VistaCalc Help

Menu Commands

Miscellaneous

Format Menu

The Format menu includes commands to format the spreadsheet.

Font	The Font dialog box is displayed, allowing you to specify the font, font style, and color of data in the selected range.
Default Font	Set the default font used to display data in worksheets.
Border	The Border dialog box is displayed, allowing you to specify the placement of borders in the selected range. You can also specify the border line style and color. The check box in the Border dialog box are three-state check boxes, allowing "as is" selections to be made.
Pattern	The Pattern dialog box is displayed, allowing you to specify the fill pattern and foreground and background colors for the selected range.
General	Formats data in the selected range using the General format.
Currency (0)	Formats data in the selected range using the Currency format and a decimal precision of 0.
Currency (2)	Formats data in the selected range using the Currency format and a decimal precision of 2.
Fixed	Formats data in the selected range using the Fixed format.
Percent	Formats data in the selected range using the Percent format. Numbers with this format are displayed as percentages with a trailing percent sign (%).
Fraction	Formats data in the selected range using the Fraction format. Numbers with this format are displayed as fractions.
Scientific	Formats data in the selected range using the Scientific format.
M/D/YY	Formats data in the selected range using the M/D/YY format. Numbers with this format are displayed as dates.

H:MM AM/PM   Formats data in the selected range using the H:MM AM/PM time format.   Numbers with this format are displayed as times.

Custom Number   The Custom Number dialog box is displayed, allowing you to define custom number formats for data in the selected range.

Color Palette   The Color Palette dialog box is displayed, allowing you to edit colors in the color palette, specify a default color, and use the default color palette.

Formula Menu

The Formula menu includes commands that enables you to copy right, copy down, and define names.

Copy Right   In the leftmost cell, data of the selected range is copied right to fill the range.

Copy Down   In the top cell, data of the selected range is copied down to fill the range.

Recalc   The worksheet is recalculated.

Calculation   The Calculation dialog box is displayed, allowing you to enable and disable automatic recalculation and to specify iteration values for calculating circular references.

Define Name   The Define Name dialog box is displayed, allowing you to add and delete names.

Options Menu

The Options menu includes commands to fix rows, fix columns, alignment, column width, row height, cell protection, and enable protection.

Fix Rows	Rows, if selected, are fixed at the left edge of the worksheet. Fixed rows do not scroll out of view.
Fix Columns	Columns, if selected are fixed at the top edge of the worksheet. Fixed columns do not scroll out of view.
Alignment	The Alignment dialog box is displayed, allowing you to specify the vertical and horizontal alignment of data in the selected range. You can also enable and disable word wrapping.
Cell Protection	The Cell Protection dialog box is displayed, allowing you to specify whether the cells in the selected range are locked and hidden.
Enable Protection	Enables protection for protected cells in the worksheet.
Column Width	The Column Width dialog box is displayed, allowing you to set the width of the selected columns, specify default column widths, and specify automatic column width. Also you can specify whether the selected columns are hidden or shown.
Row Height	The Row Height dialog box is displayed, allowing you to set the height of the selected rows, specify default row heights, and specify automatic row height. Also you can specify whether the selected rows are hidden or shown.
Allow	The Allow options dialog box is displayed.

Object Menu

The Object menu enables you create and edit objects.

To highlight an object(s), hold down the control key (Ctrl) and click on the object(s).

Object Bar

The Object Bar contains buttons that allow you to create and edit objects.

Pattern

For the selected objects, Pattern sets the fill pattern and colors.

Line Style

For the selected line object or the border surrounding the selected arcs, ovals, polygons, and rectangles, Line Style sets the line style.

Options

The Object Option dialog box is displayed allowing you to set the input/output value cell for selected check boxes and list boxes, the text displayed by check boxes, and the list of items contained by list boxes.   Separate items in list boxes with a semicolon.

Bring To Front

In the worksheet, Bring To Front, places the selected objects in front of other objects.

Send To Back

In the worksheet, Send To Back, places the selected objects behind other objects.

Object Bar

To select an object(s), hold down CTRL key and click on the object(s).

The following table describes the buttons on the object bar.

Name	Description
Check box tool	Draws check box objects.
List tool	Draws a list box.
Oval tool	Draws circles and ovals.
Arc tool	Draws arcs.
Freehand tool	Draws lines.
Line tool	Draws straight lines.
Rectangle tool	Draws rectangles and squares.
Freehand detail tool	Toggles editing points on objects drawn using the freehand tool.

View Menu

The View menu enables you to show and hide formulas, gridlines, row heading, column heading, zeros, and the status bar.

Formulas   Show or hide formulas in place of cell values.

Gridlines   Show or hide Gridlines.

Row Heading   Show or hide row heading.

Column Heading   Show or hide column heading.

Zeros   Show or hide zero value cells.

Status Bar   Show or hide the status bar.

Function Groups

To enter a function place an equals sign (=) at the beginning of the cell, then enter the formula.

Summing, Counting, and Other Statistical Functions
Indexing, Matching, and Looking Up Values
Manipulating Text
Date Functions
Time Functions
Business and Financial Functions
Math Functions
Misc. Functions
Test Functions

LEFT(text [, num_chars])

Returns text consisting of the leftmost characters from the specified text string.

Text is any text string.

Num_chars is the number of characters to return.   The value must be greater than or equal to zero.   The entire string is returned if num_chars is greater than the number of characters in text.   If num_chars is omitted it assumes a value of 1.

Examples

LEFT("six seven eight") returns "six"
LEFT("9th Inning") returns "9th"

See Also

MID
RIGHT

MID(text, start_num, num_chars)

Returns the number of characters from a text string, beginning with the character at start_num.

The position of the first character to return from text is **start_num**.

The number of characters to return is **num_chars**.

If start_num is 1, the first character in text is returned.   If start_num is greater than the number of characters in text, an empty string ("") is returned.   If start_num is less than 1, #VALUE! is returned.   If num_chars is negative, #VALUE! is returned.

Examples

MID("Shopping Expenses", 8, 8) returns "Expenses"
MID("Hello there" ,1,5) returns "Hello"

See Also

CODE
FIND
LEFT
RIGHT
SEARCH

Serial Number

Serial numbers range from 1 to 65,380, corresponding to the dates January 1, 1900, through December 31, 2078.

Numbers to the right of the decimal point in the serial number represent the time; numbers to the left represent the date. For example, the serial number 367.5 represents the date-time combination 12:00 P.M., January 1, 1901.

RIGHT(text, num_chars)

Returns text consisting of the rightmost characters from the specified text string.

Num_chars is the number of characters to return.   The value must be greater than or equal to zero.   If num_chars is greater than the length of text, the entire string is returned.   If num_chars is omitted it then assumes a value of 1.

Examples

RIGHT("9th Inning") returns "g"
RIGHT("9th Inning", 6) returns "Inning"

See Also

LEFT
MID

&

Combine two strings.

Examples

37&b1 -> if cell b1 is 5, function returns 375
"Vista"& b1 -> if cell b1 is "Calc", function returns VistaCalc

LEN(text)

Returns the number of characters in the text string.   Letters, numbers, and spaces are counted as characters.

Examples

LEN("1-3") returns 3
LEN("") returns 0

See Also

EXACT
SEARCH

LOWER(text)

Returns the characters in the text string to lowercase characters.   Numeric characters in the text are not changed.

Examples

LOWER("Fireplace") returns "fireplace"
LOWER("NEW YORK") returns "new york"

See Also

PROPER
UPPER

UPPER(text)

Returns the characters in the text string to uppercase characters.   Numeric characters in the text are not changed.

Examples

UPPER("Fireplace") returns FIREPLACE
UPPER("NEW YORK") returns NEW YORK

See Also

LOWER
PROPER

SUBSTITUTE(text, old_text, new_text [, instance_num])

Substitutes new_text for old_text in a text string.   Instance_num specifies the occurrence of old_text to replace.   If this argument is omitted, every instance of old_text is replaced with new_text.

Examples

SUBSTITUTE("cake", "c", "b") returns "bake"
SUBSTITUTE("Sixth Inning Results", "Sixth", "Ninth") returns "Ninth Inning Results"

See Also

REPLACE
TRIM

FIND(search_text, text, [start_at_num])

Searches for a text string within another text string and returns the character position at which the search string first occurs.

The text to find is **search_text**.   Search_text ("") FIND matches the first character in text.

The text to be searched is **text**.

Start_at_num is the character position in text where the search begins (which is character number 1, if omitted then the default starting position is character number 1).

FIND cannot use wildcard characters in the search_text.

Examples

FIND("o", "New York") returns 6
FIND("N", "New York") returns 1

See Also

EXACT
LEN
MID
SEARCH

SEARCH(find_text, text, [, start_position])

Searches for find_text within text.   Starts search at character specified by start_position. The search is not case-sensitive.

If text does not contain the search string, #VALUE! is returned.   If the number you specify is less than 0 or greater than the number of characters in text, #VALUE! is returned.   If this argument is omitted then the starting position is assumed to be 1.

Examples

SEARCH("o","loud voices") returns 2
SEARCH("o","loud voices",2) returns 2

See Also

FIND
MID
REPLACE
SUBSTITUTE

PROPER(text)

Returns the text string in proper -case format.

