



MÓDULO II PROGRAMACIÓN DE BASES DE DATOS RELACIONALES

Unidad Formativa 2 Definición y Manipulación de Datos

Lenguajes Relacionales

LENGUAJES RELACIONALES

| MÓDULOS | UNIDADES FORMATIVAS |
|---|---|
| PROGRAMACIÓN DE BASES DE DATOS RELACIONALES | DISEÑO DE BASES DE DATOS RELACIONALES Introducción a las bases de datos Modelos conceptuales de bases de datos El modelo relacional El ciclo de vida de un proyecto Creación y diseño de bases de datos |
| | DEFINICIÓN Y MANIPULACIÓN DE DATOS Lenguajes relacionales El lenguaje de manipulación de la base de datos |
| | DESARROLLO DE PROGRAMAS EN EL ENTORNO DE LA BASE DE DATOS Lenguajes de programación de bases de datos |

EL LENGUAJE DE MANIPULACIÓN DE DATOS

| MÓDULOS | UNIDADES FORMATIVAS |
|---|---|
| PROGRAMACIÓN DE BASES DE DATOS RELACIONALES | DISEÑO DE BASES DE DATOS RELACIONALES Introducción a las bases de datos Modelos conceptuales de bases de datos El modelo relacional El ciclo de vida de un proyecto Creación y diseño de bases de datos |
| | DEFINICIÓN Y MANIPULACIÓN DE DATOS Lenguajes relacionales El lenguaje de manipulación de la base de datos |
| | DESARROLLO DE PROGRAMAS EN EL ENTORNO DE LA BASE DE DATOS Lenguajes de programación de bases de datos |

SQL - Structured Query Language

- SQL es un lenguaje “no procedimental”
 - Se indica lo que se quiere obtener y no el “cómo”
 - El “cómo” lo decide el “optimizer” (optimizador de sentencias)

SQL (Structured Query Language)

DDL

CREATE
DROP
ALTER
RENAME
TRUNCATE

DML

SELECT
INSERT
UPDATE
DELETE

DCL

GRANT
REVOKE

TCL

START
TRANSACTION
COMMIT
ROLLBACK
SAVEPOINT

Tipos de Datos – MySQL – nros. enteros

- BIT [(M)] M indica el nro de bits. Entre 1 y 64. Por defecto 1.
(En el resto de tipos enteros, tanto M como ZEROFILL, están marcados como deprecated)
- TINYINT [UNSIGNED]. Entre (-128 a 127) ó (0 a 255)
- BOOL, BOOLEAN. Sinónimo de TINYINT (probar en consola)
- SMALLINT [UNSIGNED]. Entre (-32768 a 32767) ó (0 a 65535)
- MEDIUMINT [UNSIGNED]. Entre (-8388608 a ...)
- INT [UNSIGNED]. Entre (-2.147.483.648 a ...)
- INTEGER [UNSIGNED]. Sinónimo de INT
- BIGINT [UNSIGNED]. Entre (-9223372036854775808 a ...)

Tipos de Datos – MySQL – nros. reales

(En los tipos reales tanto `UNSIGNED` como `ZEROFILL`, están marcados como deprecated)

- `DECIMAL [(M [,D])]`.
 - `M` indica el nro total de dígitos y `D` los decimales.
- `DEC...` Sinónimo de `DECIMAL`

No estándares en SQL

- `FLOAT [(M ,D)]`.
- `FLOAT [(p)]`
- `DOUBLE [(M ,D)]`
- `DOUBLE PRECISION [(M ,D)]`

Tipos ENTEROS

| Tipo | Bytes | Valor mínimo | Valor máximo cs | UNSIGNED | |
|-----------|-------|--------------|-----------------|--------------|--------------|
| | | | | Valor mínimo | Valor máximo |
| TINYINT | 1 | -128 | 127 | 0 | 255 |
| SMALLINT | 2 | -32768 | 32767 | 0 | 65535 |
| MEDIUMINT | 3 | -8388608 | 8388607 | 0 | 16777215 |
| INT | 4 | -2147483648 | 2147483647 | 0 | 4294967295 |
| BIGINT | 8 | -2^{63} | $2^{63}-1$ | 0 | $2^{64}-1$ |

Tipos de Datos – MySQL – temporales

- DATE. De '1000-01-01' a '9999-12-31'.
- DATETIME. De '1000-01-01 00:00:00' a '9999-12-31 23:59:59'
- TIMESTAMP. '1970-01-01 00:00:00' hasta el año 2037.
- TIME. de '-838:59:59' a '838:59:59'.
- YEAR [(2 | 4)]. Por defecto 4 dígitos. Para 2: 70 a 69 (1970 a 2069).

