEDICAL_RECORD_PK PRIMARY KEY (MEDICAL_RECORD_NO),
CONSTRAINT MEDICAL_RECORD_FK FOREIGN KEY (PATIENT_ID)
REFERENCES PATIENT(PATIENT_ID))

CREATE TABLE TREATMENT
(TREATMENT_NO INT NOT NULL,
MEDICAL_RECORD_NO INT NOT NULL,
PATIENT_ID INT NOT NULL,
RESULTS VARCHAR (300),
LAB_SAMPLES VARCHAR(50),
RESTRICTIONS VARCHAR(50),
CONSTRAINT TREATMENT_PK PRIMARY KEY (TREATMENT_NO, MEDICAL_RECORD_NO),
CONSTRAINT TREATMENT_FK1 FOREIGN KEY (MEDICAL_RECORD_NO) REFERENCES
MEDICAL_RECORD(MEDICAL_RECORD_NO),
CONSTRAINT TREATMENT_FK2 FOREIGN KEY (PATIENT_ID) REFERENCES
PATIENT(PATIENT_ID))

CREATE TABLE MEDICAL_CHART
(PATIENT_ID INT NOT NULL,
DOCTOR_ID INT NOT NULL,
NURSE_ID INT NOT NULL,
MEDICAL_INFO VARCHAR(50),
CONSTRAINT MEDICAL_CHART_PK PRIMARY KEY (PATIENT_ID, DOCTOR_ID, NURSE_ID),
CONSTRAINT MEDICAL_CHART_FK1 FOREIGN KEY (PATIENT_ID) REFERENCES
PATIENT(PATIENT_ID),
CONSTRAINT MEDICAL_CHART_FK2 FOREIGN KEY (DOCTOR_ID) REFERENCES
DOCTOR(DOCTOR_ID),
CONSTRAINT MEDICAL_CHART_FK3 FOREIGN KEY (NURSE_ID) REFERENCES NURSE(NURSE_ID))
```

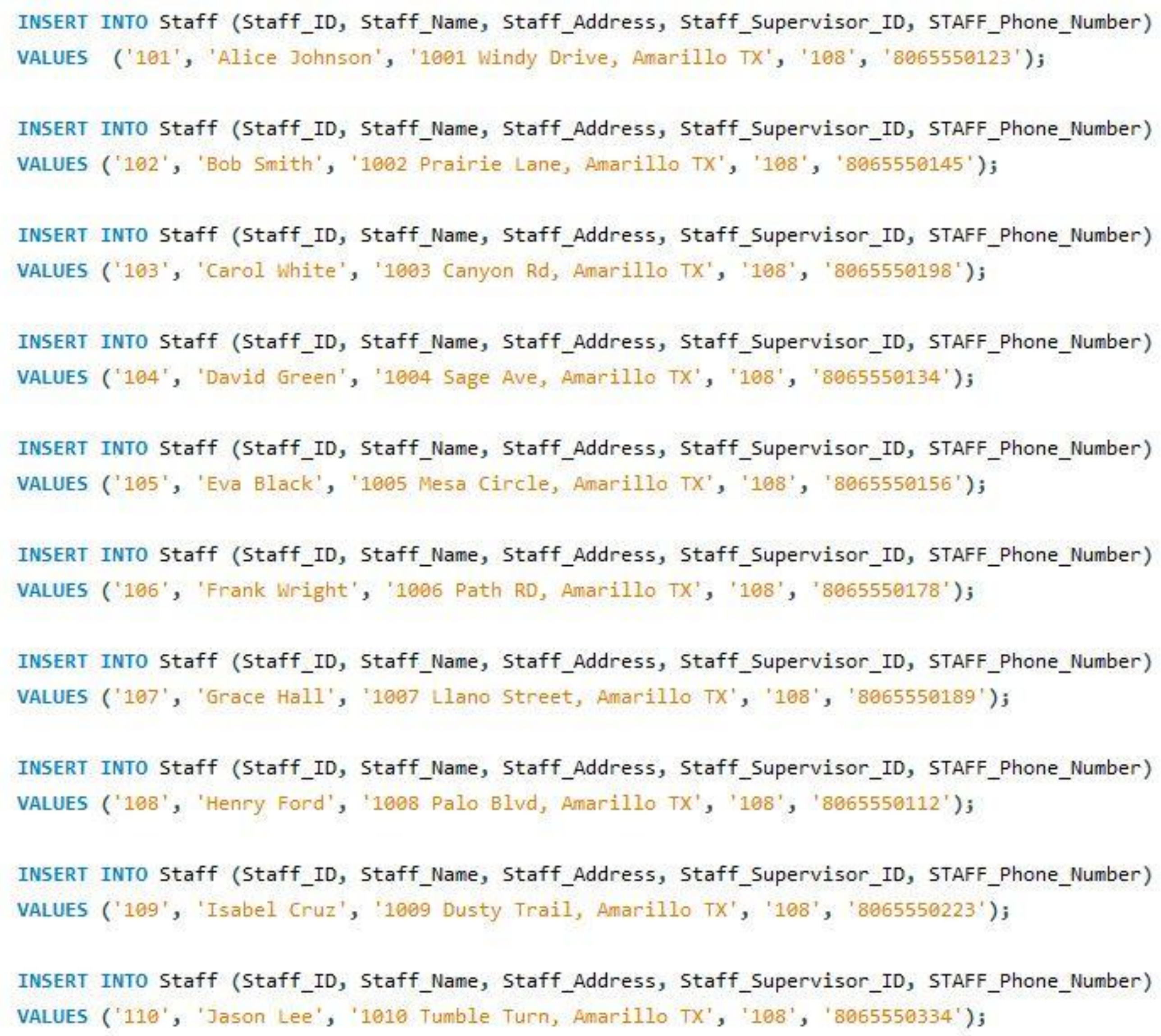


```
CREATE TABLE REGISTRATION
(PATIENT_ID INT NOT NULL,
ADMIN_ID INT NOT NULL,
REGISTRATION_DATE DATE,
CONSTRAINT REGISTRATION_PK PRIMARY KEY (PATIENT_ID),
CONSTRAINT REGISTRATION_FK1 FOREIGN KEY (PATIENT_ID) REFERENCES
PATIENT(PATIENT_ID),
CONSTRAINT REGISTRATION_FK2 FOREIGN KEY (ADMIN_ID) REFERENCES
ADMINISTRATION (ADMIN_ID))

CREATE TABLE PAYMENT
(BILLING_CODE VARCHAR(4)NOT NULL,
ADMIN_ID INT NOT NULL,
INSURANCE_ID INT,
CONSTRAINT PAYMENT_PK PRIMARY KEY (BILLING_CODE),
CONSTRAINT PAYMENT_FK1 FOREIGN KEY (ADMIN_ID) REFERENCES
ADMINISTRATION (ADMIN_ID))

ALTER TABLE MEDICAL_RECORD DROP FOREIGN KEY MEDICAL_RECORD_FK;
ALTER TABLE TREATMENT DROP FOREIGN KEY TREATMENT_FK2;
ALTER TABLE MEDICAL_CHART DROP FOREIGN KEY MEDICAL_CHART_FK1;
ALTER TABLE REGISTRATION DROP FOREIGN KEY REGISTRATION_FK1;

ALTER TABLE REGISTRATION DROP PRIMARY KEY;

ALTER TABLE REGISTRATION ADD PRIMARY KEY (PATIENT_ID, ROOM_NO);

ALTER TABLE MEDICAL_RECORD
ADD CONSTRAINT MEDICAL_RECORD_FK
FOREIGN KEY (PATIENT_ID) REFERENCES PATIENT(PATIENT_ID);

ALTER TABLE TREATMENT
ADD CONSTRAINT TREATMENT_FK2
FOREIGN KEY (PATIENT_ID) REFERENCES PATIENT (PATIENT_ID)

