Excel Function Dictionary

v1.0 Beta

What Is In The Dictionary ?

This workbook contains 157 worksheets, each explaining the purpose and usage of particular Excel functions.

There are also a number of sample worksheets which are simple models of common applications, such as Timesheet and Date Calculations.

Formatting

Each worksheet uses the same type of formatting to indicate the various types of entry.

North
100
100
100
300

Text headings are shown in grey.

Data is shown as purple text on a yellow background.

The results of Formula are shown as blue on yellow.

=SUM(C13:C15) The formula used in the calulations is shown as blue text.

The Arial font is used exclusivley throughout the workbook and should display correctly with any installation of Windows.

Each sheet has been designed to be as simple as possible, with no fancy macros to accomplish the desrired result.

Printing

Each worksheet is set to print on to A4 portrait.
The printouts will have the column headings of A,B,C... and the row numbers 1,2,3... which will assist with the reading of the formula.
The ideal printer would be a laser set at 600dpi.
If you are using a dot matrix or inkjet, it may be worth switching off the colours before printing, as these will print as dark grey. (See the sheet dealing with Colour settings).

Protection

Each sheet is unprotected so that you will be able to change values and experiment with the calculations.

Macros

There are only a few very simple macros which are used by the various buttons to naviagte through the sheets. These have been written very simply, and do not make any attempt to change your current Toolbars and Menus.

What Do The Buttons Do ?

View

View

This button will display the worksheet containing the function example.

1. Click on the function name, then
2. Click on the **View** button.

Sort

Sort

This button sorts the list of functions into alphabetical order.

Category

Category

This describes the category the function is a member of.

Click this button to sort alphabetically.

Location

Location

This shows where the function is stored in Excel.

Built-in indicates that the function is part of Excel itself.

Analysis ToolPak indicates the function is stored in the Analysis ToolPak add-in.

Click this button to sort alphabetically.

Using Different Monitor Settings

Each sheet has been designed to fit within the visible width of monitors with a low resolution of 640 x 480. This ensures that you do not need to scroll from left and right to see all the data.

The colours are best suited to monitors capable of 256 colours.
On monitors using just 16 colours the greys may look a bit rough!
You can switch colours off and on using the button below.

☒ Colour On

This may take a few minutes on any computer !

Sample Colour Scheme

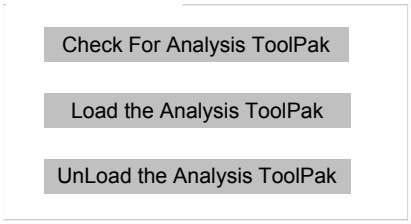
	North	South	East	West	Total
Alan	100	100	100	100	400
Bob	100	100	100	100	400
Carol	100	100	100	100	400
Total	300	300	300	300	1200

Analysis ToolPak

What Is The Analysis ToolPak ?

The Analysis ToolPak is an add-in file containing extra functions which are not built in to Excel. The functions cover areas such as Date and Mathematical operations.

The Analysis ToolPak must be added-in to Excel before these functions will be available.



Any formula using these functions without the ToolPak loaded will show the #NAME error.

