1. 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'

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) Full\_name**

**from patients**

**SELECT first\_name || ' ' || last\_name**

**FROM patients;**

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

**SELECT first\_name, last\_name, province\_name**

**FROM patients**

**JOIN province\_names ON province\_names.province\_id = patients.province\_id;**

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

**select count(patient\_id) tot\_count**

**from patients**

**where year(birth\_date) =2010**

**SELECT count(first\_name) AS total\_patients**

**FROM patients**

**WHERE birth\_date >= '2010-01-01' AND birth\_date <= '2010-12-31'**

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

**select first\_name,last\_name,max(height) 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(patient\_id) as tot\_admissions**

**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) tot\_admissions**

**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**

**from patients**

**where province\_id = 'NS'**

**--------------------------------------------------------------------------------------------------------------**

**SELECT city**

**FROM patients**

**GROUP BY city**

**HAVING province\_id = 'NS'**

1. Write a query to find the first\_name, last name and birth date of patients who have height more than 160 and weight more 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 from Hamilton where allergies are not null ?

**select first\_name, last\_name, allergies**

**from patients**

**where city = 'Hamilton'and allergies is not null**

1. Based on cities where our patient lives in, write a query to display the list of unique city starting with a vowel (a, e, i, o, u). Show the result order in ascending by city. ?

**select distinct city**

**from patients**

**where city like 'a%' or city like 'e%' or city like 'i%' or city like 'o%' or city like 'u%'**

**order by city**

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

**select distinct year(birth\_date) as birth\_year**

**from patients**

**order by year(birth\_date)**

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 first\_name like 'S\_\_\_\_%S'**

**----------------------------------------------------------------------------------------------------------**

**SELECT patient\_id,first\_name**

**FROM patients**

**WHERE first\_name LIKE 's%s' AND len(first\_name) >= 6**

1. Show patient\_id, first\_name, last\_name from patients whos 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 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 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**

1. Show patient\_id, diagnosis from admissions. Find patients admitted multiple times for the same diagnosis.?

**select patient\_id, diagnosis**

**from admissions**

**group by diagnosis, patient\_id**

**having count(\*)>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 num\_patients**

**from patients**

**group by city**

**order by num\_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(\*) as tot\_diagnosis**

**from patients**

**where allergies is not null**

**group by allergies**

**order by tot\_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**

**----------------------------------------------------------------------------------------------------------------------**

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

**FROM patients**

**WHERE year(birth\_date) LIKE '197%'**

**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 FullName**

**from patients**

**order by first\_name desc**

**-----------------------------------------------------------------------------------------------------------------------**

**SELECT UPPER(last\_name) || ',' || LOWER(first\_name) AS new\_name\_format**

**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 sum\_height**

**from patients**

**group by province\_id**

**having sum\_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\_weight**

**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) days, count(\*) tot\_admissions**

**from admissions**

**group by days**

**order by tot\_admissions desc**

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

**select \***

**from admissions**

**where patient\_id = 542**

**group by patient\_id**

**having max(admission\_date)**

**SELECT \***

**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 ( attending\_doctor\_id in (1,5,19) and patient\_id % 2 != 0) or**

**(attending\_doctor\_id like '%2%' and len(patient\_id) = 3)**

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 first\_name, last\_name ,count(\*) as tot\_number**

**from admissions join doctors**

**on admissions.attending\_doctor\_id = doctors.doctor\_id**

**group by attending\_doctor\_id**

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

**select doctor\_id, concat(first\_name,' ', last\_name) as full\_name,**

**max(admission\_date) as first\_admission\_date,min(admission\_date) as last\_admission\_date**

**from admissions join doctors**

**on admissions.attending\_doctor\_id=doctors.doctor\_id**

**group by doctor\_id**

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

**select province\_name, count(patient\_id) as tot\_count**

**from patients join province\_names**

**on patients.province\_id = province\_names.province\_id**

**group by province\_name**

**order by tot\_count 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\_name, diagnosis,**

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

**from patients p join admissions a on p.patient\_id = a.patient\_id**

**join doctors d on d.doctor\_id = a.attending\_doctor\_id**

1. display the number of duplicate patients based on their first\_name and last\_name. ?

**select first\_name, last\_name, count(\*) as num\_of\_duplicates**

**from patients**

**group by first\_name, last\_name**

**having count(\*)>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 patient\_name,**

**round(height/30.48,1) as height,**

**round(weight\*2.205,0) as weight, birth\_date,**

**case when gender ='M' then 'MALE' ELSE 'FEMALE' END as Gender**

**from patients**

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 patients\_grp, floor(weight/10)\*10 as weight\_grp**

**from patients**

**group by weight\_grp**

**order by weight\_grp desc**

**---------------------------------------------------------------------------------------------------------------------------**

**SELECT count(patient\_id) as patients\_grp, weight - weight % 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,**

**weight / power(CAST(height AS float) / 100, 2) >= 30 AS obese**

**FROM patients**

**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, specialty**

**from patients p join admissions a on p.patient\_id = a. patient\_id**

**join doctors d on a.attending\_doctor\_id = d.doctor\_id**

**where 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(p.patient\_id, concat(len(last\_name),year(birth\_date))) as Temp\_Pswd**

**from patients p 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. ?

**select has\_insurance,sum(admission\_cost) as admission\_total**

**from**

**(**

**select patient\_id,**

**case when patient\_id % 2 = 0 then 'Yes' else 'No' end as has\_insurance,**

**case when patient\_id % 2 = 0 then 10 else 50 end as admission\_cost**

**from admissions**

**)**

**group by has\_insurance**

**select 'No' as has\_insurance, count(\*) \* 50 as cost**

**from admissions where patient\_id % 2 = 1 group by has\_insurance**

**union**

**select 'Yes' as has\_insurance, count(\*) \* 10 as cost**

**from admissions where patient\_id % 2 = 0 group by has\_insurance**

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

**select pn.province\_name**

**from province\_names pn join patients p**

**on pn.province\_id = p.province\_id**

**group by pn.province\_name**

**having count(case when gender = 'M' then 1 end) >**

**count(case when gender = 'F' then 1 end)**

**SELECT pr.province\_name**

**FROM patients AS pa**

**JOIN province\_names AS pr ON pa.province\_id = pr.province\_id**

**GROUP BY pr.province\_name**

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

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 round(100 \* avg(gender = 'M'), 2) || '%' AS percent\_of\_male\_patients**

**FROM patients;**

**--------------------------------------------------------------------------------------------------------------------**

**SELECT CONCAT(ROUND(SUM(gender='M') / CAST(COUNT(\*) AS float), 4) \* 100, '%')**

**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(admission\_date) as admission\_day,**

**count(admission\_date) - LAG(count(admission\_date)) OVER(ORDER BY admission\_date) AS admission\_count\_change**

**FROM admissions**

**group by admission\_date**

**------------------------------------------------------------------------------------------------------------------------**

**WITH admission\_counts\_table AS (**

**SELECT admission\_date, COUNT(patient\_id) AS admission\_count**

**FROM admissions**

**GROUP BY admission\_date**

**ORDER BY admission\_date DESC**

**)**

**select**

**admission\_date,**

**admission\_count,**

**admission\_count - LAG(admission\_count) OVER(ORDER BY admission\_date) AS admission\_count\_change**

**from admission\_counts\_table**

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 0 else 1 end),province\_name**

**-------------------------------------------------------------------------------------------------------------------**

**select province\_name**

**from province\_names**

**order by province\_name = 'Ontario' desc, province\_name**