PROG8080
SQL Scalar Functions

Glenn Paulley

Fall 2015

Introduction to SQL scalar functions

- SQL functions can be used in the column list and in any search condition in a statement, including the WHERE and HAVING clauses, a join condition, even an ORDER BY clause
- Functions can accept zero, one, or more *arguments*
- Functions can *return* zero or one value
- Many functions [such as GETDATE() and UPPER()] are “built in” to SQL Server
  - It is possible for a developer to write their own SQL function – termed a user-defined function

SQL scalar functions

- Every database engine has functions, but the details differ between engines
- We will highlight some of the differences between Microsoft SQL Server and Oracle
  - In Microsoft SQL Server there is no “dual” table for executing arbitrary expressions
    - In Transact-SQL a FROM clause isn’t required to execute a SELECT statement and compute an expression
  - In Microsoft SQL Server some functions differ from both Oracle and the SQL standard

Scalar function examples

- Oracle

```
SELECT 2 + 2
        FROM dual
SELECT SYSDATE
        FROM dual
SELECT SQRT( 2 )
        FROM dual
SELECT SOUNDEX( 'Roselius')
        FROM dual
```

- SQL Server

```
SELECT 2 + 2

SELECT GETDATE( )

SELECT SQRT( 2 )

SELECT SOUNDEX( 'Roselius')
```

SQL Functions in the SELECT clause

- Oracle

```
SELECT a.invoiceNumber, a.“date”,
                    TO_CHAR(a.“date”, ‘MM’)
              FROM dbo.Audit a;
```

- SQL Server 2014

```
SELECT a.invoiceNumber, [date],
  DATEPART(MM, [date] ) as [month]
FROM dbo.Audit a
```

CONESTOGA
Connect Life and Learning

SQL Functions in the SELECT clause

- Oracle

```
SELECT DISTINCT
        SUBSTR(mainPhone, 1, 3) AS prefix
        FROM Person
```

- SQL Server

```
SELECT DISTINCT
        SUBSTRING(mainPhone, 1, 3) AS prefix
        FROM Person
```

SQL Functions in the WHERE clause

- Oracle

```
SELECT a.invoiceNumber, a.“date”
FROM dbo.Audit a
WHERE TO_CHAR(a.“date”, ’MM’) = ’06’
```

- SQL Server

```
SELECT a.invoiceNumber, [date]
FROM dbo.Audit a
WHERE  DATEPART(MM, [date] ) = 8
```

SQL Functions in the WHERE clause

- Oracle

```
SELECT lastName, firstName
        FROM Person
        WHERE UPPER(SUBSTR(lastName,1,2)) = 'MC'
```

- SQL Server

```
SELECT lastName, firstName
        FROM Person
        WHERE UPPER(SUBSTRING(lastName,1,2)) = 'MC'
```

STRING FUNCTION EXAMPLES

String Functions – SUBSTRING( )

- SUBSTRING(*string*, *start*, *count*):

```
SELECT SUBSTRING( city, 1, 3 ) AS [abbreviation]
FROM Person
```

- In SQL, strings are indexed starting with 1 (not 0)

String Functions – LEFT(), RIGHT()

- LEFT(*string*, *count*):

```
SELECT LEFT( city, 3 ) AS [abbreviation]
FROM Person
```

- RIGHT(string, count) does the same thing, but starting from the right side (end) of the string

String Functions – CHARINDEX( )

- Oracle: INSTR(searchIn, searchFor)

```
SELECT *
FROM Person
WHERE INSTR( lastName, ‘Mc’ ) > 0
```

- SQL Server: CHARINDEX(searchFor, searchIn)

```
SELECT *
FROM Person
WHERE CHARINDEX( ‘Mc', lastName ) > 0
```

String Concatenation

- String concatenation is done with “||” in Oracle

```
SELECT lastName ||  ‘ (‘ || firstName || ‘)’
                AS completeName
FROM Person
```

  - The || operator is defined by the ISO SQL Standard
- String concatenation is done with “+” in SQL Server

```
SELECT lastName +  ‘ (‘ + firstName + ‘)’
                AS completeName
FROM Person
```

CONESTOGA
Connect Life and Learning

Other String Functions

- Other common string functions
  - LEN(*string*) – get the length of a string
  - LTRIM(*string*) – trim a string of its leading characters
  - RTRIM(*string*) – trim a string of its trailing characters
  - LOWER(*string*) – convert a string to all lower case
  - UPPER(*string*) – convert a string to all upper case
- Note: unlike Oracle, in SQL Server (by default) all string comparisons are case insensitive
- Available in Oracle but not SQL Server:
  - PAD( ) … use concatenation instead
  - TRIM( ) … use LTRIM(RTRIM(*column*))

