**Part 29**

**ABS ( numeric\_expression )** - ABS stands for absolute and returns, the absolute (positive) number.   
  
**For example**, Select ABS(-101.5) -- returns 101.5, without the - sign.

**CEILING ( numeric\_expression ) and FLOOR ( numeric\_expression )**  
**CEILING**and **FLOOR** functions accept a numeric expression as a single parameter. CEILING() returns the smallest integer value greater than or equal to the parameter, whereas FLOOR() returns the largest integer less than or equal to the parameter.   
  
**Examples:**  
Select CEILING(15.2) -- Returns 16  
Select CEILING(-15.2) -- Returns -15  
  
Select FLOOR(15.2) -- Returns 15  
Select FLOOR(-15.2) -- Returns -16  
  
**Power(expression, power)** - Returns the power value of the specified expression to the specified power.  
  
**Example**: The following example calculates '2 TO THE POWER OF 3' = 2\*2\*2 = 8  
Select POWER(2,3) -- Returns 8  
  
**RAND([Seed\_Value])** - Returns a random float number between 0 and 1. Rand() function takes an optional seed parameter. When seed value is supplied the   
  
RADN() function always returns the same value for the same seed.  
  
**Example:**  
Select RAND(1) -- Always returns the same value  
  
**If you want to generate a random number between 1 and 100**, RAND() and FLOOR() functions can be used as shown below. Every time, you execute this query, you get a random number between 1 and 100.  
Select FLOOR(RAND() \* 100)  
  
**The following query prints 10 random numbers between 1 and 100.**  
Declare @Counter INT  
Set @Counter = 1  
While(@Counter <= 10)  
Begin  
 Print FLOOR(RAND() \* 100)  
 Set @Counter = @Counter + 1  
End  
  
**SQUARE ( Number )** - Returns the square of the given number.  
  
**Example:**  
Select SQUARE(9) -- Returns 81  
  
**SQRT ( Number )** - SQRT stands for Square Root. This function returns the square root of the given value.  
  
**Example:**  
Select SQRT(81) -- Returns 9  
  
**ROUND ( numeric\_expression , length [ ,function ] )** - Rounds the given numeric expression based on the given length. This function takes 3 parameters.   
**1. Numeric\_Expression** is the number that we want to round.

**2. Length parameter**, specifies the number of the digits that we want to round to. If the length is a positive number, then the rounding is applied for the decimal part, where as if the length is negative, then the rounding is applied to the number before the decimal.  
**3. The optional function parameter**, is used to indicate rounding or truncation operations. A value of 0, indicates rounding, where as a value of non zero indicates truncation. Default, if not specified is 0.  
  
**Examples:**  
-- Round to 2 places after (to the right) the decimal point  
Select ROUND(850.556, 2) -- Returns 850.560  
  
-- Truncate anything after 2 places, after (to the right) the decimal point  
Select ROUND(850.556, 2, 1) -- Returns 850.550  
  
-- Round to 1 place after (to the right) the decimal point  
Select ROUND(850.556, 1) -- Returns 850.600  
  
-- Truncate anything after 1 place, after (to the right) the decimal point  
Select ROUND(850.556, 1, 1) -- Returns 850.500  
  
-- Round the last 2 places before (to the left) the decimal point  
Select ROUND(850.556, -2) -- 900.000  
  
-- Round the last 1 place before (to the left) the decimal point  
Select ROUND(850.556, -1) -- 850.000

[Email This](https://www.blogger.com/share-post.g?blogID=6082652835152798567&postID=5056536853696263313&target=email)