

## Ejercicios para practicar consultas SQL, Nivel Básico e Intermedio

1 - Show first name, last name, and gender of patients whose gender is 'M'

Muestre el primer nombre, el apellido y el género de los pacientes cuyo género es 'M'

```
SELECT
  first_name,
  last_name,
  gender
FROM patients
WHERE gender = 'M';
```

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

Muestre el primer nombre y el apellido de los pacientes que no tienen alergias. (null)

```
SELECT
  first_name,
  last_name
FROM patients
WHERE allergies IS NULL;
```

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

Muestre el primer nombre de los pacientes que comienzan con la letra 'C'

```
SELECT
  first_name
FROM patients
WHERE first_name LIKE 'C%';
```

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

Muestre el primer nombre y el apellido de los pacientes que pesan entre 100 y 120 (inclusive)

```
SELECT
  first_name,
  last_name
FROM patients
WHERE weight BETWEEN 100 AND 120;
```

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

Actualice la tabla de pacientes para la columna de alergias. Si las alergias del paciente son nulas, reemplázela con 'nka'

```
UPDATE patients
SET allergies = 'NKA'
WHERE allergies IS NULL;
```

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

Muestre el primer nombre y el apellido concatenados en una sola columna para mostrar su nombre completo.

```
SELECT
  CONCAT(first_name, ' ', last_name) AS full_name
FROM patients;
```

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

Muestre el primer nombre, el apellido y el nombre completo de cada provincia de cada paciente. Ejemplo: 'Ontario' en lugar de 'ON'

```
SELECT
  first_name,
  last_name,
  province_name
FROM patients
JOIN province_names ON province_names.province_id = patients.province_id;
```

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

Muestre cuántos pacientes tienen una fecha de nacimiento con el año 2010 como la fecha de nacimiento.

```
SELECT
  COUNT(*) AS total_patients
FROM patients
WHERE YEAR(birth_date) = 2010;
```

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

Muestre el primer\_nombre, el apellido y la altura del paciente con la mayor altura.

```
SELECT
  first_name,
  last_name,
  MAX(height) AS height
FROM patients;
```

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

Muestre todas las columnas para los pacientes que tienen uno de los siguientes patient\_ids:

1,45,534,879,1000.

```
SELECT *  
FROM patients  
WHERE patient_id IN (1, 9, 11, 175, 500);
```

11 - Show the total number of admissions.

Muestre el número total de admissions.

```
SELECT  
    COUNT(*) AS total_admissions  
FROM admissions;
```

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

Muestre todas las columnas de admissions donde el paciente fue admitido y liberado el mismo día.

```
SELECT *  
FROM admissions  
WHERE admission_date = discharge_date;
```

13 - Show the patient id and the total number of admissions for patient\_id 579.

Muestre el id del paciente y el número total de admissions para el paciente\_id 579.

```
SELECT  
    patient_id,  
    COUNT(*) AS total_admissions  
FROM admissions  
WHERE patient_id = 579;
```

14 - Based on the cities that our patients live in, show unique cities that are in province\_id 'NS'?

Basado en las ciudades en las que nuestros pacientes viven, muestre ciudades únicas que están en province\_id 'NS'?

```
SELECT DISTINCT city  
FROM patients  
WHERE province_id = 'NS';
```

15 - 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.

Escriba una consulta para encontrar el primer\_nombre, el apellido y la fecha de nacimiento de los pacientes que tienen una altura mayor que 160 y un peso mayor que 70.

```
SELECT
  first_name,
  last_name,
  birth_date
FROM patients
WHERE height > 160 AND weight > 70;
```

16 - Write a query to find list of patients first\_name, last\_name, and allergies from Hamilton where allergies are not null.

Escriba una consulta para encontrar la lista de pacientes first\_name, last\_name y alergias de Hamilton donde las alergias no son nulas.

```
SELECT
  first_name,
  last_name,
  allergies
FROM patients
WHERE city = 'Hamilton' AND allergies IS NOT NULL;
```

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

Muestre años de nacimiento únicos de pacientes y ordénelos de forma ascendente.

```
SELECT DISTINCT YEAR(birth_date) AS birth_year
FROM patients
ORDER BY birth_year ASC;
```

18 - 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.

Muestre nombres únicos de los pacientes de la tabla de pacientes que solo ocurren una vez en la lista. Por ejemplo, si dos o más personas se llaman 'John' en la columna first\_name entonces no incluya su nombre en la lista de salida. Si solo hay una persona llamada 'Leo' entonces incluya a ellos en la salida.

```
SELECT first_name
FROM patients
GROUP BY first_name
HAVING COUNT(*) = 1;
```

```
SELECT first_name
FROM (
  SELECT
```

```

        first_name,
        count(first_name) AS occurrences
    FROM patients
    GROUP BY first_name
)
WHERE occurrences = 1;

```

19 - Show patient\_id and first\_name from patients where their first\_name start and ends with 's' and is at least 6 characters long.

