

Guía práctica de SQL – Carlos E. Cimino
Ejercicios resueltos

Data Manipulate Language (DML)

Selección de registros (world.sql)

Selección. Unicidad. Alias. Columnas calculadas

1. Listar código, nombre, continente y población de todos los países. (Se esperan 4 columnas y 239 registros).

```
SELECT code, name, continent, population
FROM country;
```

2. Listar solo el nombre de todos los lenguajes distintos que existan. (Se espera 1 columna y 457 registros).

```
SELECT distinct language FROM world.countrylanguage;
```

3. Listar nombre y población de cada ciudad, con los nombres de las columnas en castellano. (Se esperan 2 columnas y 4079 registros).

```
SELECT name as Ciudad, population as Población
FROM city;
```

4. Listar el nombre, el GNP como 'Producto Bruto Nacional', el GNPOld como 'Producto Bruto Nacional Anterior' y la diferencia entre estos como 'Diferencia', para todos los países. (Se esperan 4 columnas y 239 registros).

```
SELECT name as Nombre, GNP as 'Producto Bruto Nacional', GNPOld as
'Producto Bruto Nacional Anterior', GNP-GNPOld as Diferencia
FROM country;
```

5. Listar el nombre, la cantidad de habitantes, la superficie y una columna llamada 'Densidad' con el resultado de la densidad poblacional de todos los países. (Se esperan 4 columnas y 239 registros).

```
SELECT name as Nombre, population as Población, SurfaceArea as
Superficie, population/SurfaceArea as Densidad
FROM country;
```

Límites. Ordenamiento

6. Listar toda la información de los países, ordenados por población de manera ascendente. (Se esperan 15 columnas y 239 registros).

```
SELECT *
FROM country
ORDER BY population;
```

7. Listar nombre de los lenguajes en orden alfabético. (Se espera 1 columna y 984 registros).

```
SELECT language
FROM countrylanguage
ORDER BY language;
```

8. Listar nombre y cantidad de habitantes de las veinte ciudades menos pobladas. (Se esperan 2 columnas y 20 registros).

```
SELECT name, population
FROM city
ORDER BY population
LIMIT 20;
```

9. Listar código, nombre y año de independencia de todos los países, ordenados por antigüedad descendente. (Se esperan 3 columnas y 239 registros).

```
SELECT CODE, NAME, IndepYear
FROM country
ORDER BY IndepYear DESC;
```

10. Listar nombre y continente de los cien países con mayor expectativa de vida. Si hubiera países que tengan la misma expectativa de vida, mostrar primero a los de menor superficie. (Se esperan 2 columnas y 100 registros).

```
SELECT name, continent
FROM country
ORDER BY LifeExpectancy DESC, SurfaceArea ASC
LIMIT 100;
```

Filtrado de registros. Operadores relacionales y lógicos. Operadores IN y BETWEEN.

11. Listar todos los datos de los países que no cuenten con habitantes. (Se esperan 15 columnas y 7 registros).

```
SELECT *
FROM country
WHERE population = 0;
```

12. Listar todos los datos de los países cuya expectativa de vida supere los setenta y cinco años, ordenados bajo este concepto de forma ascendente. (Se esperan 15 columnas y 62 registros).

```
SELECT *
FROM country
WHERE country.LifeExpectancy > 65
ORDER BY country.LifeExpectancy;
```

13. Listar todos los datos de los países cuya independencia haya ocurrido a partir de la segunda mitad del siglo XIX y su forma de gobierno sea una monarquía constitucional. (Se esperan 15 columnas y 20 registros).

```
SELECT *
FROM country
WHERE GovernmentForm = 'Constitutional Monarchy' and IndepYear >=
1850;
```

14. Listar todos los datos de los diez países europeos de mayor PBN. (Se esperan 15 columnas y 10 registros).

```
SELECT *
FROM country
WHERE Continent = 'Europe'
ORDER BY GNP DESC
LIMIT 10;
```

15. Listar todos los datos de los países cuyo nombre registrado coincida con su nombre local. (Se esperan 15 columnas y 104 registros).

```
SELECT *
FROM country
WHERE name = LocalName;
```

16. Listar todos los datos de los países cuya independencia se haya dado a partir de la segunda mitad del siglo XX. (Se esperan 15 columnas y 110 registros).

```
SELECT *
FROM country
WHERE IndepYear >= 1950;
```

