

Database System

Functions

Muhammad Tariq Mahmood

tariq@koreatech.ac.kr

School of Computer Science and Engineering
Korea University of Technology and Education

T-SQL Functions

- ▶ **System Functions**

- SQL Server provides many system functions that you can use to perform a variety of operations.
- They cannot be modified.

- ▶ **Deterministic vs Nondeterministic**

- ▶ **User-defined functions**

- **Scalar Function**
- **Table-Valued Functions**

TSQL Built-in Functions

Function category	Description
<u>Configuration Functions</u>	Return information about the current configuration.
<u>Conversion Functions</u>	Support data type casting and converting.
<u>Cursor Functions</u>	Return information about cursors.
<u>Date and Time Data Types and Functions</u>	Perform operations on a date and time input values and return string, numeric, or date and time values.
<u>JSON Functions</u>	Validate, query, or change JSON data.
<u>Logical Functions</u>	Perform logical operations.
<u>Mathematical Functions</u>	Perform calculations based on input values provided as parameters to the functions, and return numeric values.
<u>Metadata Functions</u>	Return information about the database and database objects.
<u>Security Functions</u>	Return information about users and roles.
<u>String Functions</u>	Perform operations on a string (char or varchar) input value and return a string or numeric value.
<u>System Functions</u>	Perform operations and return information about values, objects, and settings in an instance of SQL Server.
<u>System Statistical Functions</u>	Return statistical information about the system.
<u>Text and Image Functions</u>	Perform operations on text or image input values or columns, and return information about the value.

Configuration Functions

- ▶ All configuration functions are nondeterministic. This means these functions do not always return the same results every time they are called, even with the same set of input values.

@@DATEFIRST

@@DBTS

@@LANGID

@@LANGUAGE

@@LOCK_TIMEOUT

@@MAX_CONNECTIONS

@@MAX_PRECISION

@@NESTLEVEL

@@OPTIONS

@@REMSERVER

@@SERVERNAME

@@SERVICENAME

@@SPID

@@TEXTSIZE

@@VERSION

Conversion Functions

- ▶ The following functions support data type casting and converting.

CAST and CONVERT

CAST (expression AS data_type [(length)])

CONVERT (data_type [(length)] , expression [, style])

PARSE

PARSE (string_value AS data_type [USING culture])

TRY_CAST

TRY_CAST (expression AS data_type [(length)])

TRY_CONVERT

TRY_CONVERT (data_type [(length)] , expression [, style])

TRY_PARSE

TRY_PARSE (string_value AS data_type [USING culture])

Cursor Functions

- ▶ The following scalar functions return information about cursors:

`@@CURSOR_ROWS`

PARSE

`PARSE (string_value AS data_type [USING culture])`

TRY_CAST

`TRY_CAST (expression AS data_type [(length)])`

TRY_CONVERT

`TRY_CONVERT (data_type [(length)], expression [, style])`

TRY_PARSE

`TRY_PARSE (string_value AS data_type [USING culture])`

T-SQL Date and Time Functions

► Lower-Precision system Date and Time functions

Function	Syntax	Return value
<u>CURRENT_TIMESTAMP</u>	CURRENT_TIMESTAMP	Returns a datetime value that contains the date and time of the computer on which the instance of SQL Server is running. The time zone offset is not included.
<u>GETDATE</u>	GETDATE ()	Returns a datetime value that contains the date and time of the computer on which the instance of SQL Server is running. The time zone offset is not included.
<u>GETUTCDATE</u>	GETUTCDATE ()	Returns a datetime value that contains the date and time of the computer on which the instance of SQL Server is running. The date and time is returned as UTC time (Coordinated Universal Time).

T-SQL Date and Time Functions

► Functions that get Date and Time parts

Function	Syntax	Return value
<u>DATENAME</u>	DATENAME (datepart , date)	Returns a character string that represents the specified datepart of the specified date.
<u>DATEPART</u>	DATEPART (datepart , date)	Returns an integer that represents the specified datepart of the specified date.
<u>DAY</u>	DAY (date)	Returns an integer that represents the day part of the specified date.
<u>MONTH</u>	MONTH (date)	Returns an integer that represents the month part of a specified date.
<u>YEAR</u>	YEAR (date)	Returns an integer that represents the year part of a specified date.