The first character in a word is capitalized.   If a character follows a number, punctuation mark, or space, it is capitalized.   All other characters are in lowercase format.   Characters that are not letters are unchanged.

Examples

PROPER("9th Inning") returns "9Th Inning"
PROPER("NEW YORK") returns "New York"

See Also

LOWER
UPPER

TRIM(text)

Removes all spaces from text except for a single space between words.

Examples

TRIM("   Look   at   me now!   ") returns :"Look at me now!"
TRIM("   Stop, drop, and roll   ") returns "Stop, drop, and roll"

See Also

CLEAN
MID
REPLACE
SUBSTITUTE

CLEAN(text)

Removes all non printable characters from the text.

Example

CLEAN("Payments " & CHAR(8) & "Due") returns Payments Due because the character returned by CHAR(8) is non printable.

See Also

CHAR
TRIM

REPLACE(old_text, start_num, num_chars, new_text)

Starting at start_num, removes num_chars characters from old text and replaces them with new_text.

Examples

REPLACE("1981", 3,2,"82") returns "1982"
REPLACE("fbcde", 1,1, "a") returns "abcde"

See Also

MID
SEARCH
TRIM

CHAR(number)

Returns a character that corresponds to the ANSI code.

The character and associated numeric code are defined by Windows in the ANSI character set.

Examples

CHAR(70) returns F
CHAR(65) returns A

See Also

CODE

CODE(text)

Returns a numeric code that represents the first character in text.

Text is any string.

The numeric code and associated string are defined in your computer's character set.   The character set used by Windows is the ANSI character set.

Examples

CODE("A") returns 65
CODE("b") returns 98

See Also

CHAR

DOLLAR(number, [decimals])

Returns the specified number as text, using currency format.

Decimals are the number of digits to the right of the decimal point.   If negative, then number is rounded to the left of the decimal point.   Omitting this argument, number is assumed to be 2.

Examples

DOLLAR(1674.321) returns "$1674.32"
DOLLAR(32.987) returns "$32.99"

See Also

FIXED
TEXT
VALUE

EXACT(text1, text2)

Compares text1 and text 2 for identical, case-sensitive matches.   True is returned if text1 and text2 are identical; False is returned if they are not.

Text1 is any text.

Text2 is any text.

Examples

EXACT("baby", "baby") returns True
EXACT("Baby", "baby") returns False

See Also

LEN
SEARCH

REPT(text, number)

Repeats text a specified number of times.

Text is any string.

Number is the number of times that you want text to repeat.   Number must be greater than or equal to zero.   If number is 0, empty text ("") is returned.

The result of REPT cannot exceed 255 characters.

Examples

REPT("star",4) returns "star star star star")
REPT("baby",3) returns "baby baby baby")

See Also

MANIPULATING TEXT FUNCTIONS

T(value)

Returns the value if it is text.   Any value that is not text, T returns empty text ("").

Examples

T(A1) returns "" (empty text) if A1 contains a number
T("School") returns "School"

See Also

CELL
N
VALUE

FIXED(number[,decimals][,no_commas])

Rounds a number to decimals.   Formats the number in decimal format, and returns the result as text.

Number is any number.

Decimals are the number of digits that appear to the right of the decimal place.   If decimals are negative, then number is rounded to the left of the decimal point.   You can specify a decimal as great as 127 digits.   If decimals are omitted, it is assumed to be 2.

No_commas determines if commas are used in the result.   Use 1 to exclude commas in the result.   If the argument is omitted, then commas are included (e.g., 1,000.00).

Examples

FIXED(4000.5, 3) returns "4,000.500"
FIXED(55.222) returns "55.22"

See Also

DOLLAR
ROUND
TEXT
VALUE

TEXT(number, format)

Returns the given number as text, using the specified formatting.

Number is any value, a formula that evaluates to a number that contains a value.

Format is a string representing a number format.   The string can be any valid format string. Format must be surrounded by a set of double quotation marks.   Format cannot contain an asterisk (*).

Examples

TEXT(6.145, "$0.00") returns "$6.15"
TEXT("8/23/1975","mmmm d, yyyy") returns "August 23, 1975"

See Also

DOLLAR
FIXED
T
VALUE

VLOOKUP(search_item, search_range, column_index)

Searches the first column of a table for a value, and returns the contents of a cell in that table that corresponds to the location of the search value.

Search_item is a text string, value, or reference to a cell containing a value that is matched against data in the top row of search_range.

Search_range is the reference of the range to be searched.   Cells in the first column of search_range can contain text, numbers, or logical values.   Contents of the first column must be in ascending order.   Text searches are not case-sensitive.

Column_index is the column in the search range from which returned is the matching value.
Column_index can be a number from 1 to the number of rows in the search range.

#VALUE! is returned if column_index is less than 1.

#REF! is returned when column_index is greater than the number of rows in the table.

VLOOKUP compares the information in the first column of search_range to the supplied search_item.   Column_index is returned when a match is found and information is located in the same row.

The largest value that is less than search_item is used when search_item cannot be found in the first column of search_range.

#REF! is returned when search_item is less than the smallest value in the first column of the search range.

See Also

HLOOKUP
INDEX
LOOKUP
MATCH

HLOOKUP(search_item, search_range, row_index)

Searches the top row of a table for a value.   Then returns the contents of a cell in that table that corresponds to the location of the search value.

Search_item is a text string, value, or reference to a cell containing a value that is matched against data in the top row of search_range.

Search_range is the reference of the range to be searched.   Cells in the first row of search_range can contain text, numbers, or logical values.   Contents of the first row must be in ascending order.   Text searches are not case-sensitive.

Row_index is the row in search_range from which returned is the matching value.

Row_index can be a number from 1 to the number of rows in the search range.

#VALUE! is returned if row_index is less than 1.

#REF! is returned when row_index is greater than the number of rows in the table.

HLOOKUP compares the information in the top row of search_range to the supplied search_item.   Row_index is returned when a match is found and information is located in the same column.

The largest value that is less than search_item is used when search_item cannot be found in the top row of search_range.

#REF! is returned when search_item is less than the smallest value in the first row of the search range.

See Also

INDEX
LOOKUP
MATCH
VLOOKUP

MATCH(lookup_value, lookup_range, comparison)

A specified value is compared against values in a range.   Position of the matching value in the search range is returned.

Lookup_value is the value against which to compare.   It can be text, a number, or logical value or a reference to a cell that contains one of those values.

Lookup_range is the range to search and contains just one row or one column.   The range can contain text, numbers. or logical values.

Comparison is a number that represents the type of comparison to be made between lookup_value and the values in lookup_range.   Omitting this argument, comparison method 1 is to be assumed.

When comparison is 1, then the largest value that is less than or equal to lookup_value is matched.   The values in lookup_range must be in ascending order.

When comparison is 0, then the first value that is equal to lookup_value is matched.   The values in lookup_range can be in any order.

When comparison is -1, then the smallest value that is greater than or equal to lookup_value is matched.   The values in lookup_range must be in descending order.

When using comparison method 0 and lookup_value is text, lookup_value can contain wildcard characters, asterisk (*).   The wildcard characters, asterisk (*) match any sequence of characters.   The question mark (?) matches any single character.

#N/A is returned when no match is found for lookup_value.

See Also

HLOOKUP
INDEX
LOOKUP
VLOOKUP

INDEX(reference [,row] [,column] [,range_number])

Returns the contents of a cell from a specified range.

Reference is a reference to one or more ranges.

If reference supplies more than one range, you must separate each reference with a comma and enclose reference in parentheses.

If each range in reference contains just one row or column, you can omit the row or column argument.

Row is the row number in reference from which to return data.

Column is the column number in reference from which to return data.

Range_number specifies the range from which data is returned if reference is containing more than one range.

#REF! is returned if row, column, and range_number do not point to a cell within reference. INDEX returns the range in reference specified by range_number if row and column are omitted.

See Also

CHOOSE
HLOOKUP
LOOKUP
MATCH
VLOOKUP

LOOKUP(lookup_value, lookup_range, result_range)

Searches for a value in one range and returns the contents of the corresponding position in a second range.

The value for which to search in the first range is **lookup_value**.

The first range to search, and contains only one row or one column is **lookup_range**.   The range can contain text, numbers, or logical values.

The expressions in the range must be placed in ascending order to search lookup_range correctly.   The search is not case-sensitive.

A range of one row or one column that is the same size as lookup_range is **result_range**.

If lookup_value does not have an exact match in lookup_range, the largest value that is less than or equal to lookup_value is found, and the corresponding position in result_range is returned.

#N/A is returned when lookup_value is smaller than the data in lookup_range.

See Also

HLOOKUP
INDEX
VLOOKUP

CHOOSE(index, item_list)

Based on the index number supplied, returns a value from a list of numbers.

Index is a number that refers to an item in item_list**.**
Index can be a cell reference.   Index can be a formula that returns any value from 1 to 29.

#VALUE! is returned if index is less than 1 or greater than the number of items in item_list.

Index is truncated to an integer if index is a fractional number.

