



*Prof. Flavio Bevilacqua*

*Mail: [flabevy88@gmail.com](mailto:flabevy88@gmail.com)*

*Linkedin: [www.linkedin.com/in/flabevy](http://www.linkedin.com/in/flabevy)*

SQL

# STRUCTURED QUERY LENGUAJE

➤ SQL es un lenguaje universal que permite que todas las base de datos hablen el mismo idioma a modo que ello implique operar con cualquier motor, versión y marca de bases de datos.

➤ SQL permite la interacción con una base de datos relacional, es la única manera de crear bases de datos, agregar tablas, modificar campos, ingestar registros y explotar información.





# STRUCTURED QUERY LENGUAJE

- Los motores de Base de Datos requieren de un lenguaje de definición capaz de generar el diseño del modelo relacional, establecer qué tipo de operaciones se pueden realizar con los datos, asignarle restricciones y determinar cómo se manipulan los mismos.
- Sus iniciales hacen referencia a su funcionamiento, la cual radica en una sintaxis estructurada y la parametrización de consultas.





# ***BASES DE SISTEMA***



# ***BASES DE SISTEMA***

- SQL Server contiene cuatro bases de datos con los fines de autoadministrarse y garantizar su funcionamiento y gestión.
- Estas Bases de Datos no pueden eliminarse ya que en ella se guardan metadatos de cada entidad e información interna.



# FUNDAMENTOS DE SGBD

➤ Un motor de base de datos debe valer por un procesamiento lógico y eficiente de los datos, para ello se establecen algunos parámetros a cumplir de alta relevancia:

1. **MASTER:** La Base Master es el repositorio principal del sistema ya que en ella radica su funcionamiento, inicio y ejecución. Allí, se incluyen los metadatos de cada instancia, configuración, servidores vinculados, como toda otras Bases operativa y existente.
2. **MSDB:** La Base MSBD, es una base Backend donde se almacenan procedimientos, ejecuciones, disparados, alarmas, planes de mantenimiento.
3. **MODEL:** La Base Model se utiliza como plantilla al generar una nueva Base de Datos, replicando su modelo lógico para gestionar su configuración, como contenido.
4. **TEMPDB:** Se utiliza para todos aquellos objetos temporales, como operacionales secuenciales al ejecutar una consulta.