Cadenas y binarios

- CHAR [(M)]. Cadena de largo fijo. Siempre M caracteres. Por def. 1.
- VARCHAR [(M)]. Cadena de largo variable, máximo M caracteres.
- TEXT, TINYTEXT, MEDIUMTEXT, LONGTEXT
- BLOB, TINYBLOB, MEDIUMBLOB, LONGBLOB

DDL – Data Definition Language

Definición de Bases de Datos/Esquemas

CREATE DATABASE | SCHEMA [IF NOT EXISTS] <nombre_bbdd>
[**CHARACTER SET** <charset>] [**COLLATE** <collate>]

CREATE DATABASE ejemplo **CHARACTER SET** utf8mb4;

DROP DATABASE | SCHEMA [IF EXISTS] <nombre_bbdd>

ALTER DATABASE | SCHEMA <nombre_bbdd> <cambios_especificaciones>

Seleccionar una Base de Datos

USE <nombre_bbdd>

Mostrar sentencia de creación

SHOW CREATE DATABASE <nombre_bbdd>

DDL – Data Definition Language

CREATE TABLE

CREATE TABLE <nombre_tabla> (<definiciones de columnas>)

- ```
CREATE TABLE `productos` (
 `id_producto` int unsigned NOT NULL,
 `producto` varchar(100) NOT NULL,
 `precio` decimal(10,2) NOT NULL
);
```

# CREATE TABLE

**CREATE TABLE** <nombre\_tabla> (<definiciones de columnas>, **PRIMARY KEY** (<columnas>))

- ```
CREATE TABLE `productos` (  
  `id_producto` int unsigned NOT NULL AUTO_INCREMENT,  
  `producto` varchar(100) NOT NULL,  
  `precio` decimal(10,2) NOT NULL,  
  PRIMARY KEY(id_producto)  
);
```
- ```
CREATE TABLE `productos` (
 `id_producto` int unsigned PRIMARY KEY NOT NULL AUTO_INCREMENT,
 `producto` varchar(100) NOT NULL,
 `precio` decimal(10,2) NOT NULL
);
```

# CREATE TABLE- REFERENCES

- ```
CREATE TABLE `productos` (  
  `id_producto` int unsigned NOT NULL AUTO_INCREMENT,  
  `producto` varchar(100) NOT NULL,  
  `precio` decimal(10,2) NOT NULL,  
  `fk_fabricante` int unsigned DEFAULT NULL,  
  PRIMARY KEY(id_producto),  
  CONSTRAINT `productos_ibfk_1` FOREIGN KEY (`fk_fabricante`) REFERENCES `fabricantes` (`id_fabricante`)  
);
```

CREATE [**TEMPORARY**] **TABLE** [**IF NOT EXISTS**] nombre_tabla
(**definicion**,...)

definicion:

col_nom **col_def**

| [**CONSTRAINT** [nombre_c]] **PRIMARY KEY** (col_nom,...)

| [**CONSTRAINT** [nombre_c]] **FOREIGN KEY** (col_nom,...) **reference_def**

| **CHECK** (expr)

col_def : tipo_dato [**NOT NULL** | **NULL**] [**DEFAULT** valor] [**AUTO_INCREMENT**]
[**UNIQUE** [**KEY**] | [**PRIMARY KEY**]]

reference_def: **REFERENCES** nombre_tabla (col_nom,...)

[**ON DELETE** **opciones_ref**]

[**ON UPDATE** **opciones_ref**]

opciones_ref : **RESTRICT** | **CASCADE** | **SET NULL** | **NO ACTION**

Opciones Foreign Key

ON DELETE | ON UPDATE

RESTRICT | CASCADE | SET NULL | NO ACTION

RESTRICT Impide que se puedan eliminar las filas referenciadas o cambiar su pk

CASCADE Permite que se puedan actualizar o eliminar las filas referenciadas

SET NULL Asigna NULL a las fK que apuntan a filas eliminadas o cambiada su pk

NO ACTION Equivalente a RESTRICT.

Modificación y Eliminación Tablas

DROP [**TEMPORARY**] **TABLE** [**IF EXISTS**] nombre_tabla [, nombre_tabla];

ALTER TABLE nombre_tabla **MODIFY** Permite modificar el tipo de dato y restricciones de una columna.

ALTER TABLE nombre_tabla **CHANGE** Permite renombrar una columna, modificar el tipo de dato y sus restricciones.

