

POSTGRESQL - NUMERIC FUNCTION

http://www.tutorialspoint.com/postgresql/postgresql_numeric_functions.htm

Copyright © tutorialspoint.com

PostgreSQL numeric functions are used primarily for numeric manipulation and/or mathematical calculations. The following table details the numeric functions:

Name	Description
ABS	Returns the absolute value of numeric expression.
ACOS	Returns the arccosine of numeric expression. Returns NULL if the value is not in the range -1 to 1.
ASIN	Returns the arcsine of numeric expression. Returns NULL if value is not in the range -1 to 1
ATAN	Returns the arctangent of numeric expression.
ATAN2	Returns the arctangent of the two variables passed to it.
CEIL	Returns the smallest integer value that is not less than passed numeric expression
CEILING	Returns the smallest integer value that is not less than passed numeric expression
COS	Returns the cosine of passed numeric expression. The numeric expression should be expressed in radians.
COT	Returns the cotangent of passed numeric expression.
DEGREES	Returns numeric expression converted from radians to degrees.
EXP	Returns the base of the natural logarithm e raised to the power of passed numeric expression.
FLOOR	Returns the largest integer value that is not greater than passed numeric expression.
GREATEST	Returns the largest value of the input expressions.
LEAST	Returns the minimum-valued input when given two or more.
LOG	Returns the natural logarithm of the passed numeric expression.
MOD	Returns the remainder of one expression by dividing by another expression.
PI	Returns the value of pi
POW	Returns the value of one expression raised to the power of another expression
POWER	Returns the value of one expression raised to the power of another expression
RADIANS	Returns the value of passed expression converted from degrees to radians.
ROUND	Returns numeric expression rounded to an integer. Can be used to round an expression to a number of decimal points
SIN	Returns the sine of numeric expression given in radians.

SQRT	Returns the non-negative square root of numeric expression.
TAN	Returns the tangent of numeric expression expressed in radians.

ABSX

The ABS function returns the absolute value of X. Consider the following example:

```
testdb=# SELECT ABS(2);
+-----+
| ABS(2) |
+-----+
| 2      |
+-----+
1 row in set (0.00 sec)

testdb=# SELECT ABS(-2);
+-----+
| ABS(2) |
+-----+
| 2      |
+-----+
1 row in set (0.00 sec)
```

ACOSX

This function returns the arccosine of X. The value of X must range between -1 and 1 or NULL will be returned. Consider the following example:

```
testdb=# SELECT ACOS(1);
+-----+
| ACOS(1) |
+-----+
| 0.000000 |
+-----+
1 row in set (0.00 sec)
```

ASINX

The ASIN function returns the arcsine of X. The value of X must be in the range of -1 to 1 or NULL is returned.

```
testdb=# SELECT ASIN(1);
+-----+
| ASIN(1) |
+-----+
| 1.5707963267949 |
+-----+
1 row in set (0.00 sec)
```

ATANX

This function returns the arctangent of X.

```
testdb=# SELECT ATAN(1);
+-----+
| ATAN(1) |
+-----+
| 0.78539816339745 |
+-----+
1 row in set (0.00 sec)
```

ATAN2Y,X

This function returns the arctangent of the two arguments: X and Y. It is similar to the arctangent of Y/X, except that the signs of both are used to find the quadrant of the result.

```
testdb=# SELECT ATAN2(3,6);
+-----+
| ATAN2(3,6) |
+-----+
| 0.46364760900081 |
+-----+
1 row in set (0.00 sec)
```

CEILX

CEILINGX

These function return the smallest integer value that is not smaller than X. Consider the following example:

```
testdb=# SELECT CEILING(3.46);
+-----+
| CEILING(3.46) |
+-----+
| 4 |
+-----+
1 row in set (0.00 sec)

testdb=# SELECT CEIL(-6.43);
+-----+
| CEIL(-6.43) |
+-----+
| -6 |
+-----+
1 row in set (0.00 sec)
```

COSX

This function returns the cosine of X. The value of X is given in radians.

```
testdb=#SELECT COS(90);
+-----+
| COS(90) |
+-----+
| -0.44807361612917 |
+-----+
1 row in set (0.00 sec)
```

COTX

This function returns the cotangent of X. Consider the following example:

```
testdb=#SELECT COT(1);
+-----+
| COT(1) |
+-----+
| 0.64209261593433 |
+-----+
1 row in set (0.00 sec)
```

DEGREESX

This function returns the value of X converted from radians to degrees.

```
testdb=#SELECT DEGREES(PI());
+-----+
| DEGREES(PI()) |
+-----+
```

```
| DEGREES(PI()) |
+-----+
| 180.000000 |
+-----+
1 row in set (0.00 sec)
```

