

# Windward Functions Reference

Here is a complete listing of every supported Windward function.

We start with a brief discussion about the data types used by Windward functions. Then there are tables of Windward functions by category. Links are provided that can be used to jump forward to a specific category of functions.

Windward functions and equations can be written from scratch in the [Query Tab](#) of Tags. But the [Equation Editor](#) makes writing functions easier, and should be used whenever possible.

## Data Types

Windward functions use a variety of data types. Using the correct data types is very important to ensure the functions return the expected results.

Along with function names and descriptions, function signatures (the data type returned by a function, and the data types of the function arguments) are included in the tables below. Here are descriptions of the data types referred to:

**Boolean** - the logical values *true* or *false*

**char** - a single character

**dataset** - a set of values (rows or nodes) returned by a SQL, XPath or JSONPath query

**date** - a string interpreted as a fully-typed date; e.g. [ISO 8601 date time format](#)

**datetime** - a Windward datetime object. Typed dates from the datasource, and dates stored in a template variable of type date will be of this type.

**datetimespan** - a span of time in years, months and days returned by DATESPAN()

**null** - a special data type that represents *no value*

**number** - a number with zero or more decimal places (integer or double)

**range** - an unordered, comma-delimited list of values, e.g. "1,3,2", "\${var1},\${var3},\${var2}"

**string** - a sequence of characters

**variable** - a template variable, Input Parameter or the named variable used by the ADDTOTAL() and GETTOTAL() functions. Refer to a template variable or Input Parameter by using the "\${VariableName}" syntax.

# Functions List

Use these links to jump forward to a specific category of functions:

[Date & Time](#)

[Math & Trig](#)

[Statistical](#)

[Database](#)

[Text](#)

[Logical](#)

[Custom](#)

## Date & Time

DATE	date DATE(number year, number month, number day)	Returns a fully-typed date with the given year, month and day.
DATEDIF	number DATEDIF(date start_date, date end_date, string 'D'   'M'   'MD'   'YD'   'YM')	Calculates the number of days, months, or years between two dates.
DATESPAN	datetimespan DATESPAN(number years, number months, number days)	Can be added to a fully-typed date to offset it.
DATEVALUE	number DATEVALUE(number year, number month, number day)	Returns the serial number of the given date, using the 1900 date system.
DAY	number DAY(datetime n)	Returns the day of the given date.
DAYS	number DAYS(datetime firstDate, datetime secondDate)	Returns the number of days between two given dates.
HOURL	number HOURL(datetime n)	Returns the hour of the given datetime.
MINUTE	number MINUTE(datetime n)	Returns the minutes of the datetime.
MONTH	number MONTH(datetime n)	Returns the month of the given

		datetime.
NOW	date now()	Returns the current time formatted as a fully-typed date and time.
SECOND	number SECOND(datetime n)	Returns the seconds of the given datetime.
TIME	date TIME(number hour, number minute, number second)	Returns the time of the given hour, minute and second as a fully-typed date.
TIMESPAN	datetimespan TIMESPAN(number hours, number minutes, number seconds)	Can be added to a time to offset it.
TIMEVALUE	number timevalue(number hour, number minute, number second)	Returns the decimal number of the time.
TIMEZONE	string TIMEZONE(datetime)	<ol style="list-style-type: none"> <li>1. When a datetime object has an explicit time zone, the zone ID is returned. A list of zone IDs can be found at this link (<a href="https://www.mkyong.com/java8/java-display-all-zoneid-and-its-utc-offset/">https://www.mkyong.com/java8/java-display-all-zoneid-and-its-utc-offset/</a>), EX: "America/Denver"</li> <li>2. When a datetime contains a UTC offset but no timezone, the time offset is returned.</li> <li>3. When a datetime contains no timezone or offset, UTC time is assumed as the timezone. The TIMEZONE macro will return a "Z" to indicate UTC time.</li> </ol> <p><b>Added in version 16.3.0</b></p> <p><b>Note:</b> For some datetimes with a timezone, the Java API does not return the timezone following the standard cited above. In these cases, we are restricted by the capabilities of</p>

