50 SQL Questions from https://www.sql-practice.com/ (Medium & Hard)

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

```
SELECT distinct(year(birth_date)) as year FROM patients
order by year(birth_date)
```

==========

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

```
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
  len(first_name) >= 6
  and first_name like 'S%'
  and first_name like '%s'
```

==========

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

```
select  p.patient_id, p.first_name, p.last_name
from patients as p
join admissions as a
on p.patient_id = a.patient_id
where a.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 len(first_name), first_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.

Method 1:-

```
select
count(*) as male_count,
(select count(*) as female_count from patients where gender = 'F') as female_count
from patients where gender = 'M'
```

Method 2:-

```
select
sum(case when gender = 'M' then 1 end) as male_count,
   sum(case when gender = 'F' then 1 end) as female_count
from patients
```

==========

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, diagnosis from admissions. Find patients admitted multiple times for the same diagnosis.

```
select patient_id,diagnosis from admissions
group by patient_id,diagnosis
having count(*) > 1
```

==========

Show the city and the total number of patients in the city.
Order from most to least patients and then by city name ascending.

```
select city, count(patient_id) as num_patients
from patients
group by city
order by count(patient_id) desc, city
```

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

```
select first_name,last_name,'Patient' as role from patients
union all
select first_name,last_name,'Doctor' as role from doctors
```

==========

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

```
select allergies, count(*) as total_diagnosis
from patients
where allergies is not null
group by allergies
order by count(*) 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
```

==========

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)) as new_name_format
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) as 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)) as weight_delta
from patients
where last_name = 'Maroni'
```

==========

Show all of the days of the month (1-31) and how many admission_dates occurred on that day. Sort by the day with most admissions to least admissions.

```
select day(admission_date) as daY_number,
count(*) as number_of_admissions
from admissions
group by day(admission_date)
order by count(*) desc
```

==========

Show all columns for patient_id 542's most recent admission_date.

Method 1:-

```
with cte as (
select
        *,
   row_number() over(partition by patient_id order by admission_date desc) as rn
from admissions
where patient_id = '542'
 )
select patient_id,admission_date,discharge_date,diagnosis,attending_doctor_id from cte
where rn = 1
```

Method 2:-

```
select * from admissions
where (patient_id = '542') and admission_date = (select max(admission_date) from
admissions where patient_id = '542')
```

Show patient_id, attending_doctor_id, and diagnosis for admissions that match one of the two criteria:
1. patient_id is an odd number and attending_doctor_id is either 1, 5, or 19.
2. attending_doctor_id contains a 2 and the length of patient_id is 3 characters.

```
select patient_id,attending_doctor_id,diagnosis from admissions
where
        (patient_id % 2 !=0 and attending_doctor_id in (1,5,19)) or
        (attending_doctor_id like '%2%' and len(patient_id) = 3)
```

==========

Show first_name, last_name, and the total number of admissions attended for each doctor.
Every admission has been attended by a doctor.

```
select
        d.first_name,
   d.last_name,
   count(*) as admissions_total
from doctors as d
join admissions as a
on d.doctor_id = a.attending_doctor_id
group by d.first_name,d.last_name
```

==========

For each doctor, display their id, full name, and the first and last admission date they attended.

```
select d.doctor_id,concat(d.first_name,' ',d.last_name) as full_name,
        min(a.admission_date) as first_admission_date,
   max(a.admission_date) as last_admission_date
from doctors as d
join admissions as a
on d.doctor_id = a.attending_doctor_id
group by d.doctor_id,concat(d.first_name,' ',d.last_name)
```

==========

Display the total amount of patients for each province. Order by descending.

```
select pr.province_name, count(*) as patient_count
from patients as p
join province_names as pr
on p.province_id = pr.province_id
group by pr.province_name
order by count(*) desc
```

==========

For every admission, display the patient's full name, their admission diagnosis, and their doctor's full name who diagnosed their problem.

```
select
        concat(p.first_name,' ',p.last_name) as patient_name,
         a.diagnosis as diagnosis,
        concat(d.first_name,' ',d.last_name) as doctor_name
from patients as p
join admissions as a
        on p.patient_id = a.patient_id
join doctors as d
        on d.doctor_id = a.attending_doctor_id
```