Age Calculation	Sample	Sample	Using DATEDIF()
AutoSum shortcut key	Sample	Sample	Using Alt and =
Brackets in formula	Sample	Sample	Sample
FileName formula	Sample	Sample	Using MID() CELL() and FIND()
Instant Charts	Sample	Sample	Using F11
Ordering Stock	Sample	Sample	Stock Ordering
Percentages	Sample	Sample	How to calculate various percentages
Project Dates	Sample	Sample	Example using date calculation.
Show all formula	Sample	Sample	Using Ctrl and `
Split ForenameSurname	Sample	Sample	Using LEFT() RIGHT() FIND() SUBSTITUTE()
Time Calculation	Sample	Sample	How to calculate time.
TimeSheet For Flexi	Sample	Sample	Example flexi time sheet.
ABS	Mathematical	Built-in	Returns the absolute value of a number
AND	Logical	Built-in	Returns TRUE if all its arguments are TRUE
AVERAGE	Statistical	Built-in	Returns the average of its arguments
BIN2DEC	Engineering	Analysis ToolPak	Converts a binary number to decimal
CEILING	Mathematical	Built-in	Rounds a number to the nearest integer or to the nearest multiple of signific
CELL	Information	Built-in	Returns information about the formatting, location, or contents of a cell
CHAR	Text	Built-in	Returns the character specified by the code number
CHOOSE	Lookup	Built-in	Chooses a value from a list of values
CLEAN	Text	Built-in	Removes all nonprintable characters from text
CODE	Text	Built-in	Returns a numeric code for the first character in a text string
COMBIN	Mathematical	Built-in	Returns the number of combinations for a given number of objects
CONCATENATE	Text	Built-in	Joins several text items into one text item
CONVERT	Engineering	Analysis ToolPak	Converts a number from one measurement system to another
CORREL	Statistical	Built-in	Returns the correlation coefficient between two data sets
COUNT	Statistical	Built-in	Counts how many numbers are in the list of arguments
COUNTA	Statistical	Built-in	Counts how many values are in the list of arguments
COUNTBLANK	Information	Built-in	Counts the number of blank cells within a range
COUNTIF	Mathematical	Built-in	Counts the number of nonblank cells within a range that meet the given crit
DATE	Date	Built-in	Returns the serial number of a particular date
DATEDIF	Date	Built-in	Calculates the difference between two dates. Undocumented in v5/7/97
DATEVALUE	Date	Built-in	Converts a date in the form of text to a serial number
DAVERAGE	Database	Built-in	Returns the average of selected database entries
DAY	Date	Built-in	Converts a serial number to a day of the month
DAYS360	Date	Built-in	Calculates the number of days between two dates based on a 360-day yea
DB	Financial	Built-in	Returns the depreciation of an asset for a specified period using the fixed-d
DCOUNT	Database	Built-in	Counts the cells that contain numbers in a database
DCOUNTA	Database	Built-in	Counts nonblank cells in a database
DEC2BIN	Engineering	Analysis ToolPak	Converts a decimal number to binary
DEC2HEX	Engineering	Analysis ToolPak	Converts a decimal number to hexadecimal
DELTA	Engineering	Analysis ToolPak	Tests whether two values are equal
DGET	Database	Built-in	Extracts from a database a single record that matches the specified criteria
DMAX	Database	Built-in	Returns the maximum value from selected database entries
DMIN	Database	Built-in	Returns the minimum value from selected database entries
DOLLAR	Text	Built-in	Converts a number to text, using currency format
DSUM	Database	Built-in	Adds the numbers in the field column of records in the database that match
EDATE	Date	Analysis ToolPak	Returns the serial number of the date that is the indicated number of month
EOMONTH	Date	Analysis ToolPak	Returns the serial number of the last day of the month before or after a spe
ERROR.TYPE	Information	Built-in	Returns a number corresponding to an error type
EVEN	Mathematical	Built-in	Rounds a number up to the nearest even integer
EXACT	Text	Built-in	Checks to see if two text values are identical
FACT	Mathematical	Built-in	Returns the factorial of a number
FIND	Text	Built-in	Finds one text value within another (case-sensitive)
FIXED	Text	Built-in	Formats a number as text with a fixed number of decimals
FLOOR	Mathematical	Built-in	Rounds a number down, toward zero
FORECAST	Statistical	Built-in	Returns a value along a linear trend
FREQUENCY	Statistical	Built-in	Returns a frequency distribution as a vertical array
GCD	Mathematical	Analysis ToolPak	Returns the greatest common divisor
GESTEP	Engineering	Analysis ToolPak	Tests whether a number is greater than a threshold value
GROWTH	Statistical	Built-in	Returns values along an exponential trend
HEX2DEC	Engineering	Analysis ToolPak	Converts a hexadecimal number to decimal
HLOOKUP	Lookup	Built-in	Looks in the top row of an array and returns the value of the indicated cell
HOUR	Date	Built-in	Converts a serial number to an hour
IF	Logical	Built-in	Specifies a logical test to perform
INDEX	Lookup	Built-in	Uses an index to choose a value from a reference or array
INDIRECT	Lookup	Built-in	Returns a reference indicated by a text value
INFO	Information	Built-in	Returns information about the current operating environment
INT	Mathematical	Built-in	Rounds a number down to the nearest integer
ISBLANK	Information	Built-in	Returns TRUE if the value is blank

