CISC3000 Project

PO HIU LAM DC227620

HUANG SHAO XING DC026709

KOU KIN CHON DC025244

Abstract

The database consists of seven main entities, including patients, doctors, appointments, medical records, prescriptions, admission, and wards, designed to manage hospital information. Each entity serves a specific function and is essential to the efficient management of hospital information.

The patient entity contains personal information about patients and uses the patient's ID as the primary key to identify each patient. The doctor entity stores information about employed doctors and uses the doctor's ID as the primary key to identify each doctor.

The appointments entity manages the scheduling of appointments between patients and doctors. It involves both patients' and doctors' IDs, as well as the date of the appointment. The medical records entity stores information about the symptoms and treatment dates of patients, using the record's ID as the primary key and two reference keys to connect to other schemas.

The prescriptions entity stores information about the medication prescribed to patients and the date of the prescription. The wards entity manages the allocation of beds to patients and includes the ward ID as the primary key, capacity, and the number of occupied and available beds.

Lastly, the admissions entity records when patients come and leave the hospital with admission and discharge dates.

Description of Entities

patients (patient_id, first_name, last_name, gender, date_of_birth)

This table is used to store patient information for a hospital, including unique identifiers (patient_id), name, gender, and date of birth.

This table is used to store information about doctors for a hospital, including unique identifiers (doctor_id), name, specialty field, and phone number.

This table is used to store information about doctors for a hospital, including unique identifiers (doctor_id), name, specialty field, and phone number.

doctors (doctor_id, first_name, last_name, specialty, phone_number)

This table is used to store information about doctors for a hospital, including unique identifiers (doctor_id), name, specialty field, and phone number.

appointments (patient_id, doctor_id, appointment_date, appointment_time)

This table is used to store appointment information for a hospital, including unique identifiers for each appointment (appointment_id), the patient's unique identifier (patient_id), the doctor's unique identifier (doctor_id), the appointment date, and time.

medical_records (record_id, patient_id, doctor_id, medical_date, diagnosis)

This table is used to store medical record information for a hospital, including unique identifiers for each record (record_id), the patient's unique identifier (patient_id), the doctor's unique identifier (doctor_id), the date of the medical record, and the diagnosis result.

prescriptions (prescription_id, patient_id, doctor_id, medication)

This table is used to store prescription information for a hospital, including unique identifiers for each prescription (prescription_id), the patient's unique identifier (patient_id), the doctor's unique identifier (doctor_id), the prescription date, and the medication prescribed.

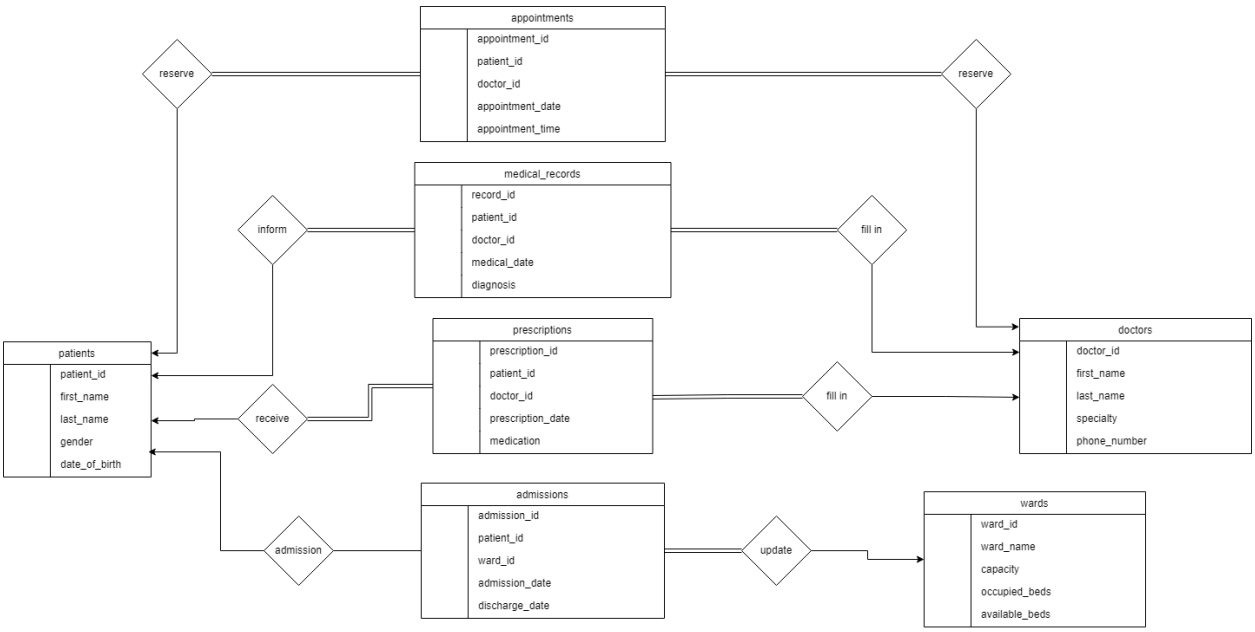
wards (ward_id, ward_name, capacity, occupied_beds, available_beds)

