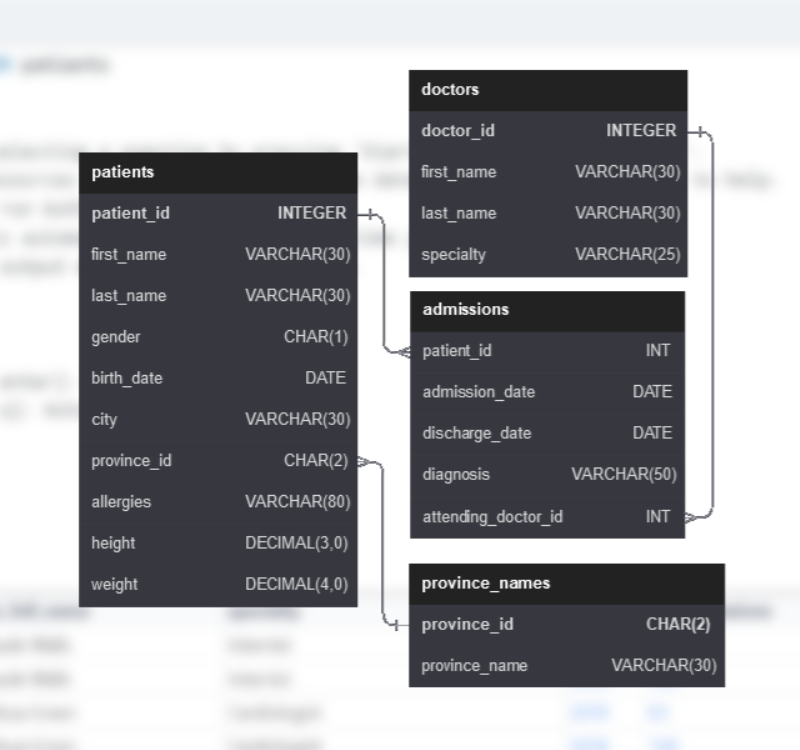
HOSPITAL SCHEMA



EASY Questions:

1. **Show first name, last name, and gender of patients whose gender is 'M':**

* SELECT first\_name, last\_name, gender

FROM patients

where gender='M';

1. **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;

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

* SELECT first\_name

FROM patients

where first\_name LIKE 'C%';

1. **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;

1. **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 allergies IS NULL;

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

* select concat(first\_name,' ',last\_name) AS full\_name from patients;

1. **Show first name, last name, and the full province name of each patient.  
     
   Example: 'Ontario' instead of 'ON'**

* select p.first\_name,p.last\_name,pr.province\_name

from patients p inner join province\_names pr

on p.province\_id=pr.province\_id;

1. **Show how many patients have a birth\_date with 2010 as the birth year.**

* select count(\*)

from patients

where year(birth\_date) = 2010;

1. **Show the first\_name, last\_name, and height of the patient with the greatest height.**

* select first\_name,last\_name, MAX(height)

from patients;

1. **Show all columns for patients who have one of the following patient\_ids:**1,45,534,879,1000

* select \*

from patients

where patient\_id IN (1,45,534,879,1000);

1. **Show the total number of admissions**

* select COUNT(\*)

from admissions;

1. **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;

1. **Show the patient id and the total number of admissions for patient\_id 579.**

* select patient\_id, count(patient\_id)

from admissions

where patient\_id=579;

1. **Based on the cities that our patients live in, show unique cities that are in province\_id 'NS'?**

* select distinct(city) as unique\_cities

from patients

where province\_id='NS';

1. **Write a query to find the first\_name, last name and birth date of patients who has height greater than 160 and weight greater than 70**

* select first\_name,last\_name, birth\_date

from patients

where height>160 and weight>70;

1. **Write a query to find list of patients first\_name, last\_name, and allergies where allergies are not null and are from the city of 'Hamilton'**

* select first\_name,last\_name, allergies

from patients

where allergies IS NOT NULL and city='Hamilton';

MEDIUM Questions:

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

* select distinct year(birth\_date)

from patients

order by year(birth\_date) asc;

1. **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 distinct first\_name

from patients

group by first\_name

having count(first\_name)=1;

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%s';

1. **Show patient\_id, first\_name, last\_name from patients whos diagnosis is 'Dementia'.  
   Primary diagnosis is stored in the admissions table.**

* select p.patient\_id, p.first\_name, p.last\_name

from patients p inner join admissions a

on p.patient\_id= a.patient\_id

where a.diagnosis='Dementia';

1. **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 asc;

1. **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(patient\_id) from patients where gender='M')as male\_count,

(select count(patient\_id) from patients where gender='F')as female\_count;

1. **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 ASC;

1. **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(diagnosis)>1;

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 no\_of\_patients

from patients

group by city

order by no\_of\_patients desc,city asc;

1. **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;

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

* select allergies, count(allergies) as total\_diagnosis

from patients

where allergies IS NOT null

group by allergies

order by total\_diagnosis desc;

1. **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;