# ***MODULOS***



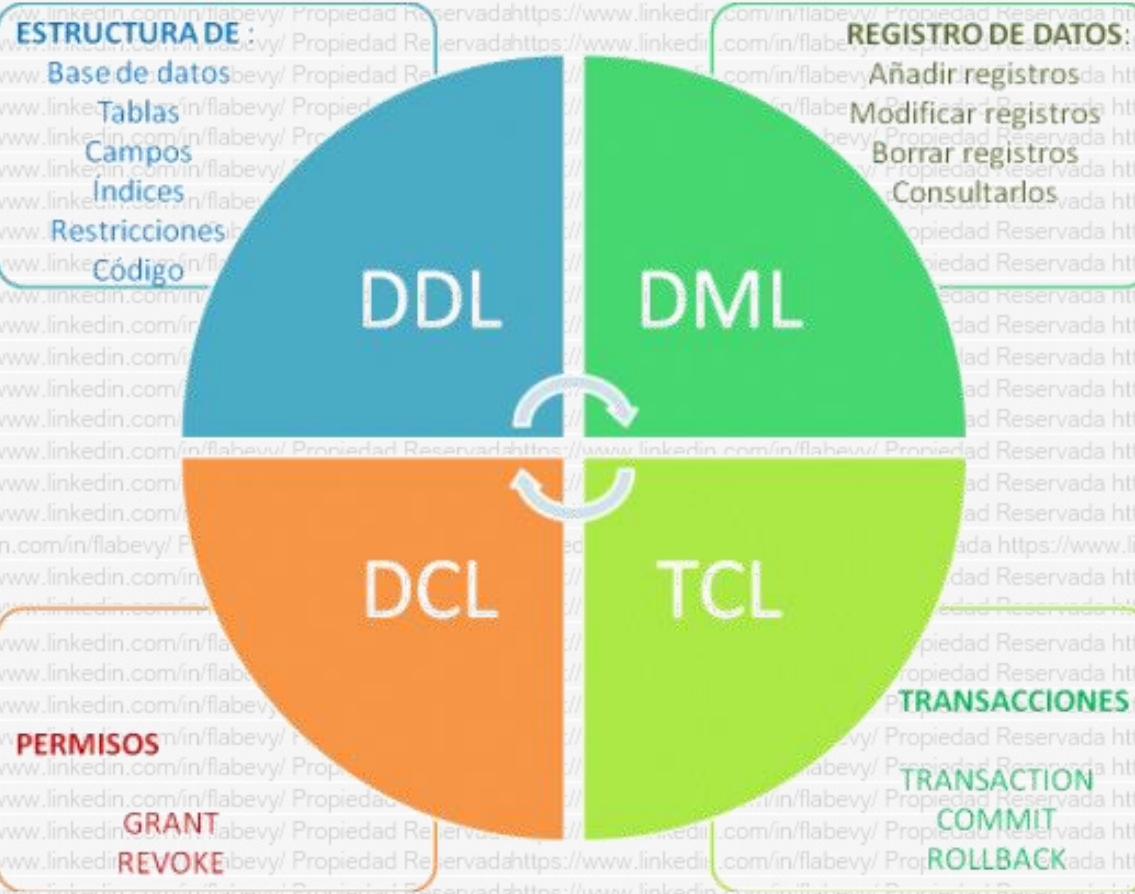
# ***MODULOS DEL LENGUAJE***

- Al ser un lenguaje declarativo orientado a Base de Datos, es necesario contar con una amplia gama de expresiones que permitan combinar la manipulación y la definición de los datos, así como también generar operaciones con los datos almacenados.



- En consecuencia, se derivan desde SQL 4 sublenguajes con diferentes finalidades, como expresiones definidas.

# MODULOS Y SUBLENGUAJES







# **DATA DEFINITION LANGUAGE**



# ***DATA DEFINITION LANGUAGE***

➤ Las Instrucciones del Lenguaje de Datos DDL hace referencia a la definición de los Objetos como a la estructura de la misma.

➤ Los objetos tales a crear pueden ser:

➤ ***DATABASE***

➤ ***SCHEMAS***

➤ ***TABLES and COLUMNS***

➤ ***PROCEDIMIENTOS ALMACENADOS***

➤ ***VISTAS***

➤ Veamos sus comandos principales:



# CREATE

➤ CREATE es una palabra reservada del lenguaje, la cual tiene como función la **CREACIÓN** de objetos dentro de nuestro servidor.

➤ La sintaxis de este comando precede al objeto a generar:

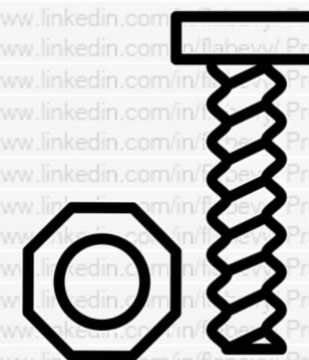
➤ **CREATE DATABASE**

➤ **CREATE SCHEMA**

➤ **CREATE TABLE**

➤ **CREATE VIEW**

➤ **CREATE PROC**





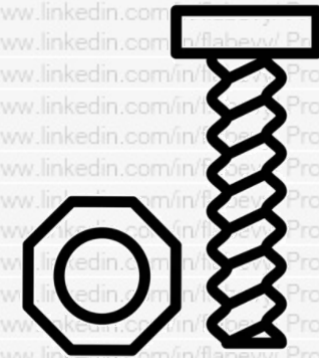
# ALTER

- ALTER es una palabra reservada del lenguaje, la cual tiene como función la **MODIFICACION** de objetos dentro de nuestro servidor.
- ALTER TABLE nos otorga la posibilidad de generar en una instancia posterior las **RELACIONES** entre las diferentes entidades de nuestra Base de Datos, asignando las FK que correspondan.
- La sintaxis de este comando precede al objeto a generar:

➤ **ALTER TABLE**

➤ **ALTER VIEW**

➤ **ALTER PROC**





# **DROP**

➤ DROP es una palabra reservada del lenguaje, la cual tiene como función la **ELIMINACION** de objetos dentro de nuestro servidor.

