Daniel Rincon

Michael Perilla

Auto estudio

A . SQL

A.1) ¿Qué es? ¿Para qué sirve?

*¿Qué es?

SQL consiste en un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de control de datos. El alcance de SQL incluye la inserción de datos, consultas, actualizaciones y borrado, la creación y modificación de esquemas y el control de acceso a los datos. También el SQL a veces se describe como un lenguaje declarativo, también incluye elementos procesales.

*¿Para qué sirve?

Es un lenguaje de dominio específico utilizado en programación, diseñado para administrar, y recuperar información de sistemas de gestión de bases de datos relacionales. Una de sus principales características es el manejo del álgebra y el cálculo relacional para efectuar consultas con el fin de recuperar, de forma sencilla, información de bases de datos, así como realizar cambios en ellas.

Originalmente basado en el álgebra relacional y en el cálculo relacional.

A.2) ¿Qué es DML, DLL,DCL,TCL?

- DML = **Lenguaje de manipulación de datos** (**D**ata **M**anipulation **L**anguage)
- DLL = **Biblioteca de enlace dinámico** (**D**ynamic-**L**ink **L**ibrary)
- DCL = **Lenguaje de control de datos** (**D**ata **C**ontrol **L**anguage)
- TCL = **Lenguaje de herramientas de comando** (**T**ool **C**ommand **L**anguage)

A.3) En este laboratorio, ¿en qué escribimos? ¿por qué?

*¿en qué escribimos?

- Calculo Relacional
- Algebra
- SQL

*¿por qué?

Para buscar y utilizar correctamente una base de datos.

B. Motor de bases de datos y bases de datos

B.1) ¿Qué son?

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.

B.2) ¿Qué motores ofrece sqlzoo.net?

* My SQL

* Oracle

* SQL server

* DB2

* Postgres

* Ingres

B.3) ¿Qué bases de datos ofrece sqlzoo?

* Module feedback

* Help desk

* Adventure works

* University timetables

* Musicians

* Dressmaker

* Congestion charging

PRACTICA

A.

- **SELECT:**
    - ✓ SELECT yr FROM games
    - ✓ SELECT city FROM games
    - ✓ SELECT yr, city FROM games

- **Functions:**
    - ✓ ABS: SELECT name, area FROM bbc WHERE ABS(area)<500
    - ✓ COUNT: SELECT region, COUNT(name) FROM bbc GROUP BY región
    - ✓ CEIL: SELECT population/1000000 AS a, CEIL(population/1000000) AS b FROM bbc
    - ✓ CONCAT: SELECT CONCAT(region, name) FROM bbc
    - ✓ LENGTH: SELECT LENGTH(name), name FROM bbc
    - ✓ MAX: SELECT region, MAX(name) FROM bbc GROUP BY región
    - ✓ MIN: SELECT region, MIN(name) FROM bbc GROUP BY región
    - ✓ SUBSTR: SELECT name, SUBSTR(name, 2, 5) FROM bbc
    - ✓ AVG: SELECT region, AVG(population) FROM bbc GROUP BY región

- **SELECT .. WHERE:**
    - ✓ SELECT yr, city FROM games WHERE city = 'Beijing'
    - ✓ SELECT yr, city FROM games WHERE yr = 2004
    - ✓ SELECT yr, city FROM games WHERE city = 'Beijing' OR city = 'Athens'

- **SELECT .. GROUP BY:**
    - ✓ SELECT continent, COUNT(yr) FROM games GROUP BY continent
    - ✓ SELECT continent, SUM(yr) FROM games GROUP BY continent

- **SELECT .. SELECT:**
    - ✓ SELECT name, ROUND(gdp_per_capita) FROM (SELECT name, gdp/population AS gdp_per_capita FROM bbc) X WHERE gdp_per_capita>20000

B.

- **¿Qué información tiene la tabla WORLD?:**
  - ✓ CÁLCULO: { x : world | : }
  - ✓ ÁLGEBRA: $\pi\ world$
  - ✓ SQL: SELECT * FROM world

- **¿Qué continentes figuran en esa tabla?:**
  - ✓ CÁLCULO: { x : world | : continent }
  - ✓ ÁLGEBRA: $\pi_{continent}\ world$
  - ✓ SQL: SELECT DISTINCT continent FROM world