Item_list is a list of formulas, numbers, or text separated by commas.   This argument can also be a range reference.   29 items can be specified in the list.

Examples

CHOOSE(-2,4,6,7) returns #VALUE!
CHOOSE(2, "red", "white", "blue") returns "white"

See Also

INDEX

TIMEVALUE(text)

Returns a serial number for the supplied text representation of time.

Serial_Number is the time as a serial number.   The decimal portion of the number represents time as a fraction of the day.

Text is a time in text format.

Examples

TIMEVALUE("2:24 AM") returns 0.1
TIMEVALUE("1:43:43 am") returns .07

See Also

HOUR
MINUTE
NOW
SECOND
TIME

OFFSET(reference, rows, columns [, height] [, width])

Returns the contents of a range that is offset from a starting point in the spreadsheet.

Reference is a reference to a cell from which the offset reference is based.   #VALUE! is returned when you specify a range reference.

The number of rows from reference that represents the upper-left cell of the offset range is **rows**.

A positive number represents rows below the starting cell; a negative number represents rows above the starting cell.   #REF! is returned if rows places the upper-left cell of the offset range outside the spreadsheet boundary.

The number of columns from reference that represents the upper-left cell of the offset range is **columns**.   A positive number represents columns right of the starting cell; a negative number represents columns left of the starting cell.   #REF! is returned if columns places the upper-left cell of the offset range outside the spreadsheet boundary.

A positive number representing the number of rows to include in the offset range is **height**. A single row is assumed if this argument is omitted.

A positive number representing the number of columns to include in the offset range is **width**.   A single column is assumed if this argument is omitted.

OFFSET does not change the current selection in the worksheet, for it returns a reference. OFFSET can be used in any function that requires or uses a range reference or a cell as an argument.

Examples

OFFSET(C3,2,3,1,1) returns F5
OFFSET(B1,3,2,1,1) returns D4

NOW()

Returns the current date and time as a serial number.

In a serial number, date is represented by numbers to the left of the decimal point; time is represented by numbers to the right of the decimal point.   When a recalculation of the worksheet occurs, then the result of this function changes.

See Also

DATE
DAY
HOUR
MINUTE
MONTH
SECOND
TODAY
WEEKDAY
YEAR

HOUR(serial_number)

Returns the hour corresponding to serial_number.

Serial_number is the time as a serial number.   The decimal portion of the number represents time as a   fraction of the day.

An integer ranging from 0 (12:00 AM) to 23 (11:00 PM) is the result.

Examples

HOUR(34259.4) returns 9
HOUR(0.7) returns 16

See Also

DAY
MINUTE
MONTH
NOW
SECOND
WEEKDAY
YEAR

MINUTE(serial_number)

Returns the minute corresponding to serial_number.

Serial_number is the time as a serial number.   The decimal portion of the number represents time as a   fraction of the day.

An integer ranging from 0 to 59 is the result.

Examples

MINUTE(34506.4) returns 36
MINUTE(.01) returns 14

See Also

DAY
HOUR
MONTH
NOW
SECOND
WEEKDAY
YEAR

SECOND(serial_number)

Returns the second corresponding to serial_number.

Serial_number is the time as a serial number.   The decimal portion of the number represents time as a fraction of the day.

Examples

SECOND(.259) returns 58
SECOND(0.01) returns 24

See Also

DAY
HOUR
MINUTE
MONTH
NOW
SECOND
WEEKDAY
YEAR

TIME(hour, minute, second)

Returns a serial number for the supplied time specified by hour, minute, and second.

A number from 0 to 23 is **hour**.

A number from 0 to 59 is **minute**.

A number from 0 to 59 is **second**.

Examples

TIME(12, 26, 24) returns .52
TIME(12, 0, 0) returns 0.5

See Also

HOUR
MINUTE
NOW
SECOND
TIMEVALUE

PMT(interest, nper, pv [,fv] [, type])

Based on regular payments and a fixed periodic interest rate, PMT returns the periodic payment of an annuity.

The fixed periodic interest rate is **interest.**

The number of period in the annuity is **nper**.

The present value, or the amount the annuity is currently worth is **pv**.

The future value, or the amount the annuity will be worth is **fv**.   Omitting this argument, 0 is assumed.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

PMT does not include taxes or other fees, it only returns the principal and interest payment.

The units used for interest must match those used for nper.   For example, if the annuity has an 8% annual interest rate over a period of 5 years, specify 8%/12 for interest and 5*12 for nper.

Cash paid out, is shown as a negative number.   Cash received, is shown as a positive number.

Examples

PMT(8%/12,10,0,10000,1) returns -$963.94
PMT(8%/12, 48, 18000, 0, 1) returns -436.52

See Also

IPMT
FV
NPER
PPMT
PV
RATE

PPMT(interest, per, nper, pv, [fv], [type])

Returns the principle paid on an annuity for a given period.

The fixed periodic interest rate is **interest**.

The period for which to return the principle is **per**.

The number of period in the annuity is **nper**.

The present value, or the amount the annuity is currently worth is **pv**.

The future value, or the amount the annuity will be worth is **fv**.   Omitting this argument, 0 is assumed.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Units used for interest must match those used for nper.   For example, if the annuity has an 8% annual interest rate over a period of 5 years, specify 8%/12 for interest and 5*12 for nper.

Examples

PPMT(8%/12, 2, 48, 18000) returns -321.56
PPMT(8%/12, 2, 48, 18000, 0, 1) returns -319.43

See Also

FV
IPMT
NPER
PMT
PV
RATE

PV(interest, nper, pmt [,fv] [,type])

Returns the present value of an annuity, considering a series of constant payments made over a regular payment period.

The fixed periodic interest rate is **interest**.

The number of payment periods in the investment is **nper**.

The fixed payment made each period is **pmt**.

The future value, or the amount the annuity will be worth is **fv**.   Omitting this argument, a future value of 0 is assumed.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Units used for interest must match those used for nper.   For example, if the annuity has an 8% annual interest rate over a period of 5 years, specify 8%/12 for interest and 5*12 for nper.

Cash paid out, is shown as a negative number.   Cash received, is shown as a positive number.

Examples

PV(8%/12, 48, 439.43) returns -17999.89
PV(8%/12, 48, -439.43) returns 17999.89

See Also

FV
IPMT
NPER
PMT
PPMT
RATE

RATE(nper, pmt, pv [,fv] [,type] [, guess])

Given a series of constant cash payments made over a regular payment period, RATE returns the interest rate per period of an annuity.

The number of period in the annuity is **nper**.

The future value, or the amount the annuity will be worth is **fv**.   Omitting this argument, a future value of 0 is assumed.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Your estimate of the interest rate is **guess**.   If no argument is supplied, a value of .1 (10%) is assumed.

RATE is calculated iteratively, cycling through the calculation until the result is accurate to .00001 percent.   If the result cannot be found after 20 iterations, #NUM! is returned. Supply a different value for guess, when this occurs.

Example

RATE(48, -439.43, 18000) returns .0067 (rounded to 4 decimals), which is the monthly interest rate.   The annual interest rate (.0067 multiplied by 12) is 8%

See Also

FV
IPMT
NPER
PMT
PPMT
PV

FV(interest, nper, payment [,pv] [,type])

Returns the future value of an annuity based on regular payments and a fixed interest rate.

The fixed interest rate is **interest**.

The number of payments in an annuity is **nper**.

The fixed payment made each period is **payment**.

The present value, or the lump sum amount, the annuity is currently worth is **pv**.   Omitting this argument, a present value of 0 is assumed.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Units used for interest must match those used for nper.   For example, if the annuity has an 8% annual interest rate over a period of 5 years, specify 8%/12 for interest and 5*12 for nper.

Cash paid out, is shown as a negative number.   Cash received, is shown as a positive number.

Examples

FV(5%, 8, -500) returns 4,774.55
FV(1%, 12, -1000) returns 12682.50

See Also

IPMT
NPER
PMT
PPMT
PV
RATE

IPMT(interest, per, nper, pv, [fv], [type])

Returns the interest payment of an annuity for a given period, based on regular payment and a fixed periodic interest rate.

The fixed periodic interest rate is **interest**.

The period for which to return the interest payment is **per**.   This number must be between 1 and nper.

The number of payments is **nper**.

The present value, or the lump sum amount the annuity is currently worth is **pv**.

The future value, or the value after all payment are made is **fv**.   Omitting this argument, the future value is assumed to be 0.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Units used for interest must match those used for nper.   For example, if the annuity has an 8% annual interest rate over a period of 5 years, specify 8%/12 for interest and 5*12 for nper.

Cash paid out, is shown as a negative number.   Cash received, is shown as a positive number.

Examples

IPMT(8%/12, 2, 48, 18000) returns -117.87
IPMT(8%/12, 2, 48, 18000, 0, 1) returns -117.09

See Also

FV
PMT
PPMT
RATE

NPER(interest, pmt pf [,fv] [,type])

Returns the number of period of an investment based on regular periodic payments and a fixed interest rate.

The fixed interest rate is **interest**.

The fixed payment made each period is **pmt**.   Pmt includes the principle and interest, not taxes or other fees.