This table is used to store ward information for a hospital, including unique identifiers for each ward (ward_id), the ward name, capacity, the number of occupied beds, and the number of available beds.

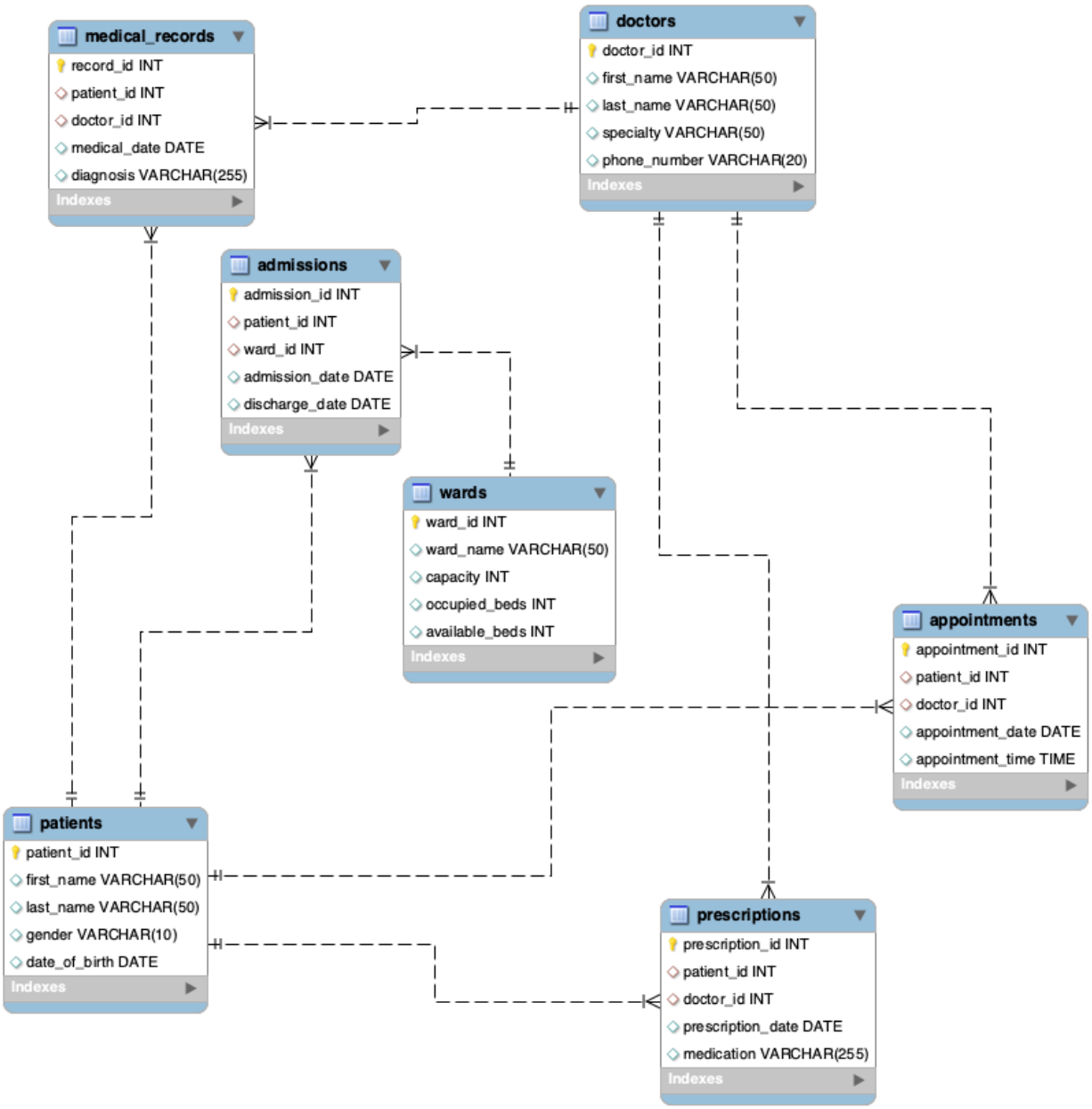
admissions (admission_id, patient_id, ward_id, admission_date, discharge_date)

