

MODELOS Y BASES DE DATOS

Guía autoestudio 1/ 6

INVESTIGACIÓN

A.SQL ¿Qué es?

El Lenguaje de Consulta Estructurado popularmente conocido por sus siglas en inglés como SQL, es un tipo de lenguaje de programación que ayuda a solucionar problemas específicos o relacionados con la definición, manipulación e integridad de la información representada por los datos que se almacenan en las bases de datos.

Bibliografía : <https://styde.net/que-es-y-para-que-sirve-sql/>

¿Para qué sirve?

SQL es un lenguaje declarativo estándar internacional de comunicación dentro de las bases de datos que nos permite a todos el acceso y manipulación de datos en una base de datos, y además se puede integrar a lenguajes de programación, por ejemplo, ASP o PHP, y en combinación con cualquier base de datos específica, por ejemplo, MySQL, SQL Server, MS Access, entre otras.

Bibliografía : <https://devcode.la/blog/que-es-sql/>

¿Qué es DML, DLL,DCL,TCL?

Lenguaje de Definición de Datos (DDL)

Es un lenguaje de programación para definir estructuras de datos, proporcionado por los sistemas gestores de bases de datos. En inglés, Data Definition Language, de ahí sus siglas DDL.

Lenguaje de Manipulación de Datos (DML)

También es un lenguaje proporcionado por los sistemas gestores de bases de datos. En inglés, Data Manipulation Language (DML). Utilizando instrucciones de SQL, permite a los usuarios introducir datos para posteriormente realizar tareas de consultas o modificación de los datos que contienen las Bases de Datos.

Lenguaje de Control de Datos (DCL)

Es un lenguaje que incluye una serie de comandos SQL. Como los anteriores, es proporcionado por los sistemas gestores de bases de datos. Sus siglas son DCL por su nombre en inglés, Data Control Language. Estos comandos permiten al Administrador del sistema gestor de base de datos, controlar el acceso a los objetos, es decir, podemos otorgar o denegar permisos a uno o más roles para realizar determinadas tareas.

CONTROL DE TRANSACCIÓN (TCL)

Es un lenguaje de programación y un subconjunto de SQL , que se utiliza para controlar el procesamiento de transacciones en una base de datos.

Bibliografía : <https://todopostgresql.com/diferencias-entre-ddl-dml-y-dcl/>

En este laboratorio, ¿en qué escribimos? ¿por qué?

Escribimos en un lenguaje de manipulación de datos puesto que el objetivo del laboratorio es Desarrollar competencias básicas para escribir consultas simples en SQL y el DML nos otorga las herramientas para lograrlo

B. Motor de bases de datos y bases de datos

Motor de base de datos

Motor de base de datos es el servicio principal para almacenar, procesar y proteger los datos. El Motor de base de datos proporciona acceso controlado y procesamiento de transacciones rápido para cumplir con los requisitos de las aplicaciones consumidoras de datos más exigentes de su empresa.

¿Para qué sirve?

Se usa el motor de base de datos para crear bases de datos relacionales para el procesamiento de transacciones en línea o datos de procesamiento analíticos en línea. Se pueden crear tablas para almacenar datos y objetos de base de datos como índices, vistas y procedimientos almacenados para ver, administrar y proteger los datos.

Bases de datos

Una base de datos es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico.

¿Para qué sirve?

Nos permite guardar grandes cantidades de información de forma organizada para poder en un futuro poder consultarla, realizar búsquedas, ingresar nuevos datos, eliminar registros, etc. Todo esto de una forma rápida y simple dentro de un ordenador.

Bibliografía : <https://docs.microsoft.com/es-es/sql/database-engine/sql-server-database-engine-overview?view=sql-server-2017>

¿Qué motores ofrece sqlzoo.net?

MySQL, Oracle, SQL

¿Qué bases de datos ofrece sqlzoo?

Las siguientes : Buses, EURO2012, Movies, Nobel, Teachers, World.

