

Funciones



¿Qué es SQL?

- **Structured Query Language** (Lenguaje de Consulta Estructurado)
- **DDL** (Lenguaje de Definición de Datos): Es el lenguaje encargado de la definición, alteración y eliminación de objetos en la base de datos. (CREATE, ALTER, DROP)
- **DML** (Lenguaje de Manipulación de Datos): Es el lenguaje encargado de la manipulación de los datos. (SELECT, INSERT, UPDATE, DELETE)
- **DCL** (Lenguaje de Control de Datos): Es el lenguaje que controla el acceso a las bases de datos. (GRANT, REVOKE)
- **TCL** (Lenguaje de Control de Transacciones): Es el lenguaje que controla la ejecución de comandos DML; indispensable para la consistencia e integridad de los datos.

Funciones

Una función es una rutina creada para tomar parámetros, procesarlos y regresar un resultado.

- Pueden o no recibir parámetros.
- Siempre regresan un valor con algún tipo de dato definido.
- Existen funciones predefinidas en MySQL.
 - (funciones de tiempo, texto, agregación)
- Se pueden crear nuevas funciones en MySQL.
- Se ejecutan con la instrucción:
 - `SELECT nombre_funcion([params]);`

Funciones de texto

Manejo de cadenas

Funciones de texto (manejo de cadenas)

1. La función CONCAT permite concatenar cadenas de texto

```
SELECT CONCAT(' " ',titulo,' " tiene ',cantidad_disponible,' discos en stock') FROM disco;
```

```
SELECT CONCAT('El disco: " ',titulo,' " cuesta $',precio)  
FROM disco;
```

2. La función CONCAT_WS permite concatenar cadenas de texto con un separador definido

```
SELECT CONCAT_WS('--',titulo,precio,cantidad_disponible) FROM disco;
```

3. La función LENGTH permite calcular el tamaño de una cadena

```
SELECT titulo,LENGTH(titulo) from disco;
```

Se pueden utilizar varias funciones anidadas

```
SELECT CONCAT('""',titulo,'" tiene ',length(titulo),' caracteres.') FROM  
disco;
```

Funciones de texto (manejo de cadenas)

4. Las funciones UCASE y LCASE permiten convertir cadenas a mayúsculas y minúsculas respectivamente

```
SELECT titulo, UCASE(titulo), LCASE(titulo) from disco;
```

5. Las funciones LPAD y RPAD permiten definir un llenado para columnas

```
SELECT LPAD(titulo,30,'.'), RPAD(titulo,30,'-') from disco;
```

6. Las funciones LEFT, RIGHT y SUBSTR, permiten extraer caracteres o subcadenas de otra cadena

```
SELECT titulo, LEFT(titulo,5) FROM disco;
```

```
SELECT titulo, RIGHT(titulo,5) FROM disco;
```

Se indica la posición inicial y la cantidad de caracteres a extraer

```
SELECT titulo, SUBSTR(titulo,4,3) FROM disco;
```

Funciones de texto (manejo de cadenas)

7. Las funciones LTRIM y RTRIM permiten quitar espacios a la izquierda y a la derecha respectivamente.

```
SELECT LTRIM("          Cadena con espacios          ");
```

```
SELECT RTRIM("          Cadena con espacios          ");
```

8. La función TRIM permite eliminar espacios en blanco tanto a la derecha como a la izquierda:

```
SELECT TRIM("          Cadena con espacios          ");
```

9. La función TRIM puede usarse para eliminar caracteres sólo a la izquierda mediante la palabra **leading**:

```
SELECT TRIM(LEADING '*' from '****Ejemplo***');
```

Funciones de texto (manejo de cadenas)

10. La función LOCATE permite localizar la posición de una subcadena dentro de otra cadena, regresa el índice

```
SELECT titulo, LOCATE('el',titulo) FROM disco;
```

```
SELECT nombre LOCATE('el',nombre) FROM artista;
```

11. La función REPLACE permite reemplazar una cadena por otra

```
SELECT REPLACE('2323 Fulerton Avenue', 'Avenue', 'Ave.');
```

Nota: El reemplazo se realiza en la cadena, no en el valor del registro.