This table is used to store admission information for a hospital, including unique identifiers for each admission (admission_id), the patient's unique identifier (patient_id), the ward's unique identifier (ward_id), the admission date, and the discharge date.

ER Diagram



EER Diagram



Table

```
DROP DATABASE IF EXISTS hospital;
CREATE DATABASE hospital;

use hospital;
CREATE TABLE patients (
patient_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
first_name VARCHAR(50),
last_name VARCHAR(50),
```

```
gender VARCHAR(10),
date_of_birth DATE
);

CREATE TABLE doctors (
doctor_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
first_name VARCHAR(50),
last_name VARCHAR(50),
specialty VARCHAR(50),
phone_number VARCHAR(20)
);

CREATE TABLE appointments (
appointment_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
patient_id INT,
doctor_id INT,
appointment_date DATE,
appointment_time TIME,
FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)
);
CREATE TABLE medical_records (
record_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
patient_id INT,
doctor_id INT,
medical_date DATE,
diagnosis VARCHAR(255),
FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)
);
CREATE TABLE prescriptions (
prescription_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
patient_id INT,
doctor_id INT,
prescription_date DATE,
medication VARCHAR(255),
FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)
);
CREATE TABLE wards (
ward_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
ward_name VARCHAR(50),
capacity INT,
occupied_beds INT,
available_beds INT
);
CREATE TABLE admissions (
admission_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
patient_id INT,
ward_id INT,
admission_date DATE,
discharge_date DATE,
FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
```

```
FOREIGN KEY (ward_id) REFERENCES wards(ward_id)
);
```