MONEY FORMATTING – CAST AND CONVERT FUNCTIONS

Formatting Money Amounts

- To display a money amount (using Canadian/U.S. conventions):
  - CAST the value to the MONEY data type
  - then use the CONVERT function to convert the MONEY type to CHAR(*n*) or VARCHAR(*n*) with style 1

```
'$' + CONVERT( CHAR(12),
               CAST( amount AS MONEY ),
               1 )
```

CONESTOGA
Connect Life and Learning

Formatting Money Amounts

- CONVERT to CHAR(*n*) to create a column aligned on the decimal point

  $ 1.00

  $ 27.50

  $ 495.61

- CONVERT to VARCHAR(*n*) if alignment is not required

  $1.00

  $27.50

  $491.65

MATH FUNCTIONS

CONESTOGA
Connect Life and Learning

Math Functions available in SQL Server

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

Math Functions: ROUND( )

- This query demonstrates rounding and truncating numeric columns:

```
SELECT id, item, amountPerSemester,
                ROUND( amountPerSemester, 0 ) AS rounded,
                ROUND( amountPerSemester, 0, 1 ) AS truncated
FROM IncidentalFee
ORDER BY amountPerSemester;
```

- The third parameter of the ROUND function toggles between rounding (missing or 0) and truncating (1)

RANDOM NUMBER GENERATION: THE RAND() FUNCTION

RAND( )

- RAND ( ) returns a pseudo-random, double-precision floating point number in the range [0,1)

Microsoft SQL Server Management Studio

File Edit View Query Debug Tools Window Community Help

New Query

master Execute

RANDDemo.sql - MCKAY\.....3))

```
1 SELECT RAND() AS 'Random Number';
2 SELECT RAND() AS 'Random Number';
```

Results

```
Random Number
----------------------
0.0955529658091138

(1 row(s) affected)

Random Number
----------------------
0.400378165503602
```

Query executed successfully. | MCKAY\SQLEXPRESS (10.50 RTM) | MCKAY\user (53) | master | 00:00:00 | 2 rows

Item(s) Saved | Ln 2 | Col 33 | Ch 33 | INS

RAND( )

- RAND ( ) can be seeded – critical for testing your application

Microsoft SQL Server Management Studio

File Edit View Query Debug Tools Window Community Help

New Query | master | Execute

RANDDemo.sql - MCKAY\.....3))

```
1 SELECT RAND(100) AS 'Seeded Random Number';
2 SELECT RAND(100) AS 'Seeded Random Number';
```

Results

```
Seeded Random Number
----------------------
0.715436657367485

(1 row(s) affected)

Seeded Random Number
----------------------
0.715436657367485
```

Query executed successfully. | MCKAY\SQLEXPRESS (10.50 RTM) | MCKAY\user (53) | master | 00:00:00 | 2 rows

Item(s) Saved | Ln 2 | Col 23 | Ch 23 | INS

RAND( )

- You can scale the value returned by `RAND()`
- To obtain a random number in the range 0-n, specify n+1
  - To obtain a random number in the range 0-999,999 specify 1000000
  - A random number in this range will be up to 6 digits long
- You can scale the value returned by `RAND()` and pad it on the right with zeroes to create a fixed-size result

Rand() Scaling and Padding Script

```
DECLARE @randomNumber REAL;
SET @randomNumber = RAND();
SELECT @randomNumber AS 'RAND()';

DECLARE @randomNumberINT INT;
SET @randomNumberINT = CAST(@randomNumber * 1000000 AS INT);
SELECT @randomNumberINT AS 'CAST(@randomNumber * 1000000 AS INT)';

DECLARE @randomNumberVARCHAR VARCHAR(7);
SET @randomNumberVARCHAR = CAST(@randomNumberINT AS VARCHAR(7));
SELECT @randomNumberVARCHAR AS 'CAST(@randomNumberINT AS VARCHAR(7))';

DECLARE @randomNumberPadded VARCHAR(14);
SET @randomNumberPadded = '0000000' + @randomNumberVARCHAR;
SELECT @randomNumberPadded AS '''0000000'' + @randomNumberVARCHAR';

DECLARE @randomNumberRIGHT VARCHAR(7);
SET @randomNumberRIGHT = RIGHT('0000000' + @randomNumberVARCHAR, 7);
SELECT @randomNumberRIGHT AS 'RIGHT(''0000000'' + @randomNumberVARCHAR, 7)';

SELECT RIGHT('0000000' + CAST(CAST(RAND() * 1000000 AS INT) AS VARCHAR(7)), 7)
    AS 'Student ID Exactly 7 digits padded with zeroes on the right'
```

Rand() Scaling and Padding Code

SQLQuery1.sql - JMCKAY\SQLEXPRESS.master (JMCKAY\John (55))* - Mi...

