(T11)討論MathFunction，包括ABS、CEILING、FLOOR、POWER、RAND、SQUARE、SQRT、ROUND  
CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc  
=======================================================================  
(T11)討論MathFunction，包括ABS、CEILING、FLOOR、POWER、RAND、SQUARE、SQRT、ROUND  
=======================================================================  
0. What to learn

1. ABS(numeric\_expression)

2. CEILING(numeric\_expression) and FLOOR(numeric\_expression)

3. POWER(f1,n)

4. RAND([seed])

5. ROUND( numeric\_expression , length [,function] )  
=======================================================================

0. What to learn

What to learn

1.

ABS(numeric\_expression)

2.

CEILING(numeric\_expression)  and  FLOOR(numeric\_expression)

3.

POWER(f1,n), f1 to the Power of n

SQUARE(f1), Square of the f1

SQRT(f1), Square root of the f1.

4.

RAND(1);

FLOOR(RAND() \* 100),  0 <= IntNumber < 100

FLOOR(RAND()\*(b-a)+a),  a <= IntNumber < b

5.

ROUND(numeric\_expression,length[,function])

Rounds the given numeric expression based on the given length.

5.1.

numeric\_expression :

numeric expression except for the bit data type.

5.2.

length int(precision length int):

If precision length int > 0 ,

then ROUND() is applied for the decimal part. (to the right)

If precision length int < 0 ,

then ROUND() is applied to the number before the decimal. (to the left)

5.3.

function (operation options):

Zero as default means operating rounding.

Non-Zero is truncated which means truncate anything after the precision length.

==================================================

1. ABS(numeric\_expression)

PRINT ABS(-123.4);

/\*

ABS(numeric\_expression)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/abs-transact-sql>

returns the absolute (positive) value of the specified numeric expression

-PRINT ABS(-123.4);

Output  :  123.4

\*/

==================================================

2. CEILING(numeric\_expression)  and  FLOOR(numeric\_expression)

--===============================================================================

--T011\_02\_CEILING(numeric\_expression)  and  FLOOR(numeric\_expression)

--===============================================================================

PRINT CEILING(26.2);

--Output  :  27

PRINT CEILING(-26.2);

--Output  :  -26

PRINT FLOOR(26.2);

--Output  :  26

PRINT FLOOR(-26.2);

--Output  :  -27

/\*

1.

CEILING(numeric\_expression)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/ceiling-transact-sql>

Returns the smallest integer greater than, or equal to,

the specified numeric expression.

2.

FLOOR(numeric\_expression)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/floor-transact-sql>

Returns the largest integer less than or equal to

the specified numeric expression.

\*/

==================================================

3. POWER(f1,n)

--===============================================================================

--T011\_03\_POWER(f1,n)

--===============================================================================

--POWER(f1,n)  , F1 to the Power of n

PRINT POWER(2, 4);

--Output  :  16

--2 to the power of 4 = 2\*2\*2\*2 = 16

PRINT POWER(8, 2);

--Output  :  64

--8 to the power of 2 = 8\*8 = 64

PRINT SQUARE(8);

--SQUARE(f1)  , Square of the f1

--Output  :  64

--Square of the 8 = 8\*8 = 64

PRINT SQRT(64);

--SQRT(f1)  , Square root of the f1.

--Output  :  8

--Square root of the 64 = 8

PRINT SQRT(5);

--Output  :  2.23607

--Square root of the 5 = 2.23607

/\*

1.

POWER(float\_expression,y)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/power-transact-sql>

Returns the value of the specified float expression to the specified power, y.

--POWER(f1,n)

F1 to the Power of n

2.

SQUARE(float\_expression)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/square-transact-sql>

Returns the square of the specified float value.

--SQUARE(f1)

Square of the f1.

3.

SQRT(float\_expression)

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/sqrt-transact-sql>

Returns the square root of the specified float value.

--SQRT(f1)

Square root of the f1.

\*/

==================================================

4. RAND([seed])

--===============================================================================

--T011\_04\_RAND([seed])

--===============================================================================

PRINT RAND(1);

--Same seed always returns the same RAND([seed]) value.

PRINT RAND();

--0 <= FloatNumber < 1

PRINT FLOOR(RAND() \* 100);

--0 <= IntNumber < 100

PRINT FLOOR(RAND() \* ( 25 - 10 ) + 10);

--10 <= IntNumber < 25

--PRINT FLOOR(RAND()\*(b-a)+a);

--a <= IntNumber < b

DECLARE @Counter INT;

SET @Counter = 1;

WHILE ( @Counter <= 10 )

    BEGIN

        PRINT FLOOR(RAND() \* 100);

        SET @Counter += 1;

    END;

--Return 10 random int value.

--and 0 <= IntNumber < 100

/\*

1.

RAND([seed])

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/rand-transact-sql>

<https://www.w3schools.com/sql/func_mysql_rand.asp>

Returns a pseudo-random float value from 0 through 1, exclusive.

0 <= ReturnNumber < 1

Same seed always returns the same RAND([seed]) value.

2.

FLOOR(RAND()\*(b-a)+a);

Where a is the smallest number and b is the largest number that you want to generate a random number for.

Reference:

<https://www.techonthenet.com/sql_server/functions/rand.php>

PRINT FLOOR(RAND()\*(25-10)+10);

10 <= IntNumber < 25

\*/

==================================================

5. ROUND( numeric\_expression , length [,function] )

--===============================================================================

--T011\_05\_ROUND( numeric\_expression , length [,function] )

--===============================================================================

PRINT ROUND(123.44500, 2);

--Output  :  123.45000

--Round to 2 places after the decimal point. (to the right)

PRINT ROUND(123.44400, 2);

--Output  :  123.44000

--Round to 2 places after the decimal point. (to the right)

PRINT ROUND(123.44500, 2, 0);

--Output  :  123.45000

--Round to 2 places after the decimal point. (to the right)

PRINT ROUND(123.44500, 2, 1);

--Output  :  123.44000

--Truncate anything after 2 places, after the decimal point. (to the right)

PRINT ROUND(123.45000, 1);

--Output  :  123.50000

--Round to 1 places after the decimal point. (to the right)

PRINT ROUND(123.44000, 1, 0);

--Output  :  123.40000

--Round to 1 places after the decimal point. (to the right)

PRINT ROUND(123.44000, 1, 1);

--Output  :  123.40000

--Truncate anything after 1 places, after the decimal point. (to the right)

PRINT ROUND(455.44500, -2);

--500.00000

--Round the last 2 places before the decimal point. (to the left)

PRINT ROUND(445.44500, -2);

-- 400.00000

--Round the last 2 places before the decimal point. (to the left)

PRINT ROUND(455.44500, -1);

-- 460.00000

--Round the last 1 places before the decimal point. (to the left)

PRINT ROUND(454.44500, -1);

-- 450.00000

--Round the last 1 places before the decimal point. (to the left)

/\*

1.

ROUND(numeric\_expression,length[,function])

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/round-transact-sql>

Rounds the given numeric expression based on the given length.

1.1.

numeric\_expression :

numeric expression except for the bit data type.

1.2.

length int(precision length int):

If precision length int > 0 ,

then ROUND() is applied for the decimal part. (to the right)

If precision length int < 0 ,

then ROUND() is applied to the number before the decimal. (to the left)

1.3.

function (operation options):

Zero as default means operating rounding.

Non-Zero is truncated which means truncate anything after the precision length.

\*/