



Funciones Transact-SQL

Este capítulo se presenta como una referencia rápida acerca del uso de las funciones escalares incorporadas en Transact-SQL.

Esta página se ha dejado en blanco intencionalmente.

Capítulo 8

Funciones Transact-SQL

Contenido

- ❑ **Ejercicio 74:** Funciones de configuración
 - ✓ @@DateFirst
 - ✓ @@DBTS
 - ✓ @@LangID
 - ✓ @@Language
 - ✓ @@Lock_timeout
 - ✓ @@Max_connections
 - ✓ @@Max_precision
 - ✓ @@NestLevel
 - ✓ @@Options
 - ✓ @@RemServer
 - ✓ @@ServerName
 - ✓ @@ServiceName
 - ✓ @@SPID
 - ✓ @@TextSize
 - ✓ @@Version
- ❑ Funciones de fecha y hora
- ❑ **Ejercicio 75:** Funciones numéricas
 - ✓ Abs(expresiónNumérica)
 - ✓ Acos(expresiónFloat)
 - ✓ Asin(expresiónFloat)
 - ✓ Atan(expresiónFloat)
 - ✓ Atn2(expresiónFloat1, expresiónFloat2)

- ✓ *Ceiling*(expresiónNumérica)
- ✓ *Cos*(expresiónFloat)
- ✓ *Cot*(expresiónFloat)
- ✓ *Degrees*(expresiónNumérica)
- ✓ *Exp*(expresiónFloat)
- ✓ *Floor*(expresiónNumérica)
- ✓ *Log*(expresiónFloat)
- ✓ *Log10*(expresiónFloat)
- ✓ *Pi*()
- ✓ *Power*(expresiónNumérica, n)
- ✓ *Radians*(expresiónNumérica)
- ✓ *Rand*([semilla])
- ✓ *Round*(expresiónNumérica, precisión [, función])
- ✓ *Sign*(expresiónNumérica)
- ✓ *Sin*(expresiónFloat)
- ✓ *Square*(expresiónFloat)
- ✓ *SQRT*(expresiónFloat)
- ✓ *Tan*(expresiónFloat)
- **Ejercicio 76:** Funciones de metadatos
 - ✓ *Col_length*(tabla, columna)
 - ✓ *Col_name*(idTabla, idColumna)
 - ✓ *ColumnProperty*(idObjeto, columna, propiedad)
 - ✓ *DatabasePropertyEx*(basedatos, propiedad)
 - ✓ *DB_ID*([basedatos])
 - ✓ *DB_name*([idBasedatos])
 - ✓ *File_IDEx*(nombreLógicoArchivo)
 - ✓ *File_name*(idNombreLógicoArchivo)
 - ✓ *FileProperty*(nombreArchivo, propiedad)

- ✓ *Index_col*(tabla, idIndice, idColumna)
- ✓ *IndexProperty*(idObjeto, nombreIndice, propiedad)
- ✓ *Object_ID*(nombreObjeto)
- ✓ *Object_name*(idObjeto)
- ✓ *ObjectPropertyEx*(idObjeto, propiedad)
- ✓ *Type_ID*(nombreTipo)
- ✓ *Type_name*(idTipo)
- ✓ *TypeProperty*(nombreTipo, propiedad)
- **Ejercicio 77:** Funciones de seguridad
 - ✓ *Is_member*(grupo_ó_rol)
 - ✓ *Is_SrvRoleMember*(rol [, login])
 - ✓ *SUser_SID*(login)
 - ✓ *SUser_SName*(server_user_ID)
- **Ejercicio 78:** Funciones de cadena
 - ✓ *ASCII*(expresiónCadena)
 - ✓ *Char*(expresiónEntera)
 - ✓ *CharIndex*(expresiónCadena1, expresiónCadena2 [, inicioBúsqueda])
 - ✓ *Difference*(expresiónCadena1, expresiónCadena2)
 - ✓ *Left*(expresiónCadena, expresiónEntera)
 - ✓ *Len*(expresiónCadena)
 - ✓ *Lower*(expresiónCadena)
 - ✓ *Ltrim*(expresiónCadena)
 - ✓ *Nchar*(expresiónEntera)
 - ✓ *PatIndex*('%patrónBúsqueda%', expresiónCadena)
 - ✓ *QuoteName*(expresiónCadena, caracterDelimitador)
 - ✓ *Replace*(expresiónCadena1, expresiónCadena2, expresiónCadena3)
 - ✓ *Replicate*(expresiónCadena, expresiónEntera)
 - ✓ *Reverse*(expresiónCadena)

