

Introducción al lenguaje de consulta estructurado (SQL)

Structured Query Language

Agenda

- Introducción general.
- Sentencias simples en SQL.

Introducción

- ¿Qué es SQL?

SQL Significa "Structured Query Language" (Lenguaje de Consulta Estructurado)

Es un lenguaje estándar para acceder a una Base de datos.

- ¿Que se puede hacer con SQL?

Se puede realizar consultas, como lo indican las siglas SQL, pero también se puede insertar, actualizar y borrar datos.

Además se puede crear y modificar el esquema de la base de datos.

Base de datos

- Base de datos
 - Es un conjunto de datos pertenecientes a un mismo contexto.

Gestores de Bases de datos

- DBMS (Data Base Managment System, Sistema Gestór de Bases de Datos)
 - Son los programas que organizan y almacenan la información de una base de datos. Estos permiten ejecutar diferentes operaciones sobre las bases de datos. Los DBMS también son llamados Motores de Bases de datos.

Características de los DBMS

- Los DBMS proveen diferentes tipos de acceso sobre los datos
 - **Masivo** (operaciones sobre cantidades masivas de información)
 - **Eficiente** (performante)
 - **Seguro** (accesibilidad mediante permisos, integridad ante fallas de software o hardware)
 - **Multi-Usuario** (acceso desde diferentes aplicaciones, sistemas y/o usuarios)
 - **Declarativo** (uso de lenguajes de alto nivel, por ejemplo SQL, solo es necesario saber el "que" y no el "como")
 - **Alta disponibilidad** (accesibilidad en cualquier momento)

Base de datos

- Base de datos relacional

- Es una base de datos que cumple con el modelo relacional (lógica de predicados y teoría de conjuntos).
- Es el modelo de datos más utilizado en la actualidad para implementar bases de datos.
- Permiten establecer relaciones entre los datos, que están guardados en tablas.

Tablas y Datos

- Tabla

- Es un conjunto registros.
- Se refiere al “lugar” donde se guardan los datos. Por ejemplo: Producto es la Tabla de los productos. Visualmente puede asemejarse a una planilla de cálculos.

- Registro

- Es cada una de las filas en que se divide la tabla. Cada registro contiene datos de los mismos tipos que los demás registros. Ejemplo: En una tabla de Productos, cada fila contendrá, obviamente, un Producto.

Tablas y Datos (cont.)

- Campo

- Es cada una de las columnas que forman la tabla. Contienen datos de tipo diferente a los de otros campos. Por ejemplo, en la tabla de Productos, un campo puede corresponder al nombre del producto mientras que otro dato puede corresponder a la fecha en que se creó el producto.