ALTER TABLE MEDICAL_CHART
ADD CONSTRAINT  MEDICAL_CHART_FK1
FOREIGN KEY (PATIENT_ID) REFERENCES PATIENT (PATIENT_ID)
```

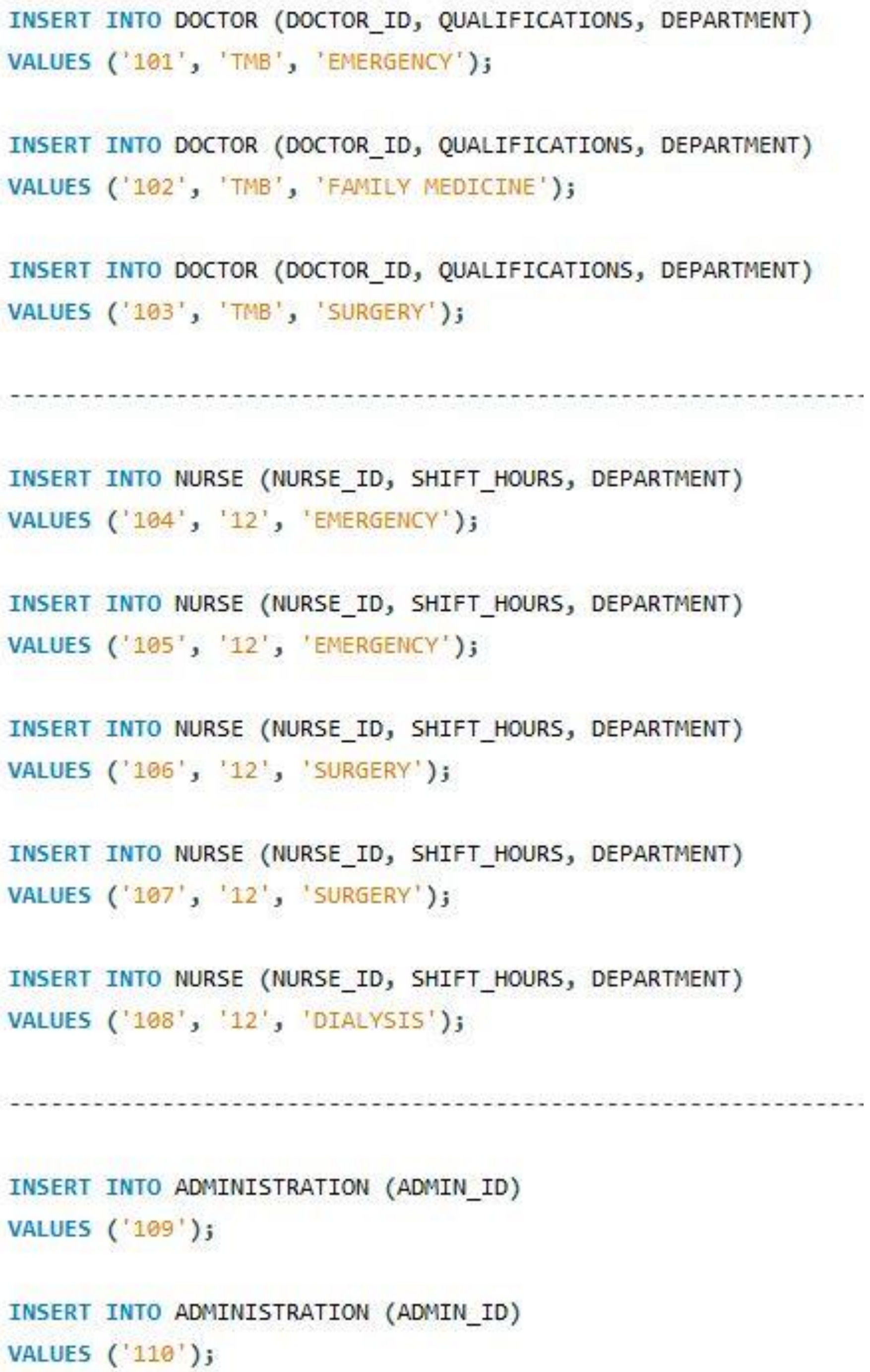


```
CREATE TABLE ROOM
(ROOM_NO INT NOT NULL,
PATIENT_ID INT NOT NULL,
CONSTRAINT ROOM_PK PRIMARY KEY (ROOM_NO),
CONSTRAINT ROOM_FK FOREIGN KEY (PATIENT_ID) REFERENCES
PATIENT (PATIENT_ID))
```

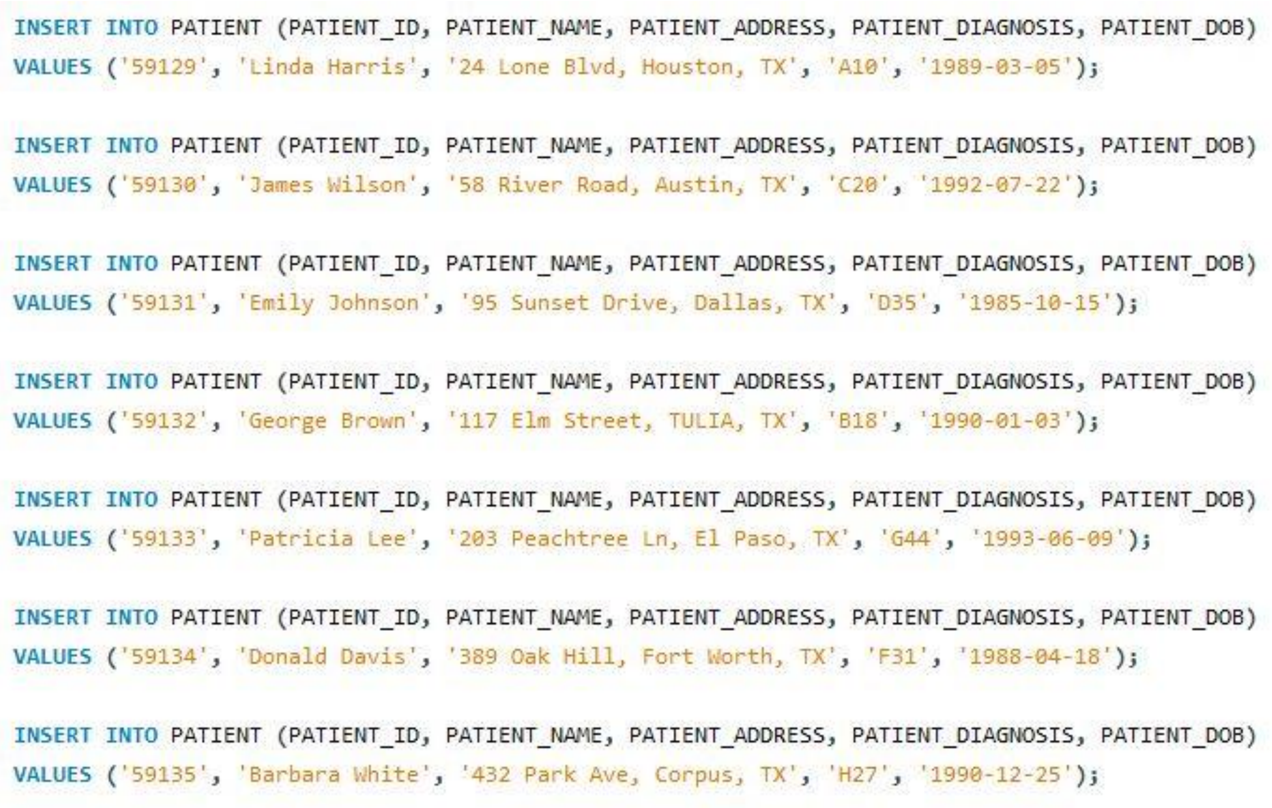
Data for SQL Tables



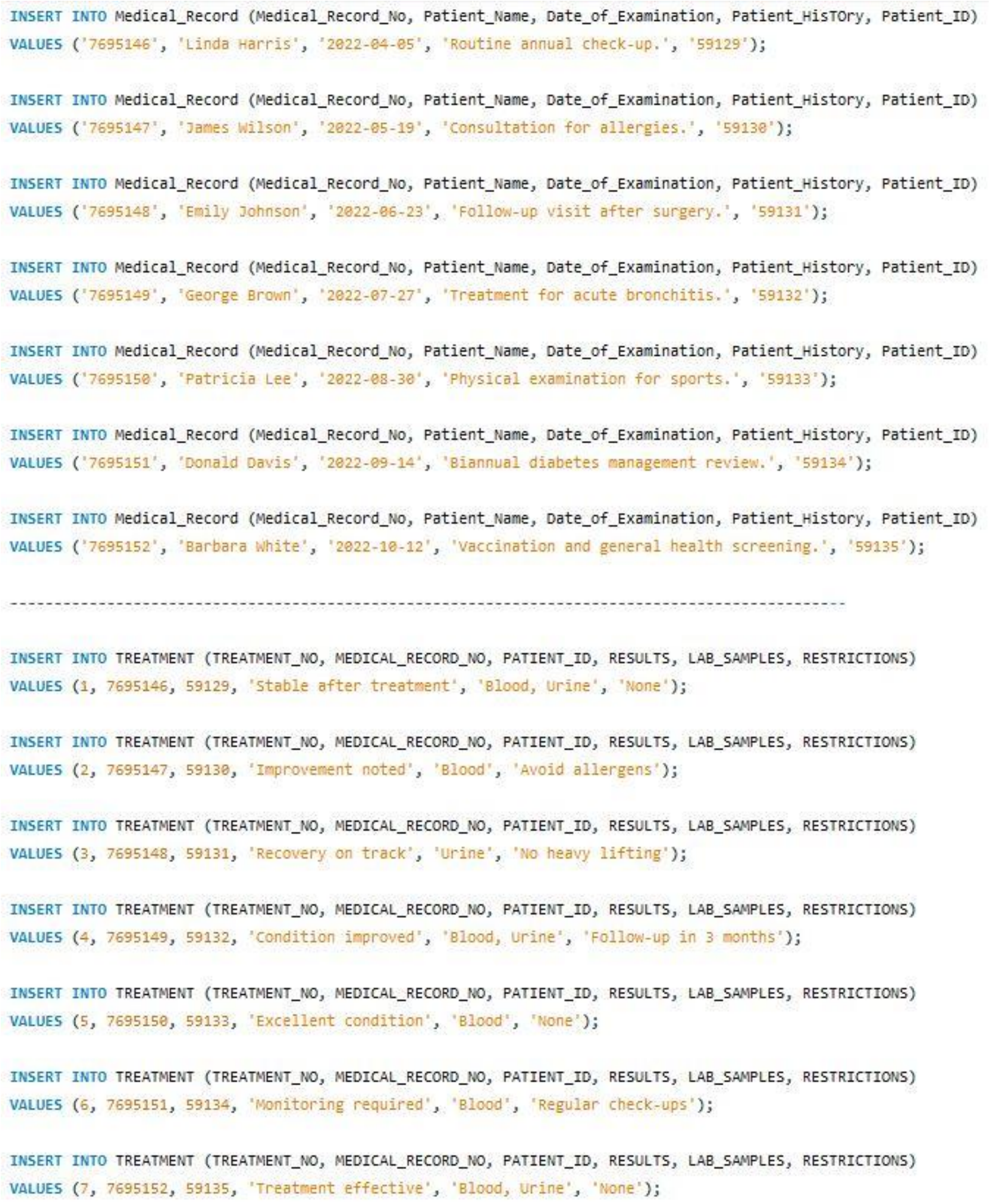
```
INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES  ('101', 'Alice Johnson', '1001 Windy Drive, Amarillo TX', '108', '8065550123');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('102', 'Bob Smith', '1002 Prairie Lane, Amarillo TX', '108', '8065550145');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('103', 'Carol White', '1003 Canyon Rd, Amarillo TX', '108', '8065550198');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('104', 'David Green', '1004 Sage Ave, Amarillo TX', '108', '8065550134');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('105', 'Eva Black', '1005 Mesa Circle, Amarillo TX', '108', '8065550156');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('106', 'Frank Wright', '1006 Path RD, Amarillo TX', '108', '8065550178');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('107', 'Grace Hall', '1007 Llano Street, Amarillo TX', '108', '8065550189');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('108', 'Henry Ford', '1008 Palo Blvd, Amarillo TX', '108', '8065550112');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('109', 'Isabel Cruz', '1009 Dusty Trail, Amarillo TX', '108', '8065550223');