ISERR	Information	Built-in	Returns TRUE if the value is any error value except #N/A
ISERROR	Information	Built-in	Returns TRUE if the value is any error value
ISEVEN	Information	Analysis ToolPak	Returns TRUE if the number is even
ISLOGICAL	Information	Built-in	Returns TRUE if the value is a logical value
ISNA	Information	Built-in	Returns TRUE if the value is the #N/A error value
ISNONTEXT	Information	Built-in	Returns TRUE if the value is not text
ISNUMBER	Information	Built-in	Returns TRUE if the value is a number
ISODD	Information	Analysis ToolPak	Returns TRUE if the number is odd
ISREF	Information	Built-in	Returns TRUE if the value is a reference
ISTEXT	Information	Built-in	Returns TRUE if the value is text
LARGE	Statistical	Built-in	Returns the k-th largest value in a data set
LCM	Mathematical	Analysis ToolPak	Returns the least common multiple
LEFT	Text	Built-in	Returns the leftmost characters from a text value
LEN	Text	Built-in	Returns the number of characters in a text string
LOOKUP (vector)	Lookup	Built-in	Looks up values in a vector or array
LOWER	Text	Built-in	Converts text to lowercase
MATCH	Lookup	Built-in	Looks up values in a reference or array
MAX	Statistical	Built-in	Returns the maximum value in a list of arguments
MEDIAN	Statistical	Built-in	Returns the median of the given numbers
MID	Text	Built-in	Returns a specific number of characters from a text string starting at the po
MIN	Statistical	Built-in	Returns the minimum value in a list of arguments
MINUTE	Date	Built-in	Converts a serial number to a minute
MINVERSE	Mathematical	Built-in	Returns the matrix inverse of an array
MMULT	Mathematical	Built-in	Returns the matrix product of two arrays
MOD	Mathematical	Built-in	Returns the remainder from division
MODE	Statistical	Built-in	Returns the most common value in a data set
MONTH	Date	Built-in	Converts a serial number to a month
MROUND	Mathematical	Analysis ToolPak	Returns a number rounded to the desired multiple
N	Information	Built-in	Returns a value converted to a number
NA	Information	Built-in	Returns the error value #N/A
NETWORKDAYS	Date	Analysis ToolPak	Returns the number of whole workdays between two dates
NOT	Logical	Built-in	Reverses the logic of its argument
NOW	Date	Built-in	Returns the serial number of the current date and time
ODD	Mathematical	Built-in	Rounds a number up to the nearest odd integer
OR	Logical	Built-in	Returns TRUE if any argument is TRUE
PERMUT	Statistical	Built-in	Returns the number of permutations for a given number of objects
PI	Mathematical	Built-in	Returns the value of Pi
POWER	Mathematical	Built-in	Returns the result of a number raised to a power
PRODUCT	Mathematical	Built-in	Multiplies its arguments
PROPER	Text	Built-in	Capitalises the first letter in each word of a text value
QUARTILE	Statistical	Built-in	Returns the quartile of a data set
QUOTIENT	Mathematical	Analysis ToolPak	Returns the integer portion of a division
RAND	Mathematical	Built-in	Returns a random number between 0 and 1
RANDBETWEEN	Mathematical	Analysis ToolPak	Returns a random number between the numbers you specify
RANK	Statistical	Built-in	Returns the rank of a number in a list of numbers
REPLACE	Text	Built-in	Replaces characters within text
REPT	Text	Built-in	Repeats text a given number of times
RIGHT	Text	Built-in	Returns the rightmost characters from a text value
ROMAN	Mathematical	Built-in	Converts an arabic numeral to roman, as text
ROUND	Mathematical	Built-in	Rounds a number to a specified number of digits
ROUNDDOWN	Mathematical	Built-in	Rounds a number down, toward zero
ROUNDUP	Mathematical	Built-in	Rounds a number up, away from zero
SECOND	Date	Built-in	Converts a serial number to a second
SIGN	Mathematical	Built-in	Returns the sign of a number
SLN	Financial	Built-in	Returns the straight-line depreciation of an asset for one period
SMALL	Statistical	Built-in	Returns the k-th smallest value in a data set
STDEV	Statistical	Built-in	Estimates standard deviation based on a sample
STDEVP	Statistical	Built-in	Calculates standard deviation based on the entire population
SUBSTITUTE	Text	Built-in	Substitutes new text for old text in a text string
SUBTOTAL	Mathematical	Built-in	Returns a subtotal in a list or database
SUM	Mathematical	Built-in	Adds its arguments
SUM_as_Running_Total	Mathematical	Built-in	Sample
SUM_using_names	Sample	Sample	Using SUM(jan)
SUM_with_OFFSET	Lookup	Built-in	Sample
SUMIF	Mathematical	Built-in	Adds the cells specified by a given criteria
SUMPRODUCT	Mathematical	Built-in	Returns the sum of the products of corresponding array components
SYD	Financial	Built-in	Returns the sum-of-years' digits depreciation of an asset for a specified per
T	Text	Built-in	Converts its arguments to text
TEXT	Text	Built-in	Formats a number and converts it to text
TIME	Date	Built-in	Returns the serial number of a particular time
-Timesheet	Sample	Sample	Sample
TIMEVALUE	Date	Built-in	Converts a time in the form of text to a serial number

TODAY	Date	Built-in	Returns the serial number of today's date
TRANSPOSE	Lookup	Built-in	Returns the transpose of an array
TREND	Statistical	Built-in	Returns values along a linear trend
TRIM	Text	Built-in	Removes spaces from text
TRUNC	Mathematical	Built-in	Truncates a number to an integer
TYPE	Information	Built-in	Returns a number indicating the data type of a value
UPPER	Text	Built-in	Converts text to uppercase
VALUE	Text	Built-in	Converts a text argument to a number
VAR	Statistical	Built-in	Estimates variance based on a sample
VARP	Statistical	Built-in	Calculates variance based on the entire population
VLOOKUP	Lookup	Built-in	Looks in the first column of an array and moves across the row to return the
WEEKDAY	Date	Built-in	Converts a serial number to a day of the week
WORKDAY	Date	Analysis ToolPak	Returns the serial number of the date before or after a specified number of
YEAR	Date	Built-in	Converts a serial number to a year
YEARFRAC	Date	Analysis ToolPak	Returns the year fraction representing the number of whole days between s

Time Calculation

Excel can work with time very easily.
Time can be entered in various different formats and calculations performed.
There are one or two oddities, but nothing which should put you off working with it.
See the **TimeSheet** example for an example.