T-SQL Date and Time Functions

► Functions that modify Date and Time values

Function	Syntax	Return value
<u>DATEADD</u>	DATEADD (datepart , number , date)	Returns a new datetime value by adding an interval to the specified datepart of the specified date.
<u>EOMONTH</u>	EOMONTH (start_date [, month_to_add])	Returns the last day of the month that contains the specified date, with an optional offset.
<u>SWITCHOFFSET</u>	SWITCHOFFSET (DATETIMEOFFSET , time_zone)	SWITCHOFFSET changes the time zone offset of a DATETIMEOFFSET value and preserves the UTC value.
<u>TODATETIMEOFFSET</u>	TODATETIMEOFFSET (expression , time_zone)	TODATETIMEOFFSET transforms a datetime2 value into a datetimeoffset value. The datetime2 value is interpreted in local time for the specified time_zone.

ISDATE ISDATE (*expression*) Determines whether a **datetime** or **smalldatetime** input expression is a valid date or time value.

T-SQL Date and Time Functions

► Functions that get or set session format

Function	Syntax	Return value
<u>@@DATEFIRST</u>	@@DATEFIRST	Returns the current value, for the session, of SET DATEFIRST.
<u>SET DATEFIRST</u>	SET DATEFIRST { number @number_var }	Sets the first day of the week to a number from 1 through 7.
<u>SET DATEFORMAT</u>	SET DATEFORMAT { format @format_var }	Sets the order of the dateparts (month/day/year) for entering datetime or smalldatetime data.
<u>@@LANGUAGE</u>	@@LANGUAGE	Returns the name of the language that is currently being used. @@LANGUAGE is not a date or time function. However, the language setting can affect the output of date functions.
<u>SET LANGUAGE</u>	SET LANGUAGE { [N] 'language' @language_var }	Sets the language environment for the session and system messages. SET LANGUAGE is not a date or time function. However, the language setting affects the output of date functions.
<u>sp_helplanguage</u>	sp_helplanguage [[@language =] 'language']	Returns information about date formats of all supported languages. sp_helplanguage is not a date or time stored procedure. However, the language setting affects the output of date functions.

JSON Functions

- ▶ Use the functions to validate or change JSON text or to extract simple or complex values.

Function	Description
<u>ISJSON</u>	Tests whether a string contains valid JSON.
<u>JSON_VALUE</u>	Extracts a scalar value from a JSON string.
<u>JSON_QUERY</u>	Extracts an object or an array from a JSON string.
<u>JSON_MODIFY</u>	Updates the value of a property in a JSON string and returns the updated JSON string.

Mathematical Functions

- ▶ The following scalar functions perform a calculation, usually based on input values that are provided as arguments, and return a numeric value:

<u>ABS</u>	<u>DEGREES</u>	<u>RAND</u>
<u>ACOS</u>	<u>EXP</u>	<u>ROUND</u>
<u>ASIN</u>	<u>FLOOR</u>	<u>SIGN</u>
<u>ATAN</u>	<u>LOG</u>	<u>SIN</u>
<u>ATN2</u>	<u>LOG10</u>	<u>SQRT</u>
<u>CEILING</u>	<u>PI</u>	<u>SQUARE</u>
<u>COS</u>	<u>POWER</u>	<u>TAN</u>
<u>COT</u>	<u>RADIANS</u>	

Metadata Functions

- ▶ The following scalar functions return information about the database and database objects:

<u>@@PROCID</u>	<u>INDEX_COL</u>
<u>APP_NAME</u>	<u>INDEXKEY_PROPERTY</u>
<u>APPLOCK_MODE</u>	<u>INDEXPROPERTY</u>
<u>APPLOCK_TEST</u>	<u>NEXT VALUE FOR</u>
<u>ASSEMBLYPROPERTY</u>	<u>OBJECT_DEFINITION</u>
<u>COL_LENGTH</u>	<u>OBJECT_ID</u>
<u>COL_NAME</u>	<u>OBJECT_NAME</u>
<u>COLUMNPROPERTY</u>	<u>OBJECT_SCHEMA_NAME</u>
<u>DATABASE_PRINCIPAL_ID</u>	<u>OBJECTPROPERTY</u>
<u>DATABASEPROPERTYEX</u>	<u>OBJECTPROPERTYEX</u>
<u>DB_ID</u>	<u>ORIGINAL_DB_NAME</u>
<u>DB_NAME</u>	<u>PARSENAME</u>
<u>FILE_ID</u>	<u>SCHEMA_ID</u>
<u>FILE_IDEX</u>	<u>SCHEMA_NAME</u>
<u>FILE_NAME</u>	<u>SCOPE_IDENTITY</u>
<u>FILEGROUP_ID</u>	<u>SERVERPROPERTY</u>
<u>FILEGROUP_NAME</u>	<u>STATS_DATE</u>
<u>FILEGROUPPROPERTY</u>	<u>TYPE_ID</u>
<u>FILEPROPERTY</u>	<u>TYPE_NAME</u>
<u>FULLTEXTCATALOGPROPERTY</u>	<u>TYPEPROPERTY</u>
<u>FULLTEXTSERVICEPROPERTY</u>	<u>VERSION</u>

