

Introducción a SQL



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Módulo 5

Introducción a SQL

Objetivo

Este curso está diseñado para que cualquier persona se introduzca en los conceptos clave del lenguaje SQL y obtenga una comprensión fundamental sobre como funciona.

Agenda del día



Introducción a SQL



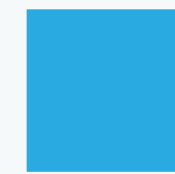
¿Qué es SQL?



Lenguaje de Definición de Datos (DDL)



Lenguaje de Manipulación de Datos (DML)



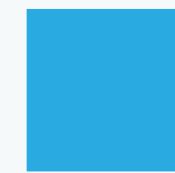
Operadores SQL



Consultas avanzadas COUNT, GROUP BY, JOINS, SUM



Procedimientos



Control de flujo

Terminología SQL



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Terminología de SQL

SQL

El lenguaje de consulta estructurada (SQL) es un lenguaje de programación para almacenar y procesar información en una base de datos relacional. Una base de datos relacional almacena información en forma de tabla, con filas y columnas que representan diferentes atributos de datos y las diversas relaciones entre los valores de datos. Puede usar las instrucciones SQL para almacenar, actualizar, eliminar, buscar y recuperar información de la base de datos. También puede usar SQL para mantener y optimizar el rendimiento de la base de datos.

Comandos SQL

Los comandos de lenguaje de consulta estructurada (SQL) son palabras clave o instrucciones SQL específicas que los desarrolladores utilizan para manipular los datos almacenados en una base de datos relacional. Puede clasificar los comandos SQL de la siguiente manera.

Definición de datos

El lenguaje de definición de datos (DDL) se refiere a comandos SQL que diseñan la estructura de la base de datos. Los ingenieros de bases de datos utilizan DDL para crear y modificar objetos de bases de datos en función de las necesidades empresariales. Por ejemplo, el ingeniero de bases de datos utiliza el comando CREATE para crear objetos de base de datos, como tablas, vistas e índices.

Terminología de SQL

Consulta de datos

El lenguaje de consulta de datos (DQL) consiste en instrucciones para recuperar datos almacenados en bases de datos relacionales. Las aplicaciones de software usan el comando SELECT para filtrar y devolver resultados específicos de una tabla SQL.

Manipulación de datos

Las instrucciones del lenguaje de manipulación de datos (DML) escriben información nueva o modifican los registros existentes en una base de datos relacional. Por ejemplo, una aplicación usa el comando INSERT para almacenar un nuevo registro en la base de datos.

Control de transacciones

El motor relacional utiliza el lenguaje de control de transacciones (TCL) para hacer cambios en la base de datos de manera automática. Por ejemplo, la base de datos usa el comando ROLLBACK para deshacer una transacción errónea.

Lenguaje de **Manipulación de Datos** (DML)