Insert data

```
INSERT INTO patients (first_name, last_name, gender, date_of_birth) VALUES
('John', 'Doe', 'Male', '1985-05-15'),
('Jane', 'Doe', 'Female', '1990-02-20'),
('Bob', 'Smith', 'Male', '1978-10-10'),
('Alice', 'Smith', 'Female', '1982-12-31'),
('Tom', 'Lee', 'Male', '1995-07-01'),
('Linda', 'Lee', 'Female', '1988-03-25'),
('David', 'Brown', 'Male', '1976-01-05'),
('Mary', 'Brown', 'Female', '1980-09-17'),
('Peter', 'Taylor', 'Male', '1992-11-22'),
('Lucy', 'Taylor', 'Female', '1986-06-13'),
('Jack', 'Johnson', 'Male', '1998-08-08'),
('Emily', 'Johnson', 'Female', '1994-04-04');

INSERT INTO doctors (first_name, last_name, specialty, phone_number) VALUES
('Michael', 'Smith', 'Cardiologist', '123-456-7890'),
('Jennifer', 'Lee', 'Dermatologist', '234-567-8901'),
('William', 'Brown', 'Neurologist', '345-678-9012'),
('Elizabeth', 'Taylor', 'Ophthalmologist', '456-789-0123'),
('Robert', 'Johnson', 'Psychiatrist', '567-890-1234'),
('Karen', 'Davis', 'Pediatrician', '678-901-2345'),
('Daniel', 'Wilson', 'General Practitioner', '789-012-3456'),
('manho', 'Wilson', 'General Practitioner', '789-012-3456'),
('jimmy', 'Wilson', 'General Practitioner', '789-012-3456'),
('Aston', 'Wilson', 'General Practitioner', '789-012-3456');

INSERT INTO appointments (patient_id, doctor_id, appointment_date, appointment_time) VALUE
S
(1, 1, '2023-05-01', '10:00:00'),
(2, 2, '2023-05-02', '11:00:00'),
(3, 3, '2023-05-03', '12:00:00'),
(4, 4, '2023-05-04', '13:00:00'),
(5, 5, '2023-05-05', '14:00:00'),
(6, 6, '2023-05-06', '15:00:00'),
(7, 7, '2023-05-07', '16:00:00'),
(8, 1, '2023-05-08', '17:00:00'),
(9, 2, '2023-05-09', '18:00:00'),
(10, 3, '2023-05-10', '19:00:00');

INSERT INTO medical_records (patient_id, doctor_id, medical_date, diagnosis) VALUES
(1, 1, '2023-05-01', 'Heart disease'),
(1, 1, '2023-05-02', 'covid 19'),
(1, 1, '2023-05-02', 'covid 19'),
(2, 2, '2023-05-02', 'Acne'),
```

```
(2, 2, '2023-05-02', 'Heart disease'),
(3, 3, '2023-05-03', 'Migraine'),
(4, 4, '2023-05-04', 'Cataract'),
(5, 5, '2023-05-05', 'Depression'),
(6, 6, '2023-05-06', 'Asthma'),
(7, 7, '2023-05-07', 'Flu'),
(8, 1, '2023-05-08', 'High blood pressure'),
(9, 2, '2023-05-09', 'Eczema'),
(10, 3, '2023-05-10', 'Alzheimer’s disease');

INSERT INTO prescriptions (patient_id, doctor_id, prescription_date, medication) VALUES
(1, 1, '2023-05-01', 'Lipitor'),
(2, 2, '2023-05-02', 'Retin-A'),
(3, 3, '2023-05-03', 'Imitrex'),
(4, 4, '2023-05-04', 'Lumigan'),
(5, 5, '2023-05-05', 'Prozac'),
(6, 6, '2023-05-06', 'Ventolin'),
(7, 7, '2023-05-07', 'Tamiflu'),
(8, 1, '2023-05-08', 'Norvasc'),
(9, 2, '2023-05-09', 'Elidel'),
(10, 3, '2023-05-10', 'Aricept');

INSERT INTO wards (ward_name, capacity, occupied_beds, available_beds) VALUES
('Cardiology', 20, 10, 10),
('Dermatology', 30, 20, 10),
('Neurology', 15, 5, 10),
('Ophthalmology', 25, 15, 10),
('Psychiatry', 10, 5, 5);

INSERT INTO admissions (patient_id, ward_id, admission_date, discharge_date) VALUES
(1, 1, '2023-05-01', '2023-05-05'),
(2, 2, '2023-05-02', '2023-05-06'),
(3, 3, '2023-05-03', '2023-05-07'),
(4, 4, '2023-05-04', '2023-05-08'),
(5, 5, '2023-05-05', '2023-05-09'),
(6, 1, '2023-05-06', '2023-05-10'),
(7, 2, '2023-05-07', '2023-05-11'),
(8, 3, '2023-05-08', '2023-05-12'),
(9, 4, '2023-05-09', '2023-05-13'),
(10, 5, '2023-05-10', '2023-05-14');
```

View & Function & Procedure

patient_appointments

— Show the corresponding information of doctor and date for each patients

```
CREATE VIEW patient_appointments AS
SELECT p.first_name, p.last_name, d.first_name AS doctor_first_name, d.last_name AS doctor
_last_name, a.appointment_date, a.appointment_time
FROM patients p
INNER JOIN appointments a ON p.patient_id = a.patient_id
INNER JOIN doctors d ON a.doctor_id = d.doctor_id;

SELECT * FROM patient_appointments;
```

get_patient_latest_diagnosis(patient_id INT)