- **¿Qué países tienen un área menor a 1000?:**
  - ✓ CÁLCULO: { z : { x : world | area < 1000 : x } | : name }
  - ✓ ÁLGEBRA: $\pi_{name}\ (\sigma_{area<1000}\ world)$
  - ✓ SQL: SELECT name FROM world WHERE área < 1000 ORDER BY área
- **¿Qué continentes tienen países con una población mayora quinientos mil de habitantes?:**
  - ✓ CÁLCULO: { z : { x : world | population > 500000 : x } | : continent }
  - ✓ ÁLGEBRA: $\pi_{continent}\ (\sigma_{population>500000}\ world)$
  - ✓ SQL: SELECT continent FROM world WHERE population>500000 GROUP BY continent ORDER BY área
- **¿Qué área tiene cada uno de los continentes?:**
  - ✓ CÁLCULO: { z : world | : continent, SUM(área) }
  - ✓ ÁLGEBRA: $\pi_{continent,SUM(area)}world$
  - ✓ SQL: SELECT continent, SUM(área) FROM world GROUP BY continent ORDER BY SUM(área)
- **¿Cuál es la población total?:**
  - ✓ CÁLCULO: { z : world | : SUM(population) }
  - ✓ ÁLGEBRA: $\pi\ _{SUM(population)}world$
  - ✓ SQL: SELECT SUM(population) FROM world
- **¿De cuántos países se tiene información?:**
  - ✓ CÁLCULO: { z : world | : COUNT(name) }
  - ✓ ÁLGEBRA: $\pi_{\mathrm{COUNT(name)}}world$
  - ✓ SQL: SELECT COUNT(name) FROM world

C.

0. **SELECT basics**
    a. SELECT population FROM world WHERE name = 'Germany'
    b. SELECT name, population FROM world WHERE name IN ('Sweden', 'Norway', 'Denmark');
    c. SELECT name, area FROM world WHERE area BETWEEN 200000 AND 250000
1. **SELECT name**
    a. SELECT name FROM world WHERE name LIKE 'Y%'
    b. SELECT name FROM world WHERE name LIKE '%y'
    c. SELECT name FROM world WHERE name LIKE '%x%'
    d. SELECT name FROM world WHERE name LIKE '%land'
    e. SELECT name FROM world WHERE name LIKE 'C%ia'
    f. SELECT name FROM world WHERE name LIKE '%oo%'
    g. SELECT name FROM world WHERE name LIKE '%a%a%a%'
    h. SELECT name FROM world WHERE name LIKE '_t%' ORDER BY name
    i. SELECT name FROM world WHERE name LIKE '%o__o%'
    j. SELECT name FROM world WHERE name LIKE '____'
    k. SELECT name FROM world WHERE name LIKE capital
    l. SELECT name FROM world WHERE capital LIKE concat(name, '%City')
    m. SELECT capital, name FROM world WHERE capital LIKE CONCAT('%', name, '%')
    n. SELECT capital, name FROM world WHERE capital LIKE CONCAT('%', name, '%') AND LENGTH(capital) > LENGTH(name)
    o. SELECT name, REPLACE(capital, name, '') AS ext FROM world WHERE capital LIKE CONCAT('%', name, '%') AND LENGTH(capital) > LENGTH(name)
2. **SELECT from World**
    a. SELECT name, continent, population FROM world
    b. SELECT name FROM world WHERE population >= 200000000
    c. SELECT name, gdp/population FROM world WHERE population >= 200000000
    d. SELECT name, population/1000000 FROM world WHERE continent = 'South America’
    e. SELECT name, population FROM world WHERE name IN ('France', 'Germany', 'Italy')C.2) SELECT from world
    f. SELECT name FROM world WHERE name LIKE '%United%'
    g. SELECT name, population, área FROM world WHERE population > 250000000 OR area > 3000000
    h. SELECT name, population, área FROM world WHERE population > 250000000 XOR area > 3000000

i. SELECT name, ROUND(population/1000000, 2), ROUND(gdp/1000000000, 2) FROM world WHERE continent = 'South America'

j. SELECT name, ROUND(gdp/population, -3) FROM world WHERE gdp >= 1000000000000

k. SELECT name, capital FROM world WHERE LENGTH(name) = LENGTH(capital)

l. SELECT name, capital FROM world WHERE LEFT(name,1) = LEFT(capital,1) AND name <> capital

m. 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 '% %'