➤ La sintaxis de este comando precede al objeto a generar:

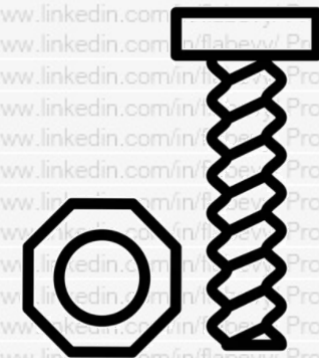
➤ **DROP TABLE**

➤ **DROP VIEW**

➤ **DROP PROC**

➤ **DROP SCHEMA**

➤ **DROP DATABASE**





# ***TIPOS DE DATOS***



# TEXTO

➤ *Estos Tipos de Datos se tienden a indicar entre comillas simples. A su vez, VARCHAR y CHAR requieren de parámetros.*

## TEXT

*Texto de Longitud variable que puede tener hasta 65,535 caracteres*

## VARCHAR

*Se utiliza para almacenar cadenas, en el mismo rango de 255 caracteres, Su particularidad radica en su LONGITUD VARIABLE. Requiere de Parámetros*

## CHAR

*Este tipo se utiliza para almacenar cadenas de texto fijas y de LONGITUD FIJA. Puede almacenar hasta 255 caracteres. Requiere de Parámetros*





# NUMÉRICOS

## INT

*Permite Almacenar Números Enteros en un rango equivalente a 4,294,967,295, pudiendo ser tantos Negativos como Positivos.*

## DECIMAL

*Almacena los Números Decimales, en los cuales deberá indicarse al momento de su asignación, la cantidad de caracteres totales como aquellos por detrás de la coma.*

## SMALLINT

*Permite Números Enteros en un rango de 65,535, pudiendo trabajar también con Números Negativos en caso de requerirlo.*

## TINYINT

*Permite Numeros Enteros en un rango de -128 a 127 o bien, un rango de valores de 255 en caso de asignarlo únicamente como positivos.*



# DATE

➤ *Para hacer referencia a una Fecha, es necesario acompañar estos tipos de datos entre comillas simples.*

## DATE

*Permite almacenar Fechas, el formato se compone de YYYY-MM-DD*

## DATETIME

*Permite almacenar tanto una fecha junto a una hora, min y segundos específicos. Su formato es Año-Mes-Días-Horas-Min-Seg*

## TIME

*Permite almacenar Horas, Minutos y Segundos. El formato de almacenamiento es de 'HH:MM:SS'.*







# ***RESTRICCIONES***



# RESTRICCIONES

- Las restricciones en SQL Server son reglas y restricciones predefinidas que se aplican a los atributos de una entidad y en consecuencia a los valores almacenados. De este modo se mantiene la estructura, integridad y consistencia de la Base de Datos.

## PRIMARY KEY

*Una llave primaria o PRIMARY KEY es una columna o un grupo de columnas que identifica de forma exclusiva cada fila de una tabla, esta restricción tiene implícitamente la restricción UNIQUE y NOT NULL siendo condiciones fundamentales de toda PK.*

## FOREIGN KEY

*La clave foránea identifica una columna o grupo de columnas en una tabla (tabla hija o referendo) que se refiere a una columna o grupo de columnas en otra tabla (tabla maestra o referenciada).*

## IDENTITY

*Mayormente se establece esta restricción sobre el campo clave primaria. Un campo identity no es editable, es decir, no se puede ingresar un valor ni actualizarlo ya que es generado por el sistema de modo automatico al ser auto-incremental*



# RESTRICCIONES

- Las restricciones en SQL Server son reglas y restricciones predefinidas que se aplican a los atributos de una entidad y en consecuencia a los valores almacenados. De este modo se mantiene la estructura, integridad y consistencia de la Base de Datos.

## NOT NULL

*Se asigna esta restricción cuando una columna debe tener un dato válido insertado sin excepciones. Dicho atributo no puede contener Nulls en su interior.*

## UNIQUE

*La restricción UNIQUE garantiza que en un atributo específico, el cual NO ES PK, no se inserten ni contengan valores duplicados.*

## CHECK

*Permite especificar los valores en una columna que debe satisfacer una expresión Booleana.*

## DEFAULT

*La restricción DEFAULT especifica un valor por defecto para un campo cuando en el proceso de Ingesta no se ha indicado uno explícitamente.*





# ***TIPS DE EXPLORACION...***

- Nos proponemos compartir información, siendo partícipes de este cambio constante..
- SQL vs NoSQL: <https://www.youtube.com/watch?v=CuAYLX6reXE>
- Instalacion: <https://www.youtube.com/watch?v=mA1qoWdNCOE&t=245s>
- Modelos: <https://www.youtube.com/watch?v=6pM0Rmj57Vk>
- Datawarehouse: <https://www.youtube.com/watch?v=jFsRdTcljeU>

FIN