TALENTO TECH

Lección 3: Lenguaje estandarizado de consultas

TIC

Universidad Tecnológica de Pereira

faceIT

Tiempo de ejecución: 4 horas

PLANTEAMIENTO DE LA SESIÓN	MATERIALES
El lenguaje de consultas SQL o structured query language es fundamental para cualquier transacción con bases de datos relacionales. Las bases de datos relacionales organizan la información en tablas y modelos de entidad-relación que es una forma de asociar información parecida sin tener que guardar duplicados ni ser redundantes. El lenguaje de consultas permite aprovechar las ventajas de la arquitectura para hacer consultas rápidas y operaciones en el servidor de bases de datos.	Conjunto de datos del Titanic, disponible en: https://github.com/datascienceedojo/datasets/blob/master/titanic.csv

DESARROLLO DE LA LECCIÓN

El lenguaje de consultas SQL permite realizar operaciones sobre un motor de bases de datos relacionales. Este lenguaje se inventó en 1970 y se compone de ciertas entidades que almacenan datos y modelan sus relaciones. Estas entidades son:

Tablas: Las tablas son los elementos básicos de almacenamiento de datos. Se componen de filas y columnas. A diferencia de sistemas de hojas de cálculo, en las tablas de una base de datos se tienen que asignar tipos de datos específicos a cada una de las columnas. Generalmente se optimizan los tipos de datos para que no se ocupe más memoria de la requerida.

Por ejemplo, para una base de datos de un sistema de nóminas, una tabla podría tener un diseño como el de la tabla 1 en donde se asignan ciertos tipos de dato a los campos necesarios. Para cada tipo de dato también se establece su longitud en bytes, determinando qué tanta información se puede guardar en cada campo.

Nombre del campo	IDENTIFICADOR	NOMBRE EMPLEADO	SALARIO
Tipo de dato	Entero	Cadena de caracteres	Número con cifras decimales
Longitud en bytes	12	200	10

Tabla 1. Ejemplo de diseño de columnas de una tabla

Instrucciones SQL

Las instrucciones SQL o consultas (queries) son instrucciones válidas para que serán interpretadas por un sistema de administración de bases de datos. Los elementos del lenguaje SQL son identificadores, variables y condiciones de búsqueda que, en conjunto, forman las instrucciones de SQL.

Procedimientos almacenados

Un procedimiento almacenado es una colección de una o más instrucciones SQL. Este procedimiento se encuentra guardado en la base de datos relacional. Estos procedimientos se usan para incrementar la eficiencia y el rendimiento de las consultas. Por ejemplo, en lugar de tener que escribir una consulta para muchos sistemas, se puede hacer un procedimiento almacenado que se llama desde cualquier sistema sin tener que escribir múltiples veces el mismo código. También ofrecen seguridad ya que los comandos no quedan expuestos en internet.

Lenguaje SQL: Es un lenguaje orientado a las consultas y de alto nivel. Tiene varios comandos y una estructura (sintaxis) que permiten realizar diferentes consultas y administrar bases de datos. Nota: Dependiendo de la implementación (programa de bases de datos) las consultas pueden variar en su estructura o incluso en algunas palabras clave. Para estar seguros se debe consultar el manual del motor en específico. En este apartado se colocarán instrucciones del estándar SQL, con lo que la mayoría de los programas de bases de datos debe funcionar apropiadamente. Sin embargo, se recomienda emplear MySQL, MariaDB o Postgres.

Base de datos relacional

Una base de datos relacional define las relaciones que pueden existir en forma de tablas. Las tablas dentro de una base de datos se suelen relacionar unas con otras, basadas en datos que tengan en común. En el siguiente ejemplo se muestran tres tablas (clientes, ordenes, envíos) de una base de datos de ejemplo

IDcliente	Nombre	Contacto	Dirección	Ciudad	Código Postal	País
1239871	Industrias Magro	Carlos Duran	Calle 74# 34-78	Cali	665005	Colombia
548972	DobleModa	Beatriz Pinzón	Complejo industrial los naranjos bodega 45	Barranquilla	6689001	Colombia

Tabla 2: Tabla de clientes

IDorden	IDcliente	Idempleado	Fecha	ID envío
41567	548972	8	2023-12-12	4
41567	548972	2	2023-10-14	3

Tabla 3: Tabla de órdenes

IDenvío	Nombre empresa envío	Teléfono
4	Servicios de entrega mundiales	3458469
3	Express entregas	3458877

Tabla 4: Tabla de envíos

En las tres tablas de ejemplo se aprecia que, en lugar de anotar toda la información de determinado empleado, se puede almacenar en cada registro simplemente el ID del empleado que hizo la orden. De la misma forma se puede almacenar el ID del envío en lugar de toda la información asociada a la empresa de envíos. En el caso de la tabla 2, se aprecia que el ID del cliente permite acceder a toda la información de un cliente y que en la tabla 3 no es necesario de guardar todos los datos de nuevo para saber qué cliente es. A este tipo de conexiones se le llaman relaciones.