— Search the diagnosis of patients using patient_id(1 - 12) as parameter

```
DROP FUNCTION IF EXISTS get_patient_latest_diagnosis;
DELIMITER $$
CREATE FUNCTION get_patient_latest_diagnosis(patient_id INT)
RETURNS VARCHAR(255)
READS SQL DATA
BEGIN
DECLARE diagnosis VARCHAR(255);
SELECT medical_records.diagnosis INTO diagnosis
FROM medical_records
WHERE patient_id = medical_records.patient_id
ORDER BY date DESC LIMIT 1;
RETURN diagnosis;
END $$
DELIMITER ;

SELECT get_patient_latest_diagnosis(1);

SELECT get_patient_latest_diagnosis(10);
```

get_appointment_count_by_date_range

— Count the number of patients reserve hospital in selected period.

```
DROP procedure IF EXISTS get_appointment_count_by_date_range;
DELIMITER //
CREATE PROCEDURE get_appointment_count_by_date_range(IN start_date DATE, IN end_date DATE,
OUT appointment_count INT)
BEGIN
SELECT COUNT(*) INTO appointment_count
FROM appointments
WHERE appointments.appointment_date BETWEEN start_date AND end_date;
```

```
END//
DELIMITER ;
```

select * from schema

```
select * from patients;
```

patient_id	first_name	last_name	gender	date_of_birth
1	John	Doe	Male	1985-05-15
2	Jane	Doe	Female	1990-02-20
3	Bob	Smith	Male	1978-10-10
4	Alice	Smith	Female	1982-12-31
5	Tom	Lee	Male	1995-07-01
6	Linda	Lee	Female	1988-03-25
7	David	Brown	Male	1976-01-05
8	Mary	Brown	Female	1980-09-17
9	Peter	Taylor	Male	1992-11-22
10	Lucy	Taylor	Female	1986-06-13
11	Jack	Johnson	Male	1998-08-08
12	Emily	Johnson	Female	1994-04-04
NULL	NULL	NULL	NULL	NULL

```
select * from doctors;
```

doctor_id	first_name	last_name	specialty	phone_number
1	Michael	Smith	Cardiologist	123-456-7890
2	Jennifer	Lee	Dermatologist	234-567-8901
3	William	Brown	Neurologist	345-678-9012
4	Elizabeth	Taylor	Ophthalmologist	456-789-0123
5	Robert	Johnson	Psychiatrist	567-890-1234
6	Karen	Davis	Pediatrician	678-901-2345
7	Daniel	Wilson	General Practitioner	789-012-3456
8	manho	Wilson	General Practitioner	789-012-3456
9	jimmy	Wilson	General Practitioner	789-012-3456
10	Aston	Wilson	General Practitioner	789-012-3456
NULL	NULL	NULL	NULL	NULL

```
select * from appointments;
```

appointment_id	patient_id	doctor_id	appointment_date	appointment_time
1	1	1	2023-05-01	10:00:00
2	2	2	2023-05-02	11:00:00
3	3	3	2023-05-03	12:00:00
4	4	4	2023-05-04	13:00:00
5	5	5	2023-05-05	14:00:00
6	6	6	2023-05-06	15:00:00
7	7	7	2023-05-07	16:00:00
8	8	1	2023-05-08	17:00:00
9	9	2	2023-05-09	18:00:00
10	10	3	2023-05-10	19:00:00
NULL	NULL	NULL	NULL	NULL

```
select * from medical_records;
```

record_id	patient_id	doctor_id	medical_date	diagnosis
1	1	1	2023-05-01	Heart disease
2	1	1	2023-05-02	covid 19
3	1	1	2023-05-02	covid 19
4	2	2	2023-05-02	Acne
5	2	2	2023-05-02	Heart disease
6	3	3	2023-05-03	Migraine
7	4	4	2023-05-04	Cataract
8	5	5	2023-05-05	Depression
9	6	6	2023-05-06	Asthma
10	7	7	2023-05-07	Flu
11	8	1	2023-05-08	High blood p…
12	9	2	2023-05-09	Eczema
13	10	3	2023-05-10	Alzheimer's…
NULL	NULL	NULL	NULL	NULL

```
select * from prescriptions;
```

prescription…	patient_id	doctor_id	prescription_date	medication
1	1	1	2023-05-01	Lipitor
2	2	2	2023-05-02	Retin-A
3	3	3	2023-05-03	Imitrex
4	4	4	2023-05-04	Lumigan
5	5	5	2023-05-05	Prozac
6	6	6	2023-05-06	Ventolin
7	7	7	2023-05-07	Tamiflu
8	8	1	2023-05-08	Norvasc
9	9	2	2023-05-09	Elidel
10	10	3	2023-05-10	Aricept
NULL	NULL	NULL	NULL	NULL

```
select * from wards;
```

ward_id	ward_name	capacity	occupied_beds	available_beds
1	Cardiology	20	10	10
2	Dermatology	30	20	10
3	Neurology	15	5	10
4	Ophthalmology	25	15	10
5	Psychiatry	10	5	5
6	Cardiology	20	10	10
7	Dermatology	30	20	10
8	Neurology	15	5	10
9	Ophthalmology	25	15	10
10	Psychiatry	10	5	5
NULL	NULL	NULL	NULL	NULL