- ✓ *Right(expresiónCadena, expresiónEntera)*
- ✓ *Rtrim(expresiónCadena)*
- ✓ *Soundex(expresiónCadena)*
- ✓ *Space(expresiónEntera)*
- ✓ *Str(expresiónFloat [, longitud [, decimal]])*
- ✓ *Stuff(expresiónCadena1, inicio, n, expresiónCadena2)*
- ✓ *Substring(expresiónCadena, inicio, n)*
- ✓ *Unicode(expresiónCadenaUnicode)*
- ✓ *Upper(expresiónCadena)*

□ **Ejercicio 79:** Otras funciones

- ✓ *Cast()*
- ✓ *Convert()*
- ✓ *Coalesce(listaExpresiones)*
- ✓ *Current_timestamp*
- ✓ *Current_user*
- ✓ *DataLength(expresión)*
- ✓ *GetAnsiNull([basedatos])*
- ✓ *Host_name()*
- ✓ *Ident_current(tabla)*
- ✓ *Ident_incr(tabla)*
- ✓ *Ident_seed(tabla)*
- ✓ *@@Identity*
- ✓ *IsDate(expresión)*
- ✓ *IsNull(expresiónEvaluada, valorReemplazo)*
- ✓ *IsDate(expresión)*
- ✓ *NullIf(expresión1, expresión2)*
- ✓ *@@RowCount*
- ✓ *RowCount_big()*

- ✓ *Scope_identity()*
- ✓ *ServerProperty(propiedad)*
- ✓ *SessionProperty(propiedad)*
- ✓ *Session_user*
- ✓ *System_user*
- ✓ *User_name([idUsuario])*

Esta página se ha dejado en blanco intencionalmente.

Funciones Transact-SQL

En este capítulo veremos las funciones escalares Transact-SQL; es decir, aquellas funciones que retornan un solo valor. Más adelante veremos otros tipos de funciones Transact-SQL, como las funciones de agregación y las funciones de conjunto de filas (RowSet).

Ejercicio 74: Funciones de configuración

Las funciones de configuración entregan el valor actual de las opciones de configuración de la sesión.

@@DateFirst

Retorna el valor actual especificado para SET DATEFIRST. Esta opción especifica cuál el primer día de la semana. Para la versión de SQL Server que estamos utilizando el primer día de la semana es el domingo (día 7).

Ejemplo

```
SELECT @@DateFirst
go
-- Retorna 7
```

@@DBTS

Retorna el valor del último dato **timestamp** utilizado en la base de datos actual.

Ejemplo

```
USE MarketPERU
go

SELECT @@DBTS
go
```

@@LangID

Retorna el identificador del idioma local ó el del idioma actualmente en uso. Use **sp_helpLanguage** para obtener información acerca de los identificadores de idioma.

Ejemplo

```
SELECT @@LangID
go
-- Retorna 0 (us_english)

sp_helpLanguage
go
```

@@Language

Retorna el nombre del idioma actualmente en uso.

Ejemplo

```
SELECT @@Language
go
-- Retorna us_english
```

@@Lock_timeout

Retorna el tiempo de espera antes del bloqueo (en milisegundos) para la sesión actual. Retorna -1 si SET LOCK_TIMEOUT no se ha ejecutado en la sesión actual.

Ejemplo

```
SELECT @@Lock_timeout
go
-- Retorna -1
```

@@Max_connections

Retorna el número máximo de conexiones de usuario simultáneas permitidas para el servidor SQL.

Ejemplo

```
SELECT @@Max_connections
go
-- Retorna 32767
```

@@Max_precision

Retorna el nivel de precisión para los datos **decimal** y **numeric** configurado actualmente en el servidor.