ALTER TABLE nombre_tabla **ADD** Permite añadir una nueva columna. Se puede agregar al principio **FIRST** o después de alguna columna **AFTER** nombre_columna

ALTER TABLE nombre_tabla **DROP** Permite eliminar una columna

Mostrar listado de tablas

SHOW TABLES

Mostrar información de la estructura

DESC|DESCRIBE nombre_tabla

Mostrar sentencia de creación

SHOW CREATE TABLE nombre_tabla

SELECT

SELECT <lista atributos>
FROM <nombre_tabla>
WHERE <condiciones>

```
SELECT ingrediente, calorías FROM ingredientes;
```

```
SELECT ingrediente, calorías FROM ingredientes  
WHERE calorías < 50;
```

SELECT <lista atributos>
FROM <lista nombre_tablas>
WHERE <condiciones>

```
SELECT plato, ingrediente FROM platos, ingredientes  
WHERE platos.id_ingrediente_principal = ingredientes.id_ingrediente;
```

```
SELECT plato, ingrediente as 'Ingrediente principal' FROM platos, ingredientes  
WHERE platos.id_ingrediente_principal = ingredientes.id_ingrediente;
```

SELECT

- Campos calculados
- Alias de campos
- Modificadores **ALL** (opcional), **DISTINCT**, **DISTINCTROW** (sinónimo)
- **[ORDER BY {nombre_col | expr} [ASC | DESC], ...]**
- **[LIMIT {[offset,] row_COUNT | row_COUNT OFFSET offset}]**

Algunas funciones

| CADENAS | |
|---------------------|----------------------------|
| CONCAT | Concatena |
| CONCAT_WS | Concatena con separador |
| LOWER | Minúsculas |
| UPPER | Mayúsculas |
| SUBSTR | Subcadena |
| LEFT - RIGHT | Subcadenas |
| CHAR_LENGTH | Cantidad de caracteres |
| LTRIM | Elimina espacios izq. |
| RTRIM | Elimina espacios der. |
| TRIM | Elimina espacios izq y der |
| REPLACE | Reemplazar caracteres |

| MATEMÁTICAS | |
|-------------------|----------------|
| ABS() | Valor absoluto |
| POW(x,y) | x elevado a y |
| SQRT() | Raíz cuadrada |
| PI() | π |
| ROUND() | Redondeo |
| TRUNCATE() | Trunca |
| CEIL | Redondeo der. |
| FLOOR | Redondeo izq. |
| MOD | Módulo |

| FECHA Y HORA | |
|------------------|---------------------|
| NOW() | Fecha y hora actual |
| CURTIME() | Hora actual |
| ADDDATE | Agrega días a fecha |
| DATEDIFF | Dif. entre 2 fechas |
| YEAR | Año de una fecha |
| MONTH | Mes... |
| MONTHNAME | Nombre del mes |
| DAY | Día de ... |
| DAYNAME | Nombre del día |
| HOURL | Horas |
| MINUTE | Minutos |
| SECOND | Segundos |

SELECT ... WHERE...

- Condiciones para comparar valores o expresiones.
- Condiciones para comprobar si un valor está dentro de un rango de valores.
- Condiciones para comprobar si un valor está dentro de un conjunto de valores.
- Condiciones para comparar cadenas con patrones.
- Condiciones para comprobar si una columna tiene valores a NULL.

Operadores más utilizados

| Aritméticos | |
|-------------|----------|
| + | Suma |
| - | Resta |
| * | Producto |
| / | División |
| % | Módulo |

| Relacionales | |
|------------------|---------------|
| < | Menor |
| <= | Menor o igual |
| > | Mayor |
| >= | Mayor o igual |
| = | Igual |
| <> ó != | Distinto |
| BETWEEN...AND... | Entre |
| IN | Incluido en |
| LIKE | Patrón (% _) |
| IS [NOT] NULL | Es o no nulo? |

| Lógicos | |
|----------|-------------|
| AND ó && | Y lógico |
| OR ó | O lógico |
| NOT ó ! | Neg. lógica |

SELECT ... WHERE...

- Condiciones para comparar valores o expresiones. **>, >=, <, ...**
- Condiciones para comprobar si un valor está dentro de un rango de valores. **BETWEEN**
- Condiciones para comprobar si un valor está dentro de un conjunto de valores. **IN ()**
- Condiciones para comparar cadenas con patrones. **LIKE**
- Condiciones para comprobar si una columna tiene valores a NULL. **IS NULL**

SELECT resumido para una Tabla

SELECT [**DISTINCT**] expr [, expr ...]
[**FROM** tabla]
[**WHERE** condicion]
[**GROUP BY** {columna | expr | posición} [**ASC** | **DESC**], ... [**WITH ROLLUP**]
[**HAVING** having_condition]
[**ORDER BY** {col_name | expr | position} [**ASC** | **DESC**], ...]
[**LIMIT** {[offset,] row_COUNT | row_COUNT OFFSET offset}]

GROUP BY

SELECT expr [, expr ...]