Metadata Functions

- ▶ `DB_ID()`: Returns a number that is id of the database.
- ▶ `DB_NAME()` returns a `nvarchar` type that is the database name.
- ▶ `OBJECT_ID ('database_name . schema_name . object_name')`: Returns the database object id number of a schema object.
- ▶ `OBJECT_DEFINITION (object_id)`: Returns the source text of the definition of a specified object.
- ▶ `OBJECT_SCHEMA_NAME (object_id [, database_id])`: Returns the database schema name of objects.

Security Functions

- ▶ The following functions return information that is useful in managing security

<u>CERTENCODED (Transact-SQL)</u>	<u>PWDCOMPARE (Transact-SQL)</u>
<u>CERTPRIVATEKEY (Transact-SQL)</u>	<u>PWDENCRYPT (Transact-SQL)</u>
<u>CURRENT_USER (Transact-SQL)</u>	<u>SCHEMA_ID (Transact-SQL)</u>
<u>DATABASE_PRINCIPAL_ID (Transact-SQL)</u>	<u>SCHEMA_NAME (Transact-SQL)</u>
<u>sys.fn_builtin_permissions (Transact-SQL)</u>	<u>SESSION_USER (Transact-SQL)</u>
<u>sys.fn_get_audit_file (Transact-SQL)</u>	<u>SUSER_ID (Transact-SQL)</u>
<u>sys.fn_my_permissions (Transact-SQL)</u>	<u>SUSER_SID (Transact-SQL)</u>
<u>HAS_PERMS_BY_NAME (Transact-SQL)</u>	<u>SUSER_SNAME (Transact-SQL)</u>
<u>IS_MEMBER (Transact-SQL)</u>	<u>SYSTEM_USER (Transact-SQL)</u>
<u>IS_ROLEMEMBER (Transact-SQL)</u>	<u>SUSER_NAME (Transact-SQL)</u>
<u>IS_SRVROLEMEMBER (Transact-SQL)</u>	<u>USER_ID (Transact-SQL)</u>
<u>ORIGINAL_LOGIN (Transact-SQL)</u>	<u>USER_NAME (Transact-SQL)</u>
<u>PERMISSIONS (Transact-SQL)</u>	

Security Functions

- ▶ `CURRENT_USER` Return the name of the current user connected.
- ▶ `ORIGINAL_LOGIN()` Return the login name connected to the instance of SQL Server.
- ▶ `SESSION_USER` is a security function and returns the session user.
- ▶ `SYSTEM_USER` is a security function and returns the system user name.
- ▶ `USER_NAME()` is part of security functions and return the user name of database.

String Functions

- ▶ The following scalar functions perform an operation on a string input value and return a string or numeric value:

<u>ASCII</u>	<u>CHAR</u>	<u>CHARINDEX</u>
<u>CONCAT</u>	<u>CONCAT_WS</u>	<u>DIFFERENCE</u>
<u>FORMAT</u>	<u>LEFT</u>	<u>LEN</u>
<u>LOWER</u>	<u>LTRIM</u>	<u>NCHAR</u>
<u>PATINDEX</u>	<u>QUOTENAME</u>	<u>REPLACE</u>
<u>REPLICATE</u>	<u>REVERSE</u>	<u>RIGHT</u>
<u>RTRIM</u>	<u>SOUNDEX</u>	<u>SPACE</u>
<u>STR</u>	<u>STRING_AGG</u>	<u>STRING_ESCAPE</u>
<u>STRING_SPLIT</u>	<u>STUFF</u>	<u>SUBSTRING</u>
<u>TRANSLATE</u>	<u>TRIM</u>	<u>UNICODE</u>
<u>UPPER</u>		