Relación

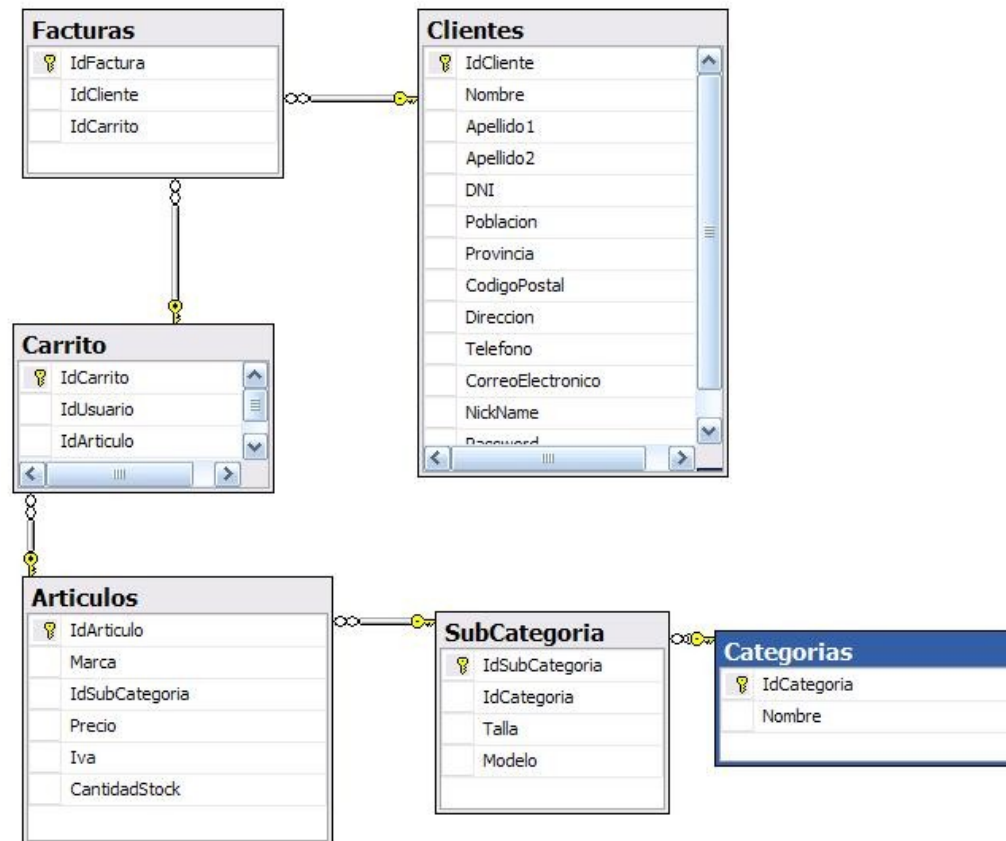
- Relación

- Puede ser el resultado de una Consulta
- Una tabla es una Relación
- Es un conjunto de campos y registros
- A diferencia de una tabla, una Relación no tiene que ser parte de la definición de la estructura de la base de datos

Schema

- Schema (Esquema o Plan)
 - Estructura y los tipos de datos a usar
 - Ejemplo: El campo “precio” de la tabla Producto es de tipo **DECIMAL** (valor decimal con punto fijo), definir si “precio” permite o no valores **NULL** tambien es parte de la definición del tipo de dato.
- DDL (Data Definition Languaje)
 - Lenguaje que permite definir el Schema

Ejemplo de un modelo de datos



Consultas y funciones

- Sentencias básicas:
 - `Select, Insert, Update, Delete`
- Sentencias select:
 - Filtros: `where, like, not in, between`
 - Joins `(left, right)`
 - Order by
 - Group by – Having
 - Consultas anidadas
 - Funciones de Agregación
 - `Count, Sum, Average, Min, Max`
- Funciones propias del motor:
 - `Date, Time, Strings`

Consultas (SELECT)

- ¿Cuál es su uso?

Sirve para obtener información de la base de datos.

- Sintaxis

`SELECT campo1, campo2, ... , campoN`

`•FROM tabla1, tabla2, ... , tablaN`

`•WHERE condicion1, condicion2, ... , condicionN`

Consultas (SELECT)

- Campos calculados

Es un campo que no se almacena físicamente en alguna tabla. Se calcula generalmente con funciones aplicadas a otros campos.

- Ejemplo sintaxis

```
SELECT campo1 + campo2
```

```
•FROM tabla
```

```
•WHERE condicion
```

Ejemplo de SELECT

- Supongamos que existe una base de datos que tiene una tabla llamada Productos.
- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos?

```
SELECT *  
FROM PRODUCTO;
```


Ejemplo de SELECT (cont.)

- Supongamos que la tabla de Productos tiene muchos campos.
- ¿Qué sentencia SQL habría que ejecutar para poder obtener sólo el nombre y el estado todos los productos?

```
SELECT nombre, estado  
FROM PRODUCTO;
```

Ejemplo de SELECT (cont.)

- Supongamos que la tabla de Productos tiene un campo con el stock del día de hoy (llamado stock_hoy) y otro campo con el stock del día de ayer (stock_ayer).
- ¿Qué sentencia SQL habría que ejecutar para poder obtener promedio de stock de los últimos 2 días?

```
SELECT (stock_hoy + stock_ayer) / 2  
FROM PRODUCTO;
```

Cláusula WHERE

- Esta cláusula permite filtrar los resultados obtenidos en una consulta por medio de condiciones o restricciones que se agregan dentro de la consulta.
- Pueden existir muchas condiciones en una misma consulta.
- Las condiciones se unen con operadores AND y OR.
- Las condiciones o restricciones pueden aplicarse a más de un campo.

Cláusula WHERE - Operadores

- Los siguientes son los operadores que pueden utilizarse en una cláusula WHERE:

- = → Igual
- <> → Distinto
- > → Mayor que
- < → Menor que
- >= → Mayor o igual
- <= → Menor o igual

Nota: Algunas versiones de base de datos SQL el operador <> se escribe como !=

Cláusula WHERE – Operadores (cont.)

- Los siguientes son los operadores que pueden utilizarse en una cláusula WHERE:
 - BETWEEN → Entre un rango de valores
 - LIKE → Busca un patrón
 - IN → Compara con varios valores
 - IS → Compara si es, ejemplo NULL

Ejemplo de WHERE

- Considerando la tabla de Productos de los ejemplos anteriores...
- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos cuyo stock de hoy sea menor a 10?

```
SELECT *  
FROM PRODUCTO  
WHERE stock_hoy < 10;
```

Ejemplo de WHERE

- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos cuyo stock de hoy sea menor a 10 y el stock de ayer sea mayor a 100?

```
SELECT *  
FROM PRODUCTO  
WHERE stock_hoy < 10 AND  
       stock_ayer > 100;
```

