

BASIC MATH			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Basic Math	average	\bar{x} returns the average of multiple values	average(value1, value2, ..., valueN)
Basic Math	sum	returns the sum of multiple values	sum(value1, value2, ..., valueN)
Basic Math	-	subtraction operator, also for dates	value1 - value2
Basic Math	+	addition operator, also for dates	value1 + value2
Basic Math	%	percentageoperator (/100)	value1%
Basic Math	*	multiplication operator	value1*value3
Basic Math	/	division operator	value1/value4
Basic Math	^	exponentiation operator	value1^value8
Basic Math	abs	calculates a number's absolute value	abs(number)
Basic Math	mod	gets the remainder following integer division of two numbers	mod(numerator, denominator)
Basic Math	quotient	Performs integer division on two numbers disregarding any remainder	quotient(numerator, denominator)
Basic Math	product	multiplies a series of numbers to return their total product	product(Value1, value2, ..., valueN)
Basic Math	power	arbitrarypower of	power(value, exponent)
Basic Math	min	returns the smallest numeric value out of a series	min(value1, value2, ..., valueN)
Basic Math	max	returns the biggest numeric value out of a series	max(value1, value2, ..., valueN)
Basic Math	exp	calculates the mathematical constant, e, raised to the specific power	exp(exponent)
Basic Math	ln	calculates the natural logarithm of a specified numeric value	ln(value)
Basic Math	log10	calculates the logarithm (base 10) of a specified numeric value	log10(value)
Basic Math	log	calculates a logarithm for a specified numeric value to the specified base	log(value, base)
Basic Math	round	rounds the fractional portion of a numeric value up or down to produce an integer	round(value, number_of_digits)
Basic Math	pi	the mathematical constant	pi0
Basic Math	sqrt	calculates the square root of a number	sqrt(value)
Basic Math	floor	calculate the next lesser whole number for a specified numeric value	floor(value)
Basic Math	ceiling	returns the smallest whole number greater than or equal to the given number	ceiling(value)
Basic Math	even	round to the nearest bigger (in abs terms) even integer	even(number)
Basic Math	odd	round to the nearest bigger (in abs terms) odd integer	odd(number)
Basic Math	fact	returns the factorial of a number	fact(number)
Basic Math	factdouble	returns the double factorial of a number	factdouble(number)
Basic Math	gcd	returns the greatest common divisor of integer values	gcd(number1, [number2, number3, ..., numberN])
Basic Math	lcm	returns the least common multiple of integer values	lcm(number1, [number2, number3, ..., numberN])
Basic Math	multinomial	returns the multinomial of a set of numbers	multinomial(number1, [number2, number3, ..., numberN])
Basic Math	seriesum	returns the sum of a power series	seriesum(inputValue, initialPower, step, coefficient1 [, coefficient2, ..., coefficientN])
Basic Math	sign	returns the sign of a number (-1, 0, or 1)	sign(number)
Basic Math	sqrtpi	Returns the square root of the specified number times PI.	sqrtpi(numner)

STRING MANIPULATION			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
String Manipulation	&	concatenates two strings	text1 & text2
String Manipulation	concatenate	concatenate multiple strings	concatenate(text1, text2, ..., textN)
String Manipulation	lower	changes case to lower	lower(text_value)
String Manipulation	upper	changes case to upper	upper(text_value)
String Manipulation	left	get leftmost characters from string	left(text_value, num_chars)
String Manipulation	right	get rightmost characters from string	right(text_value, num_chars)
String Manipulation	mid	get characters from the middle of the string	mid(text_value, starting_point, num_chars)
String Manipulation	trim	remove leading/trailing whitespaces, normalize spaces inside	trim(text_value)
String Manipulation	len	length of a string	len(text_value)
String Manipulation	find	find one piece of text within another	find(text_to_find, text_to_search, starting_point)
String Manipulation	replace	replace one piece of text within another	replace(text_value, starting_point, character_count, new_text_value)
String Manipulation	code	returns the numeric code of the first char	code(string)

LOGICAL OPERATIONS			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Logical Ops	=	equal operator	value1=value2
Logical Ops	!=	non-equal operator, also <>	value1!=value3
Logical Ops	<	less-than operator	value1<value4
Logical Ops	<=	less-equal operator	value1<=value5
Logical Ops	>	greater-than operator	value1>value6
Logical Ops	>=	greater-equal-operator	value1>=value7
Logical Ops	if	chooses between two outcomes	if(value1,aaa,bbb)
Logical Ops	TRUE	always true	true()
Logical Ops	FALSE	always false	false()
Logical Ops	not	logical not	not(boolValue)
Logical Ops	and	logical and	and(boolValue1, boolValue2, ..., boolValueN)
Logical Ops	or	logical or	or(boolValue1, boolValue2, ..., boolValueN)
Logical Ops	isblank	checks for a blank value	isblank(value)