Ejemplo

```
SELECT @@Max_precision
go
-- Retorna 38
```

@@NestLevel

Retorna el nivel de anidamiento actual durante la ejecución de un procedimiento almacenado. El valor inicial es 0 (cero).

Ejemplo

```
SELECT @@NestLevel
go
-- Retorna 0
```

@@Options

Retorna un entero que representa el conjunto de las opciones configuradas para la conexión de usuario actual.

Ejemplo

```
SELECT @@Options
go
-- Retorna 5496 (default)
```

@@RemServer

Retorna el nombre del servidor SQL remoto al que se ha conectado el usuario.

Ejemplo

```
SELECT @@RemServer
go
-- Retorna NULL
```

@@ServerName

Retorna el nombre del servidor SQL local.

Ejemplo

```
SELECT @@ServerName
go
-- Retorna DESARROLLO
```

@@ServiceName

Retorna la clave del registro bajo la que está corriendo el servidor SQL. El valor predeterminado es MSSQLSERVER para la instancia por defecto; si es una instancia nombrada, retorna el nombre de la instancia.

Ejemplo

```
SELECT @@ServiceName
go
-- Retorna MSSQLSERVER
```

@@SPID

Retorna el identificador del proceso de servidor del proceso del usuario actual.

Ejemplo

```
SELECT @@SPID
go
-- Retorna 52 (DESARROLLO\Administrator)
```

@@TextSize

Retorna el valor actual de la opción TEXTSIZE. Esta especifica la longitud máxima en bytes que los datos **varchar**, **nvarchar**, **varbinary**, **text** ó **image** pueden retornar en una instrucción SELECT.

Ejemplo

```
SELECT @@TextSize
go
-- Retorna 2147483647
```

@@Version

Retorna la fecha, versión y tipo de procesador del servidor SQL.

Ejemplo

```
SELECT @@Version
go
/*
Retorna Microsoft SQL Server 2005 - 9.00.1116 (Intel X86)
Apr  9 2005 20:56:37
Copyright (c) 1988-2004 Microsoft Corporation
Beta Edition on Windows NT 5.2 (Build 3790: )
*/
```

Funciones de fecha y hora

Las funciones de fecha y hora ejecutan una operación sobre un valor de fecha y hora, y retornan una cadena, un número, u otro valor de fecha y hora.

Vea el Capítulo 7: Consultas sencillas, páginas 240 a 243.

Ejercicio 75: Funciones numéricas

Las funciones numéricas ejecutan un cálculo en base a valores de entrada recibidos como parámetros retornando un valor numérico.

Abs(expresiónNumérica)

Retorna el valor absoluto de **expresiónNumérica**.

Ejemplo

```
SELECT abs(-5755)
go
-- Retorna 5755
```

Acos(expresiónFloat)

Retorna el ángulo en radianes cuyo coseno es especificado por **expresiónFloat**. El valor de **expresiónFloat** debe estar en el rango de -1 a 1.

Ejemplo

```
SELECT acos(-1)
go
-- Retorna 3.14159265358979
```

Asin(expresiónFloat)

Retorna el ángulo en radianes cuyo seno es especificado por **expresiónFloat**. El valor de **expresiónFloat** debe estar en el rango de -1 a 1.

Ejemplo

```
SELECT asin(1)
go
-- Retorna 1.5707963267949
```

Atan(expresiónFloat)

Retorna el ángulo en radianes cuya tangente es especificada por **expresiónFloat**.

Ejemplo

```
SELECT atan(0)
go
-- Retorna 0
```

Atn2(expresiónFloat1, expresiónFloat2)

Retorna el ángulo en radianes cuya tangente es el cociente de **expresiónFloat1** y **expresiónFloat2**.

Ejemplo

```
SELECT atn2(1, 0)
go
-- Retorna 1.5707963267949
```

Ceiling(expresiónNumérica)

Retorna el entero más pequeño que es mayor ó igual que **expresiónNumérica**.

Ejemplo

```
SELECT ceiling(27.157)
go
-- Retorna 28
```


Cos(expresiónFloat)

Retorna el coseno del ángulo en radianes especificado por **expresiónFloat**.

