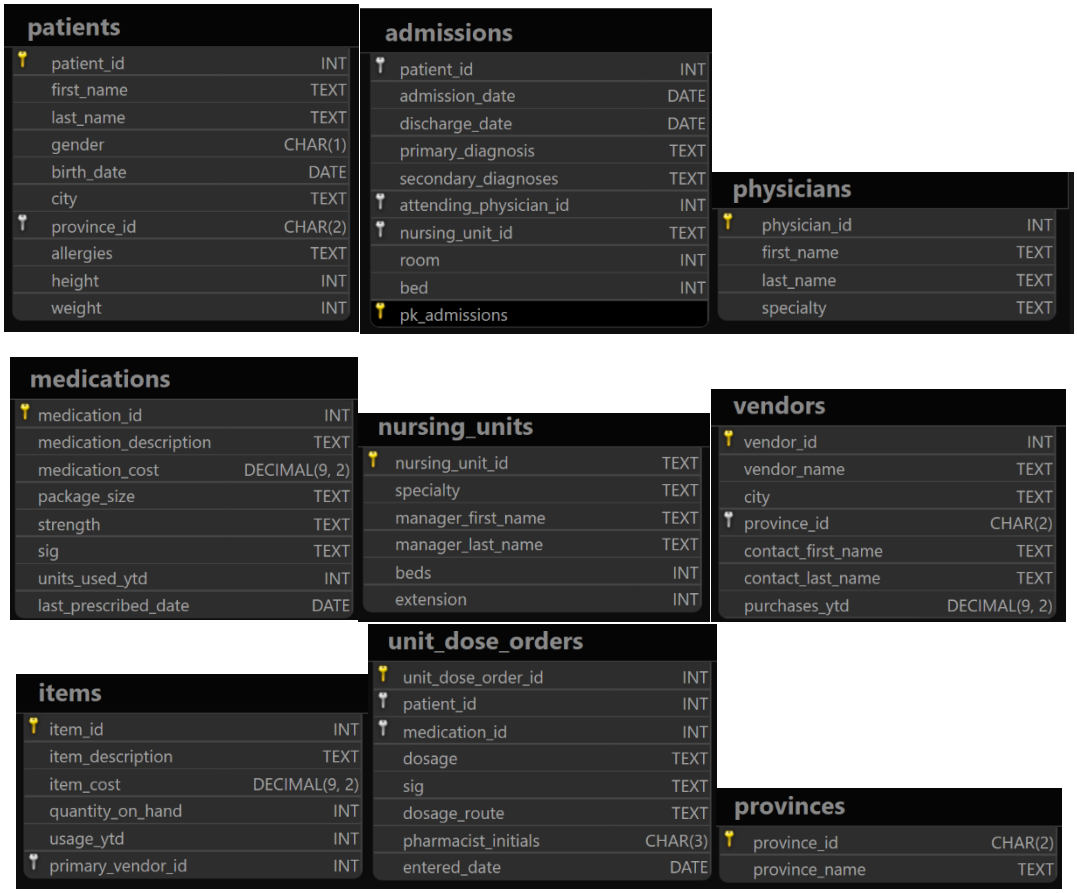
Sql-practice.com Solution

→Prepared by @balwant singh

req tables in the problems :-



Easy ->

Show first name, last name, and gender of patients who's gender is 'M'

```
select
first_name,last_name,gender
from patients
where gender = 'M'
```

Show first name and last name of patients who does not have allergies (null).

```
select
first_name,last_name
from patients
where allergies is null
```

Show first name of patients that start with the letter 'C'

```
Select
first_name
from patients
where first_name like 'C%'
```

Show first name and last name of patients that weight within the range of 100 to 120 (inclusive)

```
select
first_name,last_name
from patients
where weight between 100 and 120
```

Update the patients table for the allergies column. If the patient's allergies is null then replace it with 'NKA'

```
update patients
set allergies = 'NKA'
```

where patient_id in (select patient_id from patients where allergies is null)

Show first name and last name concatinated into one column to show their full name.

```
select concat(first_name," ",last_name) full_name
from patients
```

Show first name, last name, and the **full** province name of each patient.