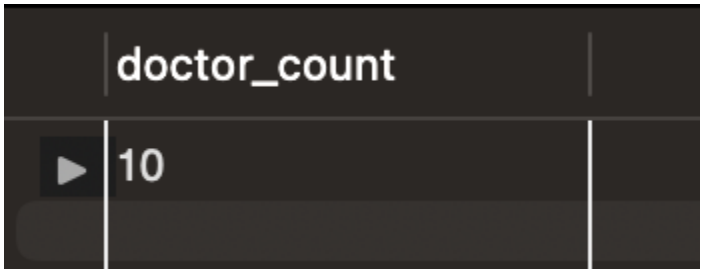
```
select * from admissions;
```

admission_id	patient_id	ward_id	admission_date	discharge_date
1	1	1	2023-05-01	2023-05-05
2	2	2	2023-05-02	2023-05-06
3	3	3	2023-05-03	2023-05-07
4	4	4	2023-05-04	2023-05-08
5	5	5	2023-05-05	2023-05-09
6	6	1	2023-05-06	2023-05-10
7	7	2	2023-05-07	2023-05-11
8	8	3	2023-05-08	2023-05-12
9	9	4	2023-05-09	2023-05-13
10	10	5	2023-05-10	2023-05-14
NULL	NULL	NULL	NULL	NULL

Example

1. show the number of doctors in hospital.

```
SELECT COUNT(*) AS doctor_count FROM doctors;
```



doctor_count
10

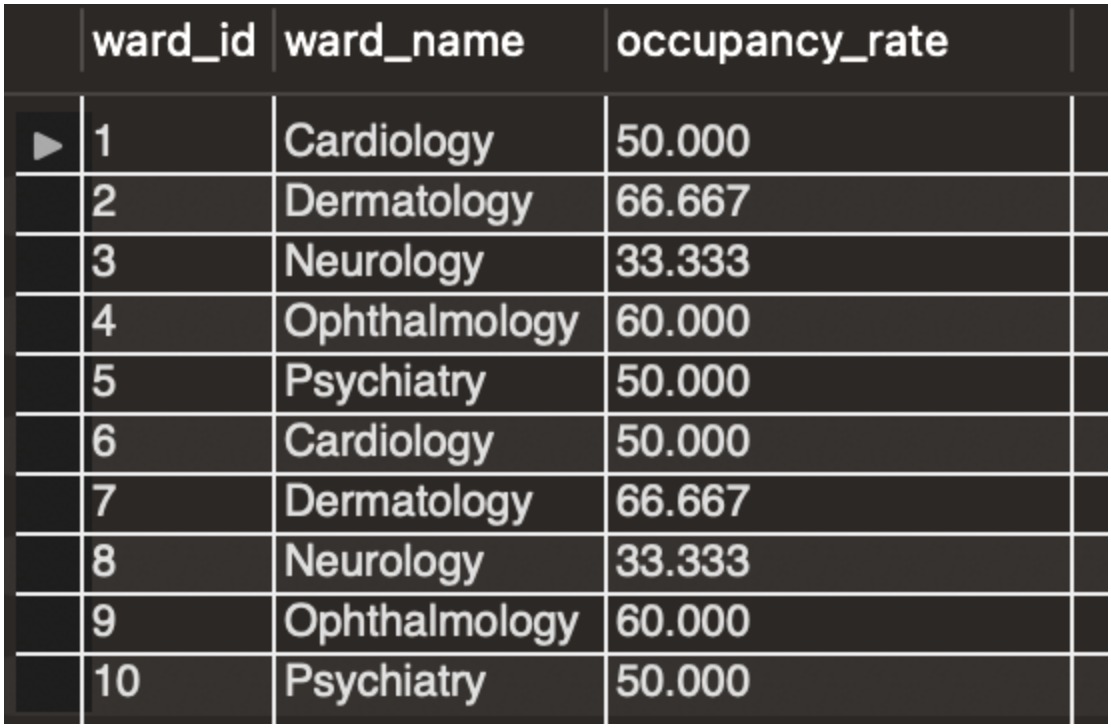
2. show the number of patients corresponding to each doctor.

```
SELECT doctors.doctor_id, doctors.first_name, doctors.last_name, COUNT(patient_id) as
patient_count
FROM doctors
LEFT OUTER JOIN appointments ON doctors.doctor_id = appointments.doctor_id
GROUP BY doctors.doctor_id;
```