Ejemplo

```
SELECT cos(3.14159265358979)
go
-- Retorna -1
```

Cot(expresiónFloat)

Retorna la cotangente del ángulo en radianes especificado por **expresiónFloat**.

Ejemplo

```
SELECT cot(pi()/4)
go
-- Retorna 1
```

Degrees(expresiónNumérica)

Retorna el ángulo en grados para el ángulo en radianes especificado por **expresiónNumérica**.

Ejemplo

```
SELECT degrees(pi()/4)
go
-- Retorna 45
```

Exp(expresiónFloat)

Retorna el valor de **e** (base de los logaritmos naturales) elevado a la potencia especificada por **expresiónFloat**.

Ejemplo

```
SELECT exp(1)
go
-- Retorna 2.71828182845905
```

Floor(expresiónNumérica)

Retorna el entero más grande que es menor ó igual que **expresiónNumérica**.

Ejemplo

```
SELECT floor(27.157)
go
-- Retorna 27
```

Log(expresiónFloat)

Retorna el logaritmo natural de **expresiónFloat**.

Ejemplo

```
SELECT log(2.71828182845905)
go
-- Retorna 1
```

Log10(expresiónFloat)

Retorna el logaritmo de base 10 de **expresiónFloat**.

Ejemplo

```
SELECT log10(1000)
go
-- Retorna 3
```

Pi()

Retorna el valor de la constante **PI**.

Ejemplo

```
SELECT pi()
go
-- Retorna 3.14159265358979
```

Power(expresiónNumérica, n)

Retorna e resultado de elevar **expresiónNumérica** a la potencia **n**.

Ejemplo

```
SELECT power(4, 5)
go
-- Retorna 1024
```

Radians(expresiónNumérica)

Retorna el ángulo en radianes para el ángulo en grados especificado por **expresiónNumérica**.

Ejemplo

```
SELECT radians(180.0)
go
-- Retorna 3.141592653589793100
```

Rand([semilla])

Genera un número aleatorio entre 0 y 1.

Ejemplo

```
SELECT rand()
go
```

Round(expresiónNumérica, precisión [, función])

Retorna el valor de **expresiónNumérica** redondeado a la **precisión** especificada. Cuando **función** es diferente a **0** (cero), trunca el resultado del redondeo.

Ejemplo

```
SELECT round(748.5845, 0) -- 749.0000
SELECT round(748.5845, 1) -- 748.6000
SELECT round(748.5845, 2) -- 748.5800
SELECT round(748.5845, 3) -- 748.5850
SELECT round(748.5845, 4) -- 748.5845
SELECT round(748.5845, -1) -- 750.0000
SELECT round(748.5845, -2) -- 700.0000
SELECT round(748.5845, -3) -- Error
SELECT round(748.5845, -4) -- 0.0000
go
```

```
SELECT round(748.5845, 0) -- 749.0000
SELECT round(748.5845, 0, 1) -- 748.0000
go

SELECT round(748.5845, 1) -- 748.6000
SELECT round(748.5845, 1, 1) -- 748.5000
go
```

Sign(expresiónNumérica)

Retorna el signo de **expresiónNumérica**. Retorna **+1**, **0**, ó **-1**.

Ejemplo

```
SELECT sign(-180)
go
-- Retorna -1
```

Sin(expresiónFloat)

Retorna el seno del ángulo en radianes especificado por **expresiónFloat**.

Ejemplo

```
SELECT sin(pi()/2)
go
-- Retorna 1
```

Square(expresiónFloat)

Retorna el cuadrado de **expresiónFloat**.

Ejemplo

```
SELECT square(15)
go
-- Retorna 225
```

SQRT(expresiónFloat)

Retorna la raíz cuadrada de **expresiónFloat**.

Ejemplo

```
SELECT sqrt(225)
go
-- Retorna 15
```

Tan(expresiónFloat)

Retorna la tangente del ángulo en radianes especificado por **expresiónFloat**.

Ejemplo

```
SELECT tan(pi()/4)
go
-- Retorna 1
```

Ejercicio 76: Funciones de metadatos

Las funciones de metadatos retornan información acerca de una base de datos y de sus objetos.