		the Java API and the output will be a timezone offset from GMT.
TODAY	string TODAY()	Returns the current date in the UTC timezone.
TOLOCALTIME	datetime TOLOCALTIME(datetime)	The local time conversion of a UTC time to the machines current timezone. Be sure that the input datetime is in the UTC "timezone." <b>Added in version 16.1.0</b>
WEEKDAY	number WEEKDAY(number year, number month, number day)	Returns a number representing the day of the week of the given date, where Monday is day 1.
WEEKNUM	number WEEKNUM(number year, number month, number day)	Returns the week number of a specific date.
YEAR	number YEAR(datetime n)	Returns the year of the given datetime.
YEARFRAC	number YEARFRAC(number year1, number month1, number day1, number year2, number month2, number day2)	Returns the fraction of the year represented by the number of whole days between two dates.
OFFSETTOTALMINUTES	OFFSETTOTALMINUTES(datetime n)	Returns the total minutes that the time is offset from UTC. Return value could be a positive or a negative integer. <b>Added in 16.5.0</b>
TOUCTTIME	datetime TOUTCTIME(datetime n)	Returns the equivalent datetime in UTC.
UTCNOW	datetime UTCNOW()	Returns the current datetime in UTC.
WORKD.INTL	number WORKD.INTL(number year, number month, number day, number days_passed, number weekend)	Returns the date (in 1900 date system), after a specified number of workdays pass with custom weekend length.
WORKDAY	datetime WORKDAY(number year, number month, number	Returns the datetime after a specified number of workdays

	day, number days_passed)	pass.
--	--------------------------	-------

[Return to Functions List.](#)

## Math & Trig

Name	Signature	Description
ABS	number ABS(number n)	Returns the absolute value of a number.
BASE	string BASE(number n, number radix, number min_length)	Converts a number into a text representation with the given radix (base).
BASE64DECODE	string BASE64DECODE(string s)	Decodes a base64-encoded image.
CEILING	number CEILING(number n, number significance)	Returns a number rounded up, away from zero to the nearest integer or to the nearest multiple of significance.
DECIMAL	number DECIMAL(string text, number radix)	Converts a text representation of a number in a given base into a decimal number.
DEGREES	number DEGREES(number radians)	Converts radians into degrees.
EVEN	number EVEN(number n)	Returns number rounded up to the nearest even integer.
FACT	number FACT(number n)	Returns the factorial of a number. The factorial is equal to $1*2*3*\dots*number$ .
FLOOR	number FLOOR(number n, number m)	Returns a number rounded down to the nearest integer or to the nearest multiple of significance.
ISEVEN	Boolean ISEVEN(number n)	Returns <i>true</i> if n is even, or <i>false</i> if n is odd.

Name	Signature	Description
ISODD	Boolean ISODD(number n)	Returns <i>true</i> if the number is odd, <i>false</i> otherwise
PI	number PI()	Returns the number pi (3.14...) accurate to 15 digits.
ODD	number ODD(number n)	Returns a number rounded up to the nearest odd integer.
POWER	number POWER(number n, number power)	Returns the result of the number raised to a power.
QUOTIENT	number QUOTIENT(number dividend, number divisor)	Returns the integer portion of a division.
RAND	number RAND()	Returns an evenly distributed random real number greater than or equal to 0 and less than 1.
RANDBETWEEN	number RANDBETWEEN(number bottom, number top)	Returns a random integer between the specified numbers.
ROUND	number ROUND(number n, number num_digits)	Returns a number rounded to a specified number of digits.
SEC	number SEC(number angle)	The given angle should be in radians, returns the secant of that angle in degrees.
STANDARDIZE	number STANDARDIZE(number n, number mean, number standard_dev)	Returns a normalized value for a given value n and a distribution characterized by mean and standard_dev. Uses the formula for Standard Score.
STDEV	number STDEV(number n, number m, )	Returns the standard deviation of group of numbers.
STDEVA	number STDEVA(string   number   Boolean n, string   number   Boolean m, ...)	Returns the standard deviation of group of numbers