Typing time

When time is entered into worksheet it should be entered with a colon between
the hour and the minutes, such as **12:30**, rather than **12.30**

1:30	12:30	20:15	22:45

Excel can cope with either the 24hour system or the am/pm system.
To use the am/pm system you must enter the **am** or **pm** after the time.
You must leave a space between the number and the text.

1:30 AM	1:30 PM	10:15 AM	10:15 PM

Finding the difference between two times

You can subtract two time values to find the length of time between.

Start	End	Duration	
1:30	2:30	1:00	=D24-C24
8:00	17:00	9:00	=D25-C25
8:00 AM	5:00 PM	9:00 AM	If the result is not shown correctly,
			You may need to reformat the answer.
			Look at the section about formatting
			further in this worksheet.

Adding time

You can add time to find a total time.
This works well until the total time goes above 24 hours.
For totals greater than 24 hours you may need to apply some special formatting.

Start	End	Duration
1:30	2:30	1:00
8:00	17:00	9:00
7:30 AM	5:45 PM	10:15
		20:15

Formatting time

When time is added together the result may go beyond 24 hours.
Usually this gives an incorrect result, as in the example below.
To correct this error, the result needs to be formatted with a Custom format.

Example 1 : Incorrect formatting

Start	End	Duration	
7:00	18:30	11:30	
8:00	17:00	9:00	
7:30	17:45	10:15	
	Total	6:45	=SUM(E49:E51)

Example 2 : Correct formatting

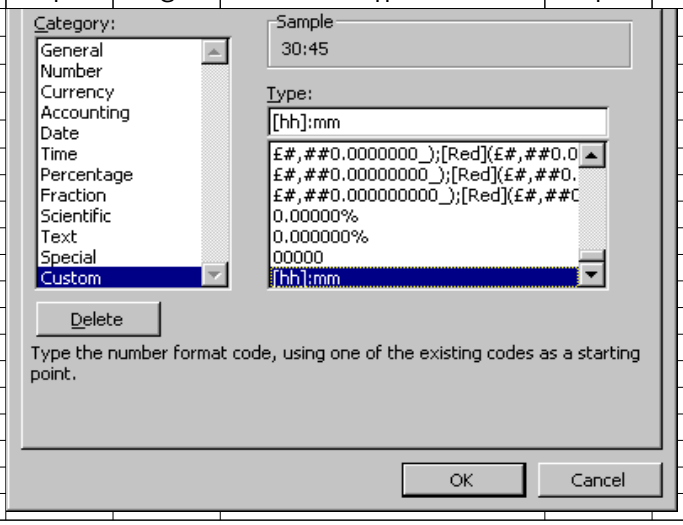
Start	End	Duration	
7:00	18:30	11:30	
8:00	17:00	9:00	
7:30	17:45	10:15	
	Total	30:45	=SUM(E56:E58)

How To Apply Custom Formatting

The custom format for time use a pair of square brackets [hh] on either side
of the hours indicators.

1. Click on the cell which needs the format.
2. Choose the **Format** menu.
3. Choose **Cells**.

	A	B	C	D	E	F	G	H	I	J
69		4. Click the **Number** tag at the top right.								
70		5. Choose **Custom**.								
71		6. Click inside the **Type**: box.								
72		7. Type **[hh]:mm** as the format.								
73		8. Click **OK** to confirm.								
74										
75										
76										
77										
78										
79										
80										
81										
82										
83										
84										
85										
86										
87										



TimeSheet for Flexi

1		**TimeSheet for Flexi**									
2											
3		Week beginning	Mon 05-Jan-98			Normal Hours	37:30				
4											
5		Day	Arrive	Lunch Out	Lunch In	Depart	Total				
6		Mon 05	8:00	13:00	14:00	17:00	8:00	=(F6-C6)+(E6-D6)			
7		Tue 06	8:45	12:30	13:30	17:00	7:15				
8		Wed 07	9:00	13:00	14:00	18:00	8:00				
9		Thu 08	8:30	13:00	14:00	17:00	7:30				
10		Fri 09	8:00	12:00	13:00	17:00	8:00				
11						Total Hours	38:45	=SUM(G6:G10)			
12											
13					Under worked by		-	=IF(G3-G11>0,G3-G11, "-")			
14					Over worked by		1:15	=IF(G3-G11<0,ABS(G3-G11),"-")			
15											
16		**This is simple example of a timesheet.**									
17											
18		**Instructions :**									
19		Type the week start date in cell C3, the Week beginning.									
20		Use the format dd/mm/yy, the name of the day will appear automatically.									
21		The date is then passed down to the Day column.									
22											
23		Type the amount of hours you are expected to work in G3, the Normal Hours.									
24		This is used later to calculate if have worked over or under the required hours.									
25											
26		Type the times you arrive and leave work in the appropriate columns.									
27		Use the format of hh:mm.									
28											
29		**Note**									
30		The Total Hours cell has been formatted as [hh]:mm.									
31		This ensures the total hours can be expressed as a value above 24 hours.									
32		If the [hh]:mm format had not been used the Total Hours would show as :					14:45				
33		If the [hh]:mm format does not show in the cell format dialog box									
34		on your computer, it can be created using Format, Cells, Number, Custom.									