Col_length(tabla, columna)

Retorna la longitud en bytes de la **columna** especificada de la **tabla** especificada.

Ejemplo

```
USE MarketPERU
go

SELECT col_length('producto', 'nombre')
go
-- Retorna 40
```

Col_name(idTabla, idColumna)

Retorna el nombre de la columna para el **idColumna** especificado en la **idTabla** especificada.

Ejemplo

```
SELECT col_name(object_id('producto'), 1)
go
-- Retorna idProducto
```

ColumnProperty(idObjeto, columna, propiedad)

Retorna el valor de la **propiedad** especificada de la **columna** de **idObjeto**, donde este último puede ser una tabla ó un procedimiento.

Ejemplo

```
SELECT columnProperty(object_id('producto'),
    'unidadMedida', 'AllowsNull')
go
-- Retorna 1 (True, si permite nulos)
```

DatabasePropertyEx(basedatos, propiedad)

Retorna el valor de la **propiedad** especificada para **basedatos**.

Ejemplo

```
SELECT databasePropertyEx('MarketPERU', 'Collation')
go
-- Retorna Latin1_General_CI_AI
```

DB_ID([basedatos])

Retorna el identificador de la base de datos actual ó de la **basedatos** especificada.

Ejemplo

```
SELECT db_id('AdventureWorks')
go
-- Retorna 8 (en mi servidor SQL)
```


DB_name([idBasedatos])

Retorna el nombre de la base de datos actual ó para la **idBasedatos** especificada.

Ejemplo

```
SELECT db_name()  
go
```

File_IDEx(nombreLógicoArchivo)

Retorna el identificador para el **nombreLógicoArchivo** de la base de datos actual.

Ejemplo

```
SELECT file_idEx('MarketPERU_log')  
go
```

File_name(idNombreLógicoArchivo)

Retorna el nombre del archivo lógico para el **idNombreLógicoArchivo** de la base de datos actual.

Ejemplo

```
SELECT file_name(2)  
go
```

FileProperty(nombreArchivo, propiedad)

Retorna el valor de **propiedad** para el **nombreArchivo** especificado.

Ejemplo

```
SELECT fileProperty('MarketPERU_log', 'isLogFile')  
go  
-- Retorna 1 (True)
```

Index_col(tabla, idIndice, idColumna)

Retorna el nombre de la columna indexada. **idColumna** es la posición de la columna en **tabla**, y **idIndice** es el identificador del índice.

Ejemplo

```
SELECT index_col('Producto', 1, 1)
go
-- Retorna idProducto
```

IndexProperty(idObjeto, nombreIndice, propiedad)

Retorna el valor de **propiedad** para el **nombreIndice** del **idObjeto** especificado.

Ejemplo

```
SELECT indexProperty(object_id('Producto'),
    'pk__producto__00551192', 'indexFillFactor')
go
-- Retorna 0 (FillFactor = 0)
```

Object_ID(nombreObjeto)

Retorna el valor del identificador de **nombreObjeto**.

Ejemplo

```
SELECT object_id('Producto')
go
-- Retorna 2137058649
```

Object_name(idObjeto)

Retorna el nombre del objeto cuyo **idObjeto** se especifica.

Ejemplo

```
SELECT object_name(2137058649)
go
-- Retorna PRODUCTO
```

ObjectPropertyEx(idObjeto, propiedad)

Retorna el valor de **propiedad** para **idObjeto**.

Ejemplo

```
SELECT objectPropertyEx(object_id('Producto'), 'isTable')
go
-- Retorna 1 (True)
```

Type_ID(nombreTipo)

Retorna el identificador para **nombreTipo**.

Ejemplo

```
SELECT type_ID('datetime')
go
-- Retorna 61
```

Type_name(idTipo)

Retorna el nombre del tipo para el **idTipo** especificado.

Ejemplo

```
SELECT type_name(61)
go
-- Retorna datetime
```

TypeProperty(nombreTipo, propiedad)

Retorna el valor de **propiedad** para el **nombreTipo** especificado.

Ejemplo

```
SELECT typeProperty('decimal', 'scale')
go
-- Retorna 38
```

Ejercicio 77: Funciones de seguridad

Las funciones de seguridad retornan información acerca de los usuarios y roles.