Logical Ops	iserr	checks for any error value except #N/A!	iserr(value)
Logical Ops	iserror	checks for any error value	iserror(value)
Logical Ops	islogical	checks for a boolean value	islogical(value)
Logical Ops	isna	checks for #N/A!	isna(value)
Logical Ops	isnontext	checks for non-text values	isnontext(value)
Logical Ops	istext	checks for text values	istext(value)
Logical Ops	isnumber	checks for numeric values	isnumber(value)
Logical Ops	isref	checks for an UltraCalc reference	isref(reference)
Logical Ops	iseven	checks for an even number	iseven(value)
Logical Ops	isodd	checks for an odd number	isodd(value)
Logical Ops	isnull	checks for null values	isnull(value)
Logical Ops	isdbnull	checks for DBNull values	isdbnull(value)
Logical Ops	delta	Compares two numbers and returns 1 if they are equal or 0 if they are not.	delta(number1, number2)
Logical Ops	gestep	Compares two numbers and returns 1 the first number is greater than or equal to the second or returns 0 if not	gestep(number, step)

CONVERSIONS			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Conversions	int	cast to integer	int(value)
Conversions	trunc	Truncates the fractional portion of a numeric value to produce an integer	trunc(value, number_of_digits)
Conversions	value	parses text into a number	value(text_value)
Conversions	mround	Rounds a number to the nearest multiple of another number	mround(number, multiple)
Conversions	roman	Converts a number into a roman number as a string	roman(number, form)
Conversions	rounddown	Rounds a number to down to the specified number of digits	rounddown(number, digits)
Conversions	roundup	Rounds a number to up to the specified number of digits	roundup(number, digits)
Conversions	dollarfr	Converts a dollary amount expressed as a decimal into a dollar amount expressed as a fraction	dollarfr(decimalDollarAmount, Fraction)
Conversions	dollarde	Converts a dollary amount expressed as a fraction into a dollar amount expressed as a decimal	dollarde(num, denom)
Conversions	n	Converts a value to a number	n(value)
Conversions	dec2x	returns a string representing a decimal value in a given base number scheme	
Conversions	dec2bin	returns a string representing a decimal value in a base 2	dec2bin(number, places)
Conversions	dec2hex	returns a string representing a decimal value in a base 16	dec2hex(number, places)
Conversions	dec2oct	returns a string representing a decimal value in a base 8	dec2oct(number, places)
Conversions	x2dec	returns a number from a string representation in a non-decimal base	x2dec(text)
Conversions	bin2dec	returns a number from a string representation in base 2	bin2dec(text)
Conversions	hex2dec	returns a number from a string representation in base 16	hex2dec(text)
Conversions	oct2dec	returns a number from a string representation in base 8	oct2dec(text)
Conversions	xbase2xbase	convert a string representation in a non-decimal base into another base	xbase2xbase(number, places)
Conversions	bin2oct	convert from binary to octal	bin2oct(number, places)
Conversions	bin2hex	convert from binary to hexadecimal	bin2hex(number, places)
Conversions	oct2bin	convert octal to binary	oct2bin(number, places)
Conversions	hex2bin	convert hexadecimal to binary	hex2bin(number, places)
Conversions	oct2hex	convert octal to hexadecimal	oct2hex(number, places)
Conversions	hex2oct	convert hexadecimal to octal	hex2oct(number, places)
Conversions	convert	Converts a value from one system of measurement to another.	convert(number, fromUnit, toUnit)

DATE TIME			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
DateTime	datevalue	converts a string into a date	datevalue(text_value)
DateTime	date	creates a date using three integers	date(year, month,day)
DateTime	days360	returns datediff in a 360-days-based year, with NASD method or european	days360(start_date,end_date, bool_Eur_method)
DateTime	day	returns day of a date	day(date_value)
DateTime	month	returns month of a date	month(date_value)
DateTime	year	returns year of a date	year(date_value)
DateTime	hour	returns hours of a date	hour(date_value)
DateTime	minute	returns minutes of a date	minute(date_value)
DateTime	second	returns seconds of a date	second(date_value)
DateTime	now	returns current datetime	now()
DateTime	timevalue	converts a string into a time	timevalue(text_value)
DateTime	time	creates a time using three integers	time(hour, minutes, seconds)
DateTime	dateadd	add an interval to a date	dateadd(interval, number,date)
DateTime	datediff	returns the difference between two dates	datediff(interval,
DateTime	today	return today's date	today()
DateTime	edate	returns a date that's a specified number of months after a start date	edate(date, months_number)
DateTime	eomonth	returns the end of month date	eomonth(date, months_number)
DateTime	weekday	returns the name of the day of the week	weekday(date, returnType)
DateTime	networkdays	returns the work days between two dates	networkdays(startDate, endDate [, holiday1, holiday2, ..., holidayN])
DateTime	weeknum	returns the week of the date	weeknum(date, returnType)
DateTime	workday	returns the first valid workday	workday(startDate, days [, holiday1, holiday2, ..., holidayN])