Example: 'Ontario' instead of 'ON'

```
select
first_name,last_name,pr.province_name
from patients
join provinces pr on pr.province_id = patients.province_id;
```

Show how many patients have a birth_date with 2010 as the birth year.

```
select count(patient_id)
from patients
where birth_date like "2010%"
```

Show the first_name, last_name, and height of the patient with the greatest height.

```
select
first_name,last_name,height
from patients
where patient_id = (select patient_id from patients order by height desc limit 1)
```

Show all columns for patients who have one of the following patient_ids: 1,45,534,879,1000

```
select *

from patients

ywhere patient_id in (1,45,534,879,1000)
```

Show the total number of admissions

```
select

count(patient_id)

from admissions
```

Show all the columns from admissions where the patient was admitted and discharged on the same day.

```
select *

from admissions

where admission_date = discharge_date
```

Show the total number of admissions for patient_id 573.

```
select

patient_id, count(admission_date) as total_admissions

from admissions

where patient_id = 573 group by patient_id
```

Based on the cities that our patients live in, show unique cities that are in province_id 'NS'?

```
select
distinct city
from patients
where province_id = "NS"
```

Medium ->

Show unique birth years from patients and order them by ascending.

```
select
distinct year(birth_date)
from patients
order by birth_date
```

Show unique first names from the patients table which only occurs once in the list.

For example, if two or more people are named 'John' in the first_name column then don't include their name in the output list. If only 1 person is named 'Leo' then include them in the output.

```
select
first_name
from patients
group by first_name
having count(*)=1
```

Show patient_id and first_name from patients where their first_name start and ends with 's' and is at least 6 characters long.

```
select
patient_id,first_name
from patients
where first_name like 's____%s'
```

Show patient_id, first_name, last_name from patients whos primary_diagnosis is 'Dementia'.

Primary diagnosis is stored in the admissions table.

```
select
patients.patient_id,first_name,last_name
from patients
join admissions on patients.patient_id = admissions.patient_id
where primary_diagnosis = 'Dementia';
```

Display every patient's first_name.
Order the list by the length of each name and then by alphabetically

```
select
first_name
from patients
order by length(first_name),concat(first_name,last_name)
```

Show the total amount of male patients and the total amount of female patients in the patients table.
Display the two results in the same row.

```
select (select count(*) from patients where gender = "M") male, (select count(*) from patients where gender = "F")
```

Show first and last name, allergies from patients which have allergies to either 'Penicillin' or 'Morphine'. Show results ordered ascending by allergies then by first_name then by last_name.

```
select

first_name,last_name,allergies

from patients where allergies in ("Penicillin","Morphine")

order by allergies,first_name,last_name
```

Show patient_id, primary_diagnosis from admissions. Find patients admitted multiple times for the same primary_diagnosis.

```
select

patient_id,primary_diagnosis

from admissions

group by primary_diagnosis,patient_id

having count(patient_id)>1
```

Show the city and the total number of patients in the city in the order from most to least patients.

```
select

city,count(*)

from patients

group by city

order by count(*) desc
```

Show first name, last name and role of every person that is either patient or physician.
The roles are either "Patient" or "Physician"

```
select

first_name,last_name,'patient' role
```

```
from patients
union
select
first_name,last_name,'physician' as role
from physicians
```

Show all allergies ordered by popularity. Remove 'NKA' and NULL values from query.

```
SELECT
  allergies,
  COUNT(*) AS total_diagnosis
FROM patients
WHERE
  NOT allergies = 'NKA'
  AND allergies NOT NULL
GROUP BY allergies
ORDER BY total_diagnosis DESC
```

Show all patient's first_name, last_name, and birth_date who were born in the 1970s decade. Sort the list starting from the earliest birth_date.

```
SELECT
  first_name,
  last_name,
  birth_date
FROM patients
WHERE
  YEAR(birth_date) BETWEEN 1970 AND 1979
ORDER BY birth_date ASC;
```

We want to display each patient's full name in a single column. Their last_name in all upper letters must appear first, then first_name in all lower case letters. Separate the last_name and first_name with a comma. Order the list by the first_name in decending order
EX: SMITH,jane

```
select
concat(upper(last_name),",",lower(first_name))
from patients
order by first_name desc
```

Show the province_id(s), sum of height; where the total sum of its patient's height is greater than or equal to 7,000.

```
select
province_id,sum(height)
from patients
group by province_id
having sum(height)>=7000
```

Show the difference between the largest weight and smallest weight for patients with the last name 'Maroni'

```
select
max(weight)-min(weight)
from patients
where last_name = 'Maroni';
```

Show all of the month's day numbers and how many admission_dates occurred on that number. Sort by the day with most admissions to least admissions.

```
select

day(admission_date) , count(admission_date)

from admissions

group by day(admission_date)

order by count(admission_date) desc
```

Show the patient_id, nursing_unit_id, room, and bed for patient_id 542's most recent admission_date.

```
select

patient_id,nursing_unit_id,room,bed

from admissions

where patient_id = 542

order by admission_date desc

limit 1
```

Show the nursing_unit_id and count of admissions for each nursing_unit_id. Exclude the following nursing_unit_ids: 'CCU', 'OR', 'ICU', 'ER'.

```
select

nursing_unit_id , count()

from admissions

where nursing_unit_id not in ('CCU','OR','ICU','ER')

group by nursing_unit_id
```

Show patient_id, attending_physician_id, and primary_diagnosis for admissions that match one of the two criteria:
1. patient_id is an odd number and attending_physician_id is either 1, 5, or 19.