Is_member(grupo_ó_rol)

Indica si la conexión actual es miembro de un **grupo** Windows ó de un **rol** SQL Server.

Ejemplo

```
SELECT is_member('db_owner')
go
-- Retorna 1 (True)
```

Is_SrvRoleMember(rol [, login])

Indica si la conexión actual es miembro del **rol** de servidor especificado. login permite especificar una cuenta a investigar.

Ejemplo

```
SELECT is_SrvRoleMember('dbCreator')
go
-- Retorna 1 (True)
```

SUser_SID(login)

Retorna el identificador de seguridad del **login** especificado.

Ejemplo

```
SELECT SUser_SID('DESARROLLO\Administrator')
go
```

SUser_SName(server_user_ID)

Retorna el login para el identificador de seguridad especificado.

Ejemplo

```
SELECT
SUser_SName(0x010500000000000051500000063F1BFBE3BA9DED47D7A6
83FF4010000)
go
-- Retorna DESARROLLO\Administrator
```

Ejercicio 78: Funciones de cadena

Las funciones de cadena ejecutan una operación sobre una cadena ingresada como parámetro, y devuelven una cadena ó valor numérico.

ASCII(expresiónCadena)

Retorna el valor ASCII del carácter más a la izquierda en **expresiónCadena**.

Ejemplo

```
SELECT ASCII('Sergio Matsukawa Maeda')
go
-- Retorna 83
```

Char(expresiónEntera)

Retorna el carácter ASCII correspondiente a **expresiónEntera**.

Ejemplo

```
SELECT char(83)
go
-- Retorna S
```

CharIndex(expresiónCadena1, expresiónCadena2 [, inicioBúsqueda])

Retorna la posición de inicio de **expresiónCadena1** dentro de **expresiónCadena2**. **inicioBúsqueda** indica la posición en **expresiónCadena2** en la que se inicia la búsqueda.

Ejemplo

```
SELECT charIndex('Cristal', 'Sporting Cristal, la Fuerza
Vencedora', 0)
go
-- Retorna 10
```

Difference(expresiónCadena1, expresiónCadena2)

Retorna un entero que indica cuan similar es el sonido de **expresiónCadena1** con el de **expresiónCadena2**. El valor retornado va de **0** (no hay similitud) a **4** (los sonidos son muy similares).

Ejemplo

```
SELECT difference('Cristal', 'Christian')
go
-- Retorna 4
```

Left(expresiónCadena, expresiónEntera)

Retorna los caracteres de la parte izquierda de **expresiónCadena**. La longitud de la cadena retornada es especificada por **expresiónEntera**.

Ejemplo

```
SELECT left('PERUDevelopers', 4)
go
-- Retorna PERU
```


Len(expresiónCadena)

Retorna la longitud de **expresiónCadena**.

Ejemplo

```
SELECT len('Microsoft SQL Server 2005')
go
-- Retorna 25
```

Lower(expresiónCadena)

Retorna **expresiónCadena**, pero todo en minúsculas.

Ejemplo

```
SELECT lower('PERU Developers')
go
-- Retorna peru developers
```

Ltrim(expresiónCadena)

Elimina los espacios a la izquierda de **expresiónCadena**, retornando la cadena resultante.

Ejemplo

```
SELECT ltrim('    Empresa Editora MACRO')
go
-- Retorna Empresa Editora MACRO
```

Nchar(expresiónEntera)

Retorna el carácter Unicode correspondiente a **expresiónEntera**.

Ejemplo

```
SELECT nchar(65)
go
-- Retorna A
```

PatIndex('%patrónBúsqueda%', expresiónCadena)

Retorna la posición de inicio de la primera ocurrencia de **patrónBúsqueda** dentro de **expresiónCadena**.

Ejemplo

```
SELECT patIndex('%sql%', 'Transact-SQL, MySQL, PostgreSQL')
go
-- Retorna 10
```

QuoteName(expresiónCadena, caracterDelimitador)

Retorna **expresiónCadena** con el **caracterDelimitador** especificado de modo que **expresiónCadena** sea una cadena válida. **caracterDelimitador** puede ser una comilla simple ('), una comilla doble ("), ó un corchete ([]). Este último es el valor predeterminado.