==========

display the first name, last name and number of duplicate patients based on their first name and last name.
Ex: A patient with an identical name can be considered a duplicate.

```
select first_name, last_name, count(*) as num_of_duplicates
from patients
group by first_name, last_name
having count(*) > 1
```

==========

Display patient's full name,
height in the units feet rounded to 1 decimal, weight in the unit pounds rounded to 0 decimals,birth_date,gender non abbreviated.
Convert CM to feet by dividing by 30.48.
Convert KG to pounds by multiplying by 2.205.

```
select
```

```
        concat(first_name,' ',last_name) as patient_name,
    round(height/30.48,1) as height_Feet,
    round(weight * 2.205,0) as weight_Pounds,
    birth_date,
    case
        WHEN gender = 'M' then 'MALE'
       WHEN gender = 'F' then 'FEMALE'
    end as gender_type
FROM patients
```

==========

Show patient_id, first_name, last_name from patients whose does not have any records in the admissions table. (Their patient_id does not exist in any admissions.patient_id rows.)

```
SELECT patient_id,first_name,last_name
FROM patients
WHERE patient_id NOT IN (SELECT distinct patient_id FROM admissions)
```

==========

Display a single row with max_visits, min_visits, average_visits where the maximum, minimum and average number of admissions per day is calculated. Average is rounded to 2 decimal places.

```
with cte as (
        SELECT
          admission_date,
          COUNT(*) AS visits
        FROM admissions
        GROUP BY admission_date
       )
select
        max(visits) as max_visits,
        min(visits) as min_visits,
        round(avg(visits),2) as average_visits
from cte
```

==========

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
        count(*) as patients_in_group,
        cast((weight/10) as int)*10 as weight_group
from patients
group by cast((weight/10) as int)*10
order by cast((weight/10) as int)*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
        when (weight/power((height/100),2)) is null then 0
    else 0
end as isobese
from patients
```

==========

Show patient_id, first_name, last_name, and attending doctor's specialty.
Show only the patients who has a diagnosis as 'Epilepsy' and the doctor's first name is 'Lisa'
Check patients, admissions, and doctors tables for required information.

```
select p.patient_id,p.first_name,p.last_name,d.specialty as attending_doctor_specality
from patients as p
join admissions as a
on p.patient_id = a.patient_id
join doctors as d
on d.doctor_id = a.attending_doctor_id
where a.diagnosis = 'Epilepsy' and d.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)) as
temp_password
from patients as p
join admissions as a
on p.patient_id = (select distinct a.patient_id from admissions)
```

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

==========

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

```
with cte as (
select pr.province_name,
        sum(case when p.gender = 'M' then 1 end) as male_count,
        sum(case when p.gender = 'F' then 1 end) as female_count
from patients as p
join province_names as pr
on p.province_id = pr.province_id
group by pr.province_name
)
select province_name from cte
where male_count > female_count
```

==========

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 'Kingston'

```
select * from patients
where
        first_name like '__r%' and
   gender = 'F' and
   month(birth_date) in (2,5,12) and
   (weight between 60 and 80) and
   patient_id % 2 != 0 and
   city = 'Kingston'
```

==========

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(cast(sum(case when gender = 'M' then 1 end) as float)*100/count(*),2),"%") as
perc
from patients
```

==========

For each day display the total amount of admissions on that day. Display the amount changed from the previous date.

```
select
        admission_date,
        count(*) as admission_day,
         count(*) - lag(count(*),1) over(order by admission_date) as admission_diff
from admissions
group by admission_date
```

==========

Sort the province names in ascending order in such a way that the province 'Ontario' is always on top.

```
select distinct province_name
from province_names
order by (case when province_name = 'Ontario' then 0 else 1 end)
```

We need a breakdown for the total amount of admissions each doctor has started each year. Show the doctor_id, doctor_full_name, specialty, year, total_admissions for that year.

```
select
d.doctor_id,
concat(d.first_name,' ',d.last_name) as doctor_name,
d.specialty as specialty,
year(a.admission_date) as selected_year,
count(*) as total_admissions
from doctors as d
join admissions as a
on d.doctor_id = a.attending_doctor_id
group by d.doctor_id,concat(d.first_name,' ',d.last_name),d.specialty,year(a.admission_date)
order by d.doctor_id,  year(a.admission_date)
```

==========