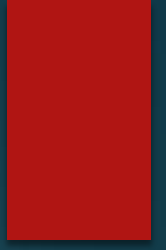


# SQL



# SGBD: SISTEMA GESTOR DE BASES DE DATOS

- MySQL
- MariaDB
- PostgreSQL
- SQLite
- Firebird
- ORACLE
- MS SQL Server
- IBM DB2
- MongoDB
- MS Azure
- Google Cloud ST
- Amazon DynamoDB

## SQL: STRUCTURED QUERY LANGUAGE

DDL

DATA DEFINITION LANGUAGE

DML

DATA MANIPULATION LANGUAGE

# Comandos DDL

Comando	Descripción
CREATE	Utilizado para crear nuevas tablas, campos e índices
DROP	Empleado para eliminar tablas e índices
ALTER	Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.

# Comandos DML

Comando	Descripción
SELECT	Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado
INSERT	Utilizado para cargar lotes de datos en la base de datos en una única operación.
UPDATE	Utilizado para modificar los valores de los campos y registros especificados
DELETE	Utilizado para eliminar registros de una tabla de una base de datos

# Cláusulas SQL

Cláusula	Descripción
FROM	Utilizada para especificar la tabla de la cual se van a seleccionar los registros
WHERE	Utilizada para especificar las condiciones que deben reunir los registros que se van a seleccionar
GROUP BY	Utilizada para separar los registros seleccionados en grupos específicos
HAVING	Utilizada para expresar la condición que debe satisfacer cada grupo
ORDER BY	Utilizada para ordenar los registros seleccionados de acuerdo con un orden específico

# Operadores Lógicos

Operador	Uso
AND	Es el "y" lógico. Evalúa dos condiciones y devuelve un valor de verdad sólo si ambas son ciertas.
OR	Es el "o" lógico. Evalúa dos condiciones y devuelve un valor de verdad si alguna de las dos es cierta.
NOT	Negación lógica. Devuelve el valor contrario de la expresión.

# Operadores de Comparación

Operador	Uso
<	Menor que
>	Mayor que
<>	Distinto de
<=	Menor ó Igual que
>=	Mayor ó Igual que
=	Igual que
BETWEEN	Utilizado para especificar un intervalo de valores.
LIKE	Utilizado en la comparación de un modelo
In	Utilizado para especificar registros de una base de datos

# Funciones de Agregado

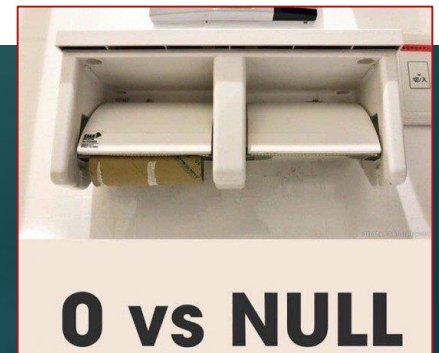
Función	Descripción
AVG	Utilizada para calcular el promedio de los valores de un campo determinado
COUNT	Utilizada para devolver el número de registros de la selección
SUM	Utilizada para devolver la suma de todos los valores de un campo determinado
MAX	Utilizada para devolver el valor más alto de un campo especificado
MIN	Utilizada para devolver el valor más bajo de un campo especificado



# Tipos de Datos

Podemos clasificar los tipos de datos en:

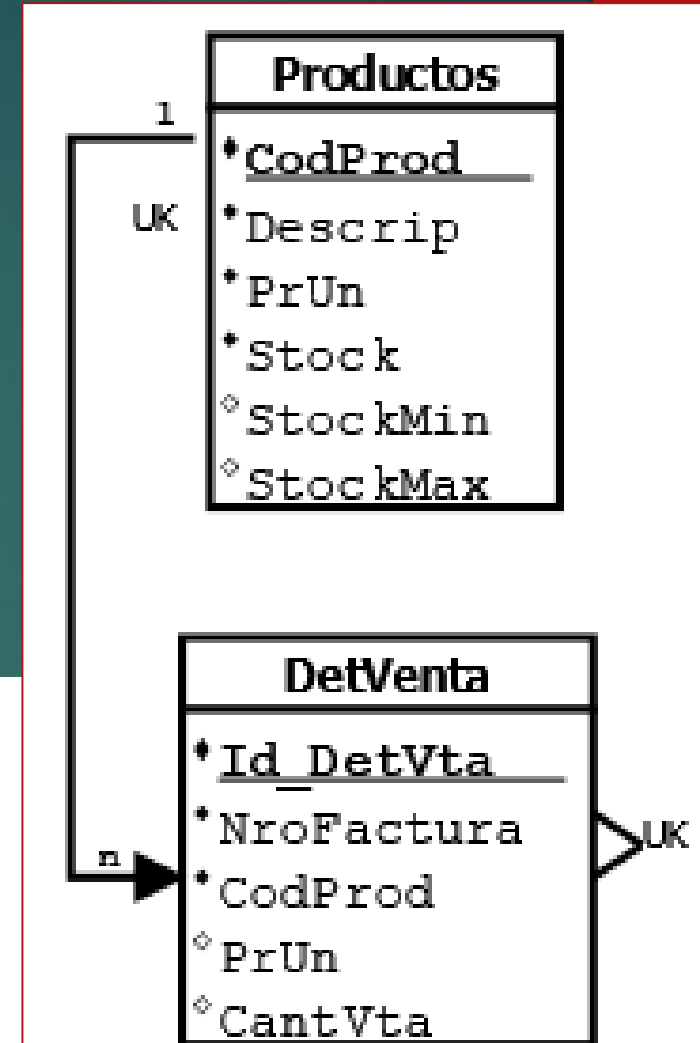
- a) **TEXTO:** Para almacenar texto usamos cadenas de caracteres. Las cadenas se colocan entre comillas simples. Podemos almacenar dígitos con los que no se realizan operaciones matemáticas, por ejemplo, códigos de identificación, números de documentos, números telefónicos. Tenemos los siguientes tipos: varchar, char y text.
- b) **NUMEROS:** Existe variedad de tipos numéricos para representar enteros, negativos, decimales. Para almacenar valores enteros, por ejemplo, en campos que hacen referencia a cantidades, precios, etc., usamos el tipo integer. Para almacenar valores con decimales utilizamos: float o decimal.
- c) **FECHAS Y HORAS:** para guardar fechas y horas dispone de varios tipos: date (fecha), datetime (fecha y hora), time (hora), year (año) y timestamp.
- d) **OTROS TIPOS:** enum y set representan una enumeración y un conjunto respectivamente.
- e) Otro valor que podemos almacenar es el valor "**null**". El valor '**null**' significa "valor desconocido" o "dato inexistente". No es lo mismo que 0 o una cadena vacía.