17. Listar todos los datos de los países situados en Europa, Asia o Sudamérica. (Se esperan 15 columnas y 111 registros).

```
SELECT *
FROM country
WHERE Continent IN ('Europe', 'Asia', 'South America');
```

18. Listar todos los datos de todos los países, excepto los africanos. (Se esperan 15 columnas y 181 registros).

```
SELECT *
FROM country
WHERE Continent <> 'Africa';
```

19. Listar todos los datos de las ciudades argentinas fuera de la provincia de Buenos Aires. (Se esperan 5 columnas y 26 registros).

```
SELECT *
FROM city
WHERE CountryCode = 'ARG' AND District <> 'Buenos Aires';
```

20. Listar todos los datos de las ciudades de entre 125 mil y 130 mil habitantes. (Se esperan 5 columnas y 138 registros).

```
SELECT *
FROM city
WHERE city.Population BETWEEN 125000 AND 130000;
```

Manejo de valores nulos. Operador LIKE. Comodines.

21. Listar todos los datos de los países donde no se cuente con datos acerca de su independencia. (Se esperan 15 columnas y 47 registros).

```
SELECT *
FROM country
WHERE IndepYear IS NULL;
```

22. Listar todos los datos de los países donde no se tengan datos acerca del PBN anterior ni de la expectativa de vida. (Se esperan 15 columnas y 17 registros).

```
SELECT *
FROM country
WHERE LifeExpectancy IS NULL AND GNPold IS NULL;
```

23. Listar todos los datos de los países cuyo nombre comience y termine con 'A'. (Se esperan 15 columnas y 13 registros).

```
SELECT *
FROM country
WHERE Name LIKE 'A%A';
```

24. Listar todos los datos de los países cuyo nombre sea compuesto (más de una palabra). (Se esperan 15 columnas y 66 registros).

```
SELECT *
FROM country
WHERE Name LIKE '% %';
```

25. Listar todos los datos de las ciudades cuyo nombre o distrito contengan un '-'. (Se esperan 5 columnas y 372 registros).

```
SELECT *
FROM city
WHERE city.Name LIKE '%-%' OR city.District LIKE '%-%';
```

Unión de tablas

26. Listar los nombres de los países sudamericanos junto a los nombres (alias 'Capital') de sus capitales. (Se esperan 2 columnas y 14 registros).

```
SELECT country.Name, city.Name AS 'Capital'
FROM country
INNER JOIN city ON country.Capital = city.ID
WHERE country.Continent = 'South America';
```

27. Listar el código de país, junto a los nombres de las ciudades y su cantidad de habitantes, de aquellos cuya expectativa de vida sea mayor a 80. (Se esperan 3 columnas y 253 registros).

```
SELECT country.Code AS CodPais, city.Name AS Ciudad,
city.Population AS CantHabitantes
FROM country
INNER JOIN city ON city.CountryCode = country.Code
WHERE LifeExpectancy > 80;
```

28. Listar las capitales de los países cuya forma de gobierno sea una República Federal. (Se esperan 2 columnas y 15 registros).

```
SELECT co.Name AS Pais, ci.Name AS Capital
FROM country AS co
INNER JOIN city ci ON co.Capital = ci.ID
WHERE GovernmentForm LIKE 'Federal Republic';
```

29. Listar los lenguajes oficiales, junto al nombre de sus respectivos países, donde la cantidad de habitantes de dicho país esté entre un millón y tres millones. (Se esperan 2 columnas y 14 registros).

```
SELECT Name AS Pais, Language AS Idioma
FROM country
INNER JOIN countrylanguage ON country.Code =
countrylanguage.CountryCode
WHERE Population BETWEEN 1000000 AND 3000000 AND IsOfficial LIKE
'T';
```

30. Listar los códigos, los nombres locales y la región a la que pertenecen aquellos países donde se hable español. (Se esperan 3 columnas y 28 registros).

```
SELECT code, localname, region
FROM country INNER JOIN countrylanguage ON
country.code=countrylanguage.countrycode
WHERE language="Spanish";
```