INSERT INTO Staff (Staff_ID, Staff_Name, Staff_Address, Staff_Supervisor_ID, STAFF_Phone_Number)
VALUES ('110', 'Jason Lee', '1010 Tumble Turn, Amarillo TX', '108', '8065550334');
```



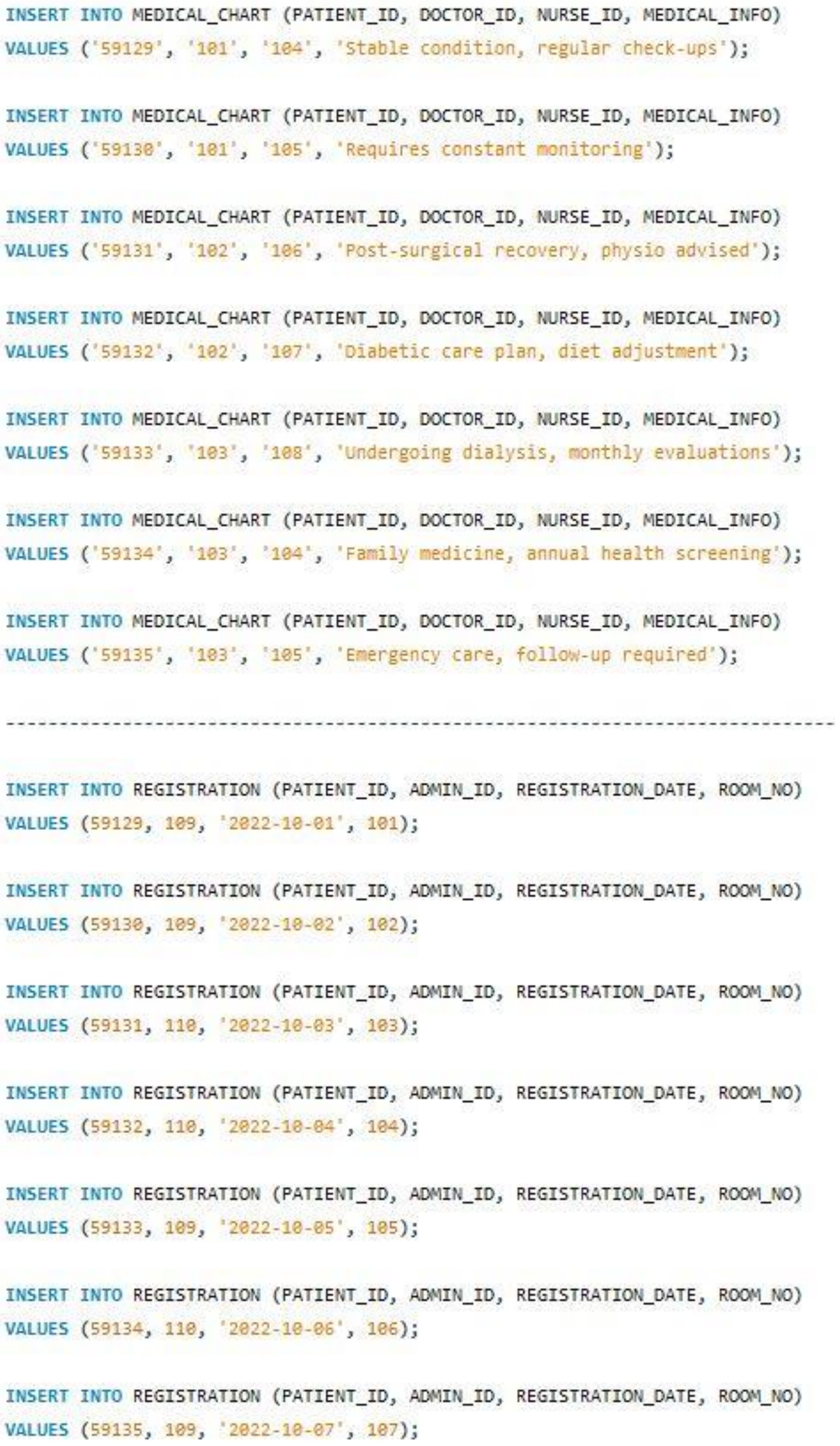
```
INSERT INTO DOCTOR (DOCTOR_ID, QUALIFICATIONS, DEPARTMENT)
VALUES ('101', 'TMB', 'EMERGENCY');

INSERT INTO DOCTOR (DOCTOR_ID, QUALIFICATIONS, DEPARTMENT)
VALUES ('102', 'TMB', 'FAMILY MEDICINE');

INSERT INTO DOCTOR (DOCTOR_ID, QUALIFICATIONS, DEPARTMENT)
VALUES ('103', 'TMB', 'SURGERY');

---------------------------------------------------------------

INSERT INTO NURSE (NURSE_ID, SHIFT_HOURS, DEPARTMENT)
VALUES ('104', '12', 'EMERGENCY');

INSERT INTO NURSE (NURSE_ID, SHIFT_HOURS, DEPARTMENT)
VALUES ('105', '12', 'EMERGENCY');

INSERT INTO NURSE (NURSE_ID, SHIFT_HOURS, DEPARTMENT)
VALUES ('106', '12', 'SURGERY');

INSERT INTO NURSE (NURSE_ID, SHIFT_HOURS, DEPARTMENT)
VALUES ('107', '12', 'SURGERY');

INSERT INTO NURSE (NURSE_ID, SHIFT_HOURS, DEPARTMENT)
VALUES ('108', '12', 'DIALYSIS');

---------------------------------------------------------------

INSERT INTO ADMINISTRATION (ADMIN_ID)
VALUES ('109');

INSERT INTO ADMINISTRATION (ADMIN_ID)
VALUES ('110');
```



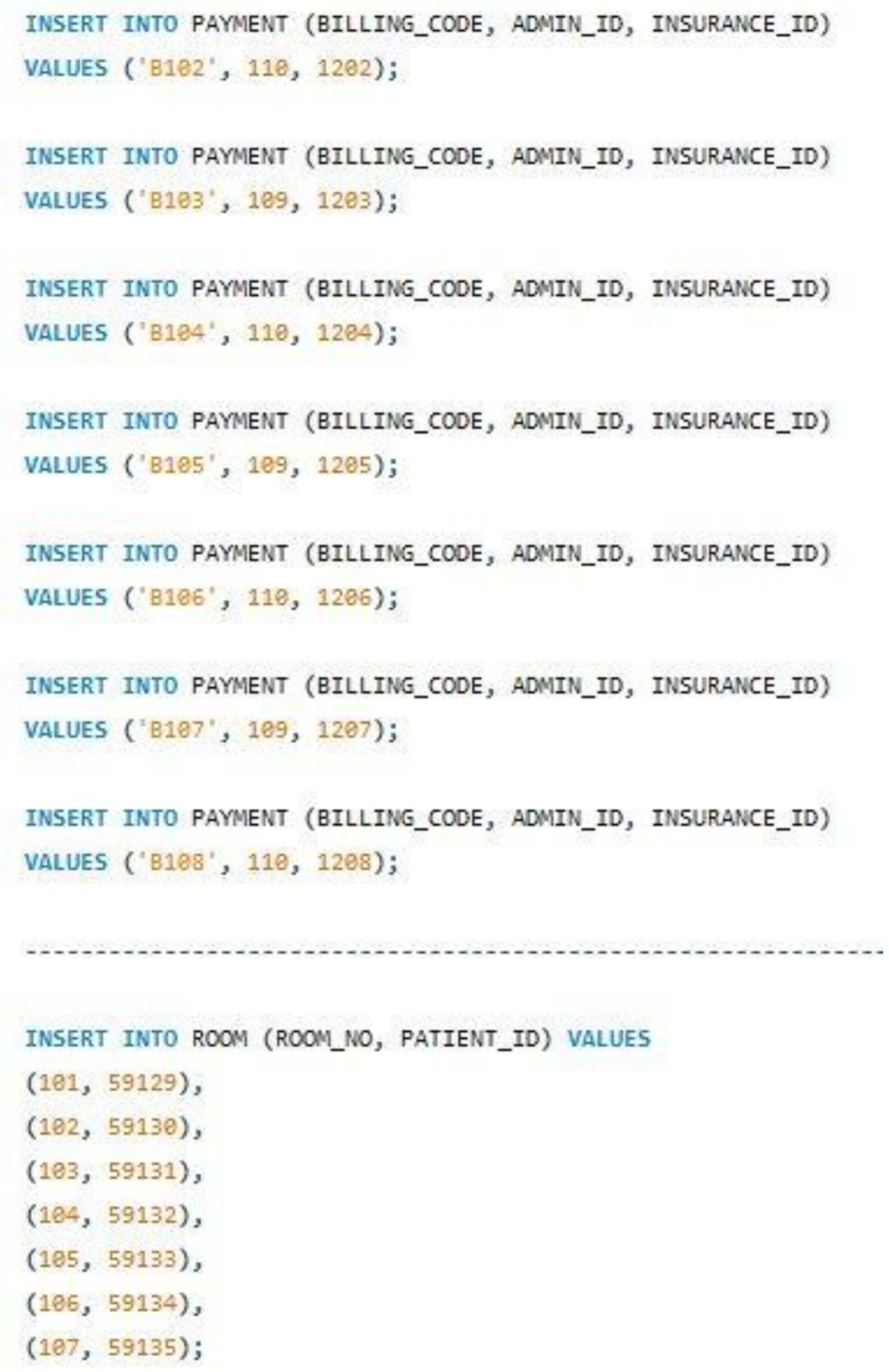
```sql
INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59129', 'Linda Harris', '24 Lone Blvd, Houston, TX', 'A10', '1989-03-05');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59130', 'James Wilson', '58 River Road, Austin, TX', 'C20', '1992-07-22');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59131', 'Emily Johnson', '95 Sunset Drive, Dallas, TX', 'D35', '1985-10-15');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59132', 'George Brown', '117 Elm Street, TULIA, TX', 'B18', '1990-01-03');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59133', 'Patricia Lee', '203 Peachtree Ln, El Paso, TX', 'G44', '1993-06-09');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59134', 'Donald Davis', '389 Oak Hill, Fort Worth, TX', 'F31', '1988-04-18');

