
Funciones para manipulación de texto

El siguiente apunte registra y clasifica las diferentes funciones de para manipular texto con T-SQL. No requiere de una base de datos para su ejemplificación ya que todos los ejemplos trabajan con cadenas de texto literales para su mejor entendimiento.

Funciones para obtener parte de un texto

LEFT()

Uso:

Devuelve los primeros **n** caracteres de un texto.

Sintaxis:

```
LEFT(expresión, número_de_caracteres)
```

Ejemplo:

```
SELECT LEFT('SQL Server', 3);
```

Resultado:

```
SQL
```

RIGHT()

Uso:

Devuelve los últimos **n** caracteres de un texto.

Sintaxis:

```
RIGHT(expresión, número_de_caracteres)
```

Ejemplo:

```
SELECT RIGHT('SQL Server', 6);
```

Resultado:

Server

SUBSTRING()

Uso:

Extrae un fragmento de texto a partir de una posición específica

Sintaxis:

```
SUBSTRING(expresión, posición_inicial, longitud)
```

Ejemplo:

```
SELECT SUBSTRING('Microsoft SQL Server', 11, 3);
```

Resultado:

SQL

Funciones para convertir texto

CAST() y CONVERT()

Uso:

Convierte un texto a otro tipo de dato o viceversa

Sintaxis:

```
CAST(expresión AS tipo_dato)
CONVERT(tipo_dato, expresión, [formato])
```

Ejemplos:

```
SELECT CAST('1234' AS VARCHAR(4));
SELECT CONVERT(VARCHAR(4), '1234');
```

Resultados:

```
1234
1234
```

FORMAT()

Uso:

Da formato a valores, útil para fechas, números o cadenas.

Sintaxis:

```
FORMAT(valor, formato, [cultura])
```

Aclaraciones:

El formato de la función format dependerá del tipo del valor enviado como primer parámetro. El mismo puede ser escapado según sea necesario como puede verse en el

Ejemplo 2.

Ejemplo 1: Fecha estándar sin la hora

```
SELECT FORMAT(getdate(), 'dd/mm/yyyy');
```

Resultado:

25/02/2025

Ejemplo 2: Fecha sin la hora y con el nombre del mes en español de Argentina

```
Select FORMAT(getdate(), 'dd \d\e MMMM \d\e yyyy', 'es-Ar');
```

Resultado:

25 de febrero de 2025

Ejemplo 3: Fecha sin la hora y con el nombre del día de la semana y mes en español de Argentina

```
Select FORMAT(getdate(), 'dddd, dd MMMM yyyy', 'es-Ar');
```

Resultado:

martes, 25 febrero 2025

Ejemplo 4: Números con formato de moneda

```
Select FORMAT(13456.44, 'C', 'es-AR');  
Select FORMAT(13456.44, 'C', 'es-ES');
```

Resultados:

\$ 13.456,44
13.456,44 €

Ejemplo 5: Formato de porcentaje

```
SELECT FORMAT(0.1234, 'P');
```

Resultado:

12.34%

Ejemplo 6: Añadir ceros a la izquierda

```
SELECT FORMAT(5, '0000');  
SELECT FORMAT(55, '0000');  
SELECT FORMAT(555, '0000');  
SELECT FORMAT(5555, '0000');  
SELECT FORMAT(55555, '0000');
```

Resultados:

0005
0055
0555
5555
55555

Ejemplo 7: Formato personalizado

```
SELECT FORMAT(1122334455, '(###) ####-####');
```

Resultado:

(11) 2233-4455

REPLACE()

Uso:

Reemplaza las ocurrencias de una subcadena dentro de un texto con otra subcadena

Sintaxis:

```
REPLACE(texto, texto_a_buscar, texto_de_reemplazo)
```

Ejemplo:

```
SELECT REPLACE('UTN FRGP', 'FRGP', 'Facultad Regional General Pacheco');  
SELECT REPLACE('UTN FRGP FRGP', ' FRGP', '');
```

Resultado:

```
UTN Facultad Regional General Pacheco  
UTN
```

STUFF()**Uso:**

Reemplaza un fragmento de texto dentro de una cadena iniciando en una posición específica y a partir de la longitud indicada.