PRACTICA

A. Estudien la información de referencia de las secciones indicadas y escriban expresiones para las consultas.

- Consultas con proyecciones, restricciones y producto cruz SELECT .. FROM .. WHERE ..

```
SELECT name, continent
FROM world
WHERE continent = 'Europe'
```

- Consultas que requieren agrupamiento GROUP BY ... HAVING ...

```
SELECT continent, COUNT(name) FROM WORLD
GROUP BY continente
```

- Orden en el resultado de consulta ORDER BY

```
SELECT name, continente
FROM world
ORDER BY name
```

- Resultados sin repeticiones DISTINCT

```
SELECT DISTINCT continent
FROM world
```

- Dar nuevos nombres a columnas AS

```
SELECT continent, COUNT(name) AS Numero_Paises
FROM WORLD
GROUP BY continente
```

B. 1. ¿ Qué información tiene la tabla WORLD?

SQL

```
SELECT * from world
```

ALGEBRA RELACIONAL

World

CALCULO RELACIONAL

$\{x : world \mid x\}$

2. ¿ Qué continentes figuran en esa tabla ? (Sin repeticiones)

SQL

```
SELECT continent  
FROM world  
GROUP BY continent
```

ALGEBRA RELACIONAL

$\sigma_{\text{continent}}(\text{world})$

CALCULO RELACIONAL

$\{x : world \mid x.\text{continent}\}$

3. ¿Qué países tienen un área menor a 1000? (ordenados de mayor a menor)?

SQL

```
SELECT name,area  
FROM world  
WHERE area<1000  
ORDER BY area desc
```

ALGEBRA RELACIONAL

$\pi_{\text{name,area}}(\sigma_{\text{area}<1000}(\text{world}))$

CALCULO RELACIONAL

$\{x : world \mid x.\text{area}<1000 : x.\text{name}\}$

4. ¿Qué continentes tienen países con una población mayor a quinientos mil de habitantes? (ordenados de mayor a menor)

SQL

```
SELECT continent  
FROM world  
WHERE population>500000  
ORDER BY population desc
```

ALGEBRA RELACIONAL

π continent (σ population > 500000 (world))

CALCULO RELACIONAL

{x : world | : x.population>500000 : x.continent}

5. ¿Qué área tiene cada una de los continentes?

SQL

```
SELECT continent,SUM(area) as area  
FROM world  
GROUP BY continent
```

ALGEBRA RELACIONAL

π continent, sum(area)(world)

CALCULO RELACIONAL

{x : world | : (x.continent,x.area)}

6. ¿Cuál es la población total?

SQL

```
SELECT sum(population) as PoblacionTotal  
FROM world
```

ALGEBRA RELACIONAL

π sum(population) (world)

CALCULO RELACIONAL

(+x : world | : x.population)

7. ¿De cuántos países se tiene información?

SQL

```
SELECT COUNT(name) as PaisesTotales  
FROM world
```

ALGEBRA RELACIONAL

```
 $\pi_{\text{count}}(\text{name})(\text{world})$ 
```

CALCULO RELACIONAL

```
(#x : world | : x.name)
```

C. SELECT BASICS

- SELECT population FROM world
WHERE name = 'Germany'
- SELECT name, population FROM world
WHERE name IN ('Sweden', 'Norway', 'Denmark')
- SELECT name, area FROM world
WHERE area BETWEEN 200000 AND 250000

• SELECT NAME

- SELECT name FROM world
WHERE name LIKE 'Y%'
- SELECT name FROM world
WHERE name LIKE '%y'
- SELECT name FROM world
WHERE name LIKE '%x%'
- SELECT name FROM world
WHERE name LIKE '%land'
- SELECT name FROM world
WHERE name LIKE 'C%ia'
- SELECT name FROM world
WHERE name LIKE '%oo%'
- SELECT name FROM world
WHERE name LIKE '%a%a%a%a%'

- SELECT name FROM world
WHERE name LIKE '_t%'
ORDER BY name
- SELECT name FROM world
WHERE name LIKE '%o__o%'
- SELECT name FROM world
WHERE name LIKE '_____'
- SELECT name
FROM world
WHERE name=capital
- SELECT name
FROM world
WHERE capital=concat(name, ' City')
- SELECT capital,name
FROM world
WHERE capital like concat('%',name,'%')
- SELECT capital,name
FROM world
WHERE capital like concat(name,'%') and name!=capital
- SELECT name,replace(capital,name,"")
FROM world
WHERE capital like concat(name,'%') and name!=capital