Name	Signature	Description
STDEVP	number STDEVP(number n, number m, ...)	Returns the standard deviation of the given arguments based on the entire population.
STDEVPA	number STDEVPA(string   number   Boolean n, string   number   Boolean m, ...)	Returns the standard deviation of the given arguments based on the entire population.
SQRT	number SQRT(number n)	Returns the square root of the number.
SUM	number SUM(number n, number m, ...)	Adds all the numbers in a range of values.
SUMPRODUCT	number SUMPRODUCT(range r1, range r2, ...)	Multiplies corresponding components in the given ranges, and returns the sum of those products.

[Return to Functions List.](#)

## Statistical

Name	Signature	Description
ADDTOTAL	ADDTOTAL(number n, variable 'var')	Adds a number to a running total held in a named variable.
AVEDEV	number AVEDEV(number n, number m, ...)	Returns the average of the absolute deviations of data points from their mean.
AVERAGE	number AVERAGE(number   range   variable n, number   range   variable m, ...)	Returns the average (arithmetic mean) of its arguments, which can be numbers, ranges or variable references that contain numbers.
FREQUENCY	<div>FREQUENCY (data_range, bin_range)</div>	Returns the number of times each element in bin_range was repeated in

Name	Signature	Description
		the specified data_range.  <b>Removed in 16.5.0</b>
COUNT	number COUNT(range r) number COUNT(number n, number m, ...)	Counts the number of values in a range or a list of numbers.
GETTOTAL	number GETTOTAL(variable 'var')	Return the value of a running total held in a named variable.
MAX	number MAX(number n, number m, ...)	Returns the largest value in a set of values.
MEDIAN	number MEDIAN(number n, number m, ...)	Returns the median of the given numbers. The median is the number in the middle of a set of numbers.
MIN	number MIN(number n, number m, ...)	Returns the smallest value in a set of values.
PERCENTILE.EXC	number PERCENTILE.EXC(number n, number m, number k)	Returns the k-th percentile of values in a range, where k is in the range 0..1, exclusive.
PERCENTILE.INC	number PERCENTILE.INC(number n, number m, number k)	Returns the k-th percentile of values in a range, where k is in the range 0..1, inclusive.
PERCENTILERANK.EXC	number PERCENTILERANK.EXC(number n, number m, number x)	Returns the rank of a value in a range as a percentage (0..1, exclusive) of the dataset.
PERCENTILERANK.INC	number PERCENTILERANK.INC(number n, number m, number x)	Returns the rank of a value in a range as a percentage (0..1, inclusive) of the dataset.
PRODUCT	number MULTIPLICATION(dataset d)	Returns the multiplication of all the values in a dataset.

[Return to Functions List.](#)



## Database

Name	Signature	Description
DATA	dataset DATA(string "text")	Returns a dataset using the string "text". Note "text" must be enclosed in double quotes.
DATEDATA	date DATEDATA(string query, string pattern)	Returns a fully-typed date using the dataset returned by query and using pattern to read in dates in a non-standard format.
NULL	null NULL()	Returns the NULL value. Primarily used to compare against the value returned by DATA() or DATEDATA().

[Return to Functions List.](#)

## Text

Name	Signature	Description
CHAR	char CHAR(number ascii_code)	Returns the character specified by the <a href="#">ASCII code</a> .
CONCATENATE	string CONCATENATE(string s, string t, ...)	Joins two or more text strings into one string and returns that string.
CONTAINS	Boolean CONTAINS(string within_text, string find_text)	Returns true if the string within_text contains the string find_text; otherwise returns false.
INDEXOF	number INDEXOF(string within_text, string find_text)	Returns the first index of one string within another string.
LASTINDEXOF	number LASTINDEXOF(string within_text, string find_text)	Returns the last index of find_text within within_text.
LEFT	string LEFT(string text, number length)	Returns length characters