INSERT INTO PATIENT (PATIENT_ID, PATIENT_NAME, PATIENT_ADDRESS, PATIENT_DIAGNOSIS, PATIENT_DOB)
VALUES ('59135', 'Barbara White', '432 Park Ave, Corpus, TX', 'H27', '1990-12-25');
```



```sql
INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_HisTOry, Patient_ID)
VALUES ('7695146', 'Linda Harris', '2022-04-05', 'Routine annual check-up.', '59129');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695147', 'James Wilson', '2022-05-19', 'Consultation for allergies.', '59130');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695148', 'Emily Johnson', '2022-06-23', 'Follow-up visit after surgery.', '59131');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695149', 'George Brown', '2022-07-27', 'Treatment for acute bronchitis.', '59132');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695150', 'Patricia Lee', '2022-08-30', 'Physical examination for sports.', '59133');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695151', 'Donald Davis', '2022-09-14', 'Biannual diabetes management review.', '59134');

INSERT INTO Medical_Record (Medical_Record_No, Patient_Name, Date_of_Examination, Patient_History, Patient_ID)
VALUES ('7695152', 'Barbara White', '2022-10-12', 'Vaccination and general health screening.', '59135');

-------------------------------------------------------------------------------------------------

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (1, 7695146, 59129, 'Stable after treatment', 'Blood, Urine', 'None');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (2, 7695147, 59130, 'Improvement noted', 'Blood', 'Avoid allergens');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (3, 7695148, 59131, 'Recovery on track', 'Urine', 'No heavy lifting');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (4, 7695149, 59132, 'Condition improved', 'Blood, Urine', 'Follow-up in 3 months');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (5, 7695150, 59133, 'Excellent condition', 'Blood', 'None');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (6, 7695151, 59134, 'Monitoring required', 'Blood', 'Regular check-ups');

INSERT INTO TREATMENT (TREATMENT_NO, MEDICAL_RECORD_NO, PATIENT_ID, RESULTS, LAB_SAMPLES, RESTRICTIONS)
VALUES (7, 7695152, 59135, 'Treatment effective', 'Blood, Urine', 'None');
```



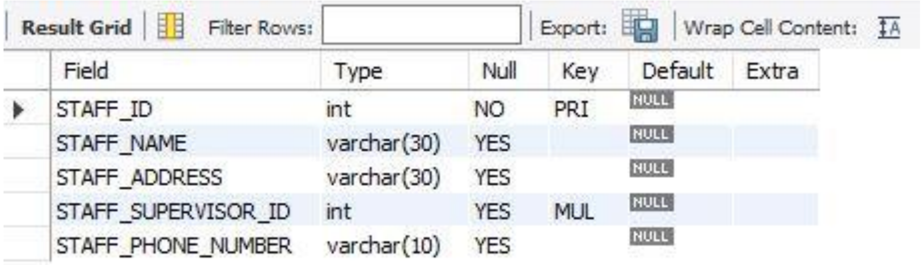
```sql
INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59129', '101', '104', 'Stable condition, regular check-ups');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59130', '101', '105', 'Requires constant monitoring');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59131', '102', '106', 'Post-surgical recovery, physio advised');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59132', '102', '107', 'Diabetic care plan, diet adjustment');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59133', '103', '108', 'Undergoing dialysis, monthly evaluations');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59134', '103', '104', 'Family medicine, annual health screening');

INSERT INTO MEDICAL_CHART (PATIENT_ID, DOCTOR_ID, NURSE_ID, MEDICAL_INFO)
VALUES ('59135', '103', '105', 'Emergency care, follow-up required');

--------------------------------------------------------------------------------

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59129, 109, '2022-10-01', 101);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59130, 109, '2022-10-02', 102);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59131, 110, '2022-10-03', 103);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59132, 110, '2022-10-04', 104);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59133, 109, '2022-10-05', 105);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59134, 110, '2022-10-06', 106);

INSERT INTO REGISTRATION (PATIENT_ID, ADMIN_ID, REGISTRATION_DATE, ROOM_NO)
VALUES (59135, 109, '2022-10-07', 107);
```



```
INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B102', 110, 1202);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B103', 109, 1203);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B104', 110, 1204);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B105', 109, 1205);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B106', 110, 1206);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B107', 109, 1207);

INSERT INTO PAYMENT (BILLING_CODE, ADMIN_ID, INSURANCE_ID)
VALUES ('B108', 110, 1208);

