

SQL BASICO ...

# DDL. Crear una tabla

```
CREATE TABLE nombre (
    campo_1          tipo_1 [restricciones_1],
    ...,
    campo_n          tipo_n [restricciones_n],
    [ PRIMARY KEY          (lista de campos), ]
    [ FOREIGN KEY          (lista de campos)
        REFERENCES      NombreTabla(clave) ],
    [restricciones de tabla]
);
```

# DDL. Tipos de datos

- Los tipos de datos dependen del SGBD que se esté utilizando (SQLite)
- **TEXT**: variable de tipo texto que se almacena en formato UTF, aquí tenemos agrupados los diferentes tipos de datos SQL para las variables de texto: CHARACTER(20), VARCHAR(255), VARYING CHARACTER(255), NCHAR(55), NATIVE CHARACTER(70), NVARCHAR(100), TEXT y CLOB.
- **NUMERIC**: NUMERIC, DECIMAL(10,5), BOOLEAN, DATE y DATETIME
- **INTEGER**: es un entero con signo que se almacena con un longitud que va en función del tipo de dato definido: INTEGER, TINYINT, SMALLINT, MEDIUMINT, BIGINT, UNSIGNED BIG INT, INT2 y INT8
- **REAL**: es un dato de tipo float, sus diferentes versiones variarán en la precisión: REAL, DOUBLE, DOUBLE PRECISION y FLOAT
- **BLOB**: los datos se almacenan en el mismo formato en que se introducen.

# DDL. Tipos de datos

```
CREATE TABLE artista
```

```
(
```

```
    id int (4),
```

```
    nombre varchar(100),
```

```
    nacionalidad varchar(100)
```

```
);
```

# INSERT, UPDATE, DELETE

- SQL permite realizar las siguientes acciones de manipulación de datos:

- Inserción de nuevos registros:

```
INSERT INTO tabla [(campo1, campo2,...)] VALUES  
    (valor11, valor12, ...),  
    (valor21, valor22,...),  
    ...
```

- Actualización de registros:

```
UPDATE tabla SET campo1 = valor1, campo2 = valor2,...  
    [WHERE condicion];
```

- Eliminación de registros:

```
DELETE FROM tabla [WHERE condicion];
```

- Vaciado de tablas:

```
TRUNCATE tabla;
```

# DML. Insertar nuevos registros

```
INSERT INTO artista (id, nombre, nacionalidad)
VALUES (2, 'Luis', 'Spain');

INSERT INTO artista VALUES (3, 'Peter', 'UK');

INSERT INTO artista (id, nombre, nacionalidad) VALUES
(1, 'Manuel', 'Spain'),
(2, 'Peter', 'UK'),
(3, 'John', 'US');
```

# DML. Actualizar registros existentes

```
UPDATE artista SET nacionalidad = 'US';

UPDATE artista SET nacionalidad = 'Spain', id = id + 10;

UPDATE artista SET nacionalidad = NULL WHERE nacionalidad = 'FR';

UPDATE artista SET nombre = NULL WHERE nacionalidad = 'Spain';
```

# DML. Borrar datos

```
DELETE FROM artista WHERE nombre = 'Luis';  
TRUNCATE artista;
```

# DQL. Consulta de datos

```
SELECT [DISTINCT] campo1, campo2, ...  
FROM tabla1, tabla2, ...  
[WHERE condición];
```

```
SELECT * FROM tabla1, tabla2, ...  
[WHERE condición];
```

# DQL. OPERADORES Y FUNCIONES

- Matemáticos: +, -, \*, /
- Comparación: =, <>, <, >, <=, >=
- Lógicos: NOT, AND, OR
- Valores nulos: IS NULL
- Rangos: BETWEEN x AND y
- Fragmentos de cadenas: x LIKE y
- Pertenencia a conjuntos: IN

# DQL. Ordenación y agregación

```
SELECT COUNT (*) FROM tablas  
[WHERE condición]
```

```
SELECT COUNT ([DISTINCT] campo) FROM tablas  
[WHERE condición]  
[GROUP BY campo1, campo2, ...];
```

```
SELECT SUM | MAX | MIN | AVG (campo) FROM tablas  
[WHERE condición]  
[GROUP BY campo1, campo2, ...];
```

```
SELECT campos FROM tablas  
[WHERE condición]  
[ORDER BY campo1, campo2, ... [DESC]];
```

# DQL. Validación de datos agregados

**SELECT** *campos, agregación* **FROM** *tablas*

[**WHERE** *condición*]

[**GROUP BY** *campos*]

[**HAVING** *condición*];

# Resumen final

- Comandos DDL
  - **CREATE** – para crear elementos de la BD
  - **ALTER** – para modificar la estructura de la BD
  - **DROP** – para eliminar elementos de la BD
  - **GRANT/REVOKE** – proporciona o quita privilegios a los usuarios de la BD
- Comandos DML
  - **SELECT** – recupera datos
  - **INSERT** – crea un nuevo dato
  - **UPDATE** – actualiza datos existentes
  - **DELETE/TRUNCATE** – borra datos
- Otros comandos:
  - **COMMIT** – finaliza una transacción consolidando las modificaciones
  - **SAVEPOINT** - marca un punto de retorno en una transacción
  - **ROLLBACK** – finaliza una transacción sin consolidar los datos
  - **EXPLAIN PLAN** – muestra el plan de acceso de una consulta