31. Listar los nombres y las capitales de los países en cuya capital se concentre más de la mitad de su población total. (Se esperan 2 columnas y 14 registros).

```
SELECT co.Name AS Pais, ci.Name AS Capital, ci.Population AS
PobCapital, co.Population AS PobPais
FROM country co
INNER JOIN city ci ON co.Capital = ci.ID
WHERE ci.Population > (co.Population / 2);
```

32. Listar los nombres y la superficie de los países africanos cuya capital coincida con el nombre del distrito a la que pertenece. (Se esperan 2 columnas y 32 registros).

```
SELECT co.Name AS Pais, co.SurfaceArea AS Superficie, ci.Name,
ci.District
FROM country co
INNER JOIN city ci ON co.Capital = ci.ID
WHERE co.Continent LIKE 'Africa' AND ci.Name = ci.District;
```

33. Listar los nombres, las capitales y el año de independencia (sin nulos) de los 20 países más antiguos. (Se esperan 3 columnas y 20 registros).

```
SELECT co.Name AS Pais, ci.Name AS Capital, co.IndepYear
FROM country co
INNER JOIN city ci ON co.Capital = ci.ID
WHERE co.IndepYear IS NOT NULL
ORDER BY co.IndepYear
LIMIT 20;
```

34. Listar las ciudades junto a sus idiomas oficiales, donde no se hable español, inglés, portugués, italiano, francés o alemán de manera oficial. (Se esperan 2 columnas y 2694 registros).

```
SELECT ci.Name AS Ciudad, cl.Language AS "Idioma Oficial"
FROM city ci
INNER JOIN countrylanguage cl ON ci.CountryCode = cl.CountryCode
WHERE cl.IsOfficial LIKE 'T' AND cl.Language NOT IN ('Spanish' ,
'English', 'Portuguese', 'Italian', 'French', 'German');
```

35. Listar nombre, población y país de las diez ciudades europeas de habla inglesa más pobladas. (Se esperan 3 columnas y 10 registros).

```
SELECT city.name AS City, city.population, country.name As Country
FROM city
INNER JOIN countrylanguage ON code=countrylanguage.CountryCode
INNER JOIN city ON code=city.CountryCode
WHERE Language="English" AND continent="Europe"
ORDER BY city.population DESC
LIMIT 10;
```

Funciones de agregación. Agrupamiento.

36. Mostrar según la tabla de países, la cantidad total de población, la población máxima, la población mínima, el promedio de población y con cuántos registros de población se cuenta. (Se esperan 5 columnas y 1 registro).

```
SELECT sum(population) AS Total,
max(population) AS Máximo, min(population) AS Mínimo,
AVG(population) AS Promedio, COUNT(population) AS 'Cantidad de
registros'
FROM city;
```

37. Mostrar según la tabla de países, la cantidad total de población, la población máxima, la población mínima y el promedio de población, por cada continente. (Se esperan 5 columnas y 7 registros).

```
SELECT Continent,
sum(population) AS Total, max(population) AS Máximo,
min(population) AS Mínimo, AVG(population) AS Promedio
FROM country
GROUP BY Continent;
```

38. Agrupar a todos los países según el continente al que pertenecen. Mostrar los continentes junto a la cantidad de naciones que pertenecen a cada uno. (Se esperan 2 columnas y 7 registros).

```
SELECT Continent AS Continente, count(*) AS 'Cant. Naciones'
FROM country
GROUP BY Continent;
```

39. Agrupar a todas las ciudades según el país al que pertenecen. Mostrar los códigos de países junto a la sumatoria total de habitantes de cada uno. (Se esperan 2 columnas y 232 registros).

```
SELECT Code AS Pais, sum(Population) AS 'Total de habitantes'
FROM country
GROUP BY Name
HAVING sum(Population)>0;
```

40. Agrupar a todos los lenguajes según su nombre. Mostrar los nombres de los lenguajes junto al porcentaje de habla mínimo registrado para cada uno. (Se esperan 2 columnas y 457 registros).

```
SELECT Language, Min(Percentage)
FROM CountryLanguage
GROUP BY Language;
```

41. Mostrar las distintas formas de gobierno posibles de los países europeos junto a su correspondiente promedio de población que vive bajo estas circunstancias. (Se esperan 2 columnas y 10 registros).

```
SELECT GovernmentForm, AVG(Population), Continent
FROM country
GROUP BY GovernmentForm
HAVING Continent='Europe';
```