-------------------------------------------------------------

INSERT INTO ROOM (ROOM_NO, PATIENT_ID) VALUES
(101, 59129),
(102, 59130),
(103, 59131),
(104, 59132),
(105, 59133),
(106, 59134),
(107, 59135);
```

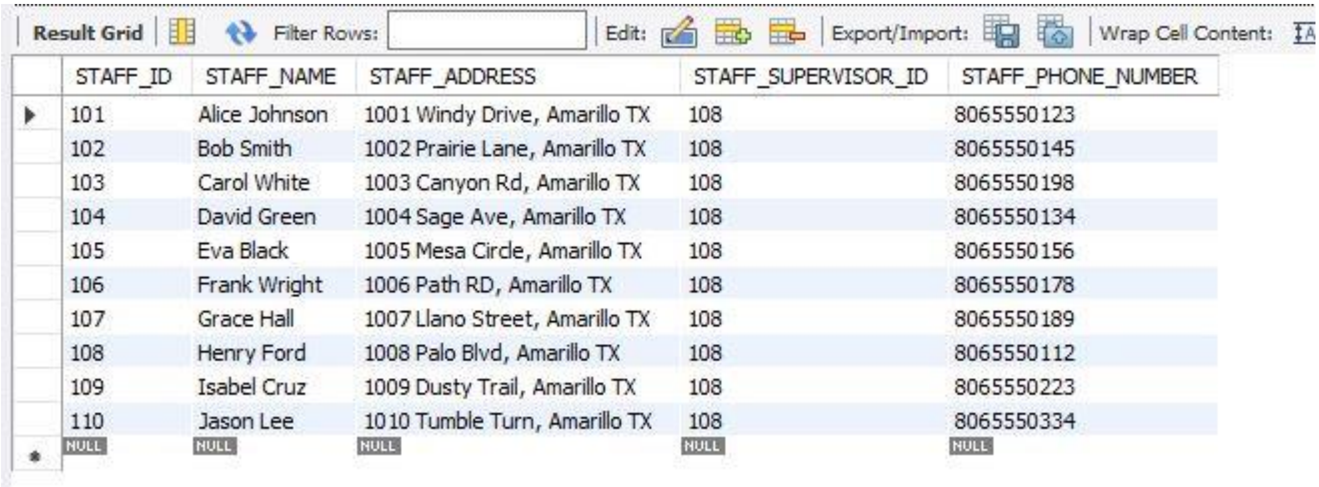
SQL Tables with Values

DESC STAFF



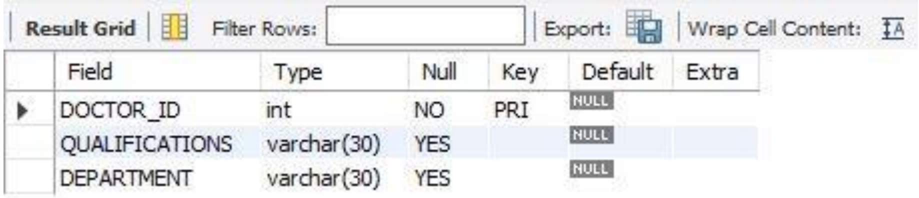
Field	Type	Null	Key	Default	Extra
STAFF_ID	int	NO	PRI	NULL	
STAFF_NAME	varchar(30)	YES		NULL	
STAFF_ADDRESS	varchar(30)	YES		NULL	
STAFF_SUPERVISOR_ID	int	YES	MUL	NULL	
STAFF_PHONE_NUMBER	varchar(10)	YES		NULL	

SELECT * FROM STAFF



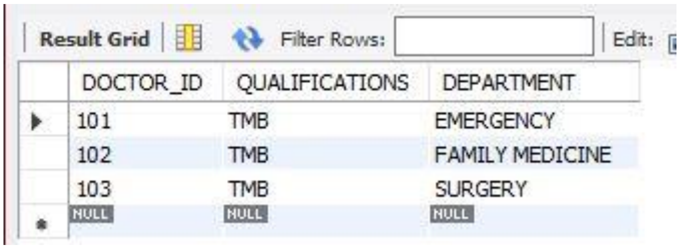
STAFF_ID	STAFF_NAME	STAFF_ADDRESS	STAFF_SUPERVISOR_ID	STAFF_PHONE_NUMBER
101	Alice Johnson	1001 Windy Drive, Amarillo TX	108	8065550123
102	Bob Smith	1002 Prairie Lane, Amarillo TX	108	8065550145
103	Carol White	1003 Canyon Rd, Amarillo TX	108	8065550198
104	David Green	1004 Sage Ave, Amarillo TX	108	8065550134
105	Eva Black	1005 Mesa Circle, Amarillo TX	108	8065550156
106	Frank Wright	1006 Path RD, Amarillo TX	108	8065550178
107	Grace Hall	1007 Llano Street, Amarillo TX	108	8065550189
108	Henry Ford	1008 Palo Blvd, Amarillo TX	108	8065550112
109	Isabel Cruz	1009 Dusty Trail, Amarillo TX	108	8065550223
110	Jason Lee	1010 Tumble Turn, Amarillo TX	108	8065550334
NULL	NULL	NULL	NULL	NULL

DESC DOCTOR



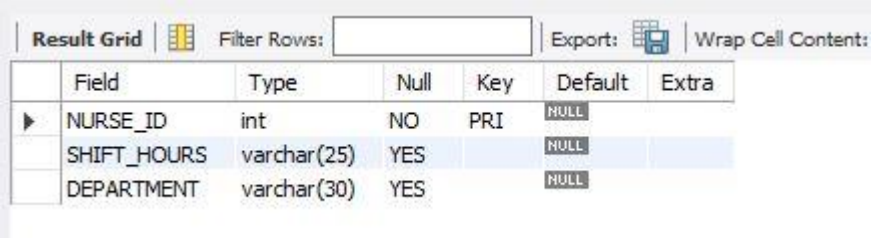
Field	Type	Null	Key	Default	Extra
DOCTOR_ID	int	NO	PRI	NULL	
QUALIFICATIONS	varchar(30)	YES		NULL	
DEPARTMENT	varchar(30)	YES		NULL	

SELECT * FROM DOCTOR



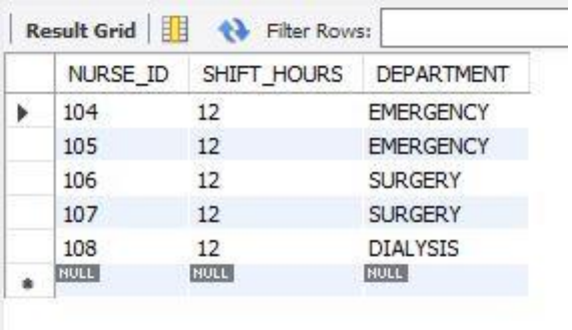
DOCTOR_ID	QUALIFICATIONS	DEPARTMENT
101	TMB	EMERGENCY
102	TMB	FAMILY MEDICINE
103	TMB	SURGERY
NULL	NULL	NULL

DESC NURSE



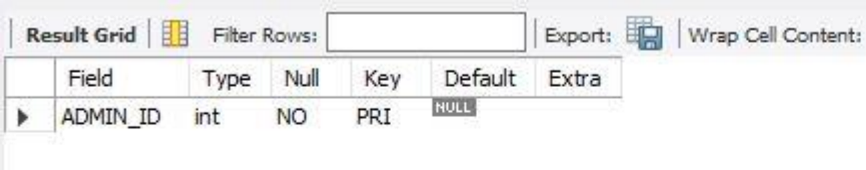
Field	Type	Null	Key	Default	Extra
NURSE_ID	int	NO	PRI	NULL	
SHIFT_HOURS	varchar(25)	YES		NULL	
DEPARTMENT	varchar(30)	YES		NULL	

SELECT * FROM NURSE



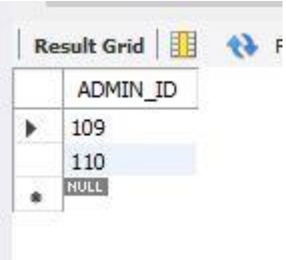
NURSE_ID	SHIFT_HOURS	DEPARTMENT
104	12	EMERGENCY
105	12	EMERGENCY
106	12	SURGERY
107	12	SURGERY
108	12	DIALYSIS
NULL	NULL	NULL

DESC ADMINISTRATION



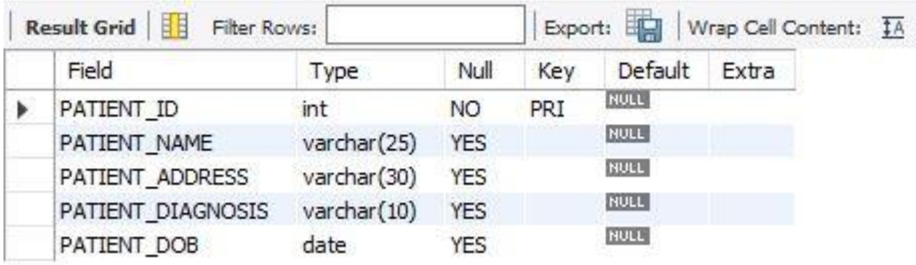
Field	Type	Null	Key	Default	Extra
ADMIN_ID	int	NO	PRI	NULL	

SELECT * FROM ADMINISTRATION



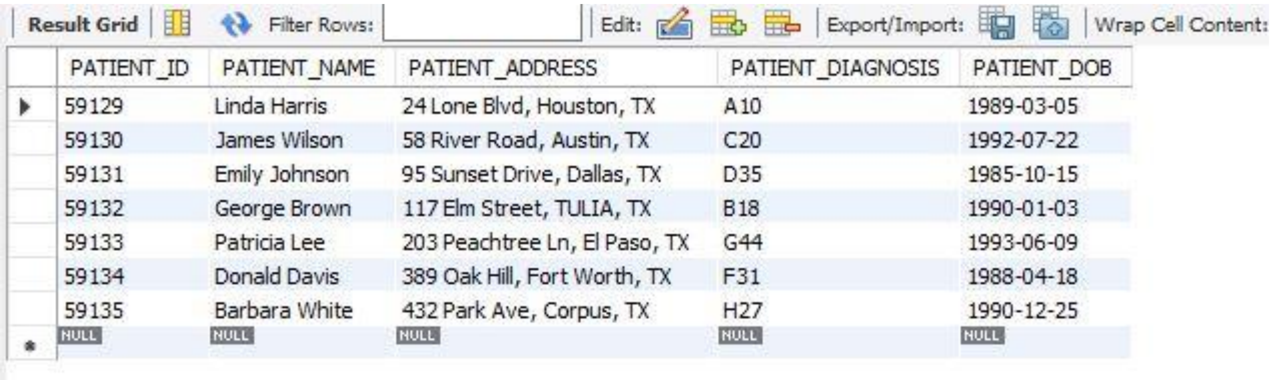
ADMIN_ID
109
110
NULL

DESC PATIENT



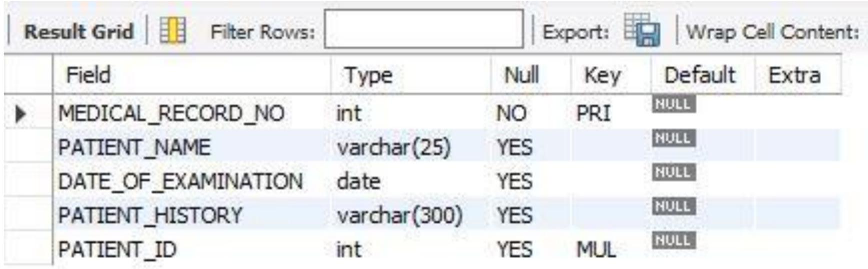
Field	Type	Null	Key	Default	Extra
PATIENT_ID	int	NO	PRI	NULL	
PATIENT_NAME	varchar(25)	YES		NULL	
PATIENT_ADDRESS	varchar(30)	YES		NULL	
PATIENT_DIAGNOSIS	varchar(10)	YES		NULL	
PATIENT_DOB	date	YES		NULL	

SELECT * FROM PATIENT



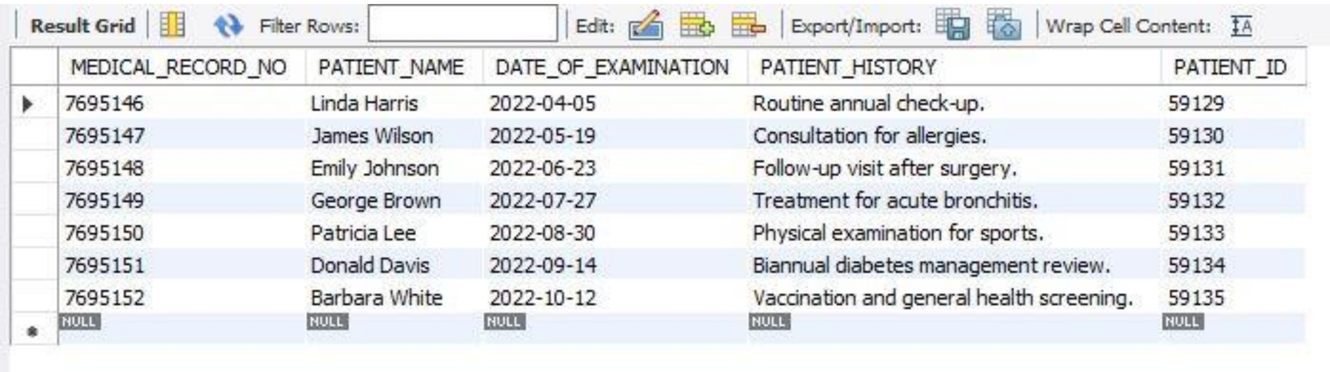
PATIENT_ID	PATIENT_NAME	PATIENT_ADDRESS	PATIENT_DIAGNOSIS	PATIENT_DOB
59129	Linda Harris	24 Lone Blvd, Houston, TX	A10	1989-03-05
59130	James Wilson	58 River Road, Austin, TX	C20	1992-07-22
59131	Emily Johnson	95 Sunset Drive, Dallas, TX	D35	1985-10-15
59132	George Brown	117 Elm Street, TULIA, TX	B18	1990-01-03
59133	Patricia Lee	203 Peachtree Ln, El Paso, TX	G44	1993-06-09
59134	Donald Davis	389 Oak Hill, Fort Worth, TX	F31	1988-04-18
59135	Barbara White	432 Park Ave, Corpus, TX	H27	1990-12-25
NULL	NULL	NULL	NULL	NULL

DESC MEDICAL_RECORD



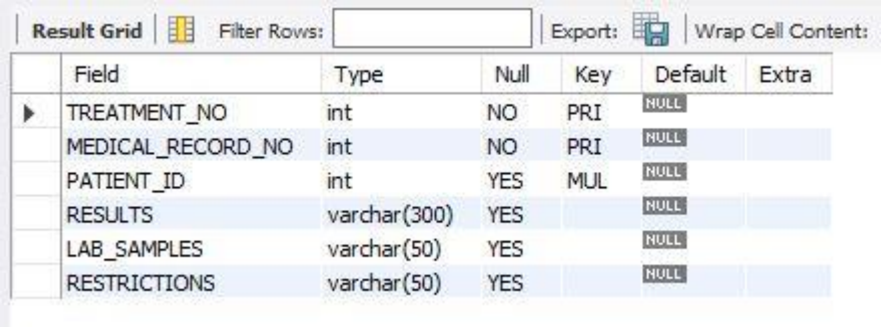
Field	Type	Null	Key	Default	Extra
MEDICAL_RECORD_NO	int	NO	PRI	NULL	
PATIENT_NAME	varchar(25)	YES		NULL	
DATE_OF_EXAMINATION	date	YES		NULL	
PATIENT_HISTORY	varchar(300)	YES		NULL	
PATIENT_ID	int	YES	MUL	NULL	

SELECT * FROM MEDICAL_RECORD



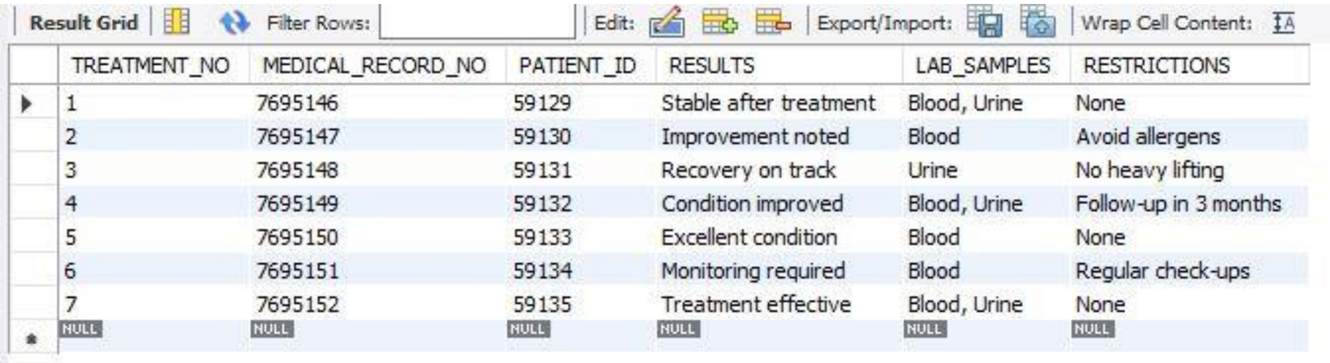