The present value, the lump-sum amount that a series of future payments is currently worth is **pf**.

The future value, the balance to attain after the final payment is **fv**.   This argument omitted assumes a future balance of 0.

Type indicates when payments are due.   0 is used if payments are due at the end of the period or 1 if payments are due at the beginning of the period.   Omitting this argument, 0 is assumed.

Examples

NPER(12%/12, -350, -300, 16000, 1) returns 36.67
NPER(1%, -350, -300, 16000) returns 36.98

See Also

FV
IPMT
PMT
PPMT
PV
RATE

NPV(discount_rate, value_list)

Returns the net present value of an investment based on a series of periodic payments and a discount rate.

The rate of discount for one period is **discount_rate**.

A list of as many as 29 arguments or a reference to a range that contains values that represent payments and income is **value_list**.

NPV, during calculation, uses the order in which the values appear to determine the order of cash flow.

Included in the calculation are numbers, empty cells, and text representations of numbers.

Only numeric data in the range is included in the calculation, if value_list is a range reference.   Other types of data in the range are ignored.

The time span NPV uses for calculation begins one period before the first cash flow date and ends when the last cash flow payment is made.   This function is based on future cash flows. When your first cash flow occurs at the beginning of the first period, the first value must be added to the NPV result, not supplied as a value in value_list.

Example

NPV(8%, -1200, 3000, 3000, 3000, 7000) returns 8115.57

See Also

FV
IRR
PV

IRR(cash_flow [,guess])

Returns internal rate of return for a series of periodic cash flows.

A reference to a range that contains values for which to calculate the internal rate of return is **cash_flow**.   The values must contain at least one positive and one negative value.

IRR use the order in which the values appear to determine the order of the cash flow, during calculation.

Empty cells in the rage are ignored.   Text and logical values also are ignored.

The estimate of the internal rate of return is **guess**.   If no argument is supplied, a rate of return of 10 percent is assumed.

The interest rate received for an investment consisting of payments and investments is the internal rate of return.

IRR is calculated iteratively, cycling through the calculation until the result is accurate to .00001 percent.   #NUM! is returned if the result cannot be found after 20 iterations. Supply a different value for guess, when this occurs.

See Also

MIRR
NPV
RATE

MIRR(cash_flows, finance_rate, reinvest_rate)

Returns the modified internal rate of return for a series of periodic cash flows.

A reference to a range that contains values for which to calculate the modified internal rate of returns is **cash_flow**.   The values must contain at least one negative and one positive value.

MIRR uses the order in which the values appear to determine the order of cash flow, during calculation

Positive values represent cash receive; negative values represent cash paid.

Empty cells in the range are ignored.   Text and logical values are also ignored.

The interest rate paid on money used in the cash flow is **finance_rate**.

The interest rate received on money reinvested from the cash flow is **reinvest_rate**.

The modified internal rate of return considers the cost of the investment and the interest received on the reinvestment of cash.

See Also

IRR
NPV
RATE

DB(cost, salvage, life, period [,months])

Using the fixed-declining balance method, DB returns the real depreciation of an asset for a specific period of time.

The initial cost of the asset is **cost**.

The salvage value of the asset is **salvage**.

The number of periods in the useful life of the asset is **life**.

The period for which to calculate the depreciation is **period**.   The time units used to determine period and life must match.

The number of months in the first year of the item's life is **months**.   Omitting this argument assumes there are 12 months in the first year.

Example

DB(1000000, 100000, 6, 1, 7) returns $186,083
DB(10000, 1000, 7, 3) returns 1451.52

See Also

DDB
SLN
SYD
VDB

DDB(cost, salvage, life, period [,factor])

Returns the depreciation of an asset for a specific period of time using the double declining balance method or a declining balance factor that you supply.

The initial cost of the asset is **cost**.

The salvage value of the asset is **salvage**.

The number of periods in the useful life of the asset is **life**.

The period for which to calculate the depreciation is **period**.   The time units used to determine period and life must match.

The rate at which the balance declines is **factor**.   Omitting this argument assumes a default factor of 2, the double-declining balance factor.

All arguments for this function must be positive.

Example

DDB(10000, 1000, 7, 3) returns 1457.73

See Also

DB
SLN
SYD
VDB

SLN(cost, salvage, life)

Returns the depreciation of an asset for a specific period of time using the straight-line balance method.

The initial cost of the asset is **cost**.

The salvage value of the asset is **salvage**.

The number of periods in the useful life of the asset is **life**.

Example

SLN(10000, 1000, 7) returns 1285.71

See Also

DDB
SYD
VDB

SYD(cost, salvage, life, per)

Returns the depreciation of an asset for a specified period using the sum-of-years method. This depreciation method uses an accelerated rate, where the greatest depreciation occurs early in the useful life of the asset.

The initial cost of the asset is **cost**.

The salvage value of the asset is **salvage**.

The number of periods in the useful life of the asset is **life**.

The period for which to calculate the depreciation is **period**.   The time units used to determine period and life must match.

Example

SYD(10000, 1000, 7, 3) returns 1607.14

See Also

DDB
SLN
VDB

VDB(cost, salvage, life, start_period, end_period [,factor] [,method])

Returns the depreciation of assets for a specified period using a variable method of depreciation.

The initial cost of the asset is **cost**.

The salvage value of the asset is **salvage**.

The number of periods in the useful life of the asset is **life**.

The beginning period for which to calculate the depreciation is **start_period**.   The time units used to determine start_period and life must match.

The ending period for which to calculate the depreciation is **end_period**.   The time units used to determine end_period and life must match.

The rate at which the balance declines is **factor**.   Omitting this argument assumes a default of 2, which is the double-declining balance factor.

A logical value that determines if you want to switch to straight-line depreciation when depreciation is greater than the declining balance calculation.   Use True to maintain declining balance calculation; use False or omit the argument to switch to straight-line depreciation calculation.

Example

VDB(10000, 1000, 7, 3, 4) returns 1041.23

See Also

DDB
SLN
SYD

SUM(number_list)

Returns the sum of the supplied numbers.

A list of as many as 30 numbers, separated by commas is a **number_list**.

The list can contain logical values, numbers, text representation of numbers, or a reference to a range containing those values.

Text or error values that cannot be translated into numbers return errors.

If a range reference is included in the text, logical expressions, list, and empty cells in the range are ignored.

Examples

SUM(A1:A3) returns 15
SUM(1000, 3000, 5000) returns 9000

See Also

AVERAGE
COUNT
COUNTA
PRODUCT
SUMSQ

COUNT(value_list)

Returns the number of values in the supplied list.

A list of values is a **value_list**.   30 values may be contained in the list.

COUNT only numerates numbers or numerical values.   If you supply a range, only numbers and numerical values in the range are counted.   Empty cells, logical values, error values, and text in the range are ignored.

Example

COUNT(5, 6, "02") returns 2

See Also

AVERAGE
COUNTA
SUM

COUNTA(expression_list)

Returns the number of non-blank values in the supplied list.

A list of expressions is an **expression_list**.   30 expressions can be included in the list.

COUNTA returns the number of cells that contain data in a range.   Null values (" ") are counted, but references to empty cells are ignored.

Example

COUNTA(32, 45, "Earnings", "") returns 4

See Also

AVERAGE
COUNT
PRODUCT
SUM

AVERAGE(number_list)

Returns the average of the supplied numbers.   The result of AVERAGE is also known as the arithmetic mean.

A list of numbers separated by commas is a **number_list**.   30 numbers can be included in the list, and the list can contain numbers or a reference to a range that contains numbers. Logical expressions, text, or empty cells in a referenced range are ignored.   All numeric values (including 0) are used.

Example

AVERAGE(5, 6, 8, 14) returns 8.25

See Also

MIN
MAX

MIN(number_list)

Returns the smallest value in the specified list of numbers.

A list of as many as 30 numbers, separated by commas is a **number_list**.

The list can contain logical values, numbers, text representation of numbers, or a reference to a range containing those values.

Text or error values that cannot be translated into numbers return errors.

If a range reference is included in the text, logical expressions, list, and empty cells in the range are ignored.

0 is returned if there are no numbers in the list.

Example

MIN(50, 100, 150, 500, 200) returns 50

See Also

AVERAGE
MAX

MAX(number_list)

Returns the largest value in the specified lust of numbers.

A list of as many as 30 numbers, separated by commas is a **number_list**.

The list can contain logical values, text representation of numbers, numbers, or a reference to a range containing those values.

Text or error values that cannot be translated into numbers return errors.

If a range reference is included in the text, logical expressions, list, and empty cells in the range are ignored.

0 is returned if there are no numbers in the list.

Example

MAX(50, 100, 150, 500, 200) returns 500

See Also

AVERAGE
MIN

FACT(number)

Returns the factorial of a specified number.

Any non-negative integer is **number**.   If you supply a real number, FACT truncates the number to an inter before calculation.

Examples

FACT(2.5) returns 2
FACT(1) returns 1

See Also

PRODUCT

STDEV(number_list)

Returns the standard deviation of a population based on a sample of supplied values.   The standard deviation of a population represents an average of deviations from the population mean within a list of values.