T-SQL String Functions

- ▶ CONCAT(): Returns a string that is the result of concatenating two or more string values.

```
SELECT CONCAT ( 'Happy ', 'Birthday ', 11, '/', '25' ) AS Result;
```

- ▶ LEFT(): Returns the left part of a character string with the specified number of characters.

```
SELECT LEFT('abcdefg',2);
```

- ▶ LTRIM: Returns a character expression after it removes leading blanks.

```
DECLARE @string_to_trim varchar(60);
```

```
SET @string_to_trim = '    Five spaces are at the beginning of  
this
```

```
string.';
```

```
SELECT 'Here is the string without the leading spaces: ' +
```

```
    LTRIM(@string_to_trim);
```

```
GO
```

T-SQL String Functions

- ▶ CONCAT(): Returns a string that is the result of concatenating two or more string values.

```
SELECT CONCAT ( 'Happy ', 'Birthday ', 11, '/', '25' ) AS Result;
```

- ▶ LEFT(): Returns the left part of a character string with the specified number of characters.

```
SELECT LEFT('abcdefg',2);
```

- ▶ LTRIM: Returns a character expression after it removes leading blanks.

```
DECLARE @string_to_trim varchar(60);
```

```
SET @string_to_trim = '    Five spaces are at the beginning of  
this
```

```
string.';
```

```
SELECT 'Here is the string without the leading spaces: ' +
```

```
    LTRIM(@string_to_trim);
```

```
GO
```

T-SQL String Functions

- ▶ REPLACE(): Replaces all occurrences of a specified string value with another string value.

```
SELECT REPLACE('abcdefghicde','cde','xxx');
```

- ▶ REPLICATE(): Returns the reverse order of a string value.

```
DECLARE @myvar varchar(10);  
SET @myvar = 'sdrawkcaB';  
SELECT REVERSE(@myvar) AS Reversed ;  
GO
```

- ▶ UPPER(): Returns a character expression with lowercase character data converted to uppercase.

```
SELECT UPPER(RTRIM(LastName)) + ', ' + FirstName AS Name  
FROM Person.Person  
ORDER BY LastName;  
GO
```

System Functions

- ▶ The following scalar functions return statistical information about the system:

<u>@@CONNECTIONS</u>	<u>@@PACK_RECEIVED</u>
<u>@@CPU_BUSY</u>	<u>@@PACK_SENT</u>
<u>fn_virtualfilestats</u>	<u>@@TIMETICKS</u>
<u>@@IDLE</u>	<u>@@TOTAL_ERRORS</u>
<u>@@IO_BUSY</u>	<u>@@TOTAL_READ</u>
<u>@@PACKET_ERRORS</u>	<u>@@TOTAL_WRITE</u>

T-SQL System Functions

- ▶ @@Error: Return the error number of the Transact-SQL statement executed.

```
USE model;  
GO  
UPDATE products  
  SET product_type = 'DEDE'  
  WHERE product_id = 2;  
  
IF @@ERROR <> 0  
  PRINT N'Error: Product type.';  
GO
```

- ▶ ERROR_NUMBER(): Return the code number of the error that use the CATCH block of a TRY ... CATCH.

```
USE model;  
GO  
BEGIN TRY  
  SELECT 18/0;  
END TRY  
BEGIN CATCH  
  SELECT ERROR_NUMBER() AS Error_Number;  
END CATCH;  
GO
```

T-SQL System Functions

- ▶ **ERROR_MESSAGE()**: Return the message text of the error that use the CATCH block of a TRY ... CATCH

```
USE model;  
GO  
BEGIN TRY  
    SELECT -5*9/0;  
END TRY  
BEGIN CATCH  
    SELECT ERROR_MESSAGE() AS Error_Message;  
END CATCH;  
GO
```

- ▶ **HOST_NAME()**: Returns the workstation name.

```
select HOST_NAME() as Host_name;
```

- ▶ **HOST_ID ()**: Returns the workstation identification number.

```
select HOST_ID () as Host_ID;
```

- ▶ **@@Connections**: It is an system statistical function and return the number of connection attempts successful or unsuccessful since SQL Server was last started.

```
SELECT @@CONNECTIONS AS 'Login id';
```