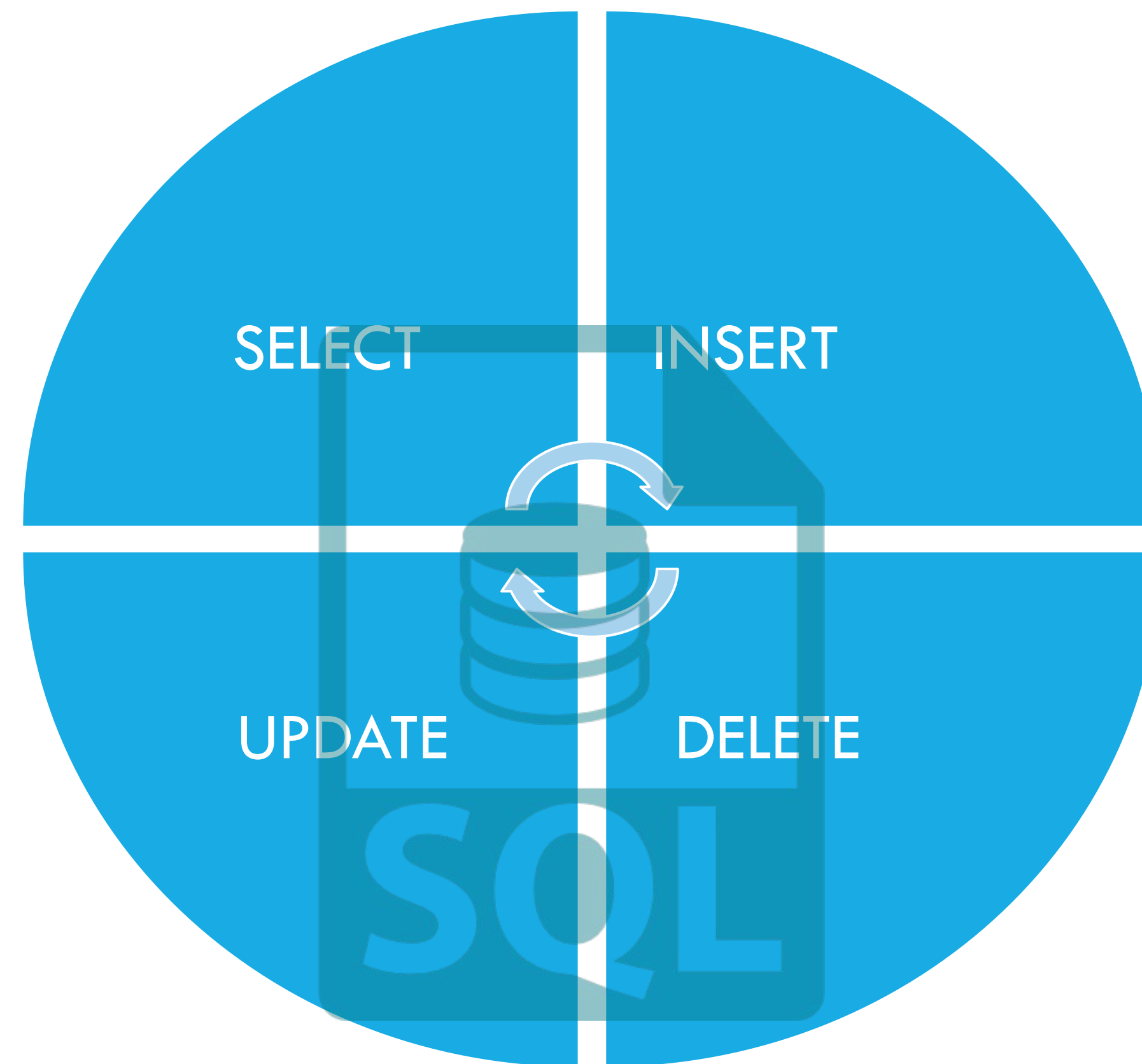
Presentado por **Nelson Aguayo**

www.consultec-ti.com

Lenguaje de Manipulación de Datos

Esta sentencia se utiliza para realizar consultas sobre los datos.

Sirve para modificar los valores de uno o varios registros.



Con esta instrucción podemos insertar los valores en una base de datos.

Se utiliza para eliminar las filas de una tabla.

Break **15 minutos**



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Lenguaje de **Definición de Datos (DDL)**