A list of as many as 30 numbers, separated by commas is a **number_list**.   The list can contain numbers or a reference to a range that contains numbers.

Example

STDEV(4.0, 3.0, 3.0, 3.5, 2.5, 4.0, 3.5) returns .56

See Also

STDEVP
VAR
VARP

STDEVP(number_list)

Returns the standard deviation of a population based on an entire population of values.   The standard deviation of a population represents an average of deviations from the population mean within a list of values.

A list of as many as 30 numbers, separated by commas is a **number_list**.   The list can contain numbers or a reference to a range that contains numbers.

Example

STDEVP(4.0, 3.0, 3.0, 3.5, 2.5, 4.0, 3.5) returns .52

See Also

STDEV
VAR
VARP

SUMSQ(number_list)

Squares each of the supplied numbers and returns the sum of the squares.

A list of as many as 30 numbers, separated by commas is a **number_list**.

The list can contain logical values, text representation of numbers, numbers, or a reference to a range containing those values.

Text or error values that cannot be translated into numbers return errors.

If a range reference is included in the text, logical expressions, list, and empty cells in the range are ignored.

Example

SUMSQ(9, 10, 11) returns 302

See Also

SUM

VAR(number_list)

Returns the variance of a population based on a sample of values.

A list of as many as 30 numbers, separated by commas is a number_list.   The list can contain numbers or a reference to a range that contains numbers.

Examples

VAR(4.0, 3.0, 3.0, 3.5, 2.5, 4.0, 3.5) returns .31

See Also

STDEV
STDEVP
VARP

VARP(number_list)

Returns the variance of a population based on an entire population of values.

A list of as many as 30 numbers, separated by commas is a **number_list**.   The list can contain numbers or a reference to a range that contains numbers.

Examples

VARP(4.0, 3.0, 3.0, 3.5, 2.5, 4.0, 3.5) returns .27

See Also

STDEV
STDEVP
VAR

DATEVALUE(text)

Returns the serial number of a date supplied as a text string.

In text format, **text** is a date, between January 1, 1900 and December 31, 2078.   By omitting the year, then the current year is used.

Examples

DATEVALUE("12/25/95") returns 35058
DATEVALUE("8/22/55") returns 20323

See Also

NOW
TIMEVALUE
TODAY

<u>Summing, Counting, and Other Statistical Functions</u>

<u>SUM</u>
<u>COUNT</u>
<u>COUNTA</u>
<u>AVERAGE</u>
<u>MIN</u>
<u>MAX</u>
<u>FACT</u>
<u>STDEV</u>
<u>STDEVP</u>
<u>SUMSQ</u>
<u>VAR</u>
<u>VARP</u>

Indexing, Matching, and Looking Up Values

VLOOKUP
HLOOKUP
MATCH
INDEX
LOOKUP
CHOOSE
OFFSET

Time Functions

NOW
HOUR
MINUTE
SECOND
TIME
TIMEVALUE

Manipulating Text

LEFT
MID
RIGHT
&
LEN
LOWER
UPPER
SUBSTITUTE
FIND
SEARCH
REPLACE
PROPER
TRIM
CLEAN
REPLACE
CHAR
CODE
DOLLAR
EXACT
REPT
FIXED
T
TEXT

Date Functions

NOW
TODAY
WEEKDAY
MONTH
DAY
DATEVALUE
YEAR
DATE

TODAY()

Returns the current date as a serial number.

TODAY does not use arguments.   You must include empty parentheses to correctly reference the function.

See Also

DATE
DAY
NOW

WEEKDAY(serial_number)

Returns the day of the week that corresponds to the supplied date.

Serial_number is the date as text or as a serial number.

WEEKDAY returns a number ranging from 1 to 7. (Sunday to Saturday).

Example

WEEKDAY("06/05/95") returns 2, indicating Monday

See Also

DAY
NOW
TEXT
TODAY

Print dialog box

The following options allow you to specify how the document should be printed:

Printer
    This is the active printer and printer connection.   Choose the Setup option to change the printer and printer connection.

Setup
    Displays a Print Setup dialog box, so   you can select a printer and printer connection.

Print Range
    Specify the pages you want to print:
    All   Prints the entire document.
    Selectio   Prints the currently selected text.
    n
    Pages   Prints the range of pages you specify in the From and To boxes.

Copies
    Specify the number of copies you want to print for the above page range.

Collate Copies
    Prints copies in page number order, instead of separated multiple copies of each page.

Print Quality
    Select the quality of the printing.   Generally, lower quality printing takes less time to produce.

Print Progress Dialog

The Printing dialog box is shown during the time that VistaCalc is sending output to the printer.   The page number indicates the progress of the printing.

To abort printing, choose Cancel.

Print Preview command (File menu)

Use this command to display the active document as it would appear when printed.   When you choose this command, the main window will be replaced with a print preview window in which one or two pages will be displayed in their printed format.   The <u>print preview toolbar</u> offers you options to view either one or two pages at a time; move back and forth through the document; zoom in and out of pages; and initiate a print job.

Print Preview toolbar

The print preview toolbar offers you the following options:

Print
Bring up the print dialog box, to start a print job.

Next Page
Preview the next printed page.

Prev Page
Preview the previous printed page.

One Page / Two Page
Preview one or two printed pages at a time.

Zoom In
Take a closer look at the printed page.

Zoom Out
Take a larger look at the printed page.

Close
Return from print preview to the editing window.

Print Setup command (File menu)

Use this command to select a printer and a printer connection.   This command presents a Print Setup dialog box, where you specify the printer and its connection.

Print Setup dialog box

The following options allow you to select the destination printer and its connection.

Printer
Select the printer you want to use.   Choose the Default Printer; or choose the Specific Printer option and select one of the current installed printers shown in the box.   You install printers and configure ports using the Windows Control Panel.

Orientation
Choose Portrait or Landscape.

Paper Size
Select the size of paper that the document is to be printed on.

Paper Source
Some printers offer multiple trays for different paper sources.   Specify the tray here.

Options
Displays a dialog box where you can make additional choices about printing, specific to the type of printer you have selected.

Network...
Choose this button to connect to a network location, assigning it a new drive letter.

Page Setup

The Page Setup Command allows you to specify a per-page header and/or footer and margins.   If you don't want a header or footer, leave the header or footer blank. You can specify any text in the header and footer.   You can also specify the following format codes:

Format Code	Description
&R	Right-aligns the characters.
&C	Centers the characters.
&L	Left-aligns the characters.
&F	Prints worksheet name.
&T	Prints current time.
&D	Prints current date.
&P	Prints page number.
&P+*number*	Prints page number plus *number.*
&P-*number*	Prints page number minus *number.*
&&	Prints an ampersand.
&N	Prints total number of pages in the document.

Text and codes ,by default, are centered unless &L or &R is specified.

Font codes must appear before other codes and text or they are ignored.   The alignment codes restart each section; new font codes can be specified after an alignment code.   Here is a list of the following font codes:

Format Code	Description
&B	Uses a bold font.
&I	Uses an italic font.

&U	Underlines the header.
&S	Strikeout the header.
&O	Ignored.
&H	Ignored.
&"fontname"	Uses the specified font.
&nn	Uses the specified font sized - must be a two digit number.

Activating Menus with the Keyboard

Press the ALT key.

To activate an item, hold down the ALT key and press the underlined letter of the item you wish to activate.

Toolbar

The toolbar is displayed across the top of the application window, below the menu bar.   The toolbar
provides quick mouse access to many tools used in VistaCalc.

Click	To
	Open a new document.
	Open an existing document.
	Save the active document with its current name.
	Save the active document with a different name.
	Remove selected data from the document and stores it on the clipboard.
	Copy the selection to the clipboard.
	Insert the contents of the clipboard at the insertion point.
	Specify a cell to display in the worksheet window.
	Change the font in the active document.
	Sets the fill pattern and colors.
	Specify the border line style and color.
	Deletes the current selection.
	Objects data from the current selection.
	Sorts keys for data sorting.
	Recalculates the worksheet.
	Show or hide formulas in place of cell values.
	Formats data using the Currency format, and a decimal precision of 0.
	Uses the Percent format, numbers in this format are displayed as percentages.
	Formats data using the Fixed format.
	Uses the Fraction format, numbers in this format are displayed as fractions.
	Defines footer and header text, page margins, worksheet-related print options, etc.
	Prints the worksheet.
	Information about VistaCalc.
	Quit VistaCalc; prompts to save the modified documents.

DATE(year, month, day)

Returns the serial number corresponding to year, month, and day.

Year is a number from 1900 to 2078.   If year is between 1920 to 2019, you can specify two digits to represent the year.   To specify a year before 1920 or after 2019, specify all four digits of the year.

Month is a number representing the month.   If the number supplied is greater than 12, the number is added to the first month of the specified year.

Day is a number representing the day of the month.   If the number specified for day exceeds the number of days in that month, the number is then added to the first day of the specified month.

Examples

DATE(99, 3, 6) returns 36225
DATE(94, 6, 21) returns 34506

See Also

DAY
MONTH
NOW
TIMEVALUE
TODAY
YEAR

ABS(number)

Returns the absolute value of a number.