SELECT FROM WORLD

- SELECT name, continent, population
FROM world
- SELECT name
FROM world
WHERE population >= 200000000
- SELECT name,gdp/population
FROM world
WHERE population>=200000000
- SELECT name,population/1000000
FROM world
WHERE continent= 'South America'

- ```
SELECT name,population
FROM world
WHERE name='France'or name='Germany'or name='Italy'
```
- ```
SELECT name
FROM world
WHERE name like '%United%'
```
- ```
SELECT name,population,area
FROM world
WHERE area>3000000 or population>250000000
```
- ```
SELECT name,population,area
FROM world
WHERE (area>3000000 and population<=250000000) or (area<3000000
and population>250000000)
```
- ```
SELECT
name,ROUND(population/1000000,2),ROUND(gdp/1000000000,2)
FROM world
WHERE continent='South America'
```
- ```
SELECT name,ROUND(gdp/population,-3)
FROM world
WHERE gdp>=1000000000000
```
- ```
SELECT name,capital
FROM world
WHERE LENGTH(name)=LENGTH(capital)
```
- ```
SELECT name,capital
FROM world
WHERE LEFT(name,1)=LEFT(capital,1) and name<>capital
```
- ```
SELECT name
FROM world
WHERE name LIKE '%a%'and name LIKE '%e%'and name LIKE '%i%' and
name LIKE '%o%' and name LIKE '%u%' and name NOT LIKE '% %'
```



## SELECT FROM NOBEL

- ```
SELECT yr, subject, winner
FROM nobel
WHERE yr = 1950
```
- ```
SELECT winner
FROM nobel
WHERE yr = 1962
AND subject = 'Literature'
```
- ```
SELECT yr,subject
FROM nobel
WHERE winner = 'Albert Einstein'
```
- ```
SELECT winner
FROM nobel
WHERE yr >= 2000 and subject = 'Peace'
```
- ```
SELECT *
FROM nobel
WHERE yr>=1980 and yr<=1989 and subject = 'Literature'
```
- ```
SELECT * FROM nobel
WHERE winner = 'Woodrow Wilson' or winner = 'Jimmy Carter' or winner =
'Barack Obama' or winner = 'Theodore Roosevelt'
```
- ```
SELECT winner
FROM nobel
WHERE winner like 'John%'
```
- ```
SELECT *
FROM nobel
WHERE (subject = 'Physics' and yr = 1980) or (subject = 'Chemistry' and yr
= 1984)
```
- ```
SELECT *
FROM nobel
WHERE subject != 'Medicine' and subject != 'Chemistry' and yr=1980
```
- ```
SELECT *
FROM nobel
WHERE (subject = 'Medicine' and yr <1910) or (subject = 'Literature' and yr
>= 2004)
```

- `SELECT *`  
`FROM nobel`  
`WHERE winner = 'PETER GRÜNBERG'`
- `SELECT *`  
`FROM nobel`  
`WHERE winner like 'EUGENE O"NEILL'`
- `SELECT winner,yr,subject`  
`FROM nobel`  
`WHERE winner like 'Sir%'`  
`ORDER BY yr desc`
- `SELECT winner,subject`  
`FROM nobel`  
`WHERE yr = 1984`  
`ORDER BY subject IN ('Chemistry','Physics') , subject , winner`

### **SELECT within SELECT Tutorial**

- `SELECT name FROM world`  
`WHERE population >`  
`(SELECT population FROM world`  
`WHERE name='Russia')`
- `SELECT name`  
`FROM world`  
`WHERE continent = 'Europe' and gdp/population > (SELECT gdp/population`  
`FROM world`  
`WHERE name = 'United Kingdom')`
- `SELECT name , continent`  
`FROM world`  
`WHERE continent = 'South America' or continent = 'Oceania'`  
`ORDER BY name`
- `SELECT name,population`  
`FROM world`  
`WHERE population > (SELECT population`  
`FROM world`  
`WHERE name = 'Canada') and population < (SELECT population`  
`FROM world`  
`WHERE name = 'Poland')`

- SELECT name,CONCAT(ROUND(population/(SELECT population  
FROM world  
WHERE name = 'Germany')\*100),'%')  
FROM world  
WHERE continent = 'Europe'
- SELECT name  
FROM world  
WHERE gdp > ALL (SELECT gdp  
FROM world  
WHERE gdp > 0 and continent = 'Europe')
- SELECT continent, name, area FROM world x  
WHERE area >= ALL  
(SELECT area FROM world y  
WHERE y.continent=x.continent  
AND area>0)
- SELECT continent , name  
FROM world x  
WHERE name <= ALL (SELECT name  
FROM world y  
WHERE x.continent = y.continent)
- SELECT name,continent,population  
FROM world x  
WHERE 25000000 >= ALL (SELECT population  
FROM world y  
WHERE x.continent = y.continent)
- SELECT name,continent  
FROM world x  
WHERE population/3 > ALL ( SELECT population  
FROM world y  
where x.continent = y.continent and y.name!= x.name)