Presentado por **Nelson Aguayo**

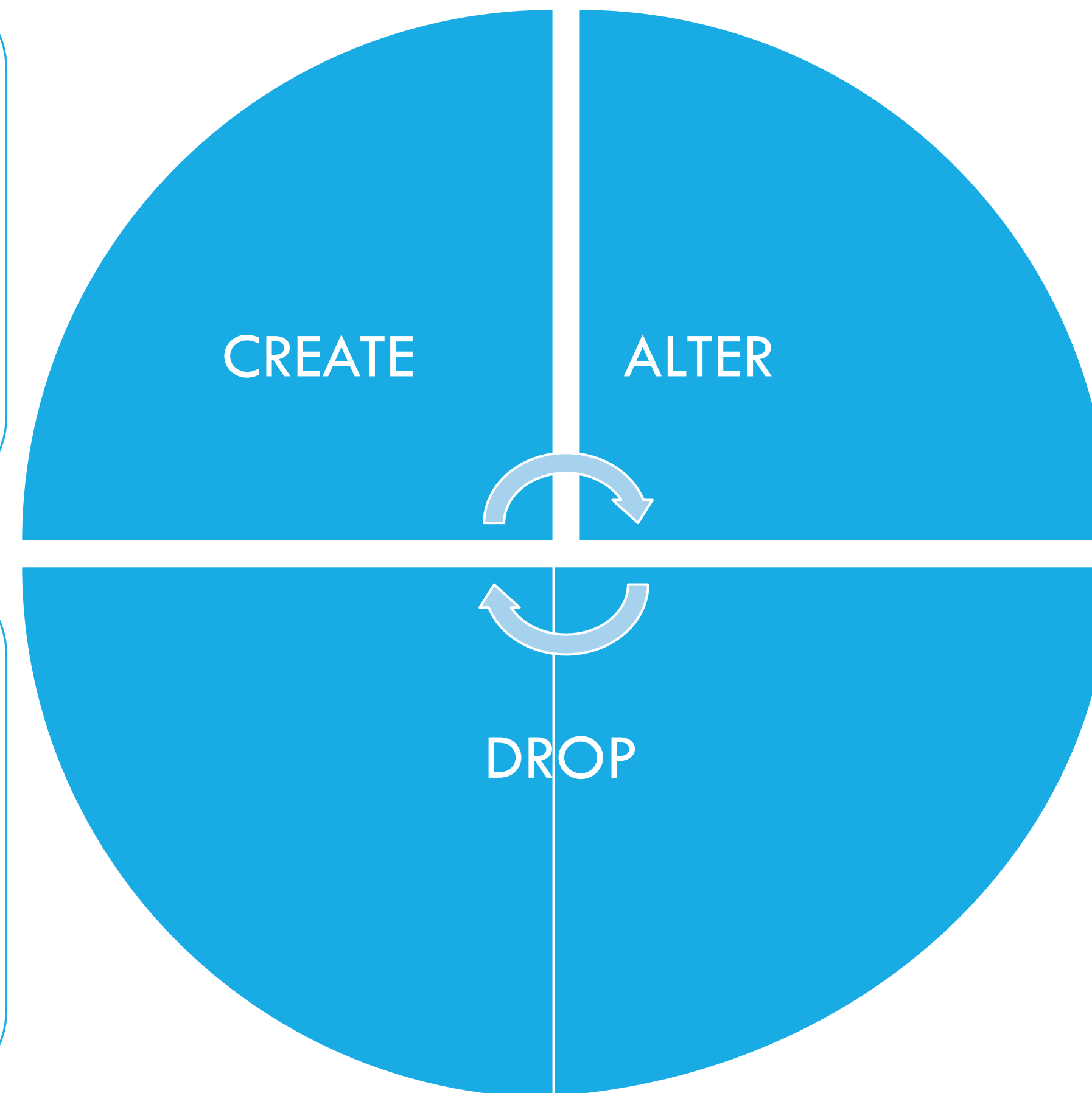
www.consultec-ti.com

Lenguaje de Definición de Datos

Principalmente utilizamos el objeto tabla para crear nuestra estructura, esto lo hacemos utilizando el comando CREATE TABLE y escribiendo los nombres de las columnas con sus respectivos tipos de datos.

El funcionamiento del comando DROP es la inversa del comando CREATE. Por lo tanto podemos eliminar los objetos que hemos creado para almacenar nuestros datos. Para eliminar una estructura es tan simple como indicar el nombre de nuestro objeto.

```
DROP table tabla;
```



Modifica la estructura de la base de datos.

Commit y rollback



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Commit y rollback

Commit y rollback

Si la base de datos permite la gestión de transacciones, se puede utilizar 'Commit' para confirmar una 'Insert', 'Update', o 'Delete', o 'Rollback' para cancelarlos.



Hora de almuerzo



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Operadores SQL



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Operadores SQL

Ahora se describirán qué operaciones y con qué operadores se realizan.