FROM tabla

[**WHERE** condicion]

[**GROUP BY** {columna | expr | posición} [**ASC** | **DESC**], ...]

| Funciones de agregación | |
|-------------------------|-----------------------------|
| AVG() | Promedio / Media aritmética |
| SUM() | Sumatorio |
| COUNT() | Cantidad de elementos |
| MAX() | Máximo valor |
| MIN() | Mínimo valor |

SELECT varias tablas

- Producto cartesiano
 - ... **FROM** tabla1, tabla2; **SQL 1 (SQL-86)**
 - ... **FROM** tabla1 **JOIN** tabla2; (**INNER JOIN = JOIN**) **SQL 2 (SQL-92)**
- Filtrando resultados
 - ... **FROM** tabla1, tabla2 **WHERE** fk_1 = pk_2;
 - ... **FROM** tabla1 **JOIN** tabla2 **ON** fk_1 = pk_2;

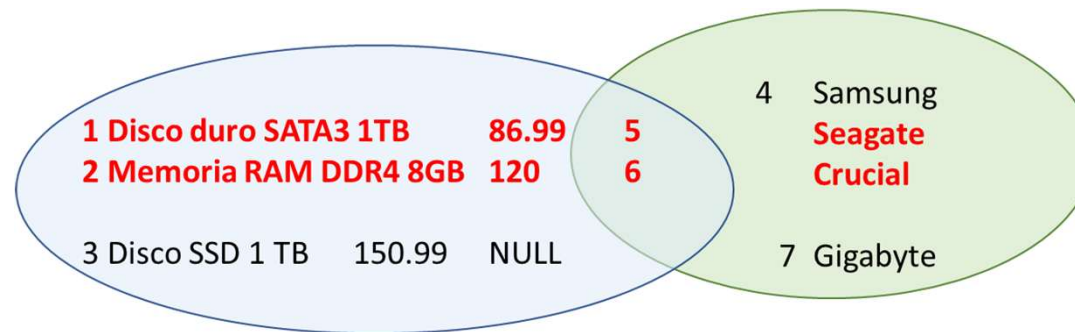
JOIN

| | | |
|------------------------|--------|------|
| 1 Disco duro SATA3 1TB | 86.99 | 5 |
| 2 Memoria RAM DDR4 8GB | 120 | 6 |
| 3 Disco SSD 1 TB | 150.99 | NULL |

| | |
|---|----------|
| 4 | Samsung |
| 5 | Seagate |
| 6 | Crucial |
| 7 | Gigabyte |

| | | | |
|------------------------|--------|------|------------|
| 1 Disco duro SATA3 1TB | 86.99 | 5 | 4 Samsung |
| 2 Memoria RAM DDR4 8GB | 120 | 6 | 5 Seagate |
| 3 Disco SSD 1 TB | 150.99 | NULL | 6 Crucial |
| | | | 7 Gigabyte |

JOIN INTERNO



SELECT ... **FROM** productos **INNER JOIN** fabricantes
ON productos.fk_fabricante = fabricante.id_fabricante;

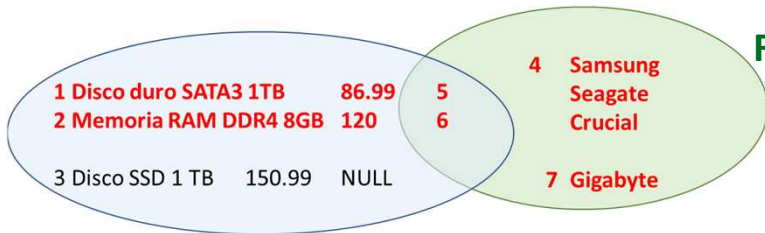
| Id_prod | producto | precio | Id_fabr | fabricante |
|---------|---------------|--------|---------|------------|
| 1 | Disco duro... | 86.99 | 5 | Seagate |
| 2 | Memoria ... | 120 | 6 | Crucial |