Split Forename and Surname

The following formula are useful when you have one cell containing text which needs to be split up.
One of the most common examples of this is when a persons Forename and Surname are entered in full into a cell.

The formula use various text functions to accomplish the task.
Each of the techniques uses the space between the names to identify where to split.

Finding the First Name

Full Name	First Name	
Alan Jones	Alan	=LEFT(C14,FIND(" ",C14,1))
Bob Smith	Bob	=LEFT(C15,FIND(" ",C15,1))
Carol Williams	Carol	=LEFT(C16,FIND(" ",C16,1))

Finding the Last Name

Full Name	Last Name	
Alan Jones	Jones	=RIGHT(C22,LEN(C22)-FIND(" ",C22))
Bob Smith	Smith	=RIGHT(C23,LEN(C23)-FIND(" ",C23))
Carol Williams	Williams	=RIGHT(C24,LEN(C24)-FIND(" ",C24))

Finding the Last name when a Middle name is present

The formula above cannot handle any more than two names.
If there is also a middle name, the last name formula will be incorrect.
To solve the problem you have to use a much longer calculation.

Full Name	Last Name
Alan David Jones	Jones
Bob John Smith	Smith
Carol Susan Williams	Williams

```
=RIGHT(C37,LEN(C37)-FIND("#",SUBSTITUTE(C37," ","#",LEN(C37)-LEN(SUBSTITUTE(C37," ",""))))
```

Finding the Middle name

Full Name	Middle Name
Alan David Jones	David
Bob John Smith	John
Carol Susan Williams	Susan

```
=LEFT(RIGHT(C45,LEN(C45)-FIND(" ",C45,1)),FIND(" ",RIGHT(C45,LEN(C45)-FIND(" ",C45,1)),1))
```

Percentages

There are no specific functions for calculating percentages.
You have to use the skills you were taught in your maths class at school!

Finding a percentage of a value

Initial value	120
% to find	25%
Percentage value	30

=D8*D9

Example 1

A company is about to give its staff a pay rise.
The wages department need to calculate the increases.
Staff on different grades get different pay rises.

Grade	% Rise
A	10%
B	15%
C	20%

Name	Grade	Old Salary	Increase	
Alan	A	£10,000	£1,000	=E23*LOOKUP(D23,$C$18:$C$20,$D$18:$D$20)
Bob	B	£20,000	£3,000	=E24*LOOKUP(D24,$C$18:$C$20,$D$18:$D$20)
Carol	C	£30,000	£6,000	=E25*LOOKUP(D25,$C$18:$C$20,$D$18:$D$20)
David	B	£25,000	£3,750	=E26*LOOKUP(D26,$C$18:$C$20,$D$18:$D$20)
Elaine	C	£32,000	£6,400	=E27*LOOKUP(D27,$C$18:$C$20,$D$18:$D$20)
Frank	A	£12,000	£1,200	=E28*LOOKUP(D28,$C$18:$C$20,$D$18:$D$20)

Finding a percentage increase

Initial value	120
% increase	25%
Increased value	150

=D33*D34+D33

Example 2

A company is about to give its staff a pay rise.
The wages department need to calculate the new salary including the % increase.
Staff on different grades get different pay rises.

Grade	% Rise
A	10%
B	15%
C	20%

Name	Grade	Old Salary	Increase	
Alan	A	£10,000	£11,000	=E48*LOOKUP(D48,$C$18:$C$20,$D$18:$D$20)+E48
Bob	B	£20,000	£23,000	=E49*LOOKUP(D49,$C$18:$C$20,$D$18:$D$20)+E49
Carol	C	£30,000	£36,000	=E50*LOOKUP(D50,$C$18:$C$20,$D$18:$D$20)+E50
David	B	£25,000	£28,750	=E51*LOOKUP(D51,$C$18:$C$20,$D$18:$D$20)+E51
Elaine	C	£32,000	£38,400	=E52*LOOKUP(D52,$C$18:$C$20,$D$18:$D$20)+E52
Frank	A	£12,000	£13,200	=E53*LOOKUP(D53,$C$18:$C$20,$D$18:$D$20)+E53

Finding one value as percentage of another

Value A	120
Value B	60
A as % of B	50%

=D59/D58

You will need to format the result as % by using the % button
on the toolbar.