MEDICAL_RECORD_NO	PATIENT_NAME	DATE_OF_EXAMINATION	PATIENT_HISTORY	PATIENT_ID
7695146	Linda Harris	2022-04-05	Routine annual check-up.	59129
7695147	James Wilson	2022-05-19	Consultation for allergies.	59130
7695148	Emily Johnson	2022-06-23	Follow-up visit after surgery.	59131
7695149	George Brown	2022-07-27	Treatment for acute bronchitis.	59132
7695150	Patricia Lee	2022-08-30	Physical examination for sports.	59133
7695151	Donald Davis	2022-09-14	Biannual diabetes management review.	59134
7695152	Barbara White	2022-10-12	Vaccination and general health screening.	59135
NULL	NULL	NULL	NULL	NULL

DESC TREATMENT



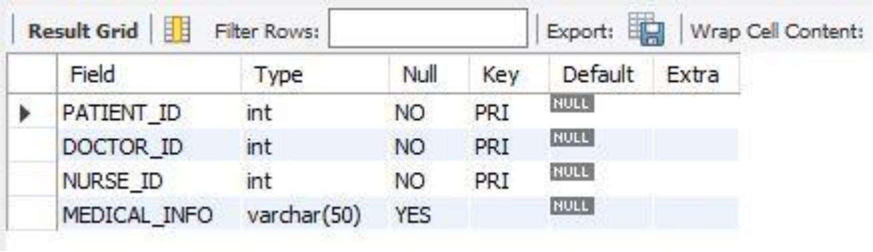
Field	Type	Null	Key	Default	Extra
TREATMENT_NO	int	NO	PRI	NULL	
MEDICAL_RECORD_NO	int	NO	PRI	NULL	
PATIENT_ID	int	YES	MUL	NULL	
RESULTS	varchar(300)	YES		NULL	
LAB_SAMPLES	varchar(50)	YES		NULL	
RESTRICTIONS	varchar(50)	YES		NULL	

SELECT * FROM TREATMENT



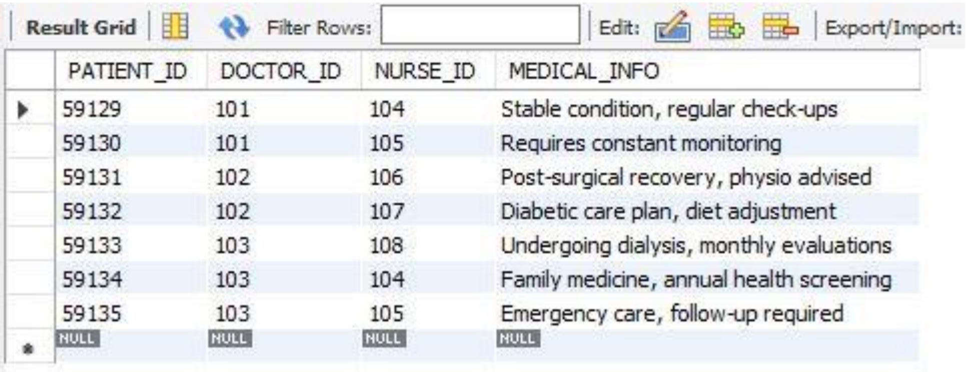
TREATMENT_NO	MEDICAL_RECORD_NO	PATIENT_ID	RESULTS	LAB_SAMPLES	RESTRICTIONS
1	7695146	59129	Stable after treatment	Blood, Urine	None
2	7695147	59130	Improvement noted	Blood	Avoid allergens
3	7695148	59131	Recovery on track	Urine	No heavy lifting
4	7695149	59132	Condition improved	Blood, Urine	Follow-up in 3 months
5	7695150	59133	Excellent condition	Blood	None
6	7695151	59134	Monitoring required	Blood	Regular check-ups
7	7695152	59135	Treatment effective	Blood, Urine	None
NULL	NULL	NULL	NULL	NULL	NULL

DESC MEDICAL_CHART



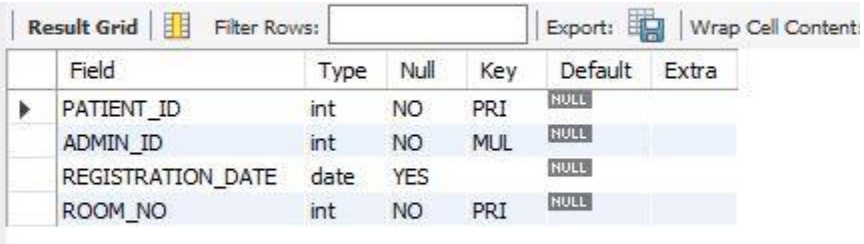
Field	Type	Null	Key	Default	Extra
PATIENT_ID	int	NO	PRI	NULL	
DOCTOR_ID	int	NO	PRI	NULL	
NURSE_ID	int	NO	PRI	NULL	
MEDICAL_INFO	varchar(50)	YES		NULL	

SELECT * FROM MEDICAL_CHART



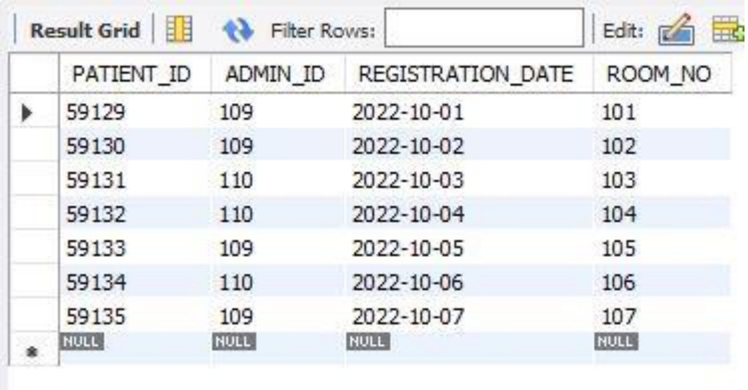
PATIENT_ID	DOCTOR_ID	NURSE_ID	MEDICAL_INFO
59129	101	104	Stable condition, regular check-ups
59130	101	105	Requires constant monitoring
59131	102	106	Post-surgical recovery, physio advised
59132	102	107	Diabetic care plan, diet adjustment
59133	103	108	Undergoing dialysis, monthly evaluations
59134	103	104	Family medicine, annual health screening
59135	103	105	Emergency care, follow-up required
NULL	NULL	NULL	NULL

DESC REGISTRATION



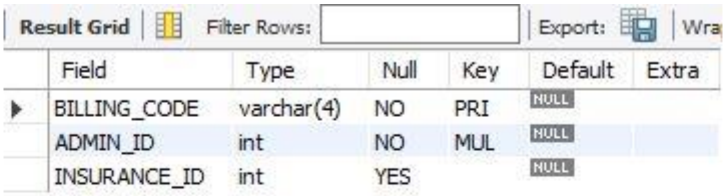
Field	Type	Null	Key	Default	Extra
PATIENT_ID	int	NO	PRI	NULL	
ADMIN_ID	int	NO	MUL	NULL	
REGISTRATION_DATE	date	YES		NULL	
ROOM_NO	int	NO	PRI	NULL	

SELECT * FROM REGISTRATION



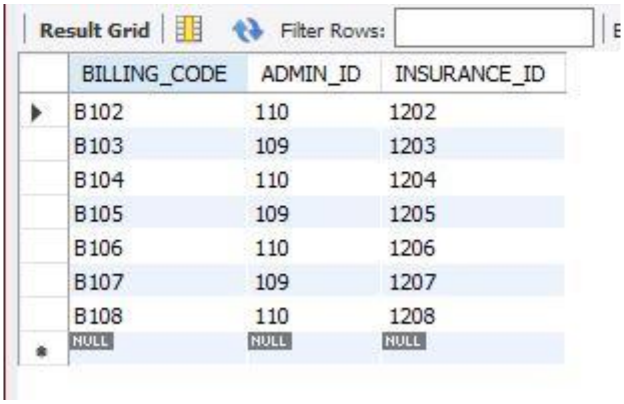
PATIENT_ID	ADMIN_ID	REGISTRATION_DATE	ROOM_NO
59129	109	2022-10-01	101
59130	109	2022-10-02	102
59131	110	2022-10-03	103
59132	110	2022-10-04	104
59133	109	2022-10-05	105
59134	110	2022-10-06	106
59135	109	2022-10-07	107
NULL	NULL	NULL	NULL

DESC PAYMENT



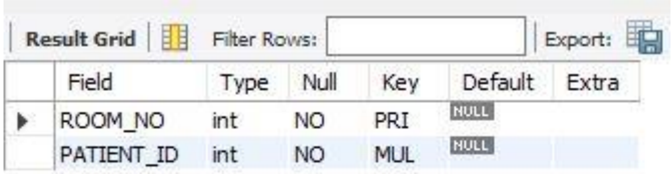
Field	Type	Null	Key	Default	Extra
BILLING_CODE	varchar(4)	NO	PRI	NULL	
ADMIN_ID	int	NO	MUL	NULL	
INSURANCE_ID	int	YES		NULL	

SELECT * FROM PAYMENT



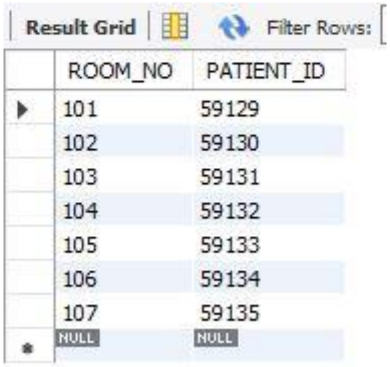
BILLING_CODE	ADMIN_ID	INSURANCE_ID
B102	110	1202
B103	109	1203
B104	110	1204
B105	109	1205
B106	110	1206
B107	109	1207
B108	110	1208
NULL	NULL	NULL

DESC ROOM



	Field	Type	Null	Key	Default	Extra
▶	ROOM_NO	int	NO	PRI	NULL	
	PATIENT_ID	int	NO	MUL	NULL	

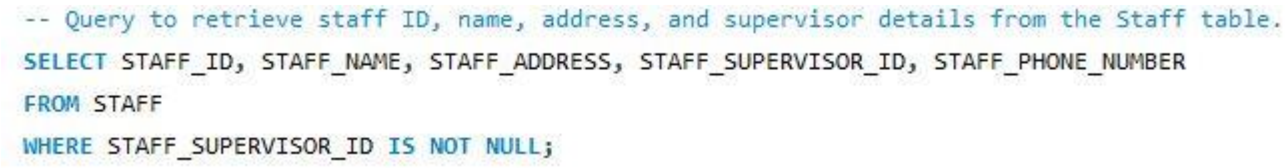
SELECT * FROM ROOM



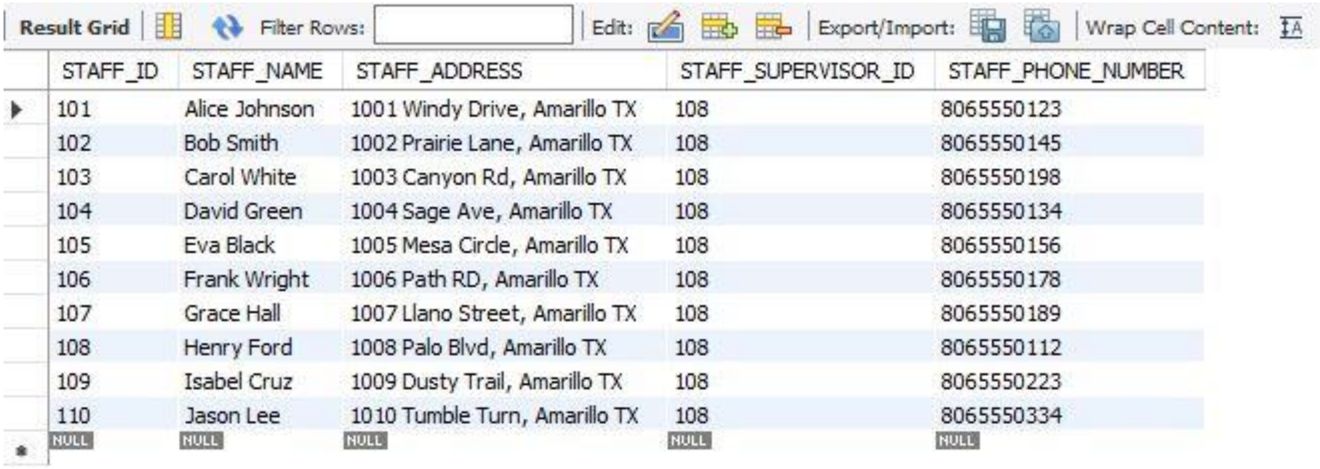
	ROOM_NO	PATIENT_ID
▶	101	59129
	102	59130
	103	59131
	104	59132
	105	59133
	106	59134
	107	59135
*	NULL	NULL

Queries

Q 1 -

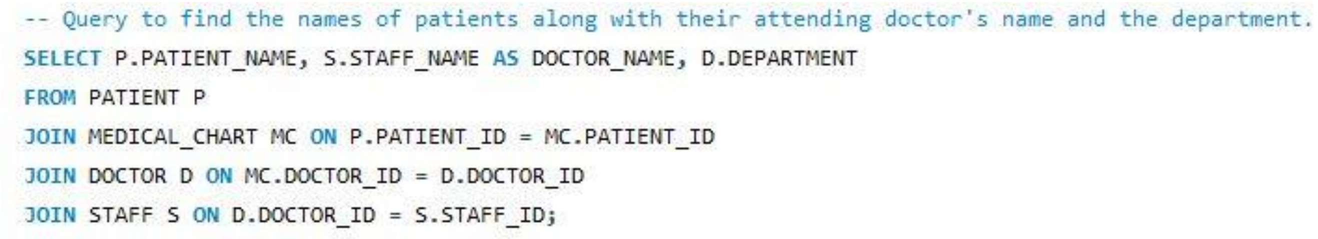