Examples

ABS(-1)   returns 1
ABS(1) returns 1

See Also

SIGN

INT(number)

Rounds the supplied number down to the nearest integer.

Examples

INT(1.99)   returns 1
INT(-1.99)   returns -2

See Also

CEILING
FLOOR
MOD
ROUND
TRUNC

LN(number)

Returns the natural logarithm of a number, based on the constant e.

Examples

LN(20.09) returns 3.00
LN(86) returns 4.45

See Also

EXP
LOG
LOG10

LOG(number(,base))

Returns the logarithm of a number to base.

Examples

LOG(10) returns 1
LOG(1) returns 0

See Also

EXP
LN
LOG10

LOG10(number)

Returns base 10 logarithm of a number.

Examples

LOG10(10) returns 1
LOG10(100) returns 2

See Also

EXP
LN
LOG

SQRT(number)

Returns square root of a number.

Number is a positive number. If a negative number is specified, #NUM! returns the error value.

Examples

SQRT(25) returns 5
SQRT(-25) returns #NUM!

See Also

SUMSQ

EXP(number)

Returns e raised to the power of specified number.   The base of the natural logarithm is the constant e, 2.71828182845904.

Examples

EXP(3) returns 20.09
EXP(1) returns 2.71828

See Also

LN
LOG

ROUND(number,precision)

Rounds a given number to the specified decimal places.

Examples

ROUND(3.15) returns 3.2
ROUND(3.22) returns 3.2

See Also

CEILING
FLOOR
INT
MOD
TRUNC

TRUNC(number,precision)

Rounds number down to nearest integer.

Precision argument is optional, if omitted it is assumed to be zero.

Examples

TRUNC(9.975,0) returns 9
TRUNC(6899.435, -2) returns 6800

See Also

CEILING
FLOOR
INT
MOD
ROUND

FLOOR(number, significance)

Rounds number down to nearest multiple of specified significance.

Number is the value to which you round.

Significance is the multiple to which you round.

Examples

FLOOR(1.5, 0.1) returns 1.5
FLOOR(1.23459, 0.5) returns 1.2

See Also

CEILING
EVEN
INT
ODD
ROUND
TRUNC

CEILING(number, significance)

Rounds number to the nearest multiple of significance.

Number is the value to which you round.

Significance is the multiple to which you round.

Examples

CEILING(1.23459, .05) returns 1.25
CEILING(-2.5, -2) returns -4

See Also

EVEN
FLOOR
INT
ODD
ROUND
TRUNC

ODD(number)

Rounds number up to the nearest odd integer.

Examples

ODD(3) returns 3
ODD(6) returns 7

See Also

CEILING
EVEN
FLOOR
INT
ROUND
TRUNC

EVEN(number)

Rounds number up to the nearest even integer.

Examples

EVEN(2.5) returns 4
EVEN(3) returns 4

See Also

CEILING
FLOOR
INT
ODD
ROUND
TRUNC

TYPE(expression)

Returns a number indicating the type of the given expression.

Expression types and numbers:

Expression type	Number
Number	1
Text string	2
Logical value	4
Error value	16
An array	64

Examples

TYPE(A1) returns 1 if cell A1 contains a number.
TYPE("Customer") returns 2

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

N(value)

If value is a number, returns that number.

Value is a reference to a cell containing a value to test.

Examples

N(32467) returns 32467
N(A4) returns 1 if A4 contains the logical function True

See Also

T
VALUE

SIGN(number)

Determines the sign of a number.

Number is any number.   If number is positive, returns 1.   If number is zero, returns 0. If number is negative, returns -1.

Examples

SIGN(123) returns 1
SIGN(4-4) returns 0

See Also

ABS

VALUE(text)

Returns the text as a number.

Text can be any text string, a cell reference that contains a text string, or a formula that evaluates to a text string.   Value can also return the text in date or time formats.   If the format is not recognized then #VALUE! is returned.

Examples

VALUE(9800) returns 9800
VALUE("$2,000") returns 2000

See Also

DOLLAR
TEXT

SIN(number)

Returns the sine of number.

Number is the angle in radians.   If the angle is in degrees, multiply the angle by PI()180, to convert to radians.

Examples

SIN(PI()/2) returns 1
SIN(90) returns .89

See Also

ASIN
PI

COS(number)

Returns the cosine of a given angle.

Number is the angle in radians.   If the angle is in degrees, multiply the angle by PI()/180, to convert the angle to radians.

Examples

COS(5) returns .28
COS(PI()/2) returns 0

See Also

ACOS
ASINH
ATANH
PI

TAN(number)

Returns the tangent of a given angle.

Number is the angle in radians.   Multiply the degrees by 180/PI() to convert a number expressed as degrees.

Examples

TAN(45) returns 1.62
TAN(0.785) returns   0.99204

See Also

ATAN
PI

ASIN(number)

Returns the arcsine of a number in radians (ranging from -pi/2 to pi/2).

Number is the sine of the resulting angle (ranging from -1 to 1).   Multiply the radians by 180/PI() to convert the resulting radians to degrees.

Examples

ASIN(-1) returns -1.57
ASIN(-0.5) returns -0.52 (pi/6 radians)

See Also

ASINH
PI
SIN

ACOS(number)

Returns the arc cosine of a number.

Number is the cosine of the angle ranging from 1 to -1.   The   resulting angle is returned in radians from 0 to pi.   Multiply the radians by 180/PI() to convert the resulting radians to degrees.

Examples

ACOS(.5) returns 1.05
ACOS(-0.5)*180/PI() returns 120 degrees

See Also

COS
PI

ATAN(number)

Returns the arc tangent of number.

Number is the tangent of the angle.   The resulting angle is returned in radians from -pi/2 to pi/2.   Multiply the radians by 180/PI() to convert the resulting radians to degrees.

Examples

ATAN(3.5) returns 1.29
ATAN(1) returns 0.785 (pi/4 radians)

See Also

ATANH
PI
TAN

ASINH(number)

Returns the inverse hyperbolic sine of a number.

Number is any number.

Examples

ASINH(-4) returns -2.09
ASINH(10) returns 2.998223

See Also

ASIN
ATANH

ACOSH(number)

Returns the inverse hyperbolic cosine of number.

Number is any number that is equal to or greater than 1.

Examples

ACOSH(3) returns 1.76
ACOSH(1.2) returns .62

See Also

ASINH
ATANH
COSH

ATANH(number)

Returns the inverse hyperbolic tangent of a number.

Number must range between -1 and 1.

Examples

ATANH(0.50) returns 0.549306
ATANH(.5) returns .55

See Also

ACOS
ASINH

IF(condition, true_value, false_value)

Tests the condition and returns the specified value.

Condition is any logical expression.

True_value is the value to be returned if condition evaluates to True.

False_value is the value to be returned if condition evaluated to False.

Examples

IF(TRUE,1,2) returns 1
IF(FALSE,1,2) returns 2

See Also

AND
FALSE
NOT
OR
TRUE

INDIRECT(ref_text [,a1])

Returns the contents of the cell referenced by the ref_text.

Ref_text is a reference to a cell that references a third cell.   If ref_text is not a valid cell reference, then #REF! is returned.

A1 is the reference format.   TRUE() must be represented for an A1 reference format.

Example

INDIRECT (C1) returns the contents of the cell that c1 references If C1 contains "D1", then INDIRECT returns the contents of D1.

See Also

OFFSET

ATAN2(x,y)

Returns the arc tangent of the x and y coordinates.

The angle is returned in radians, ranging from -pi to pi, excluding -pi.

Examples

ATAN2(-1, .1) returns 3.04
ATAN2(3, 6) returns 1.11

See Also

ATAN
ATANH
PI
TAN

ERROR.TYPE(error_ref)

Returns a number corresponding to an error.

Error_ref is a cell reference.

The following lists the error text and associated error numbers returned by the function Error.type:

Error text	Number
#NULL!	1
#DIV/0!	2
#VALUE!	3
#REF!	4
#NAME?	5
#NUM!	6
#N/A	7
Other	#N/A

Example

ERROR.TYPE(A1) returns 2 if the formula in cell A1 attempts to divide by zero.

See Also

ISERR
ISERROR

MOD(number, divisor)

Returns the remainder after a number is divided by a specified divisor.

Divisor is any non-zero number.   #DIV/0! is returned if divisor is 0.

Examples

MOD(3,2) returns 1
MOD(-3,-2) returns -1

See Also

INT
ROUND
TRUNC

ROW(reference)

Returns the row number of the supplied reference.

Reference is a cell.   If reference is omitted it is assumed to be the reference of the cells in which the function ROW is entered.

Example

ROW(C3) returns 3

See Also

COLUMN
ROWS

ROWS(range)

Returns the number of rows in a range reference.

Examples

ROWS(A1:D5) returns 5
ROWS(A1:C4) returns 4

See Also

COLUMNS
ROW

COLUMN(reference)

Returns the column number of reference.

Reference is a reference to a cell.   Reference cannot be a reference to multiple areas. Omitting the argument returns the number of the column in which COLUMN is replaced.