42. Mostrar las diez regiones de mayor expectativa de vida promedio. (Se esperan 2 columnas y 10 registros).

```
SELECT Region, AVG(LifeExpectancy)
FROM country
GROUP BY Region
ORDER BY AVG(LifeExpectancy) DESC
LIMIT 10;
```

43. Mostrar los nombres de los diez distritos de mayor cantidad de ciudades con cantidad de habitantes mayor al medio millón, junto a la cantidad de ciudades. (Se esperan 3 columnas y 10 registros).

```
SELECT District AS 'Distrito', count(District) as 'Cant. Ciudades',
Population
FROM city
GROUP BY District
HAVING Population > 500000
ORDER BY count(District) DESC
LIMIT 10;
```

44. Mostrar los nombres de los países que tengan ciudades en el Caribe, junto al número de las mismas por país. (Se esperan 2 columnas y 24 registros).

```
SELECT co.Name, count(ci.Name) AS 'Cant. Ciudades', co.Region
FROM country co
INNER JOIN city ci ON co.Code = ci.CountryCode
GROUP BY co.Name
HAVING co.Region = 'Caribbean';
```

45. Mostrar los lenguajes existentes junto a la cantidad de países que lo hablan de manera oficial. (Se esperan 2 columnas y 102 registros).

```
SELECT cl.Language AS Lenguaje, COUNT(*) AS 'Cant. paises'
FROM country co
INNER JOIN countrylanguage cl ON co.Code = cl.CountryCode
WHERE IsOfficial LIKE 'T'
GROUP BY cl.Language;
```

Filtrado de grupos.

46. Mostrar listados los años de independencia (sin nulos) junto a la cantidad de países que la hayan conseguido en ese año. Se desea visualizar aquellos años donde más de un país se haya independizado. (Se esperan 2 columnas y 39 registros).

```
SELECT IndepYear, COUNT(IndepYear) as Cantidad
FROM country
GROUP BY IndepYear
HAVING cantidad >= 1;
```

47. Listar los países junto a la cantidad de idiomas diferentes hablados, pero solo aquellos donde se hablen entre tres y cinco idiomas diferentes. (Se esperan 2 columnas y 80 registros).

```
SELECT CountryCode, count(CountryCode) AS 'Cantidad de idiomas'
FROM countrylanguage
GROUP BY CountryCode
HAVING count(CountryCode) BETWEEN 3 AND 5;
```

48. Mostrar los distritos, junto al nombre del país al que pertenecen, cuya población total (también listada) no supere los diez mil habitantes. (Se esperan 3 columnas y 35 registros).

```
SELECT District, co.Name AS Pais, SUM(ci.Population) AS Poblacion
FROM city ci
INNER JOIN country co ON ci.CountryCode = co.Code
GROUP BY co.Name , District
HAVING SUM(ci.Population) < 10000;
```

49. Mostrar las regiones junto a su densidad poblacional promedio, donde ésta supere a la mitad de la densidad poblacional máxima. (Se esperan 2 columnas y 3 registros).

```
SELECT Region, AVG(Population / SurfaceArea) AS Densidad
FROM country
GROUP BY Region
HAVING AVG(Population / SurfaceArea) > MAX(Population /
SurfaceArea) / 2
ORDER BY Region;
```

50. Mostrar los lenguajes oficiales junto a su porcentaje promedio de habla, cuyo promedio no supere un dígito entero. (Se esperan 2 columnas y 7 registros).

```
SELECT cl.Language, AVG(cl.Percentage)
FROM countrylanguage AS cl
WHERE cl.IsOfficial = 'T'
GROUP BY cl.Language
HAVING AVG(cl.Percentage) < 10;
```


Inserción de registros (inupde.sql)

51. Insertar un nuevo cliente con los siguientes datos: '12169851', 'Luis Enrique', 'Pérez', 'HOMBRE', '1954-06-01', 'perez_luisenrique@ma1l.com', '1157319468', 'Pujol 416', 'Caballito', 'Uruguay'

```
INSERT INTO `inupde`.`clientes`  
(`dni`, `nombre`, `apellido`, `genero`, `fecha_nacimiento`, `mail`,  
`movil`, `domicilio`, `barrio`, `nacionalidad`)  
VALUES  
( '12169851', 'Luis Enrique', 'Pérez', 'HOMBRE', '1954-06-01',  
'perez_luisenrique@ma1l.com', '1157319468', 'Pujol 416',  
'Caballito', 'Uruguay');
```