Ejemplo

```
SELECT quoteName('Richard O'Hara')
go
-- Retorna [Richard O'Hara]
```

Replace(expresiónCadena1, expresiónCadena2, expresiónCadena3)

Reemplaza en **expresiónCadena1** todas las ocurrencias de **expresiónCadena2** con **expresiónCadena3**.

Ejemplo

```
SELECT replace('la triste noche del triste día', 'triste',  
              'alegre')  
go  
-- Retorna la alegre noche del alegre día
```

Replicate(expresiónCadena, expresiónEntera)

Repite **expresiónCadena** el número de veces indicado por **expresiónEntera**.

Ejemplo

```
SELECT replicate('Cristal Campeón ', 3)  
go  
-- Retorna Cristal Campeón Cristal Campeón Cristal Campeón
```

Reverse(expresiónCadena)

Retorna **expresiónCadena**, pero invertida.

Ejemplo

```
SELECT reverse('Bellavista')  
go  
-- Retorna atsivalleB
```

Right(expresiónCadena, expresiónEntera)

Retorna los caracteres de la parte derecha de **expresiónCadena**. La longitud de la cadena retornada es especificada por **expresiónEntera**.

Ejemplo

```
SELECT right('PERUDevelopers', 4)
go
-- Retorna pers
```

Rtrim(expresiónCadena)

Elimina los espacios a la derecha de **expresiónCadena**, retornando la cadena resultante.

Ejemplo

```
SELECT rtrim('Empresa Editora MACRO      ')
go
-- Retorna Empresa Editora MACRO
```

Soundex(expresiónCadena)

Retorna un código de cuatro caracteres que establece la fonética de **expresiónCadena**. Se utiliza para comparar si dos cadenas son fonéticamente similares.

Ejemplo

```
SELECT soundex('MACRO'), soundex('MICRO')
go
-- Retorna M260, M260
```

Space(expresiónEntera)

Retorna una cadena que contiene los espacios indicados por **expresiónEntera**.

Ejemplo

```
SELECT 'Hola' + space(1) + 'Latinoamérica'
go
-- Retorna Hola Latinoamérica
```

Str(expresiónFloat [, longitud [, decimal]])

Convierte **expresiónFloat** a una cadena con la **longitud** especificada y con los decimales especificados por **decimal**.

Ejemplo

```
SELECT str(7567.3412, 7, 1)
go
-- Retorna 7567.3
```

Stuff(expresiónCadena1, inicio, n, expresiónCadena2)

En **expresiónCadena1** elimina **n** caracteres a partir de la posición **inicio**, y los reemplaza con **expresiónCadena2**.

Ejemplo

```
SELECT stuff('El campeón 2005 es Alianza Lima', 20, 12,
'Sporting Cristal')
go
-- Retorna El campeón 2005 es Sporting Cristal
```

Substring(expresiónCadena, inicio, n)

A partir de **expresiónCadena** retorna una subcadena de **n** caracteres de longitud generada a partir de la posición **inicio**.

Ejemplo

```
SELECT substring('Sporting Cristal Campeón 2005', 10, 7)
go
-- Retorna Cristal
```

Unicode(expresiónCadenaUnicode)

Retorna el valor Unicode del carácter más a la izquierda en **expresiónCadenaUnicode**.

Ejemplo

```
SELECT Unicode('Ñusta')
go
-- Retorna 209
```

Upper(expresiónCadena)

Retorna **expresiónCadena**, pero todo en mayúsculas.

Ejemplo

```
SELECT upper('PERU Developers')
go
-- Retorna PERU DEVELOPERS
```

Ejercicio 79: Otras funciones

Cast()

Vea el Capítulo 7: Consultas sencillas, página 244.

Convert()

Vea el Capítulo 7: Consultas sencillas, páginas 245 y 246.

Coalesce(listaExpresiones)

Retorna el primer valor NOT NULL de **listaExpresioónes**.