Examples

COLUMN(B3) returns 2
COLUMN(A3) returns 1

See Also

COLUMNS
ROW

COLUMNS(range)

Returns the number of columns in a range reference.

Range is a reference to a range of cells.

Examples

COLUMNS(A1:D5) returns 4
COLUMNS(A1:C4) returns 3

See Also

COLUMN
ROWS

ADDRESS(row, column, ref_type [,a1] [,sheet])

Creates a cell address as text.

Row is the row number for a cell address.

Column is the column number for an address.

Ref_type is the cell reference type.   The following lists the values for this argument:

Reference type	Argument
Absolute	1
Absolute row, relative column	2
Relative row, absolute column	3
Relative	4

a1 is the reference format.   This argument must be TRUE() to represent an A1 reference format.

Sheet_text is the name of an external spreadsheet.   If omitted, then there is no sheet name used.

Examples

ADDRESS(5, 6, 1) returns "F5"
ADDRESS(2,3) returns "C2"

See Also

COLUMN
OFFSET
ROW

AND(logical_list)

Returns the logical value True if all arguments are true.   Returns False if any argument is false.

Logical_list is a list of conditions separated by commas.   Logical_list can have 1 to 30 conditions in the list.   The list can contain logical values or a reference to a range containing logical values.   Text and empty cells are ignored.

#VALUE! is returned if there are no logical values in the list.

Examples

AND(TRUE(), FALSE()) returns False
AND(TRUE(), TRUE()) returns True

See Also

IF
NOT
OR

OR(logical_list)

Returns True if any of a series of logical arguments is true.   Returns False if any of a series of logical arguments is false.

Logical_list is a list of conditions separated by commas.   Logical_list can have 1 to 30 conditions in the list.   The list can contain logical values or a reference to a range containing logical values.   Text and empty cells are ignored.

#VALUE! is returned if there are no logical values in the list.

Examples

OR(TRUE()) returns true
OR(1+1=1, 2+2=5) returns false

See Also

AND
IF
NOT

NOT(logical)

Returns a logical value that is the opposite of its value.

Examples

NOT(TRUE) returns False
NOT(2+2=4) returns False

See Also

AND
IF
OR

PI()

Returns the value of pi, which is approximately the number 3.14159 when calculated to 15 significant digits.

PI does not use arguments.   You must include empty parentheses to correctly reference the function.

Example

PI()/2 returns 1.57079

See Also

COS
SIN
TAN

RAND()

Returns a number selected randomly greater than or equal to 0 and less than 1.

RAND does not use arguments.   You must include empty parentheses to correctly reference the function.

Example

RAND()*10 returns a random number greater than or equal to 0 and less than 10.

TRUE()

Returns the logical value True.   You must include the parentheses when using this function.

See Also

FALSE

FALSE()

Returns the logical value False.   You must include the parentheses when using this function.

See Also

TRUE

COSH(number)

Returns the hyperbolic of number.

Number is any number.

Examples

COSH(2.10) returns 4.14
COSH(4) returns 27.30823

See Also

ASINH
ATANH
COS

NA()

Returns the error value #N/A, which represents "no value is available."

NA marks cells that lack data without leaving them empty.   Empty cells may not be correctly represented in some calculations.

See Also

ISNA

PRODUCT(number_list)

Multiplies a list of numbers and returns the result.

A list of as many as 30 numbers, separated by commas is **number_list**.

The list can contain logical values, numbers, text representation of numbers, or a reference to a range containing those values.

Text or error values that cannot be translated into numbers return errors.

If a range reference is included in the text, logical expression, list, and empty cells in the range are ignored.

All numeric values, including 0, are used in the calculation.

Example

PRODUCT(2, 3, 4) returns 24

See Also

FACT
SUM

Business and Financial Functions

PMT
PPMT
PV
RATE
FV
IPMT
NPER
NPV
IRR
MIRR
DB
DDB
SLN
SYD
VDB

Math Functions

ABS
INT
LN
LOG
LOG10
SQRT
EXP
ROUND
TRUNC
FLOOR
CEILING
ODD
EVEN
TYPE
N
SIGN
VALUE
SIN
COS
COSH
TAN
ASIN
ACOSH
ATAN
ATAN2
ASINH
ACOH
ATANH
PRODUCT

Misc. Functions

[IF](#)
[INDIRECT](#)
[ERROR.TYPE](#)
[MOD](#)
[ROW](#)
[ROWS](#)
[COLUMN](#)
[COLUMNS](#)
[ADDRESS](#)
[AND](#)
[OR](#)
[NOT](#)
[PI](#)
[RAND](#)
[TRUE](#)
[FALSE](#)
[NA](#)

Error Values

VistaCalc displays an error value in a cell when it cannot properly calculate the formula for that cell.

If a formula includes a reference to a cell that contains an error value, that formula also produces an error value (unless you are using the special worksheet functions ISERR, ISERROR, or ISNA, which look for error values).   You may have to trace the references back through a series of cells to discover the source of the error.

Error value	Meaning
#DIV/0!	The formula is trying to divide by zero.
#N/A!	No value is available. Usually, you enter this value directly into worksheet cells that will eventually contain data that is not yet available. Formula referring to those cells will return #N/A! instead of calculating a value.
#NAME?	VistaCalc does not recognize a name used in the formula.
#NULL!	You specified an intersection of two areas that do not intersect.
#NUM!	There is a problem with a number.
#REF!	The formula refers to a cell that is not valid.
#VALUE!	An argument or operand is of the wrong type.

Test Functions

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

MONTH(serial_number)

Returns the month that corresponds to the supplied date.

Serial_number is the date as a text or as a serial number.

MONTH returns a number ranging from 1 to 12 (January to December).

Examples

MONTH("06/05/95") returns 6
MONTH("08/23/75") returns 8

See Also

DAY
NOW
HOUR
MINUTE
SECOND
TODAY
WEEKDAY
YEAR

DAY(serial_number)

Returns the day of the month corresponding to the date represented by the serial_number.

Serial_number is the date as a text or as a serial number.

Examples

DAY("8/23/75") returns 23
DAY("5-Feb") returns 5

See Also

NOW
HOUR
MINUTE
MONTH
SECOND
TODAY
WEEKDAY
YEAR

YEAR(serial_number)

Returns the year corresponding to the supplied date.

Serial_number is the date as a text or as a serial number.

Examples

YEAR("08/23/75") returns 1975
YEAR(34328) returns 1993

See Also

DAY
NOW
HOUR
MINUTE
MONTH
SECOND
TODAY
WEEKDAY

ISBLANK(reference)

Returns True if the referenced cell is blank.   Returns False if the referenced cell is not blank.

Reference is a reference to any cell.

Example

ISBLANK(A1) returns True if A1 is a blank cell.

See Also

ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

ISERR(expression)

Determines if the specified expression returns an error value.

True is returned if the expression returns any error except #N/A!.   Otherwise, False is returned.

For a list of error values, see Error Values in VistaCalc Help.

Example

ISERR(A1) returns True if the formula in A1 returns an error (e.g., #NUM!).

See Also

ISBLANK
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

ISERROR(expression)

Determines if the specified expression returns an error value.

True is returned if the expression returns any error value.   Otherwise, False is returned.

For a list of error values, see Error Values in VistaCalc Help.

Example

ISERROR(A1) returns False if the formula in A1 does not return an error.

See Also

ISBLANK
ISERR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

ISLOGICAL(expression)

Determines if the specified expression returns a logical value.

True is returned if the expression returns a logical value.   Otherwise, False is returned.

Example

ISLOGICAL(ISBLANK(A1)) returns True, ISBLANK returns a logical value.

See Also

ISBLANK
ISERR
ISERROR
ISNA
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

ISNA(expression)

Determines if the specified expression returns the value not available error.

True is returned if the expression returns the #N/A! error.   Otherwise, False is returned.

Example

ISNA(A1) returns True if the NA() function is contained in cell A1 or returns the error value #N/A!.

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNONTEXT
ISNUMBER
ISREF
ISTEXT

ISNONTEXT(expression)

Determines if the specified expression is not text.

True is returned if the expression returns any value that is not text.   Otherwise, False is returned.

Examples

ISNONTEXT("text") returns False
ISNONTEXT(A3) returns True if cell A3 is a blank cell or contains a number.

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNUMBER
ISREF
ISTEXT

ISNUMBER(expression)

Determines if the specified expression is a number.

True is returned if the expression returns a number.   Otherwise, False is returned.

False is returned if expression returns a number represented as text.

Examples

ISNUMBER("525") returns False
ISNUMBER(525.55) returns True

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISREF
ISTEXT

ISREF(expression)

Determines if the specified expression is a range reference.

True is returned if the expression returns a range reference.   Otherwise, False is returned.

Example

ISREF(A3) returns True

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISTEXT

VistaCalc 1.1 Registration Form

To print this form, in the "File" menu choose "Print Topic".

If you cannot print this form, write a note with the same information on it as is on this form.

If you have CompuServe, you can register online.
GO SWREG.   Registration ID: 7142

If you plan to use VistaCalc on more than one computer, a site license is required.   Volume discounts and site licenses can be obtained by contacting Brandon Fridley.