3. **SELECT from nobel**

a. SELECT yr, subject, winner FROM nobel WHERE yr = 1950

b. SELECT winner FROM nobel WHERE yr = 1962 AND subject = 'Literature'

c. SELECT yr, subject FROM nobel WHERE winner = 'Albert Einstein'

d. SELECT winner FROM nobel WHERE subject = 'Peace' AND yr >= 2000

e. SELECT * FROM nobel WHERE yr >= 1980 AND yr <= 1989 AND subject = 'Literature'

f. SELECT * FROM nobel WHERE winner IN ('Theodore Roosevelt', 'Woodrow Wilson', 'Jimmy Carter', 'Barack Obama')

g. SELECT winner FROM nobel WHERE winner LIKE 'John%'

h. SELECT * FROM nobel WHERE (subject='Physics' AND yr=1980) OR (subject='Chemistry' AND yr=1984)

i. SELECT * FROM nobel WHERE yr=1980 AND subject<>'Chemistry' AND subject<>'Medicine'

j. SELECT * FROM nobel WHERE (subject='Medicine' AND yr<1910) OR (subject='Literature' AND yr>=2004)

k. SELECT * FROM nobel WHERE winner='PETER GRÜNBERG'

l. SELECT * FROM nobel WHERE winner='EUGENE O''NEILL'

m. SELECT winner, yr, subject FROM nobel WHERE winner LIKE 'Sir%' ORDER BY yr DESC, winner

4. **SELECT within SELECT**

a. SELECT name FROM world WHERE population > (SELECT population FROM world WHERE name='Russia')

b. SELECT name FROM world WHERE continent = 'Europe' AND gdp/population > (SELECT gdp/population FROM world WHERE name='United Kingdom')

c. SELECT name, continent FROM world WHERE continent IN (SELECT continent FROM world WHERE name IN ('Argentina', 'Australia')) ORDER BY name

    d. SELECT name, population FROM world WHERE population >(SELECT population FROM world WHERE name = ('Canada')) AND population < (SELECT population FROM world WHERE name = ('Poland'))
    e. SELECT name, CONCAT(ROUND(population / (SELECT population FROM world WHERE name = 'Germany') * 100), '%') FROM world WHERE continent = 'Europe'
    f. SELECT name FROM world WHERE gdp > ALL(SELECT gdp FROM world WHERE continent='Europe' AND gdp <> NULL)
    g. SELECT continent, name, area FROM world x WHERE area >= ALL (SELECT area FROM world y WHERE y.area>x.area AND x.continent = y.continent)
    h. SELECT continent, name FROM world x WHERE LEFT(name, 1) < ALL (SELECT name FROM world y WHERE x.continent = y.continent)

5. **SUM and COUNT**
    a. SELECT SUM(population) FROM world
    b. SELECT continent FROM world GROUP BY continent
    c. SELECT SUM(gdp) FROM world WHERE continent = 'Africa'
    d. SELECT COUNT(name) FROM world WHERE area >= 1000000
    e. SELECT SUM(population) FROM world WHERE name IN ('Estonia', 'Latvia', 'Lithuania')
    f. SELECT continent, COUNT(name) FROM world GROUP BY continent
    g. SELECT continent, COUNT(name) FROM world WHERE population >= 10000000 GROUP BY continent
    h. SELECT continent FROM world GROUP BY continent HAVING SUM(population) >= 100000000

D.

- Busque las ciudades que tengan un area entre 100000 a 200000
    - SELECT name,area
    - FROM world
    - Where area BETWEEN 100000 AND 200000
- Busque ciudades que empiecen por M
    - SELECT name
    - FROM world
    - WHERE name LIKE 'm%'
- Busque ciudades que su nombre tenga 6 caracteres
    - SELECT name
    - FROM world
    - WHERE name LIKE '______'
- Busque ciudades que tengan la palabra col en el nombre
    - SELECT name
    - FROM world

    - WHERE name LIKE '%col%'
- Busque continente con una población total mayor a 100 millones
    - SELECT continent
    - FROM world
    - GROUP BY continent
    - HAVING SUM(population) >= 100000000
- Cuantas ciudades comienzan por ‘B’
    - SELECT COUNT(name)
    - FROM world
    - WHERE name LIKE ‘B%’
- Dé el GDP total de EUROPA
    - SELECT SUM(gdp)
    - FROM world
    - WHERE continent = ‘Europe’
- Diga todas las ciudades con una población mayor a la de Bogotá
    - SELECT name FROM world
    - WHERE population >
    - (SELECT population FROM world
    - WHERE name=’Bogotá’)

Bibliografía:

- https://es.wikipedia.org/wiki/SQL
- https://docs.microsoft.com/es-es/sql/database-engine/sql-server-database-engine-overview?view=sql-server-2017