File Edit View Query Project Debug Tools Window Help

New Query

master | Execute | Debug

SQLQuery1.sql - J...(JMCKAY\John (55))*

```sql
DECLARE @randomNumber REAL;
SET @randomNumber = RAND();
SELECT @randomNumber AS 'RAND()';

DECLARE @randomNumberINT INT;
SET @randomNumberINT = CAST(@randomNumber * 1000000 AS INT);
SELECT @randomNumberINT AS 'CAST(@randomNumber * 1000000 AS INT)';

DECLARE @randomNumberVARCHAR VARCHAR(7);
SET @randomNumberVARCHAR = CAST(@randomNumberINT AS VARCHAR(7));
SELECT @randomNumberVARCHAR AS 'CAST(@randomNumberINT AS VARCHAR(7))';

DECLARE @randomNumberPadded VARCHAR(14);
SET @randomNumberPadded = '0000000' + @randomNumberVARCHAR;
SELECT @randomNumberPadded AS '''0000000'' + @randomNumberVARCHAR';

DECLARE @randomNumberRIGHT VARCHAR(7);
SET @randomNumberRIGHT = RIGHT('0000000' + @randomNumberVARCHAR, 7);
SELECT @randomNumberRIGHT AS 'RIGHT(''0000000'' + @randomNumberVARCHAR, 7)';
```

100 %

Query executed s... | JMCKAY\SQLEXPRESS (12.0 RTM) | JMCKAY\John (55) | master | 00:00:00 | 1 rows

Ready Ln 1 Col 1 INS

Rand() Scaling and Padding Output

SQLQuery1.sql - JMCKAY\SQLEXPRESS....

File Edit View Query Project Debug Tools Window Help

New Query

master

Execute Debug

SQLQuery1.sql - J...(JMCKAY\John (55))*

100 %

Results Messages

	RAND()
1	0.02777647

	CAST(@randomNumber * 1000000 AS INT)
1	27776

	CAST(@randomNumberINT AS VARCHAR(7))
1	27776

	'0000000' + @randomNumberVARCHAR
1	000000027776

	RIGHT('0000000' + @randomNumberVARCHAR, 7)
1	0027776

EXPRESS (12.0 RTM) | JMCKAY\John (55) | master | 00:00:00 | 5 rows

Ln 1 Col 1 Ch 1 INS

Rand() Scaling and Padding Combined

SQLQuery1.sql - JMCKAY\SQLEXPRESS.master (JMCKAY\John

File Edit View Query Project Debug Tools Window Help

New Query

master

Execute Debug

SQLQuery1.sql - J...(JMCKAY\John (55))*

```
DECLARE @randomNumber REAL;
SET @randomNumber = RAND();
SELECT @randomNumber AS 'RAND()';

DECLARE @randomNumberINT INT;
SET @randomNumberINT = CAST(@randomNumber * 1000000 AS
SELECT @randomNumberINT AS 'CAST(@randomNumber * 100000

DECLARE @randomNumberVARCHAR VARCHAR(7);
SET @randomNumberVARCHAR = CAST(@randomNumberINT AS VAR
SELECT @randomNumberVARCHAR AS 'CAST(@randomNumberINT A

DECLARE @randomNumberPadded VARCHAR(14);
SET @randomNumberPadded = '0000000' + @randomNumberVARC
SELECT @randomNumberPadded AS '''0000000'' + @randomNum

DECLARE @randomNumberRIGHT VARCHAR(7);
SET @randomNumberRIGHT = RIGHT('0000000' + @randomNumbe
SELECT @randomNumberRIGHT AS 'RIGHT(''0000000'' + @rand
```

100 %

Query executed s... | JMCKAY\SQLEXPRESS (12.0 RTM) | JMCKAY\John (5

Ready Ln 1

SQLQuery1.sql - JMCKAY\SQLEXPRESS....

File Edit View Query Project Debug Tools Window Help

New Query

master

Execute Debug

SQLQuery1.sql - J...(JMCKAY\John (55))*

100 %

Results | Messages

	RAND()
1	0.02777647

	CAST(@randomNumber * 1000000 AS INT)
1	27776

	CAST(@randomNumberINT AS VARCHAR(7))
1	27776

	'0000000' + @randomNumberVARCHAR
1	000000027776

	RIGHT('0000000' + @randomNumberVARCHAR, 7)
1	0027776

EXPRESS (12.0 RTM) | JMCKAY\John (55) | master | 00:00:00 | 5 rows

Ln 1 Col 1 Ch 1 INS