Sintaxis:

```
STUFF(texto, posición, longitud, nuevo_texto)
```

Ejemplo:

```
SELECT STUFF('UTN FRGP: Partido de Tigre', 5, 4, 'Facultad Regional  
General Pacheco');
```

Resultado:

```
UTN Facultad Regional General Pacheco: Partido de Tigre
```

REPLICATE()

Uso:

Repite un texto un número especificado de veces

Sintaxis:

```
REPLICATE(texto, cantidad_repeticiones);
```

Ejemplo:

```
SELECT REPLICATE('UTN', 5);
```

Resultado:

```
UTNUTNUTNUTNUTN
```

Upper() y Lower()

Uso:

Convierte una cadena todo a mayúsculas o todo a minúsculas

Sintaxis:

```
UPPER(texto)  
LOWER(texto)
```

Ejemplo:

```
SELECT UPPER('Base de Datos 2');  
SELECT LOWER('Base de Datos 2');
```

Resultado:

```
BASE DE DATOS 2
```

Otras funciones de texto útiles

LTrim(), RTrim() y Trim()

Uso:

Elimina espacios en blanco. LTrim elimina los espacios en blanco desde el inicio y RTrim elimina los espacios en blanco del final. Trim elimina los espacios en blanco del inicio y del final.

Sintaxis:

```
LTRIM(texto)
RTRIM(texto)
TRIM(texto)
```

Ejemplo:

```
SELECT LTRIM('   Base de Datos 2   ');
SELECT RTRIM('   Base de Datos 2   ');
SELECT TRIM('   Base de Datos 2   ');
```

Resultado:

```
Base de Datos 2   -- Espacios a la derecha
Base de Datos 2
Base de Datos 2
```

CHARINDEX()

Uso:

Busca la posición de una subcadena dentro de un texto desde la posición inicial deseada

Sintaxis:

```
CHARINDEX(subcadena, texto, [posicion_inicial])
```


Ejemplo:

```
SELECT CHARINDEX('Datos', 'Base de Datos II: La materia Base de Datos II es correlativa con Base de Datos I'); -- Busca la subcadena 'Datos' desde el principio de la cadena.
```

```
SELECT CHARINDEX('Datos', 'Base de Datos II: La materia Base de Datos II es correlativa con Base de Datos I', 15); -- Busca la subcadena 'Datos' desde la posición 15 de la cadena en adelante.
```

```
SELECT CHARINDEX('Datos', 'Base de Datos II: La materia Base de Datos II es correlativa con Base de Datos I', 150); -- Busca la subcadena 'Datos' desde la posición 150 de la cadena en adelante.
```

Resultado:

```
9
38
0
```

El primer resultado significa que en la novena posición se encuentra la primera ocurrencia de la palabra Datos dentro de la cadena de búsqueda. Partiendo desde 1.

El segundo resultado significa que en la posición 38 de la cadena de búsqueda se encontró la palabra Datos partiendo desde la posición 15.

El tercer resultado significa que no se encontró la palabra Datos en la cadena de búsqueda partiendo de la posición 150. También ocurriría lo mismo si se proporciona un término de búsqueda que no se encuentra en la cadena. El resultado devuelto cuando no hay ocurrencias es la posición 0 si el texto es una cadena y NULL si el texto es NULL.

LEN() y DATALENGTH()

Uso:

Len devuelve la longitud del texto sin contar los espacios al final. DataLength devuelve el tamaño en bytes de una cadena.

Sintaxis:

LEN(expresión)
DATALENGTH(expresión)

Ejemplo:

```
SELECT LEN('SQL Server ');  
SELECT DATALENGTH('SQL Server ');  
SELECT DATALENGTH(N'SQL Server ');  
SELECT DATALENGTH('🐱🚀');  
SELECT DATALENGTH('🐱');
```

Resultados:

10
14
28
6
2

El primero de los resultados devuelve 10 porque 'SQL Server ' tiene 10 caracteres sin contar los espacios del final.

El segundo de los resultados devuelve 14 porque 'SQL Server ' ocupa 14 bytes en memoria.

El tercero de los resultados devuelve 28 porque 'SQL Server ' expresado como una cadena Unicode ocupa 28 bytes en memoria.

El cuarto y quinto de los resultados devuelven valores distintos a pesar de ser emojis ambos. Es porque la cantidad de bytes dependerá del emoji debido a su composición.