EXPX

This function returns the value of e *the base of the natural logarithm* raised to the power of X.

```
testdb=#SELECT EXP(3);
+-----+
| EXP(3) |
+-----+
| 20.085537 |
+-----+
1 row in set (0.00 sec)
```

FLOORX

This function returns the largest integer value that is not greater than X.

```
testdb=#SELECT FLOOR(7.55);
+-----+
| FLOOR(7.55) |
+-----+
| 7 |
+-----+
1 row in set (0.00 sec)
```

GREATEST $n_1, n_2, n_3, \dots$

The GREATEST function returns the greatest value in the set of input parameters n_1, n_2, n_3 , and soon. The following example uses the GREATEST function to return the largest number from a set of numeric values:

```
testdb=#SELECT GREATEST(3, 5, 1, 8, 33, 99, 34, 55, 67, 43);
+-----+
| GREATEST(3, 5, 1, 8, 33, 99, 34, 55, 67, 43) |
+-----+
| 99 |
+-----+
1 row in set (0.00 sec)
```

LEAST $N_1, N_2, N_3, N_4, \dots$

The LEAST function is the opposite of the GREATEST function. Its purpose is to return the least-valued item from the value list N_1, N_2, N_3 , and soon. The following example shows the proper usage and output for the LEAST function:

```
testdb=#SELECT LEAST(3, 5, 1, 8, 33, 99, 34, 55, 67, 43);
+-----+
| LEAST(3, 5, 1, 8, 33, 99, 34, 55, 67, 43) |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)
```

LOGX

LOG B, X

The single argument version of the function will return the natural logarithm of X. If it is called with two arguments, it returns the logarithm of X for an arbitrary base B. Consider the following

example:

```
testdb=#SELECT LOG(45);
+-----+
| LOG(45) |
+-----+
| 1.65321251377534 |
+-----+
1 row in set (0.00 sec)

testdb=#SELECT LOG(2,65536);
+-----+
| LOG(2,65536) |
+-----+
| 16.0000000 |
+-----+
1 row in set (0.00 sec)
```

MODN,M

This function returns the remainder of N divided by M. Consider the following example:

```
testdb=#SELECT MOD(29,3);
+-----+
| MOD(29,3) |
+-----+
| 2 |
+-----+
1 row in set (0.00 sec)
```

PI

This function simply returns the value of pi. SQL internally stores the full double-precision value of pi.

```
testdb=#SELECT PI();
+-----+
| PI() |
+-----+
| 3.141593 |
+-----+
1 row in set (0.00 sec)
```

POWX,Y

POWERX,Y

These two functions return the value of X raised to the power of Y.

```
testdb=# SELECT POWER(3,3);
+-----+
| POWER(3,3) |
+-----+
| 27 |
+-----+
1 row in set (0.00 sec)
```

RADIANSX

This function returns the value of X, converted from degrees to radians.

```
testdb=#SELECT RADIANS(90);
+-----+
| RADIANS(90) |
+-----+
```

```

1.570796
+-----+
1 row in set (0.00 sec)

```

ROUNDX

ROUNDX,D

This function returns X rounded to the nearest integer. If a second argument, D, is supplied, then the function returns X rounded to D decimal places. D must be positive or all digits to the right of the decimal point will be removed. Consider the following example:

```

testdb=#SELECT ROUND(5.693893);
+-----+
| ROUND(5.693893) |
+-----+
| 6 |
+-----+
1 row in set (0.00 sec)

testdb=#SELECT ROUND(5.693893,2);
+-----+
| ROUND(5.693893,2) |
+-----+
| 5.69 |
+-----+
1 row in set (0.00 sec)

```

SIGNX

This function returns the sign of X *negative, zero, or positive* as -1, 0, or 1.

```

testdb=#SELECT SIGN(-4.65);
+-----+
| SIGN(-4.65) |
+-----+
| -1 |
+-----+
1 row in set (0.00 sec)

testdb=#SELECT SIGN(0);
+-----+
| SIGN(0) |
+-----+
| 0 |
+-----+
1 row in set (0.00 sec)

testdb=#SELECT SIGN(4.65);
+-----+
| SIGN(4.65) |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)

```

SINX

This function returns the sine of X. Consider the following example:

```

testdb=#SELECT SIN(90);
+-----+
| SIN(90) |
+-----+
| 0.893997 |
+-----+

```

```
1 row in set (0.00 sec)
```

SQRTX

This function returns the non-negative square root of X. Consider the following example:

```
testdb=#SELECT SQRT(49);
+-----+
| SQRT(49) |
+-----+
| 7 |
+-----+
1 row in set (0.00 sec)
```

TANX

This function returns the tangent of the argument X, which is expressed in radians.

```
testdb=#SELECT TAN(45);
+-----+
| TAN(45) |
+-----+
| 1.619775 |
+-----+
1 row in set (0.00 sec)
```

Processing math: 100%