Muestre patient\_id y first\_name de pacientes donde su first\_name comienza y termina con 's' y es al menos de 6 caracteres de longitud.

```

SELECT
    patient_id,
    first_name
FROM patients
WHERE first_name LIKE 's%s' AND LEN(first_name) >= 6;

```

```

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 LENGTH(first_name) >= 6;

```

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

Mostrar paciente\_id, first\_name, último de los pacientes que el diagnóstico es 'demencia'. El diagnóstico principal se almacena en la tabla de admissions.

```

SELECT
    p.patient_id,
    p.first_name,
    p.last_name
FROM patients AS p
    JOIN admissions AS a ON a.patient_id = p.patient_id
WHERE a.diagnosis = 'Dementia';

```

21 - Display every patient's first\_name. Order the list by the length of each name and then by alphabetically. Muestre el primer\_name de cada paciente. Ordene la lista por la longitud de cada nombre y luego por alfabéticamente.

```
SELECT first_name
FROM patients
ORDER BY LEN(first_name), first_name;
```

22 - 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. En la tabla de pacientes, muestre el total de pacientes masculinos y el total de pacientes femeninos. Muestre los dos resultados en la misma fila.

```
SELECT
    SUM(Gender = 'M') AS male_count,
    SUM(Gender = 'F') AS female_count
FROM patients
```

23 - 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. Muestre el primer y apellido, las alergias de los pacientes que tienen alergias a 'penicilina' o 'morfina'. Mostrar resultados ordenados ascendentes por alergias luego por First\_Name y luego por Last\_Name.

```
SELECT
    first_name,
    last_name,
    allergies
FROM patients
WHERE allergies IN ('Penicillin', 'Morphine')
ORDER BY allergies, first_name, last_name;
```

24 - Show patient\_id, diagnosis from admissions. Find patients admitted multiple times for the same diagnosis. Mostrar paciente\_id, diagnóstico de admisiones. Encuentre pacientes admitidos varias veces para el mismo diagnóstico.

```
SELECT
    patient_id,
    diagnosis
FROM admissions
GROUP BY patient_id, diagnosis
HAVING COUNT(*) > 1;
```

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

Muestre la ciudad y el número total de pacientes en la ciudad. Ordene de más a menos pacientes y luego por nombre de ciudad ascendente.

```
SELECT
  city,
  COUNT(*) AS num_patients
FROM patients
GROUP BY city
ORDER BY num_patients DESC, city;
```

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

Muestre el nombre, el apellido y el papel de cada persona que sea paciente o médico. Los roles son "paciente" o "médico".

```
SELECT
  first_name, last_name, 'Patient' AS role FROM patients
UNION ALL
SELECT first_name, last_name, 'Doctor' AS role FROM doctors;
```

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

Mostrar todas las alergias ordenadas por popularidad. Elimine los valores NULL de la consulta.

```
SELECT allergies, COUNT(*) AS total_diagnosis
FROM patients
WHERE allergies IS NOT NULL
GROUP BY allergies
ORDER BY total_diagnosis DESC;
```

28 - Show all patient's first\_name, last\_name, and birth\_date who were born in the 1970s decade.

Mostrar todos los pacientes first\_name, last\_name y birth\_date que nacieron en los 1970s.

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

```
SELECT
  first_name,
```

```

    last_name,
    birth_date
FROM patients
WHERE birth_date >= '1970-01-01' AND birth_date < '1980-01-01';

```

```

SELECT
    first_name,
    last_name,
    birth_date
FROM patients
WHERE year(birth_date) LIKE '197%'
ORDER BY birth_date ASC

```

29 - 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

Queremos mostrar el nombre completo de cada paciente en una sola columna. Su último nombre en todas las letras superiores debe aparecer primero, luego primero en todas las letras minúsculas. Separe el Last\_Name y First\_Name con una coma. Ordene la lista por First\_Name en orden DiCending Ej: Smith, jane

```

SELECT
    CONCAT(UPPER(last_name), ',', LOWER(first_name)) AS new_name_format
FROM patients
ORDER BY first_name DESC;

```

No lo pide el enunciado pero queda mas prolijo.

```

SELECT
    CONCAT(
        UPPER(last_name), ', ',
        UPPER(LEFT(first_name, 1)),
        LOWER(SUBSTRING(first_name, 2, LEN(first_name) - 1))
    ) AS new_name_format
FROM patients
ORDER BY first_name DESC;

```

30 - 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.

Mostrar la provincia\_id (s), suma de altura; donde la suma total de la altura de su paciente es mayor o igual a 7,000.

```

SELECT
    province_id,
    SUM(height) AS total_height

```



```
FROM patients  
GROUP BY province_id  
HAVING SUM(height) >= 7000;
```