Please fill out the following information then send it to:

    Brandon Fridley
    P.O. Box 6174
    Vancouver, WA 98668
    U.S.A.

Name   _____

Company   _____

Address   _____

City/State/Zip   _____

Country   _____

Telephone Number   _____

E-mail Address   _____

Quantity _____ * each license $29.95   _____

Washington State residents, add sales tax   _____

Shipping and handling
(see rates below)   _____

                   Total   _____

Shipping and handling rates:

Country	Amount
United States	$2.65
Other Countries	$3.95

Make checks or money orders (U.S.) payable to: Brandon Fridley.

Credit card orders only:

You can order with MC, Visa, Amex, or Discover from Public (software) Library by calling 800-2424-PsL or 713-524-6394 or by FAX to 713-524-6398 or by CIS Email to 71355,470. You can also mail credit card orders to PsL at P.O.Box 35705, Houston, TX 77235-5705.

VistaCalc - item/product #14205

The above numbers are for orders only.

To insure that you get the latest version, PsL will notify us the day of your order and we will ship the product directly to you.

Any questions about the status of the shipment of the order, refunds, registration options, product details, technical support, volume discounts, dealer pricing, site licenses, etc, must be directed to:

    Brandon Fridley
    P.O. Box 6174
    Vancouver, WA 98668
    U.S.A.

E-mail:
CompuServe - 74012,1635
Internet - 74012.1635@compuserve.com

Thank you for your support.

Questionnaire:

How did you hear of VistaCalc (store, BBS, friend, magazine, etc.)?

Comments:

ISTEXT(expression)

Determines if the specified expression is text.

True is returned if the expression returns text.   Otherwise, False is returned.

Examples

ISTEXT("9th Inning") returns True

See Also

ISBLANK
ISERR
ISERROR
ISLOGICAL
ISNA
ISNONTEXT
ISNUMBER
ISREF

Registering VistaCalc

VistaCalc Version 1.1
Copyright © 1995 Brandon Fridley
All rights reserved.

Brandon Fridley
P.O. Box 6174
Vancouver, WA 98668
U.S.A.

E-mail:
CompuServe - 74012,1635
Internet - 74012.1635@compuserve.com

DESCRIPTION:

A fast, powerful, and easy to use spreadsheet program for the home and office.   VistaCalc's user friendly interface enables you to create a wide range of documents.   Everything from your own personalized budget plan to financial and accounting presentations for the office. Full spreadsheet program with many special features and functions, that only programs, such as Excel, can offer.

REQUIRES: Windows 3.1 or greater.

INSTALLING:

Run install.exe

GENERAL INFORMATION:

VistaCalc is shareware. Try before you buy software. If you would like to continue to use VistaCalc after the 30 day evaluation period you must register your copy. To register, fill out the <u>order form</u> in the file form.txt and send it to Brandon Fridley. If form.txt is missing print the order form in the help file or contact Brandon Fridley.   Registering will remove the notice at the ending of VistaCalc. You also will receive the latest version of VistaCalc, and will be entitled to technical support.

A limited license is granted to copy and distribute VistaCalc for trial use, subject to the following conditions:

I. VistaCalc must be copied in unmodified form, complete with the following files:

vista.exe
install.exe
ctl3d.dll
appsetup.inf
vista.hlp
vista.ini
readme.txt
form.txt
file_id.diz

II. VistaCalc may not be distributed in conjunction with any other product without a specific license to do so from Brandon Fridley.

LIMITED LICENSE

THE AUTHOR (BRANDON FRIDLEY) DISCLAIMS ALL WARRANTIES RELATING TO THIS SOFTWARE (VISTACALC), WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NEITHER THE AUTHOR NOR ANYONE ELSE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DELIVERY OF THE SOFTWARE SHALL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, OR INCIDENTAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE EVEN IF THE AUTHOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR CLAIMS. IN NO EVENT SHALL THE AUTHOR'S LIABILITY FOR ANY DAMAGES EVER EXCEED THE PRICE PAID FOR THE LICENSE TO USE THE SOFTWARE, REGARDLESS OF THE FORM OF CLAIM. THE PERSON USING THE SOFTWARE BEARS ALL RISK AS TO THE QUALITY AND PERFORMANCE OF THE SOFTWARE. SOME STATES DO NOT ALLOW THE EXCLUSION OF THE LIMIT OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MIGHT NOT APPLY TO YOU.

NON-LICENSED USERS ARE GRANTED A LIMITED LICENSE TO USE THE SOFTWARE FOR A 30-DAY TRIAL PERIOD FOR THE PURPOSE OF DETERMINING WHETHER THE SOFTWARE IS SUITABLE FOR THEIR NEEDS.   THE USE OF THE SOFTWARE AFTER THE 30-DAY TRIAL PERIOD REQUIRES REGISTRATION.   THE USE OF UNLICENSED COPIES OF THE SOFTWARE, AFTER THE 30-DAY TRIAL PERIOD, BY ANY INDIVIDUAL OR ENTITY IS PROHIBITED.   A SINGLE USER LICENSE PERMITS A USER TO USE THE SOFTWARE ONLY ON A SINGLE COMPUTER. LICENSED USERS MAY USE THE SOFTWARE ON DIFFERENT COMPUTERS, BUT MAY NOT USE THE SOFTWARE ON MORE THAN ONE COMPUTER AT THE SAME TIME.   NO ONE MAY MODIFY THE SOFTWARE IN ANY WAY.

USE OF THE SOFTWARE CONSTITUTES YOUR ACCEPTANCE OF THIS AGREEMENT AND SUBJECTS YOU TO ITS CONTENTS.

THIS AGREEMENT IS GOVERNED BY THE LAWS OF THE STATE OF WASHINGTON AND SHALL INURE TO THE BENEFIT OF THE AUTHOR AND ANY SUCCESSORS, ADMINISTRATORS, HEIRS AND ASSIGNS.   ANY ACTION OR PROCEEDING BROUGHT BY EITHER PARTY AGAINST THE OTHER ARISING OUT OF OR RELATED TO THIS AGREEMENT SHALL BE BROUGHT ONLY IN A STATE OR FEDERAL COURT OF COMPETENT JURISDICTION LOCATED IN CLARK COUNTY, WASHINGTON, U.S.A.   THE PARTIES HEREBY CONSENT TO IN PERSONAL JURISDICTION OF SAID COURTS.

Edit Menu

The Edit menu includes commands that enable you to move text to and from the clipboard, to delete text, and to undo a previous editing operation.

Cut   Cuts current worksheet selection to the clipboard.

Copy   Copies current worksheet selection to the clipboard.

Paste   Pastes contents of the clipboard to the   current worksheet selection.

Paste Values   Pastes Values from clipboard to the current worksheet selection.   Formatting applied to the values is ignored. Only formula results are pasted, formulas are ignored.

Clear   The Clear dialog box is displayed, allowing you to object the data from the current selection.   You can only clear formats, only values, or both.

Insert   Inserts cells at the location of the current selection.   To make room for new cells, cells adjacent to the insertion are shifted.

Delete   Deletes the current selection.   Cells adjacent to the deleted cells are shifted to fill the space left by the vacated cells.

Goto   The Goto dialog box is displayed, allowing you to specify a cell to display in the worksheet window.   The specified cell is made the active cell.

Sort   The Sort dialog box is displayed, allowing you to set the sorting method and sort keys for data sorting.

File Menu

The File menu includes commands that enable you to open, save, and create new files, print, and exit.

New   Creates a new worksheet.

Open   Opens a worksheet from disk.

Save   Saves the current worksheet.

Save As   Saves the current worksheet with a different format or name.

Print Area   Defines the worksheet range(s) to be printed.

Print Titles   Specifies the rows and columns printed on each new page.

Add Page Break   Places a vertical page break adjacent to the left edge of the active cell, and a horizontal page break adjacent to the top edge of the active cell.   If a column or row is selected, then a page break is placed adjacent to the selected column or row.

Remove Page Break Replaces Set Page Breaks if page breaks are adjacent to the active cell.   Removes page breaks adjacent to the top edge and left edge of the active cell.

Print   Prints the worksheet.

Page Setup   The Page Setup dialog box is displayed, allowing you to define footer and header text, page margins, page print order, page centering, work sheet-related print options.

Print Setup   The standard Windows Print Setup dialog box is displayed, allowing you to select the printer to which the worksheet is sent, the page orientation, and paper size.

Exit   Quits VistaCalc; prompts to save the modified documents.

Window Menu

The Window menu includes commands that enable you to arrange and change to open windows.

New Window	An additional window is created, that displays the current worksheet.
Cascade	Windows are placed in cascading arrangement, if multiple worksheet windows are displayed.
Tile	Windows are tiled so that each worksheet is displayed, if multiple worksheet windows are displayed.
Arrange Icons	Arranges the icons of minimized worksheets.

Help Menu

The Help menu includes commands that enable you to receive help on VistaCalc and help on using help.   You can also display program information such as version number and copyright.

Index   Lists all topics.

Search   Allows you to search for help on words and topics.

Using Help   Displays instructions on using help.

About VistaCalc   Displays program information, version number, and copyright.