Name	Signature	Description
		beginning from the left end of string text.
LEN	number LEN(string text)	Returns the number of characters in a string.
LOWER	string LOWER(string text)	Converts all letters in a string to lowercase.
MID	string MID(string text, number start, number length)	Returns the characters from the middle of a text string, give a starting position and length.
NUMBERVALUE	number NUMBERVALUE(string text, char decimal_separator, char thousands_separator)	Converts text to a number in a locale-independent way.
PROPER	string PROPER(string text)	Capitalizes the first letter in a text string and any other letters in text that follow a character other than a letter. Converts all other letters to lowercase letters.
REGEXEXTRACT	string REGEXEXTRACT(string text, string regex)	Extracts matching substrings according to a regular expression.
REGEXMATCH	Boolean REGEXMATCH(string text, string regex)	Returns true if a piece of text matches a regular expression, false otherwise.
REGEXREPLACE	string REGEXREPLACE(string text, string regex, string replace_text)	Replaces a part of a text string with a different text string using a regular expression.
REPLACE	string REPLACE(string text, string pattern, string replacement, Boolean ignoreCase)	Replaces each substring of the text with the replacement.
RIGHT	string RIGHT(string text, number num_chars)	Returns length characters beginning from the right end of string text.
SEARCH	number SEARCH(string search_text,	Returns the position of the

Name	Signature	Description
	string text, number start_pos)	character at which a specific character or text string is first found, beginning with start_pos (optional).
SUBSTITUTE	string SUBSTITUTE(string text, string old_text, string new_text, number instance_num)	Replaces each substring of the text with the replacement. instance_num (optional) specifies which occurrence of old_text to replace.
SUBSTRING	string SUBSTRING(string text, number start_num, number end_num)	Returns the characters from the middle of a text string, given a starting position and an ending position.
TEXT	string TEXT(number   date value, string format)	<p>Converts a numeric or date value to text and lets you specify the display formatting by using special format strings.</p> <p><b>Note:</b> This macro does not support fractions as inputs. instead, use the decimal form of the number.</p>
TRIM	string TRIM(string s)	Returns the string with the whitespace removed from the beginning and the end.
UPPER	string UPPER(string s)	Converts a text string to all uppercase letters.
URLDECODE	string URLDECODE(string url)	Decodes the URL according to <a href="#">RFC2396</a> . Returns the decoded URL.
URLENCODE	string URLENCODE(string url)	Encodes the URL according to <a href="#">RFC2396</a> . Returns the encoded URL.
VALUE	number VALUE(string text)	Converts a text string to a number.
XPATH	string XPATH(string xml, string xpath)	Performs an XPath select on

Name	Signature	Description
		the given XML.

[Return to Functions List.](#)

## Logical

Name	Signature	Description
FALSE	Boolean FALSE()	Returns the logical value <i>false</i> .
IF	number   string IF(Boolean expr, number   string value_if_true, number   string value_if_false)	Evaluates expr and returns one value if expr is <i>true</i> , or returns another value if expr is <i>false</i> .
IFERROR	number   string   dataset IFERROR(number   string   dataset value, number   string   dataset value_if_error)	Tests if evaluating the first argument returns an error; and if so, returns the second argument; else returns the first argument.
ISNUMBER	Boolean ISNUMBER(<any data type> value)	Returns <i>true</i> if value is a number; otherwise returns <i>false</i> .
TRUE	Boolean TRUE()	Returns the logical value <i>true</i> .

[Return to Functions List.](#)

## Custom

Name	Signature	Description
RANGE	range RANGE(string   number   variable x, string   number   variable y, ...)	Returns a range (unordered, comma-delimited list) of values.
RANGEARRAY	range RANGEARRAY(number   date start, number   date end, (optional) number step_size, (optional) unit)	Returns an array of values from start (inclusive) to end (inclusive), default step size is 1.

Name	Signature	Description
DISTINCT	array DISTINCT(array source)	Returns an identical array to the source array with all duplicate values removed.

[Return to Functions List.](#)