# Crear Tabla

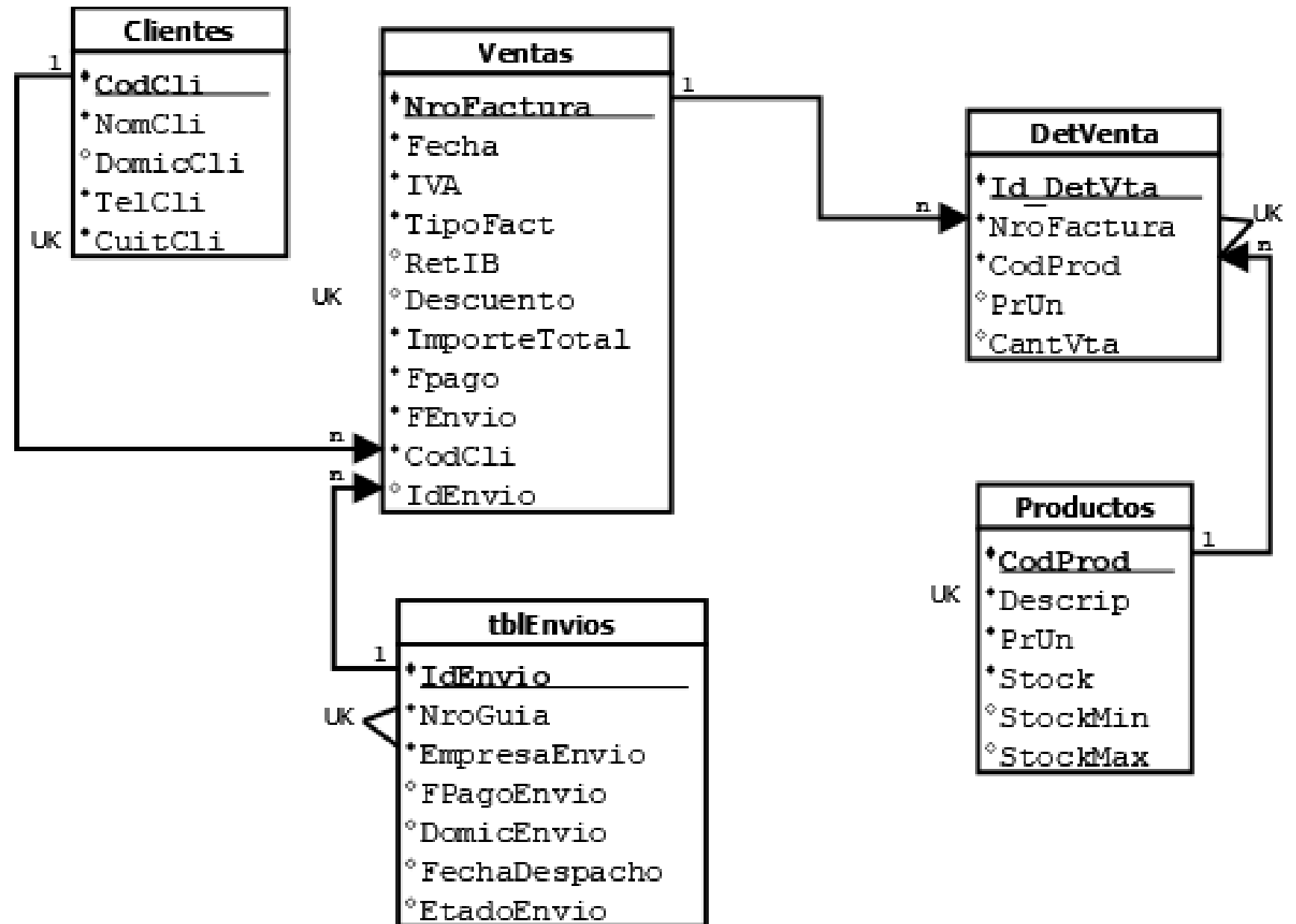
```
CREATE TABLE Productos (  
  CodProd VARCHAR(5) NOT NULL,  
  Descrip VARCHAR(100) NOT NULL,  
  PrUn FLOAT NOT NULL,  
  Stock INT NOT NULL,  
  StockMin INT,  
  StockMax INT,  
  PRIMARY KEY (CodProd),  
  UNIQUE uk_Descrip (Descrip)  
);
```

```
CREATE TABLE DetVenta (  
  id_DetVenta INT AUTO_INCREMENT,  
  NroFactura INT NOT NULL,  
  CodProd VARCHAR(5) NOT NULL,  
  PrUn FLOAT NOT NULL,  
  CantVta INT NOT NULL,  
  PRIMARY KEY (id_DetVenta),  
  UNIQUE uk_Factura_CodProd (NroFactura, CodProd),  
  FOREIGN KEY fk_DetVenta_Productos (CodProd) REFERENCES Productos(CodProd)  
);
```





## Ejercicio



# Ejercicio