Cláusula WHERE – BETWEEN

- El operador BETWEEN se utiliza en la cláusula WHERE para elegir todos los registros que se encuentran entre dos valores.
- Los valores pueden ser numéricos, de texto o fechas.
- Las bases de datos tratan distinto las inclusiones de los rangos.
- Sintaxis

```
SELECT *
```

```
FROM tabla
```

```
WHERE campo BETWEEN valor1 AND valor2
```


Cláusula WHERE – (NOT) IN

- El operador IN (o NOT IN) se utiliza en la cláusula WHERE para elegir todos los registros cuyos datos se encuentran en un conjunto de valores.
- Los valores pueden ser numéricos, de texto o fechas.
- Sintaxis

```
SELECT *  
  
FROM tabla  
  
WHERE campo IN (valor1, valor2, ... , valorN)
```

Cláusula WHERE – (NOT) IN

- Las siguientes sentencias son equivalentes

```
SELECT *  
FROM Producto  
WHERE nombre IN ('producto1', 'producto2',  
                'producto3')
```

```
SELECT *  
FROM Producto  
WHERE nombre = 'producto1' OR  
•         nombre = 'producto2' OR  
•         Nombre = 'producto3'
```

Cláusula WHERE – LIKE

- El operador LIKE se utiliza en la cláusula WHERE para elegir todos los registros cuyos datos cumplen con un cierto patrón.
- Patrones con caracteres especiales
 - El signo "%" sirve para coincidir con cualquier cadena de caracteres. Se utiliza en conjunto con otros caracteres.
Por ejemplo '%cadena a buscar'.
 - El signo "_" se utiliza para coincidir con cualquier caracter.
Por ejemplo "_ruta".

Cláusula WHERE – LIKE (cont.)

- Patrones con caracteres especiales
 - Los signos "[]" sirven para coincidir con un caracter de un rango ([a-f]) o un conjunto de caracteres ([abcdef]). Por ejemplo '[PC]edro'
 - Los signos "[^]" es el caso inverso al anterior. Es decir, coincide con todos los caracteres que no se encuentren el conjunto de caracteres específico. Por ejemplo "pa[^sl]o" coincidirá con *pato* pero no con *paso* ni *pato*.

Cláusula WHERE – LIKE (cont.)

- Sintaxis

```
SELECT *  
  
FROM tabla  
  
WHERE campo LIKE patrón
```

Ejemplo de WHERE - BETWEEN

- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos cuyo stock del día de hoy esté entre 1 y 10 unidades?

```
SELECT *  
  FROM PRODUCTO  
 WHERE stock_hoy BETWEEN 1 AND 10;
```

Ejemplo de WHERE - IN

- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos cuyo nombre sea alguno de los de esta lista?
 - Caja de Ahorro
 - Cuenta Corriente
 - Plazo Fijo

```
SELECT *  
FROM PRODUCTO  
WHERE nombre IN ('Caja de Ahorro',  
                 'Cuenta Corriente',  
                 'Plazo Fijo');
```

Ejemplo de WHERE - LIKE

- ¿Qué sentencia SQL habría que ejecutar para poder obtener todos los productos cuyo nombre contenga la palabra 'gratis' en cualquier parte del mismo?

```
SELECT *  
FROM PRODUCTO  
WHERE nombre LIKE '%gratis%';
```


Ejemplo de WHERE - IS

- ¿Qué sentencia SQL habría que ejecutar para poder obtener aquellos empleados que no tengan un jefe asignado?

```
SELECT *  
FROM EMPLEADO  
WHERE idJefe IS NULL;
```

Ejemplo de WHERE – IS NOT

- ¿Qué sentencia SQL habría que ejecutar para poder obtener aquellos empleados que tengan un jefe asignado?

```
SELECT *  
FROM EMPLEADO  
WHERE idJefe IS NOT NULL;
```

Ejercicios

- E01: Obtener todos los registros de los clientes de la base de datos.
- E02: Obtener solo el nombre de todos los clientes de la base de datos.
- E03: Obtener el nombre y el apellido de todos los clientes de la base de datos.
- E04: Obtener los clientes que hayan nacido en la década de los 80'.
- E05: Obtener los clientes cuyo nombre sea “Pedro”.

Ejercicios (cont.)

- E06: Obtener los clientes cuyo apellido sea “Gutierrez”.
- E07: Obtener los clientes cuyo nombre figure en la siguiente lista:
 - Antonio
 - Gabriel
 - Javier
 - Lorenzo

Nota: Hacerlo con la cláusula IN y sin la misma.

Ejercicios (cont.)

- E08: Obtener la lista de clientes cuyo apellido comience con la A, B, C o D.
- E09: Obtener los clientes cuyo nombre comience con “Ana”.
- E10: Obtener los clientes cuyo nombre sea “Lorenzo” o “Lorenza”.

FIN