## **SUM and COUNT**

- SELECT SUM(population)  
FROM world
- SELECT continent  
FROM world  
GROUP BY continent

- `SELECT SUM(gdp) as suma  
FROM world  
WHERE continent = 'Africa'`
- `SELECT COUNT(name)  
FROM world  
WHERE area >= 1000000`
- `SELECT SUM(population) as suma  
FROM world  
WHERE name = 'Estonia' or name = 'Latvia' or name = 'Lithuania'`
- `SELECT continent,COUNT(name) as ciudades  
FROM world  
GROUP BY continent`
- `SELECT continent,COUNT(name) as ciudades  
FROM world  
WHERE population >= 10000000  
GROUP BY continent`
- `SELECT continent  
FROM world x  
WHERE 100000000 < (SELECT SUM(population)  
FROM world y  
where x.continent = y.continent)  
GROUP BY continent`

**D. 8 Consultas: una para cada uno de los tipos de operadores.**

- PARA EUROPA MUESTRE CADA PAIS ,LA POBLACION EN BILLONES Y EL PIB/POBLACION EN MILLONES, ESTOS DOS ULTIMOS CON TRES DECIMALES

```
SELECT name, ROUND (population/1000000000,3),
ROUND(GDP/1000000,3)
FROM world
WHERE continent='Europe'
```

- MUESTRE LOS PAISES DE AMERICA DEL SUR Y EUROPA EN LOS CUALES LA POBLACION SEA DE AL MENOS 100 MILLONES.

```
SELECT name
FROM world
WHERE population>=100000000 and (continent='Europe'or
continent='South America')
```

- MUESTRE TODOS LOS PAISES QUE PERTENEZCAN AL CONTINENTE ASIATICO Y QUE AL MENOS TENGAN 2 'A' EN SU NOMBRE

```
SELECT name
FROM world
WHERE continent='Asia' and name LIKE '%a%a%'
```

- MUESTRE LA POBLACION DE LOS PAISES QUE TIENEN AL MENOS 6 CARACTERES EN SU NOMBRE Y ASI MISMO MUESTRE EL CONTINENTE AL QUE PERTENECES Y EL GDP/POBLACION

```
SELECT name,population,continent,gdp/population
FROM world
WHERE LENGTH (name)>=6
```

- MUESTRE LOS PAISES EN LOS CUALES LA POBLACION SEA MENOR A 150 MILLONES Y AGREGUE LA FECHA ACTUAL A CADA UNO DE ELLOS

```
SELECT name,CURRENT_DATE()
FROM world
WHERE population<150000000
```

- MUESTRE LA POBLACION TOTAL DEL MUNDO EXCLUYENDO EL CONTINENTE ASIATICO

```
SELECT SUM(POPULATION)
FROM world
WHERE continent!='Asia'
```

- MUESTRE EL NOMBRE CON LA PALABRA 'YES' SI UN PAIS PERTENECE AL CONTINENTE DE AMERICA DEL SUR DE LO CONTRARIO MUESTRE NO

```
SELECT continent,CASE continent
WHEN continent='South America' THEN 'YES'
ELSE 'NO'
END
'PAISES SUR-AME'
FROM world
```

- MUESTRE LA POBLACION DE LOS PAISES DE SUR AMERICA QUE SEA AL MENOS DE 150 EN BILLONES CONVERTIDOS A ENTEROS

```
SELECT name, CAST(population/1000000000) As int
FROM world
WHERE population<150000000000
```

**3 consultas anidadas que usen otra consulta: 1) (SELECT ...) en FROM, 2) SELECT en WHERE y 3) SELECT ... en SELECT**

- **(SELECT ...) en FROM**

```
SELECT name
FROM (SELECT name, population
FROM world WHERE continent= 'Europe') As MayorQueColombia
WHERE population > (SELECT population FROM world WHERE name=
'Colombia')
```

- **SELECT en WHERE**

```
SELECT name,continent
FROM world x
WHERE gdp >= ALL (SELECT gdp
FROM world y
WHERE x.continent = y.continent)
```

- **SELECT ... en SELECT**

```
SELECT continent , name
FROM world
where gdp > (SELECT SUM(gdp)
FROM world
WHERE name = 'Colombia')
```