1. **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 full\_name

from patients

order by first\_name desc;

1. **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 total\_height

from patients

group by province\_id

having total\_height >=7000;

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

* select MAX(weight)-min(weight) as diff

from patients

where last\_name ='Maroni';

1. **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 days, count(day(admission\_date)) as admissions

from admissions

group by day(admission\_date)

order by admissions desc;

1. **Show all columns for patient\_id 542's most recent admission\_date.**

* select \*

from admissions

where patient\_id=542 and admission\_date IN (

select admission\_date from admissions

where patient\_id=542

order by admission\_date desc

limit 1);

1. **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=1 and attending\_doctor\_id IN (1,5,19))

or (len(patient\_id)=3 and attending\_doctor\_id LIKE ('%2%'));

1. **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(a.attending\_doctor\_id)

from doctors d inner join admissions a

on d.doctor\_id=a.attending\_doctor\_id

group by doctor\_id;

1. **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 d inner join admissions a

on d.doctor\_id=a.attending\_doctor\_id

group by doctor\_id;

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

* select p.province\_name, COUNT(pa.patient\_id) as amt\_of\_patients

from patients pa inner join province\_names p

on pa.province\_id=p.province\_id

group by pa.province\_id

order by amt\_of\_patients DESC;

1. **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\_full\_name,

a.diagnosis,

concat(d.first\_name,' ',d.last\_name) as doctor\_name

from patients p inner join admissions a

on p.patient\_id=a.patient\_id

inner join doctors d

on a.attending\_doctor\_id=d.doctor\_id;

1. **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(concat(first\_name,' ', last\_name)) as duplicates

from patients

group by concat(first\_name,' ', last\_name)

having duplicates>1;

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 full\_name,

round(height/30.48,1) as height\_in\_feet,

round(weight\*2.205) as weight\_in\_pounds,

birth\_date,

(case

when gender='M' then 'Male'

Else 'Female'

END) AS patient\_gender

from patients;

1. **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 patient\_id from admissions);

HARD Questions:

1. **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(patient\_id) as no\_of\_patients,

floor(weight/10) \* 10 as weight\_group

from patients

group by weight\_group

order by weight\_group DESC;

1. **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/100.0,2)) >= 30 then 1

ELSE 0

END) AS isObese

from patients;

1. **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

from patients p inner join admissions a

on p.patient\_id=a.patient\_id

inner join doctors d

on a.attending\_doctor\_id=d.doctor\_id

where a.diagnosis= 'Epilepsy' and d.first\_name='Lisa';

1. **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(a.patient\_id,len(p.last\_name),year(p.birth\_date)) as temp\_password

from patients p inner join admissions a

on p.patient\_id=a.patient\_id;

1. **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.**

* with total\_cost as

(select patient\_id,

(case

when patient\_id%2=0 then 'Yes'

else 'No'

end) as has\_insurance

from admissions),

admission\_cost as

(select has\_insurance,

(case

when has\_insurance= 'Yes' then 10

else 50

end) as cost

from total\_cost)

select has\_insurance,SUM(cost) as cost\_after\_insurance

from admission\_cost

group by has\_insurance;

1. **Show the provinces that has more patients identified as 'M' than 'F'. Must only show full province\_name:**

* SELECT pr.province\_name

FROM patients pa JOIN province\_names pr

ON pa.province\_id = pr.province\_id

GROUP BY pr.province\_name

HAVING SUM(gender = 'M') > SUM(gender = 'F');

OR

* with gender\_count as

(

select province\_id,

sum(case

when gender='M' then 1

else 0

END) as male\_count,

sum(case

when gender='F' then 1

else 0

END) as female\_count

from patients

group by province\_id)

select pr.province\_name

from gender\_count gc inner join province\_names pr

on gc.province\_id=pr.province\_id

where gc.male\_count>gc.female\_count;

1. **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=1

and city='Kingston';

1. **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(sum(gender='M')/cast(count(\*) as float),4)\*100,'%') as percent

from patients;

OR

* SELECT concat(round(sum(gender='M')/cast(Count(\*) as float) \*100,2),'%')as per

from patients;

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

* select admission\_date,

count(patient\_id) as tot\_amt,

count(patient\_id) - LAG(Count(patient\_id)) OVER (order by admission\_date ASC) AS amt\_changed

from admissions

group by admission\_date;

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

* select province\_name

from province\_names

order by (case

when province\_name='Ontario' then '1'

else '2'

end) asc;

OR

* select province\_name

from province\_names

order by

(not province\_name = 'Ontario'),

province\_name ASC;

OR

* select province\_name

from province\_names

order by province\_name = 'Ontario' desc,

province\_name;

1. **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 doc\_full\_name,

d.specialty,

year(a.admission\_date) as year,

count(a.patient\_id) as tot\_admissions

from admissions a inner join doctors d

on a.attending\_doctor\_id=d.doctor\_id

group by year(a.admission\_date),d.doctor\_id

order by d.doctor\_id,year(a.admission\_date) desc;