Funciones de tiempo

Manejo de fechas

Funciones de tiempo (manejo de fechas)

1. Muestra el día de la semana empezando por lunes en 0:

```
SELECT WEEKDAY('1964-10-12');
```

2. Muestra el nombre del día:

```
SELECT DAYNAME('1964-10-12');
```

3. Muestra el día de la semana empezando por domingo en 1

```
SELECT DAYOFWEEK('1964-10-12');
```

WEEKDAY value	DAYOFWEEK value	Day of the week
0	2	Monday
1	3	Tuesday
2	4	Wednesday
3	5	Thursday
4	6	Friday
5	7	Saturday
6	1	Sunday

Funciones de tiempo (manejo de fechas)

```
select fecha_lanzamiento,  
weekday(fecha_lanzamiento),  
dayname(fecha_lanzamiento),  
dayofweek(fecha_lanzamiento)  
from disco;
```

```
mysql> select fecha_lanzamiento,weekday(fecha_lanzamiento),dayname(fecha_lanzamiento),dayofweek(fecha_lanzamiento) from disco;
```

fecha_lanzamiento	weekday(fecha_lanzamiento)	dayname(fecha_lanzamiento)	dayofweek(fecha_lanzamiento)
2020-10-30	4	Friday	6
2004-09-14	1	Tuesday	3
2017-04-28	4	Friday	6
2016-12-09	4	Friday	6
2007-12-03	0	Monday	2
2011-01-19	2	Wednesday	4
2019-03-29	4	Friday	6
2020-03-27	4	Friday	6
2013-10-18	4	Friday	6
2017-11-03	4	Friday	6
2015-01-01	3	Thursday	5
2020-01-01	2	Wednesday	4
2007-01-01	0	Monday	2
2019-01-01	1	Tuesday	3
2019-01-01	1	Tuesday	3

Funciones de tiempo (manejo de fechas)

4. Muestra el número correspondiente al mes:

```
SELECT MONTH('1964-10-12');
```

5. Muestra el nombre del mes:

```
SELECT MONTHNAME('1964-10-12');
```

Mostrar el mes y el nombre del mes de las fechas de lanzamiento de los primeros 10 discos:

```
SELECT fecha_lanzamiento, MONTH(fecha_lanzamiento),  
MONTHNAME(fecha_lanzamiento)  
FROM disco limit 10;
```

fecha_lanzamiento	MONTH(fecha_lanzamiento)	MONTHNAME(fecha_lanzamiento)
2020-10-30	10	October
2004-09-14	9	September
2017-04-28	4	April
2016-12-09	12	December
2007-12-03	12	December
2011-01-19	1	January
2019-03-29	3	March
2020-03-27	3	March
2013-10-18	10	October
2017-11-03	11	November

Funciones de tiempo (manejo de fechas)

```
SELECT fecha_lanzamiento,  
MONTH(fecha_lanzamiento),  
MONTHNAME(fecha_lanzamiento),  
WEEKDAY(fecha_lanzamiento),  
DAYOFWEEK(fecha_lanzamiento),  
DAYNAME(fecha_lanzamiento)  
FROM disco limit 10;
```

```
mysql> SELECT fecha_lanzamiento,MONTH(fecha_lanzamiento),MONTHNAME(fecha_lanzamiento),WEEKDAY(fecha_lanzamiento),DAYOFWEEK(fecha_lanzamiento),DAYNAME(fecha_lanzamiento) FROM di
```

fecha_lanzamiento	MONTH(fecha_lanzamiento)	MONTHNAME(fecha_lanzamiento)	WEEKDAY(fecha_lanzamiento)	DAYOFWEEK(fecha_lanzamiento)	DAYNAME(fecha_lanzamiento)
2020-10-30	10	October	4	6	Friday
2004-09-14	9	September	1	3	Tuesday
2017-04-28	4	April	4	6	Friday
2016-12-09	12	December	4	6	Friday
2007-12-03	12	December	0	2	Monday
2011-01-19	1	January	2	4	Wednesday
2019-03-29	3	March	4	6	Friday
2020-03-27	3	March	4	6	Friday
2013-10-18	10	October	4	6	Friday
2017-11-03	11	November	4	6	Friday