Algunos comandos de SQL:

SELECT: Permite extraer datos de una base de datos (en forma de tabla, fila o columna)

UPDATE: Actualiza registros en una base de datos
INSERT INTO: Permite almacenar nuevos datos en una tabla
CREATE DATABASE: comando para crear una nueva base de datos
ALTER DATABASE: modifica una base de datos existente
CREATE TABLE: permite adicionar una tabla a una base de datos
ALTER TABLE: permite cambiar la estructura de una tabla
DROP TABLE: Elimina una tabla
CREATE INDEX: permite crear una llave de búsqueda (índice) como el caso de los identificadores en las tablas 2, 3 y 4.
DROP INDEX: borra un índice.
Instrucción select: Permite seleccionar datos en una base de datos. Los datos retornados se guardan en una tabla conocida como el conjunto resultante (result set).

Para ejecutar la instrucción select la sintaxis es:

SELECT columna1, columna2....
FROM nombre_de_la_tabla;

En este caso se retornará el contenido de columna1 y columna2 de la tabla seleccionada. Si se quiere consultar toda una tabla, se puede usar el comando:

SELECT * **FROM** nombre_de_la_tabla;

En donde el símbolo * (star) significa todo.

La instrucción select puede ser complementada con otras palabras clave que permiten alterar el comportamiento (filtrar la información resultante).
Por ejemplo, para encontrar los valores únicos en una base de datos, se puede ejecutar el comando:

```
SELECT DISCTINT columna1, columna2...
FROM nombre_de_la_tabla;
```

También se puede contar utilizando SELECT COUNT, el resultado será un número con el conteo de filas resultantes.

No siempre se quiere consultar toda la tabla, de hecho, en la mayoría de casos se quiere filtrar la información resultante basados en condiciones. Para esto se emplea la palabra clave where:

```
SELECT columnas...
FROM nombre_de_la_tabla
WHERE condición;
```

La instrucción where también puede ser utilizada con muchos otros comandos para establecer condiciones para filtrar los resultados, por ejemplo con UPDATE o con DELETE.

Para la instrucción WHERE se requiere algún operador lógico para establecer si una condición es verdadera o falsa.

Operador	Descripción
=	Igualdad
>, >=	Mayor que, mayor o igual que
< , <=	Menor que, menor o igual que
<>	Diferente de (en algunas versiones de SQL se puede emplear !=)
BETWEEN	Valores comprendidos en un rango
LIKE	Valores que cumplen un patrón
IN	Especifica múltiples valores en una columna

Las condiciones se pueden complementar empleando lógica simple, para eso SQL define las operaciones AND, OR y NOT. Esto permite juntar varias condiciones en una sola cláusula de SQL. Por ejemplo, seleccionar los pedidos que fueron hecho entre enero y febrero por el empleado con Id 8. (múltiples condiciones). Existen otras instrucciones que permiten filtrar como es el caso de NOT IN, LIMIT, IS NULL que se pueden emplear complementando consultas.

Instrucciones de ordenamiento

En las consultas es usual requerir los datos ordenados por un criterio (el valor de una columna, por ejemplo) esto se logra con la instrucción ORDER BY. El orden puede ser ascendente o descendente según se requiera.

Instrucciones para juntar tablas: Son instrucciones que permiten unir los datos contenidos en varias tablas, conservando cierta estructura y basados en criterios, por ejemplo, unir dos tablas si el valor de una columna es igual. Las instrucciones de unión son: **JOIN, INNER JOIN, LEFT JOIN, RIGHT JOIN, SELF JOIN, CROSS JOIN**.

INSTRUCCIONES DE AGRUPAMIENTO: Son instrucciones que permiten agrupar datos por tipos o condiciones. Estas instrucciones son **HAVING, GROUP BY, HAVING COUNT, ROLLUP.**

Sub consultas: Son consultas que se hacen sobre una consulta existente. Por ejemplo **SUBQUERY, DERIVED TABLE y EXISTS.**

Operadores de conjuntos: Permiten manipular datos en forma de conjuntos, son **UNION, EXCEPT e INTERSECT.**