FINANCIAL			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Financial	npv	net present value	npv(discountRate, value1, value2, ..., valueN)
Financial	fv	future value	fv(interestRate, nPeriod, payment, presentValue, paymentDue)
Financial	pv	present value	pv(interestRate, nPeriods, amount, futureValue, paymentDue)
Financial	pmt	payment for a loan at fixed rate and fixed payments	pmt(interestRate, nPeriods, presentValue, futureValue, paymentDue)
Financial	nper	number of payments with steady freq, amt, rate	nper(interestRate, amount, presentValue, futureValue, paymentDue)
Financial	ppmt	principal portion of payment of a loan	ppmt(interestRate, periodNumber, nPeriods, presentValue, futureValue, paymentDue)
Financial	ipmt	interest portion of payment of a loan	ipmt(interestRate, periodNumber, nPeriods, presentValue, futureValue, paymentDue)
Financial	syd	depreciation of an asset using sum of years' digit method	syd(assetCost, salvageValue, lifespan, periodNumber)
Financial	sln	straight-line depreciation of an asset per period	sln(assetCost, salvageValue, lifespan)
Financial	db	depreciation of an asset using fixed declining balance method	db(assetCost, salvageValue, lifespan, period, months)
Financial	ddb	depreciation of an asset using double declining balance method	ddb(assetCost, salvageValue, lifespan, period, weight)

Financial	intrate	interest rate yielded by a security investment redeemable in the future	intrate(settlementDate, maturityDate, amount, redemptionValue, basis)
Financial	irr	internal rate of return for a series of cashflows	irr(value_reference, estimate)
Financial	rate	per-period interest rate for a series of cashflows	rate(nPeriods, amount, presentValue, paymentDue, futureValue, estimate)
Statistical	median	median of a list of values	median(Value1, value2, ..., valueN)
Statistical	var	estimated variance	var(Value1, value2, ..., valueN)
Statistical	stdev	standard deviation	stdev(Value1, value2, ..., valueN)
Statistical	combin	Returns the number of possible combinations given a set of items and a number of chosen items from that set	combin(number, numberChosen)

TRIGONOMETRY			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Trigonometry	cos	Calculates the trigonometric cosine of a specified angle (measured in radians)	cos(value)
Trigonometry	acos	Returns the angle (measured in radians) having the specified value of the trigonometric cosine function	acos(value)
Trigonometry	cosh	Calculates the hyperbolic cosine of a specified angle measured in radians	cosh(value)
Trigonometry	acosh	Returns the angle (measured in radians) having the specified value of it's hyperbolic cosine function	acosh(value)
Trigonometry	sin	Calculates the trigonometric sine of a specified angle (measured in radians)	sin(value)
Trigonometry	asin	Returns the angle (measured in radians) having the specified value of the trigonometric sine function	asin(value)
Trigonometry	sinh	Calculates the hyperbolic sine of a specified angle measured in radians	sinh(value)
Trigonometry	asinh	Returns the angle (measured in radians) having the specified value of it's hyperbolic sine function	asinh(value)
Trigonometry	tan	Calculates the trigonometric tangent of a specified angle (measured in radians)	tan(value)
Trigonometry	atan	Returns the angle (measured in radians) having the specified value of the trigonometric tangent function	atan(value)
Trigonometry	atan2	Calculates the angle made with the x-axis on a Cartesian coordinate plane by the specified (x, y) coordinates	atan2(x, y)
Trigonometry	tanh	Calculates the hyperbolic tangent of a specified angle measured in radians	tanh(value)
Trigonometry	atanh	Returns the angle (measured in radians) having the specified value of it's hyperbolic tangent function	atanh(value)
Trigonometry	degrees	Converts radians to degrees	degrees(radians)
Trigonometry	radians	Converts degrees to radians	radians(degrees)