Example 3

	A	B	C	D	E	F	G	H	I	J	K
66			An manager has been asked to submit budget requirements for next year.								
67			The manger needs to specify what will be required each quarter.								
68			The manager knows what has been spent by each region in the previous year.								
69			By analysing the past years spending, the manager hopes to predict								
70			what will need to be spent in the next year.								
71											
72			**Last years figures**								
73			Region	Q1	Q2	Q3	Q4				
74			North	9,000	2,000	9,000	7,000				
75			South	7,000	4,000	9,000	5,000				
76			East	2,000	8,000	7,000	3,000				
77			West	8,000	9,000	6,000	5,000	Total			
78			Total	26,000	23,000	31,000	20,000	100,000			
79											
80			**Last years Quarters as % of last years Total**								
81			Region	Q1	Q2	Q3	Q4				
82			North	9%	2%	9%	7%	=G74/$H$78			
83			South	7%	4%	9%	5%	=G75/$H$78			
84			East	2%	8%	7%	3%	=G76/$H$78			
85			West	8%	9%	6%	5%	=G77/$H$78			
86			Total	26%	23%	31%	20%	=G78/$H$78			
87											
88			**Next years budget**		150,000						
89			**Next years estimated budget requirements**								
90			Region	Q1	Q2	Q3	Q4				
91			North	13,500	3,000	13,500	10,500	=G82*$E$88			
92			South	10,500	6,000	13,500	7,500	=G83*$E$88			
93			East	3,000	12,000	10,500	4,500	=G84*$E$88			
94			West	12,000	13,500	9,000	7,500	Total			
95			Total	39,000	34,500	46,500	30,000	150,000			
96											
97											
98		**Finding an original value after an increase has been applied**									
99											
100			Increased value	150							
101			% increase	25%							
102			Original value	120	=D100/(100%+D101)						
103											
104			**Example 4**								
105			An employ has to submit an expenses claim for travelling and accommodation.								
106			The claim needs to show the VAT tax portion of each receipt.								
107			Unfortunately the receipts held by the employee only show the total amount.								
108			The employee needs to split this total to show the original value and the VAT amount.								
109											
110			VAT rate	17.50%							
111											
112			Receipt	Total	Actual Value	Vat Value					
113			Petrol	£10.00	£8.51	£1.49	=D113-D113/(100%+$D$110)				
114			Hotel	£235.00	£200.00	£35.00					
115			Petrol	£117.50	£100.00	£17.50					
116				=D115/(100%+$D$110)							

Show all formula

You can view all the formula on the worksheet by pressing **Ctrl** and `.

The ' is the left single quote usually found on the key to left of number 1.

Press **Ctrl** and ` to see the formula below. *(The screen may look a bit odd.)*

Press the same combination to see the original view.

10	20	30
30	40	70
50	60	60
70	80	30

SUM using names

You can use the names typed at the top of columns or side of rows in calculations simply by typing the name into the formula.

Try this example:

Go to cell **C16** and then enter the formula **=SUM(jan)**

The result will show.

This formula can be copied to **D16** and **E16**, and the names change to **Feb** and **Mar**.

	Jan	Feb	Mar
North	45	50	50
South	30	25	35
East	35	10	50
West	20	50	5
Total			

If it does not work !

The feature may have been switched off on your computer.

You can switch it on by using **Tools**, **Options**, **Calculation**, **Accept Labels in Formula**.

Instant Charts

You can create a chart quickly without having to use the chart button on the toolbar by pressing the function key F11 whilst inside a range of data.

	Jan	Feb	Mar
North	45	50	50
South	30	25	35
East	35	10	50
West	20	50	5

Click anywhere inside the table above.
Then press F11.

Filename formula

There may be times when you need to insert the name of the current workbook or worksheet in to a cell.

This can be done by using the CELL() function, shown below.

'file:///var/www/apps/scribd/scribd/tmp/scratch6/9568157.xls'#$_Filename formula

```
=CELL("filename")
```

The problem with this is that it gives the complete path including drive letter and folders.

To just pick out the workbook or worksheet name you need to use text functions.

To pick the Path.

#VALUE!

```
=MID(CELL("filename"),1,FIND("[",CELL("filename"))-1)
```

To pick the Workbook name.

#VALUE!

```
=MID(CELL("filename"),FIND("[",CELL("filename"))+1,FIND("]",CELL("filename"))-FIND("[",CELL("filename"))-1)
```

To pick the Worksheet name.

#VALUE!

```
=MID(CELL("filename"),FIND("]",CELL("filename"))+1,255)
```

Brackets in formula

Sometimes you will need to use brackets, (also known as 'braces'), in formula.
This is to ensure that the calculations are performed in the order that you need.
The need for brackets occurs when you mix plus or minus with divide or multiply.

Mathematically speaking the * and / are more important than + and - .
The * and / operations will be calculated before + and - .

