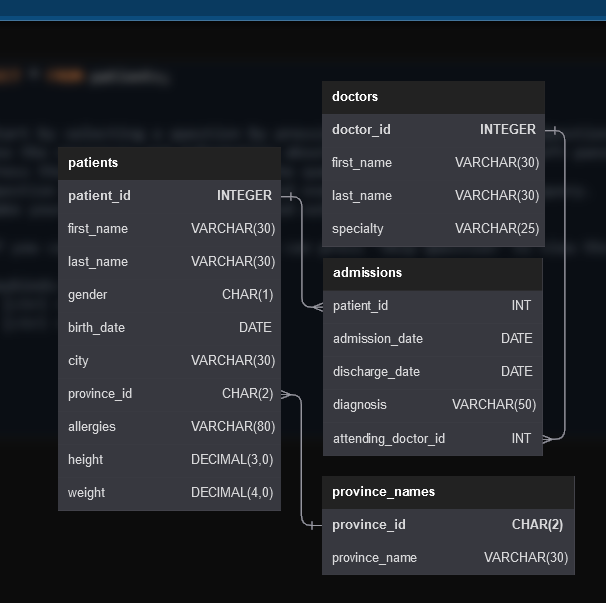
SQL PRACTICE WEBSITE QUESTIONS AND ANSWERS

SCHEMA INFORMATION

SCHEMA NAME : Hospital.db

TABLE NAMES : patients, admissions, doctors, province names



**EASY LEVEL**

QUESTION 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';

QUESTION 2:

**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;

QUESTION 3:

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

select first\_name from patients where first\_name like 'C%';

QUESTION 4:

**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;

QUESTION 5:

**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;

QUESTION 6:

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

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

QUESTION 7:

**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

where province\_names.province\_id = patients.province\_id;

QUESTION 8:

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

select count(\*) from patients

where year(birth\_date) = 2010;

QUESTION 9:

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

select first\_name,last\_name,height

from patients

where height = (select max(height) from patients);

QUESTION 10:

**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);

QUESTION 11:

**Show the total number of admissions.**

select count(\*) as TotalAdmissions from admissions;

QUESTION 12:

**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;

QUESTION 13:

**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;

QUESTION 14:

**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';

QUESTION 15:

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

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

from patients

where height > 160 AND weight > 70;

QUESTION 16:

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

select first\_name,last\_name,allergies

from patients

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

QUESTION 17:

**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 asc;

MEDIUM LEVEL

QUESTION 1:

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

select distinct year(birth\_date)

from patients

order by year(birth\_date);

QUESTION 2:

**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;

QUESTION 3:

**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'

and len(first\_name)>=6;

QUESTION 4:

**Show patient\_id, first\_name, last\_name from patients whose diagnosis is 'Dementia'.**

**Primary diagnosis is stored in the admissions table.**

select patient\_id,first\_name,last\_name

from patients

join admissions

using(patient\_id)

where admissions.diagnosis = 'Dementia';

QUESTION 5:

**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;

QUESTION 6:

**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') as Male,

(select count(\*) from patients where gender = 'F') as Female;

QUESTION 7:

**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 = 'Penicillin' or allergies = 'Morphine'

order by allergies,first\_name,last\_name ;

QUESTION 8:

**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;

QUESTION 9:

**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 NoOfPatients

from patients

group by city

order by 2 desc,city;

QUESTION 10:

**show first name, last name and role of every person that is either a 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;

QUESTION 11:

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

select allergies,count (allergies)

from patients

group by allergies

having allergies is not null

order by 2 desc;

QUESTION 12:

**Show all patients'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 3 ;

QUESTION 13:

**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;

QUESTION 14:

**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

order by 2;

QUESTION 15:

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

select (max(weight) - min(weight)) as WeightDifference

from patients

where last\_name = 'Maroni';

QUESTION 16:

**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 DaysNum,

count(admission\_date) as NoOfAdmissions

from admissions

group by day(admission\_date)

order by 2 desc;

QUESTION 17:

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

select \* from admissions

where patient\_id = 542

group by patient\_id

having admission\_date = max(admission\_date);

QUESTION 18:

**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

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

QUESTION 19:

**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.admission\_date) as TotalAdmissions

from doctors d join admissions a

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

group by attending\_doctor\_id;

QUESTION 20:

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

select d.doctor\_id,d.first\_name||' '||d.last\_name as FullNmae,

min(a.admission\_date) as firstAdmissionDate,

max(a.admission\_date) as lastAdmissionDate

from doctors d join admissions a

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

group by d.doctor\_id;

QUESTION 21:

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

select pn.province\_name,count(p.patient\_id) as TotalPatients

from patients p join province\_names pn

where p.province\_id = pn.province\_id

group by p.province\_id

order by TotalPatients desc;

QUESTION 22:

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

select p.first\_name||' '||p.last\_name as PatientName,

a.diagnosis,

d.first\_name||' '||d.last\_name as DoctorName

from admissions a

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

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

QUESTION 23:

**Display the number of duplicate patients based on their first\_name and last\_name.**

select first\_name,last\_name,count(\*) as duplicates

from patients

group by first\_name,last\_name

having count(first\_name and last\_name) > 1;

QUESTION 24:

**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

first\_name||' '||last\_name as FullName,

round(height/30.48,1) as 'height"feet"',

round(weight\*2.205,0) as 'weight"pounds"',

birth\_date,

case

when gender = 'M' then 'Male'

else 'Female'

end as gender

from patients;

**HARD LEVEL**

QUESTION 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 descending.**

**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;

QUESTION 2:

**Show patient\_id, weight, height, is Obese from the patients table.**

**Display is Obese 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\*.01),2)) >= 30

then 1 else 0

end as IsObese

from patients;

QUESTION 3:

**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 join admissions a

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

join doctors d

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

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

QUESTION 4:

**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 p join admissions a

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

QUESTION 5:

**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\_of\_Insurance

from admissions

group by has\_insurance;

QUESTION 6:

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

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

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

QUESTION 7:

**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 of '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';

QUESTION 8:

**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 '%\_of\_male\_patients'

from patients;

QUESTION 9:

**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(\*) - lag (count (\*)) over (order by admission\_date))

as admissions\_change

from admissions

group by admission\_date;

QUESTION 10:

**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 province\_name = 'Ontario' desc,

province\_name;