COMPLEX MATH			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Complex Math	complex	Returns a complex number	complex(realNum, imagNum, [suffix])
Complex Math	imabs	Returns the absolute value of a complex number	imabs(complexNumber)
Complex Math	imaginary	Returns the imaginary coefficient of a complex number	imaginary(complexNumber)

Complex Math	imargument	Returns the argument theta, an angle expressed in radians.	imargument(complexNumber)
Complex Math	imcos	Returns the cosine of a complex number	imcos(complexNumber)
Complex Math	imdiv	Returns the quotient of two complex numbers	imdiv(dividend, divisor)
Complex Math	imexp	Returns the exponential of a complex number	imexp(complexNumber)
Complex Math	imln	Returns the natural logarithm of a complex number	imln(complexNumber)
Complex Math	imreal	Returns the real coefficient of a complex number	imreal(complexNumber)
Complex Math	imsub	Returns the difference between two complex numbers	imsub(complexNumber1, complexNumber2)
Complex Math	improduct	Returns the product of 1 to n complex numbers	improduct(complexNumber1 [, complexNumber2, ..., complexNumberN])
Complex Math	imsum	Returns the sum of 1 to n complex numbers	imsum(complexNumber1 [, complexNumber2, ..., complexNumberN])
Complex Math	imlog10	Returns the base-10 logarithm of a complex number	imlog10(complexNumber)
Complex Math	imlog2	Returns the base-2 logarithm of a complex number	imlog2(complexNumber)
Complex Math	imsin	Returns the sine of a complex number	imsin(complexNumber)
Complex Math	imsqrt	Returns the square root of a complex number	imsqrt(complexNumber)
Complex Math	impower	Returns the complex number raised to the specified power	impower(complexNumber, power)

MISC			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Misc	rand	Generates a pseudorandom number from zero up to (but not including) one	rand()
Misc	type	return UltraCalc data type	type(value)
Misc	errortype	numeric value corresponding to a specific error	errortype(value)
Misc	na	not applicable	na()
Misc	null	return a null value	null()
Misc	dbnull	return a DBNull value	dbnull()
Misc	char	returns the char of an integer	char(int_value)
Misc	fixed	formats a numeric value	fixed(value, decimal_places, exclude_thousands_separators)
Misc	count	Counts how many cells have numeric or date/time values	count(value1, value2, ..., valueN)
Misc	info	Returns information about the current operating environment	info(type_of_information)
Misc	randbetween	Generates a pseudorandom integer between two specified numbers	randbetween(bottom, top)
Misc	choose	Returns one of the values provided in its arguments based on the number in the first argument	choose(index_num, value1, [value2, ..., valueN])

ATHENA SPECIFIC			
CATEGORY	FUNCTION	DESCRIPTION	SYNTAX
Athena-Specific	aum_func	Returns the static AUM value pulled from a specific data source	aum_func(column) / aum_func(TOTAL)
Athena-Specific	aum_rt_func	Returns the realtime AUM value as calculated from incoming data	aum_rt_func(value_column, categ_column) / aum_rt_func(value_column, TOTAL)
Athena-Specific	bb_func	Requests data to the Bloomberg Terminal API	bb_func(symbol, field)
Athena-Specific	buckets	Returns the bucket the input value belongs to	buckets(column, limit1, label1, limit2, label2...)
Athena-Specific	current_cross_rate	Returns the latest FX rate with respect to base currency	current_cross_rate(currency)
Athena-Specific	indexof	Returns the first instance of txt1 in txt2 starting from position pos, or -1 if not included	indexof(txt1, txt2, pos)

Athena-Specific	ntile	Returns the dynamic n-tile the input value belongs to	ntile(column, #_of_tiles)
Athena-Specific	old_cross_rate	Returns the last close FX rate with respect to base currency	old_cross_rate(currency)
Athena-Specific	rank	Assigns an incremental integer value to cells within a range, depending on their relative value	rank(column, key, ascending, tied)
Athena-Specific	ref	Access another column via a column's content	ref(column)
Athena-Specific	rtd	Requests data to a generic RTD server data source	rtd(rtd_server, rtd_location, symbol, field)
Athena-Specific	substitute	Substitutes to_txt to from_txt in source_text	substitute(source_text, from_txt, to_txt)
Athena-Specific	url	Returns a clickable text to a browsable document	url(caption, link)
Athena-Specific	yh_func	Requests data to the Yahoo Web API	yh_func(symbol, field)
Athena-Specific	FormatString	Format the output following string.Format definition. See: ref: https://msdn.microsoft.com/en-us/library/txafckwd.aspx	FormatString("–format string–", arg1, optArg2, ..., optArg10)