Los operadores se pueden dividir en TRES conjuntos:



Aritméticos: utilizan valores numéricos



Lógicos (o booleanos o de comparación): utilizan valores booleanos o lógicos



Concatenación: para unir cadenas de caracteres.

Consultas avanzadas:

COUNT | GROUP BY | JOINS | SUM



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Consultas avanzadas

Count:

Devuelve un recuento del número de valores no NULL de expr en las filas recuperadas por una instrucción SELECT . El resultado es un valor GRANDE (BIGINT). Es una función agregada, por lo que se puede utilizar con la cláusula GROUP BY.

COUNT(*) cuenta el número total de filas en una tabla.

Count(expr) devuelve 0 si no hay filas que coincidan..

COUNT(COLUMNA) devuelve el número total de registros para una columna.

Consultas avanzadas

Group By:




Use la cláusula GROUP BY en una instrucción SELECT para agrupar filas que tengan el mismo valor en una o más columnas, o el mismo valor calculado usando expresiones con cualquier función y operador excepto las funciones de agrupación. Cuando usa una cláusula GROUP BY, obtendrá una única fila de resultado para cada grupo de filas que tengan el mismo valor para la expresión dada en GROUP BY .

Consultas avanzadas

JOIN:

Se utilizan para recuperar datos de varias tablas. Se realiza un JOIN de MariaDB cada vez que se unen dos o más tablas en una instrucción SQL.

Hay diferentes tipos de uniones:

-  **INNER JOIN (unión simple):** Es el tipo de unión más común. INNER JOINS devuelve todas las filas de varias tablas donde se cumple la condición de combinación.
-  **LEFT OUTER JOIN (o algunas veces llamado LEFT JOIN):** Este tipo de combinación devuelve todas las filas de la tabla de la IZQUIERDA especificada en la condición ON y solo aquellas filas de la otra tabla donde los campos combinados son iguales (se cumple la condición de combinación).
-  **RIGHT OUTER JOIN (o algunas veces llamado RIGHT JOIN):** Este tipo de combinación devuelve todas las filas de la tabla de la DERECHA especificada en la condición ON y solo aquellas filas de la otra tabla donde los campos combinados son iguales (se cumple la condición de combinación).

Consultas avanzadas

SUM:

La función SUM(expr) devuelve el valor sumado de una expresión.

Break **15 minutos**



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Procedimientos



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Procedimientos

Los procedimientos son códigos compilados en el servidor, que se ejecutan en el servidor, y que pueden aceptar parámetros de entrada y/o de salida. Un procedimiento se puede usar en un select, pero no en un where clause.

Un procedimiento tiene un nombre, un conjunto de parámetros (opcional) y un bloque de código.

Control de flujo



Presentado por **Nelson Aguayo**

www.consultec-ti.com

Control de flujo

IF:

If..elsif...else

Se usa el IF..ELSIF..ELSE para controlar el flujo en PL/SQL.

```
IF (expresion) THEN
-- Instrucciones
ELSIF (expresion) THEN
-- Instrucciones
ELSE END IF;
```

Control de flujo

Bucles

Un bucle permite repetir una acción un sin número de veces. Hay que tener cuidado en no crear bucles infinitos.

En PL/SQL tenemos a nuestra disposición los siguientes iteradores o bucles:

- * LOOP
- * WHILE
- * FOR

Control de flujo

LOOP

El bucle LOOP, se repite tantas veces como sea necesario hasta que se fuerza su salida con la instrucción EXIT. Su sintaxis es la siguiente

```
LOOP
-- Instrucciones
IF (expresion) THEN
-- Instrucciones
EXIT;
END IF;
END LOOP;
```

Control de flujo

WHILE

El bucle WHILE, se repite mientras que se cumpla la expresión.
WHILE (expresión)

LOOP
-- Instrucciones
END LOOP;

Control de flujo

FOR

El bucle FOR, se repite tantas veces como le indiquemos en los identificadores inicio y final.

```
FOR contador IN [REVERSE] inicio..final LOOP
```

```
-- Instrucciones
```

```
END LOOP;
```

En el caso de especificar REVERSE el bucle se recorre en sentido inverso.



Gracias
¡Nos vemos pronto!