52. Insertar los siguientes clientes en una sola consulta:

DNI	Nombre	Apellido	Género	Fecha de nacimiento	Mail	Móvil	Dirección	Barrio	Nacido en
17278290	Florencia	Santos	MUJER	1963-07-12	flor.santos@ma1l.com	1148434346	Piran 3015	Villa Urquiza	Argentina
16839283	Luisa Nina	Escudero	MUJER	1962-03-22	lulu62@hushma1l.com	1175940343	Superi 348 1º A	Colegiales	Colombia
19164274	Carlos Alberto	Medina	HOMBRE	1966-04-26	medinacarlosalberto@z0h0.com	1123423451	Mexico 2121	Balvanera	Argentina

```
INSERT INTO inupde.clientes(dni ,nombre, apellido, genero,  
Fecha_nacimiento, mail, movil, domicilio, barrio, nacionalidad)  
VALUES('17278290', 'Florencia', 'Santos', 'MUJER', '1963-07-12',  
'flor.santos@ma1l.com', '1148434346', 'Piran 3015', 'Villa  
Urquiza', 'Argentina'),  
( '16839283', 'Luisa Nina', 'Escudero', 'MUJER', '1962-03-  
22', 'lulu62@hushma1l.com', '1175940343', 'Superi 348 1º  
A', 'Colegiales', 'Colombia'),  
( '19164274', 'Carlos Alberto', 'Medina', 'HOMBRE', '1966-04-  
26', 'medinacarlosalberto@z0h0.com', '1123423451', 'Mexico  
2121', 'Balvanera', 'Argentina');
```

53. Insertar una cliente llamada María Sol González, con DNI 36.427.254, nacida en Argentina el 15 de agosto de 1992. Domiciliada en Gurruchaga 243, Villa Crespo. Su móvil es 136782013 y su mail es ma.gon54@gmail.com.

```
INSERT INTO inupde.clientes(dni ,nombre, apellido, genero,  
Fecha_nacimiento, mail, movil, domicilio, barrio, nacionalidad)  
VALUES('36.427.254', 'María Sol', 'González', 'MUJER', '1992-08-  
15', 'ma.gon54@gmail.com', '1136782013', 'Gurruchaga 243', 'Villa  
Crespo', 'Argentina');
```

Modificación de registros

54. Modificar el apellido de la cliente 85 por 'Chaves'. (Se espera 1 registro afectado).

```
UPDATE `inupde`.`clientes`  
SET `apellido`='Chaves'  
WHERE `id_cliente`='85';
```

55. Agregar un cero a la izquierda de todos los DNI con exactamente siete cifras. (Se esperan 1173 registros afectados).

```
UPDATE `inupde`.`clientes`  
SET clientes.dni = CONCAT('0', clientes.dni)  
WHERE length(clientes.dni)=7;
```

56. Agregar una tilde a todos los clientes llamados 'Nicolas' como único nombre para que el mismo pase a ser 'Nicolás'. (Se esperan 3 registros afectados).

```
UPDATE `inupde`.`clientes`  
SET `nombre`='Nicolás'  
WHERE `nombre`='Nicolas';
```

57. Poner en mayúsculas los apellidos de los clientes cuya nacionalidad no sea 'Argentina'. (Se esperan 4450 registros afectados).

```
UPDATE `inupde`.`clientes`  
SET apellido=upper(apellido)  
WHERE nacionalidad NOT LIKE 'Argentina';
```

Eliminación de registros

58. Eliminar al cliente 9804. (Se espera 1 registro afectado).

```
DELETE FROM `inupde`.`clientes`  
WHERE id_cliente=9804;
```

59. Eliminar a los clientes cuyo servidor de mail sea 'z0h0'. (Se esperan 2127 registros afectados).

```
DELETE FROM `inupde`.`clientes`  
WHERE mail LIKE '%@z0h0%';
```

60. Eliminar a todos los clientes que residan en el barrio de 'Flores'. (Se esperan 803 registros afectados).

```
DELETE FROM `inupde`.`clientes`  
WHERE barrio LIKE 'Flores';
```