10 rows in set (0.00 sec)

Funciones de tiempo (manejo de fechas)

6. Muestra la semana del año:

```
SELECT WEEK('2006-12-24');
```

7. Muestra el año:

```
SELECT YEAR('1964-10-12');
```

Muestra la semana y el año:

```
select fecha_lanzamiento,  
week(fecha_lanzamiento),  
year(fecha_lanzamiento) from disco limit 10;
```

```
mysql> select fecha_lanzamiento, week(fecha_lanzamiento), year(fecha_lanzamiento)
+-----+-----+-----+
| fecha_lanzamiento | week(fecha_lanzamiento) | year(fecha_lanzamiento) |
+-----+-----+-----+
| 2020-10-30        | 43                      | 2020                    |
| 2004-09-14        | 37                      | 2004                    |
| 2017-04-28        | 17                      | 2017                    |
| 2016-12-09        | 49                      | 2016                    |
| 2007-12-03        | 48                      | 2007                    |
| 2011-01-19        | 3                       | 2011                    |
| 2019-03-29        | 12                      | 2019                    |
| 2020-03-27        | 12                      | 2020                    |
| 2013-10-18        | 41                      | 2013                    |
| 2017-11-03        | 44                      | 2017                    |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

Funciones de tiempo (manejo de fechas)

8. Agregar días a una fecha determinada:

```
select fecha_lanzamiento,  
DATE_ADD(fecha_lanzamiento,INTERVAL 10 DAY)  
from disco;
```

9. Quitar días a una fecha determinada:

```
select fecha_lanzamiento,  
DATE_SUB(fecha_lanzamiento,INTERVAL 10 DAY)  
from disco;
```

10. Diferencia entre fechas y devuelve el número de días entre dos fechas:

```
SELECT id_disco,titulo,fecha_lanzamiento,datediff('2022-10-10',fecha_lanzamiento) as dias_transcurridos from disco limit 10;
```

Funciones de tiempo

1. Muestra la hora y los minutos:

```
SELECT hour('04:46:45'),MINUTE('04:46:45');
```

2. Concatena la hora y los minutos:

```
SELECT CONCAT_WS("-",hour('04:46:45'),MINUTE('04:46:45'));
```

3. Muestra la fecha y hora actual

```
SELECT NOW( );
```

```
SELECT SYSDATE();
```

```
SELECT CURRENT_TIMESTAMP();
```

Todos son equivalentes

```
mysql> SELECT NOW(), SYSDATE(),CURRENT_TIMESTAMP();
```

NOW()	SYSDATE()	CURRENT_TIMESTAMP()
2020-11-12 14:23:08	2020-11-12 14:23:08	2020-11-12 14:23:08

```
1 row in set (0.00 sec)
```


Funciones de tiempo

4. Muestra sólo la fecha actual

```
SELECT CURDATE( );
```

5. Muestra sólo la hora actual

```
SELECT CURTIME( );
```

6. Agregar días a la fecha actual

```
SELECT DATE_ADD(NOW( ), INTERVAL 10 DAY);
```

7. Quitar días a la fecha actual

```
SELECT DATE_SUB(NOW( ), INTERVAL 10 DAY);
```

Funciones aritméticas

Funciones aritméticas

1. La función ROUND permite redondear la cantidad especificada

```
SELECT ROUND(28.5), ROUND(-28.5);  
SELECT titulo,precio,ROUND(precio) FROM disco;
```

Otro uso de ROUND es para establecer el número de decimales

```
SELECT titulo,precio,ROUND(precio,1) FROM disco;
```

2. La función ABS permite obtener el valor absoluto de la cantidad especificada

```
SELECT ABS(-14.7), ABS(14.7);
```

3. La función **FORMAT** permite dar formato a una cantidad especificada

```
select titulo,format(datediff('2022-10-10',fecha_lanzamiento),0) as  
dias_transcurridos from disco limit 10;
```

¿Preguntas?