



Cheat Sheet: Numeric Functions

TRUNC:

Returns the number or expression passed as parameter truncated to *n* decimal digits.

Syntax:

TRUNC(original_number[,n])

The parameter for the number of decimals (*n*) is optional. It is omitted it defaults to 0, which means that the function truncates the original number to convert it into an integer.

If *n* is negative, the function truncates *n* digits to the left of the decimal point.

ROUND:

Returns the number or expression passed as parameter rounded to *n* decimal digits.

Syntax:

ROUND(original_number[,n])

The parameter for the number of decimals (*n*) is optional. It is omitted it defaults to 0, which means that the function rounds the original number to convert it into an integer.

If *n* is negative, the function rounds off *n* digits to the left of the decimal point.

FLOOR:

Returns the largest integer that is equal to or less than the number or expression passed as parameter. In other words, it removes any decimals from the original number, if it is not an integer. If the original number is an integer, the FLOOR function returns the same integer.

Syntax:

FLOOR(original_number)

CEIL:

Returns the smallest integer that is greater than or equal to the number or expression passed as parameter. In other words, if the number passed has decimals it returns the next integer. If the original number is an integer, the CEIL function returns the same integer.

Syntax:

CEIL(original_number)

SIGN:

Returns a number that represents the sign of the number passed as parameter. If the number passed is negative it returns -1, if the number passed is positive it returns 1, and it returns 0 if the number passed is 0.

Syntax:

SIGN(number)

MOD:

Returns the remainder of the *first_number* divided by the *second_number*.

Syntax:

MOD(first_number,second_number)