doctor_id	first_name	last_name	patient_count
1	Michael	Smith	2
2	Jennifer	Lee	2
3	William	Brown	2
4	Elizabeth	Taylor	1
5	Robert	Johnson	1
6	Karen	Davis	1
7	Daniel	Wilson	1
8	manho	Wilson	0
9	jimmy	Wilson	0
10	Aston	Wilson	0

3. check the percentage rate of ward occupation of corresponding hospital departments, occupancy rate round off to 3 decimal places.

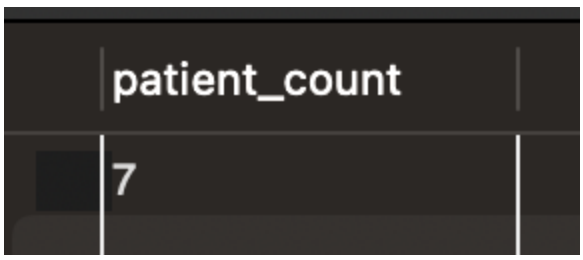
```
SELECT wards.ward_id, wards.ward_name, ROUND((occupied_beds / capacity) * 100, 3) AS
 occupancy_rate
FROM wards;
```



ward_id	ward_name	occupancy_rate
1	Cardiology	50.000
2	Dermatology	66.667
3	Neurology	33.333
4	Ophthalmology	60.000
5	Psychiatry	50.000
6	Cardiology	50.000
7	Dermatology	66.667
8	Neurology	33.333
9	Ophthalmology	60.000
10	Psychiatry	50.000

4. show the number of patients whose date of birth is before 1990.

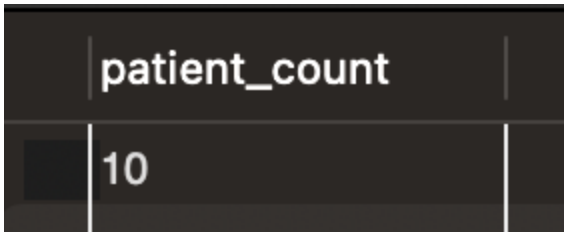
```
SELECT COUNT(*) AS patient_count
FROM patients
WHERE date_of_birth < '1990-01-01';
```



patient_count
7

5. show the number of patients whose medication are recorded.

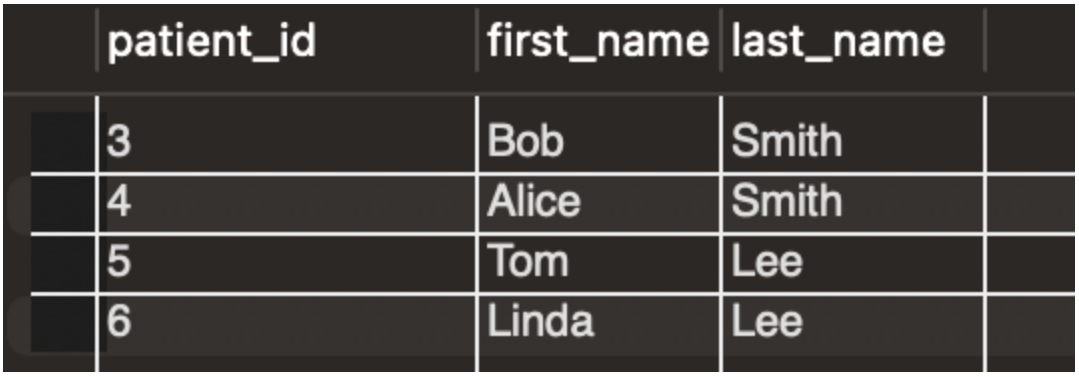
```
SELECT COUNT(DISTINCT patient_id) AS patient_count
FROM medical_records;
```