2. attending_physician_id contains a 2 and the length of patient_id is 3 characters.

```
select

patient_id,attending_physician_id,primary_diagnosis

from admissions

where (patient_id%2<>0 and attending_physician_id in (1,5,19)) or (attending_physician_id like "%2%" and len(patient_id) =3)
```

Hard ->

Show all of the patients grouped into weight groups.
Show the total amount of patients in each weight group.
Order the list by the weight group decending.

For example, if they weight 100 to 109 they are placed in the 100 weight group, 110-119 = 110 weight group, etc.

```
SELECT

FLOOR(t.weight/10) * 10 as weightGroup, count(*) as cnt

FROM patients t

GROUP BY FLOOR(t.weight/10)

ORDER BY FLOOR(t.weight/10) DESC
```

Show patient_id, weight, height, isObese from the patients table.

Display isObese as a boolean 0 or 1.

Obese is defined as weight(kg)/(height(m)^2) >= 30.

weight is in units kg.

height is in units cm.

```
select

patient_id,weight,height,
```

```
case
when (weight/power(height/100.,2))>=30 then 1
else 0
end
as isObese
from
patients
```

Show patient_id, first_name, last_name, and attending physician's specialty.
Show only the patients who has a primary_diagnosis as 'Dementia' and the physician's first name is 'Lisa'

Check patients, admissions, and physicians tables for required information.

```
select
patients.patient_id,patients.first_name,patients.last_name,ph.specialty
from patients
join admissions on admissions.patient_id = patients.patient_id
join physicians ph on ph.physician_id = admissions.attending_physician_id
where primary_diagnosis = "Dementia" and ph.first_name = "Lisa"
```

All patients who have gone through admissions, can see their medical documents on our site. Those patients are given a temporary password after their first admission. Show the patient_id and temp_password.

The password must be the following, in order:
1. patient_id
2. the numerical length of patient's last_name
3. year of patient's birth_date

```
select
distinct p.patient_id,concat(p.patient_id,len(p.last_name),year(p.birth_date))
from patients p
```

```
join
admissions a
on a.patient_id = p.patient_id
```

Each admission costs $50 for patients without insurance, and $10 for patients with insurance. All patients with an even patient_id have insurance.

Give each patient a 'Yes' if they have insurance, and a 'No' if they don't have insurance. Add up the admission_total cost for each has_insurance group.

```
select
case
when patient_id%2=0 then 'Yes'
when patient_id%2<>0 then 'No'
end
as has_insurance,
sum(
case
when patient_id%2=0 then 10
when patient_id%2<>0 then 50
end
 )
as cost_after_insurance
from admissions
group by has_insurance
```

Show the provinces that has more patients identified as 'M' than 'F'. Must only show full province_name

```
Select
p.province_name
```

```
from provinces p
join patients pa on pa.province_id = p.province_id
group by p.province_name
having count(case when gender = "M" then 1 end)>count(case when gender = "F" then 1 end)
```

We are looking for a specific patient. Pull all columns for the patient who matches the following criteria:
- First_name contains an 'r' after the first two letters.
- Identifies their gender as 'F'
- Born in February, May, or December
- Their weight would be between 60kg and 80kg
- Their patient_id is an odd number
- They are from the city 'Halifax'

```
select *
from patients
where first_name like "__r%" and gender="F"
and month(birth_date) in (02,05,12)
and weight between 60 and 80
and patient_id%2<>0
and city = 'Halifax'
```

Show the percent of patients that have 'M' as their gender. Round the answer to the nearest hundreth number and in percent form.

```
SELECT
CONCAT( ROUND((
    SELECT COUNT(*)
    FROM patients
    WHERE gender = 'M'
) / CAST(COUNT(*) as float),
4
```

```
    ) * 100,
    '%'
  ) as percent_of_male_patients
FROM patients;
```

Show the patient_id and total_spent for patients who spent over 150 in medication_cost. Sort by most total_spent to least total_spent.

```
select
patient_id,sum(medication_cost)
from medications m join unit_dose_orders u on  m.medication_id = u.medication_id
group by patient_id
having sum(medication_cost)>150
order by sum(medication_cost) desc
```

Provide the description of each item, along with the total cost of the quantity on hand (rounded to the nearest whole dollar), and the associated primary vendor. Sort the output by the most spent to the least spent on inventory.

```
SELECT
  i.item_description,
  ROUND(i.item_cost * i.quantity_on_hand, 0) as total_cost,
  v.vendor_name
FROM items i
  JOIN vendors v ON i.primary_vendor_id = v.vendor_id
GROUP BY i.item_description
ORDER BY total_cost DESC;
```