Example 1 : The wrong answer !

10	
20	
2	
50	=C12+C13*C14

You may expect that 10 + 20 would equal 30
And then 30 * 2 would equal 60

But because the * is calculated first Excel sees the
calculation as 20 * 2 resulting in 40
And then 10 + 40 resulting in 50

Example 2 : The correct answer.

10	
20	
2	
60	=(C27+C28)*C29

By placing brackets around (10+20) Excel performs this
part of the calulation first, resulting in 30
Then the 30 is multipled by 2 resulting in 60

	A	B	C	D	E	F	G	H	I
1		**Age Calculation**							
2									
3		You can calculate a persons age based on their birthday and todays date.							
4		The calculation uses the DATEDIF() function.							
5		The DATEDIF() is not documented in Excel 5, 7 or 97, but it is in 2000.							
6		(Makes you wonder what else Microsoft forgot to tell us!)							
7									
8		Birth date :	1-Jan-60						
9									
10		Years lived :	#NAME?	=DATEDIF(C8,TODAY(),"y")					
11		and the months :	#NAME?	=DATEDIF(C8,TODAY(),"ym")					
12		and the days :	#NAME?	=DATEDIF(C8,TODAY(),"md")					
13									
14		You can put this all together in one calculation, which creates a text version.							
15		#NAME?							
16		="Age is "&DATEDIF(C8,TODAY(),"y")&" Years, "&DATEDIF(C8,TODAY(),"ym")&" Months and "&DATEDIF(C8,TODAY(),"md")&" Days"							
17									
18									
19		**Another way to calculate age**							
20		This method gives you an age which may potentially have decimal places representing the months.							
21		If the age is 20.5, the .5 represents 6 months.							
22									
23		Birth date :	1-Jan-60						
24									
25		Age is :	48.81	=(TODAY()-C23)/365.25					

AutoSum Shortcut Key

Instead of using the AutoSum button from the toolbar,
you can press **Alt** and **=** to achieve the same result.

Try it here :

Move to a blank cell in the Total row or column, then press **Alt** and **=**.

or

Select a row, column or all cells and then press **Alt** and **=**.

	Jan	Feb	Mar	Total
North	10	50	90	
South	20	60	100	
East	30	70	200	
West	40	80	300	
Total				

ABS

Number	Absolute Value	
10	10	=ABS(C4)
-10	10	=ABS(C5)
1.25	1.25	=ABS(C6)
-1.25	1.25	=ABS(C7)

What Does it Do ?

This function calculates the value of a number, irrespective of whether it is positive or negative.

Syntax

=ABS(CellAddress or Number)

Formatting

The result will be shown as a number, no special formatting is needed.

Example

The following table was used by a company testing a machine which cuts timber.
The machine needs to cut timber to an exact length.
Three pieces of timber were cut and then measured.
In calculating the difference between the Required Length and the Actual Length it does not matter if the wood was cut too long or short, the measurement needs to be expressed as an absolute value.

Table 1 shows the original calculations.
The Difference for Test 3 is shown as negative, which has a knock on effect when the Error Percentage is calculated.
Whether the wood was too long or short, the percentage should still be expressed as an absolute value.

Table 1

Test Cut	Required Length	Actual Length	Difference	Error Percentage
Test 1	120	120	0	0%
Test 2	120	90	30	25%
Test 3	120	150	-30	-25%
			=D36-E36	

Table 2 shows the same data but using the =ABS() function to correct the calculations.

Table 2

Test Cut	Required Length	Actual Length	Difference	Error Percentage
Test 1	120	120	0	0%
Test 2	120	90	30	25%
Test 3	120	150	30	25%
		=ABS(D45-E45)		

ADDRESS

Type a column number :	2	
Type a row number :	3	
Type a sheet name :	Hello	

$B$3	=ADDRESS(F4,F3,1,TRUE)
B$3	=ADDRESS(F4,F3,2,TRUE)
$B3	=ADDRESS(F4,F3,3,TRUE)
B3	=ADDRESS(F4,F3,4,TRUE)
R3C2	=ADDRESS(F4,F3,1,FALSE)
R3C[2]	=ADDRESS(F4,F3,2,FALSE)
R[3]C2	=ADDRESS(F4,F3,3,FALSE)
R[3]C[2]	=ADDRESS(F4,F3,4,FALSE)
Hello!$B$3	=ADDRESS(F4,F3,1,TRUE,F5)
Hello!B$3	=ADDRESS(F4,F3,2,TRUE,F5)
Hello!$B3	=ADDRESS(F4,F3,3,TRUE,F5)
Hello!B3	=ADDRESS(F4,F3,4,TRUE,F5)

What Does It Do ?

This function creates a cell reference as a piece of text, based on a row and column numbers given by the user.

This type of function is used in macros rather than on the actual worksheet.

Syntax

=ADDRESS(RowNumber,ColNumber,Absolute,A1orR1C1,SheetName)

The RowNumber is the normal row number from 1 to 16384.