patient_count
10

6. Find the patients who enter hospital before 4th May 2023 and leave after 7th May 2023

```
SELECT DISTINCT patient_id, first_name, last_name
FROM admissions natural join patients
WHERE admission_date <= '2023-05-06' and discharge_date >= '2023-05-07';
```



patient_id	first_name	last_name
3	Bob	Smith
4	Alice	Smith
5	Tom	Lee
6	Linda	Lee

7. Find the most common diagnosis in hospital and number of occurrence.

```
SELECT diagnosis, diagnosis_count
from (SELECT diagnosis, COUNT(*) AS diagnosis_count
FROM medical_records
GROUP BY diagnosis) as cnt
where diagnosis_count > 0;
```

diagnosis	diagnosis_count
Heart disease	2
covid 19	2
Acne	1
Migraine	1
Cataract	1
Depression	1
Asthma	1
Flu	1
High blood pressure	1
Eczema	1
Alzheimer’s disease	1

8. show the information of patients who do not reserve hospital after March 2023 and do not have appointments records.

```
SELECT *
FROM patients
WHERE patients.patient_id NOT IN(
SELECT patient_id
FROM appointments
WHERE appointments.appointment_date > '2023-05-05'
);
```

patient_id	first_name	last_name	gender	date_of_birth
1	John	Doe	Male	1985-05-15
2	Jane	Doe	Female	1990-02-20
3	Bob	Smith	Male	1978-10-10
4	Alice	Smith	Female	1982-12-31
5	Tom	Lee	Male	1995-07-01
11	Jack	Johnson	Male	1998-08-08
12	Emily	Johnson	Female	1994-04-04
NULL	NULL	NULL	NULL	NULL

9. Show the number of prescriptions written after February 1, 2023 from the patient and prescription table, grouped by patient first and last name.

```
SELECT patients.first_name, patients.last_name, COUNT(*) AS prescription_count
FROM patients
JOIN prescriptions ON patients.patient_id = prescriptions.patient_id
WHERE prescriptions.prescription_date > '2023-02-01'
GROUP BY patients.first_name, patients.last_name;
// result    \  /
//.           \/
```

first_name	last_name	prescription_count
Alice	Smith	1
Bob	Smith	1
David	Brown	1
Jane	Doe	1
John	Doe	1
Linda	Lee	1
Lucy	Taylor	1
Mary	Brown	1
Peter	Taylor	1
Tom	Lee	1

10. Show patients from the patients table who do not have any appointments after May 5th, 2023.

```
SELECT *
FROM patients
WHERE patients.patient_id NOT IN(
    SELECT patient_id
    FROM appointments
    WHERE appointments.appointment_date > '2023-05-05'
    );
// result    \  /
//.           \/
```

←T→	patient_id	first_name	last_name	gender	date_of_birth
編輯 複製 刪除	1	John	Doe	Male	1985-05-15
編輯 複製 刪除	2	Jane	Doe	Female	1990-02-20
編輯 複製 刪除	3	Bob	Smith	Male	1978-10-10
編輯 複製 刪除	4	Alice	Smith	Female	1982-12-31
編輯 複製 刪除	5	Tom	Lee	Male	1995-07-01
編輯 複製 刪除	11	Jack	Johnson	Male	1998-08-08
編輯 複製 刪除	12	Emily	Johnson	Female	1994-04-04
編輯 複製 刪除	13	John	Doe	Male	1985-05-15
編輯 複製 刪除	14	Jane	Doe	Female	1990-02-20
編輯 複製 刪除	15	Bob	Smith	Male	1978-10-10
編輯 複製 刪除	16	Alice	Smith	Female	1982-12-31
編輯 複製 刪除	17	Tom	Lee	Male	1995-07-01
編輯 複製 刪除	18	Linda	Lee	Female	1988-03-25
編輯 複製 刪除	19	David	Brown	Male	1976-01-05
編輯 複製 刪除	20	Mary	Brown	Female	1980-09-17
編輯 複製 刪除	21	Peter	Taylor	Male	1992-11-22
編輯 複製 刪除	22	Lucy	Taylor	Female	1986-06-13
編輯 複製 刪除	23	Jack	Johnson	Male	1998-08-08
編輯 複製 刪除	24	Emily	Johnson	Female	1994-04-04