```
-- Query to retrieve staff ID, name, address, and supervisor details from the Staff table.
SELECT STAFF_ID, STAFF_NAME, STAFF_ADDRESS, STAFF_SUPERVISOR_ID, STAFF_PHONE_NUMBER
FROM STAFF
WHERE STAFF_SUPERVISOR_ID IS NOT NULL;
```

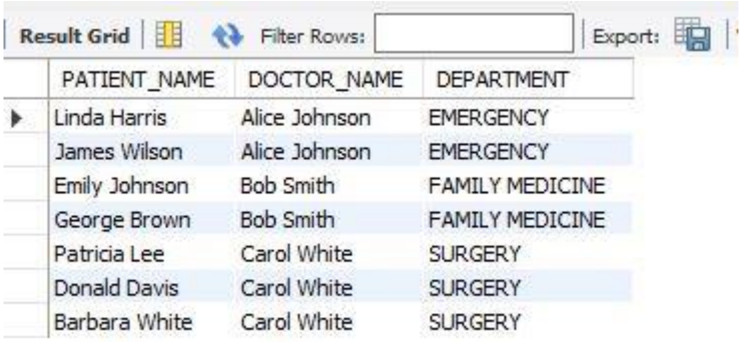


STAFF_ID	STAFF_NAME	STAFF_ADDRESS	STAFF_SUPERVISOR_ID	STAFF_PHONE_NUMBER
101	Alice Johnson	1001 Windy Drive, Amarillo TX	108	8065550123
102	Bob Smith	1002 Prairie Lane, Amarillo TX	108	8065550145
103	Carol White	1003 Canyon Rd, Amarillo TX	108	8065550198
104	David Green	1004 Sage Ave, Amarillo TX	108	8065550134
105	Eva Black	1005 Mesa Circle, Amarillo TX	108	8065550156
106	Frank Wright	1006 Path RD, Amarillo TX	108	8065550178
107	Grace Hall	1007 Llano Street, Amarillo TX	108	8065550189
108	Henry Ford	1008 Palo Blvd, Amarillo TX	108	8065550112
109	Isabel Cruz	1009 Dusty Trail, Amarillo TX	108	8065550223
110	Jason Lee	1010 Tumble Turn, Amarillo TX	108	8065550334
NULL	NULL	NULL	NULL	NULL

Q 2 -

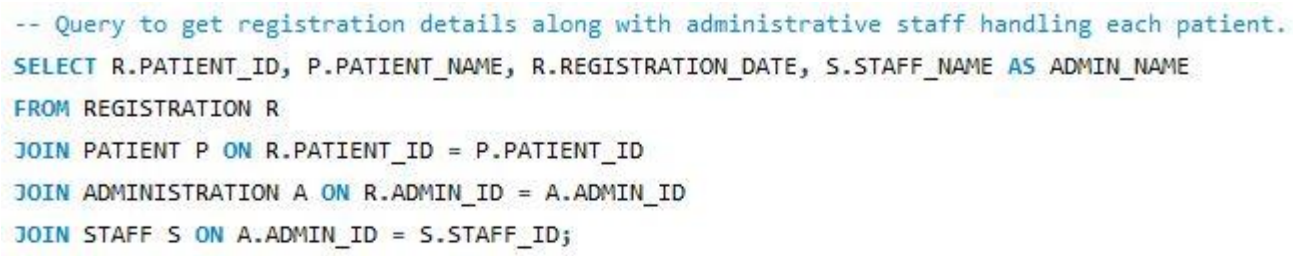