Ejemplo

```
SELECT coalesce(null, null, 45, null, 34)
go
-- Retorna 45
```

Current_timestamp

Retorna la fecha y hora actuales. Es equivalente a Getdate().

Ejemplo

```
SELECT current_timestamp
go
```

Current_user

Retorna el usuario actual.

Ejemplo

```
SELECT current_user
go
```

DataLength(expresión)

Retorna el número de bytes utilizado para representar expresión.

Ejemplo

```
SELECT dataLength(123456789.123456789)
go
-- Retorna 9
```

GetAnsiNull([basedatos])

Retorna el valor predeterminado para la nulidad de la base de datos actual ó de **basedatos**.

Ejemplo

```
SELECT getAnsiNull('MarketPERU')
go
-- Retorna 1 (por defecto permite valores nulos)
```


Host_name()

Retorna el nombre de la estación de trabajo para la conexión actual.

Ejemplo

```
SELECT host_name()  
go  
-- Retorna DESARROLLO (el nombre de mi PC)
```

Ident_current(tabla)

Retorna el último valor de identidad generado para **tabla**.

Ejemplo

```
SELECT ident_current('Producto')  
go  
-- Retorna 138
```

Ident_incr(tabla)

Retorna el valor del incremento para la propiedad identidad de **tabla**.

Ejemplo

```
SELECT ident_incr('Producto')  
go  
-- Retorna 1
```

Ident_seed(tabla)

Retorna el valor inicial para la propiedad identidad de **tabla**.

Ejemplo

```
SELECT ident_seed('Producto')
go
-- Retorna 1
```

@@Identity

Retorna el último valor de identidad insertado en la sesión.

Ejemplo

```
SELECT @@identity
go
```

IsDate(expresión)

Determina si expresión es una fecha válida.

Ejemplo

```
SELECT isDate('31/08/2005')
go
-- Retorna 0 (False)
```

IsNull(expresiónEvaluada, valorReemplazo)

Reemplaza **expresiónEvaluada** con **valorReemplazo** si **expresiónEvaluada** es NULL.

Ejemplo

```
DECLARE @var1 AS varchar(25)
SET @var1 = NULL

SELECT isNull(@var1, 'Ahora no es NULL')
go
-- Retorna Ahora no es NULL
```

IsDate(expresión)

Determina si expresión es un número.

Ejemplo

```
SELECT isNumeric('31/08/2005')
go
-- Retorna 0 (False)
```

NullIf(expresión1, expresión2)

Retorna NULL si **expresión1** es igual a **expresión2**; en caso contrario, retorna **expresión1**.

Ejemplo

```
SELECT nullIf(345, 346)
go
-- Retorna 345
```

@@RowCount

Retorna el número de filas afectadas por la última instrucción ejecutada.

Ejemplo

```
SELECT * FROM Producto
go

SELECT @@RowCount
go
-- Retorna 138
```

RowCount_big()

Retorna el número de filas afectadas por la última instrucción ejecutada. Es similar a @@RowCount solo que retorna un dato **bigint**.

Ejemplo

```
SELECT * FROM Producto
go

SELECT RowCount_big()
go
-- Retorna 138
```

Scope_identity()

Retorna el último valor de identidad insertado en el alcance actual, donde el alcance puede ser un procedimiento almacenado, desencadenante, función, ó lote.

Ejemplo

```
SELECT Scope_identity()
go
```

ServerProperty(propiedad)

Retorna información sobre la **propiedad** del servidor.

Ejemplo

```
SELECT ServerProperty('Collation')
go
-- Retorna Latin1_General_CI_AI
```

SessionProperty(propiedad)

Retorna información sobre la **propiedad** de la sesión.

Ejemplo

```
SELECT SessionProperty('Concat_Null_Yields_Null')
go
-- Retorna 1 (True)
```

Session_user

Retorna el nombre del usuario actual.

Ejemplo

```
SELECT Session_user
go
```

System_user

Retorna el nombre de la conexión actual.

Ejemplo

```
SELECT System_user
go
-- Retorna DESARROLLO\Administrator
```

User_name([idUsuario])

Retorna el nombre del usuario actual ó del usuario para el idUsuario especificado.

Ejemplo

```
SELECT User_name( )
go
```