JOIN EXTERNOS



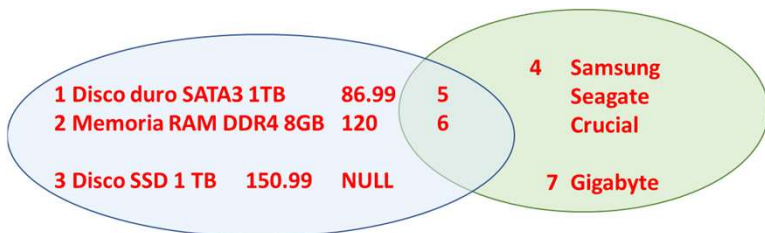
LEFT OUTER JOIN
LEFT JOIN

| Id_prod | producto | precio | Id_fabr | fabricante |
|---------|---------------|--------|---------|------------|
| 1 | Disco duro... | 86.99 | 5 | Seagate |
| 2 | Memoria ... | 120 | 6 | Crucial |
| 3 | Disco SSD ... | 150.99 | NULL | NULL |



RIGHT OUTER JOIN
RIGHT JOIN

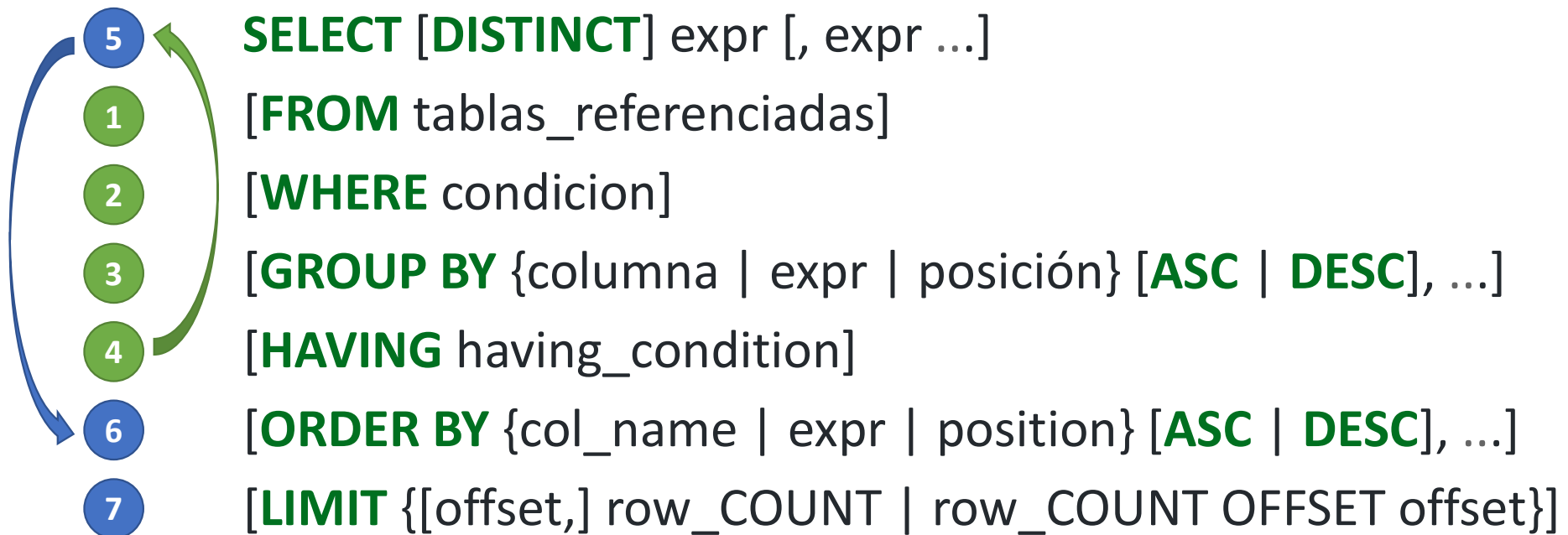
| Id_prod | producto | precio | Id_fabr | fabricante |
|---------|---------------|--------|---------|------------|
| 1 | Disco duro... | 86.99 | 5 | Seagate |
| 2 | Memoria ... | 120 | 6 | Crucial |
| NULL | NULL | NULL | 4 | Samsung |
| NULL | NULL | NULL | 7 | Gigabyte |



FULL OUTER JOIN
no soportado
MySQL

| Id_prod | producto | Precio | Id_fabr | fabricante |
|---------|---------------|--------|---------|------------|
| 1 | Disco duro... | 86.99 | 5 | Seagate |
| 2 | Memoria ... | 120 | 6 | Crucial |
| NULL | NULL | NULL | 4 | Samsung |
| NULL | NULL | NULL | 7 | Gigabyte |
| 3 | Disco SSD ... | 150.99 | NULL | NULL |

SELECT resumido



¿En qué orden se ejecuta el SELECT?