```
-- Query to find the names of patients along with their attending doctor's name and the department.
SELECT P.PATIENT_NAME, S.STAFF_NAME AS DOCTOR_NAME, D.DEPARTMENT
FROM PATIENT P
JOIN MEDICAL_CHART MC ON P.PATIENT_ID = MC.PATIENT_ID
JOIN DOCTOR D ON MC.DOCTOR_ID = D.DOCTOR_ID
JOIN STAFF S ON D.DOCTOR_ID = S.STAFF_ID;
```

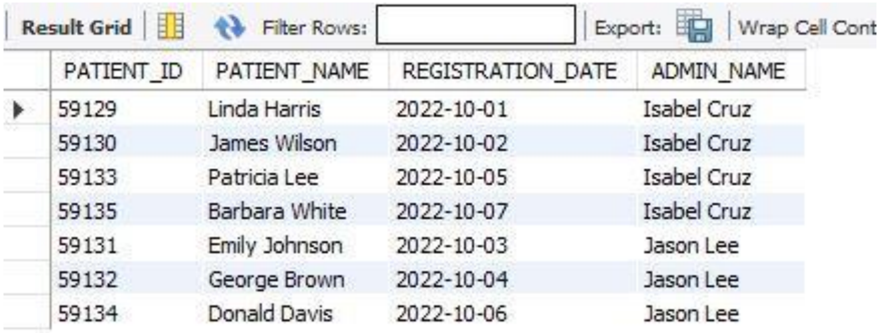


PATIENT_NAME	DOCTOR_NAME	DEPARTMENT
Linda Harris	Alice Johnson	EMERGENCY
James Wilson	Alice Johnson	EMERGENCY
Emily Johnson	Bob Smith	FAMILY MEDICINE
George Brown	Bob Smith	FAMILY MEDICINE
Patricia Lee	Carol White	SURGERY
Donald Davis	Carol White	SURGERY
Barbara White	Carol White	SURGERY

Q 3 -

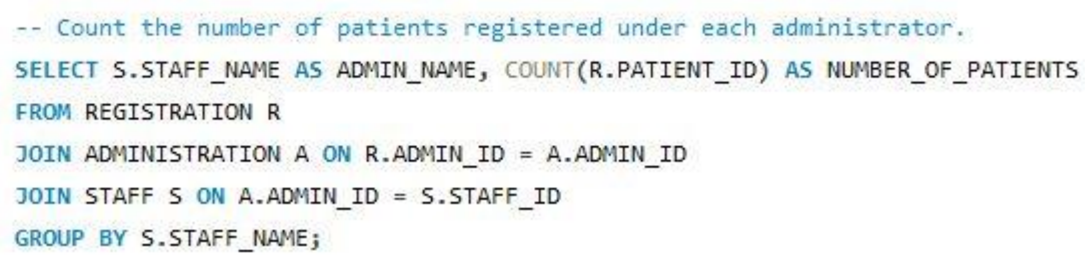


```
-- Query to get registration details along with administrative staff handling each patient.
SELECT R.PATIENT_ID, P.PATIENT_NAME, R.REGISTRATION_DATE, S.STAFF_NAME AS ADMIN_NAME
FROM REGISTRATION R
JOIN PATIENT P ON R.PATIENT_ID = P.PATIENT_ID
JOIN ADMINISTRATION A ON R.ADMIN_ID = A.ADMIN_ID
JOIN STAFF S ON A.ADMIN_ID = S.STAFF_ID;
```

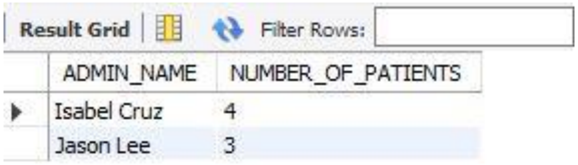


PATIENT_ID	PATIENT_NAME	REGISTRATION_DATE	ADMIN_NAME
59129	Linda Harris	2022-10-01	Isabel Cruz
59130	James Wilson	2022-10-02	Isabel Cruz
59133	Patricia Lee	2022-10-05	Isabel Cruz
59135	Barbara White	2022-10-07	Isabel Cruz
59131	Emily Johnson	2022-10-03	Jason Lee
59132	George Brown	2022-10-04	Jason Lee
59134	Donald Davis	2022-10-06	Jason Lee

Q 4 –

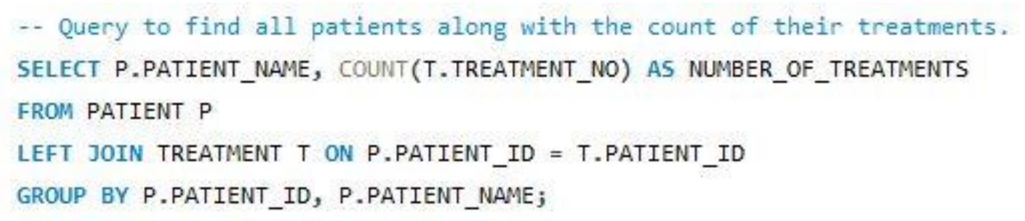


```
-- Count the number of patients registered under each administrator.
SELECT S.STAFF_NAME AS ADMIN_NAME, COUNT(R.PATIENT_ID) AS NUMBER_OF_PATIENTS
FROM REGISTRATION R
JOIN ADMINISTRATION A ON R.ADMIN_ID = A.ADMIN_ID
JOIN STAFF S ON A.ADMIN_ID = S.STAFF_ID
GROUP BY S.STAFF_NAME;
```

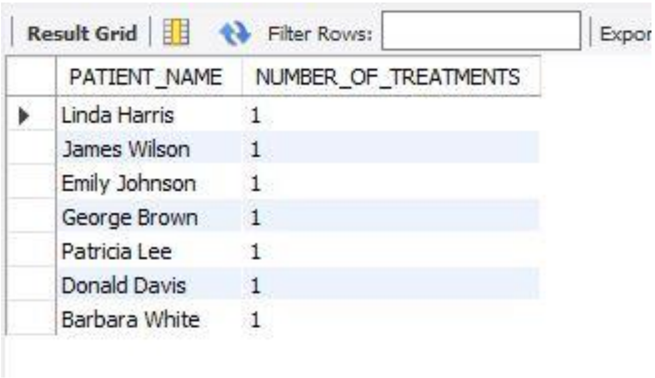


ADMIN_NAME	NUMBER_OF_PATIENTS
Isabel Cruz	4
Jason Lee	3

Q 5 -



```
-- Query to find all patients along with the count of their treatments.
SELECT P.PATIENT_NAME, COUNT(T.TREATMENT_NO) AS NUMBER_OF_TREATMENTS
FROM PATIENT P
LEFT JOIN TREATMENT T ON P.PATIENT_ID = T.PATIENT_ID
GROUP BY P.PATIENT_ID, P.PATIENT_NAME;
```



PATIENT_NAME	NUMBER_OF_TREATMENTS
Linda Harris	1
James Wilson	1
Emily Johnson	1
George Brown	1
Patricia Lee	1
Donald Davis	1
Barbara White	1