The ColNumber is from 1 to 256, cols A to IV.

The Absolute can be 1,2,3 or 4.

- When 1 the reference will be in the form $A$1, column and row absolute.
- When 2 the reference will be in the form A$1, only the row absolute.
- When 3 the reference will be in the form $A1, only the column absolute.
- When 4 the reference will be in the form A1, neither col or row absolute.

The A1orR1C1 is either TRUE of FALSE.

- When TRUE the reference will be in the form A1, the normal style for cell addresses.
- When FALSE the reference will be in the form R1C1, the alternative style of cell address.

The SheetName is a piece of text to be used as the worksheet name in the reference.

- The SheetName does not actually have to exist.

AND

Items To Test		Result	
500	800	TRUE	=AND(C4>=100,D4>=100)
500	25	FALSE	=AND(C5>=100,D5>=100)
25	500	FALSE	=AND(C6>=100,D6>=100)
	12	TRUE	=AND(D7>=1,D7<=52)

What Does It Do?

This function tests two or more conditions to see if they are all true.
It can be used to test that a series of numbers meet certain conditions.
It can be used to test that a number or a date falls between an upper and lower limit.
Normally the AND() function would be used in conjunction with a function such as =IF().

Syntax

=AND(Test1,Test2)
Note that there can be up to 30 possible tests.

Formatting

When used by itself it will show TRUE or FALSE.

Example 1

The following example shows a list of examination results.
The teacher wants to find the pupils who scored above average in all three exams.
The =AND() function has been used to test that each score is above the average.
The result of TRUE is shown for pupils who have scored above average in all three exams.

Name	Maths	English	Physics	Passed
Alan	80	75	85	TRUE
Bob	50	30	40	FALSE
Carol	60	70	50	FALSE
David	90	85	95	TRUE
Eric	20	30	Absent	FALSE
Fred	40	60	80	FALSE
Gail	10	90	80	FALSE
Harry	80	70	60	TRUE
Ian	30	10	20	FALSE
Janice	10	20	30	FALSE

=AND(C38>=AVERAGE($C$29:$C$38),D38>=AVERAGE($D$29:$D$38),E38>=AVERAGE($E$29:$E$38))

Averages	47	54	60

AREAS

	Name	Age
Pink	Alan	18
	Bob	17
	Carol	20

	Name	Age
Green	David	20
	Eric	16
	Fred	19

2 =AREAS(PeopleLists)

What Does It Do?

This function tests a range to determine whether it is a single block of data, or whether it is a multiple selection.
If it is a single block the result will be 1.
If it is a multiple block the result will be the number of ranges selected.
The function is designed to be used in macros.

Syntax

=AREAS(RangeToTest)

Formatting

The result will be shown as a number.

Example

The example at the top of this page shows two ranges coloured pink and green.
These ranges have been given the name PeopleLists.
The =AREAS(PeopleLists) gives a result of 2 indicating that there are two separate selections which form the PeopleLists range.

Note

To name multiple ranges the CTRL key must be used.
In the above example the pink range was selected as normal, then the Ctrl key was held down before selecting the green range.
When a Range Name is created it will consider both Pink and Green as being one range.

AVERAGE

	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Average	
Temp	30	31	32	29	26	28	27	29	=AVERAGE(D4:J4)
Rain	0	0	0	4	6	3	1	2	=AVERAGE(D5:J5)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Average	
Temp	30		32	29	26	28	27	28.67	=AVERAGE(D8:J8)
Rain	0		0	4	6	3	1	2.33	=AVERAGE(D9:J9)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Average	
Temp	30	No	32	29	26	28	27	28.67	=AVERAGE(D12:J12)
Rain	0	Reading	0	4	6	3	1	2.33	=AVERAGE(D13:J13)

What Does It Do ?

This function calculates the average from a list of numbers.
If the cell is blank or contains text, the cell will not be used in the average calculation.
If the cell contains zero 0, the cell will be included in the average calculation.

Syntax

=AVERAGE(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Note

To calculate the average of cells which contain text or blanks use =SUM() to get the total and then divide by the count of the entries using =COUNTA().

	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Average	
Temp	30	No	32	29	26	28	27	24.57	=SUM(D31:J31)/COUNTA(D31:J31)
Rain	0	Reading	0	4	6	3	1	2	=SUM(D32:J32)/COUNTA(D32:J32)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Average	
Temp	30		32	29	26	28	27	28.67	=SUM(D35:J35)/COUNTA(D35:J35)
Rain	0		0	4	6	3	1	2.33	=SUM(D36:J36)/COUNTA(D36:J36)

Further Usage

BIN2DEC

Binary Number	Decimal Equivalent	
0	0	=BIN2DEC(C4)
1	1	=BIN2DEC(C5)
10	2	=BIN2DEC(C6)
11	3	=BIN2DEC(C7)
111111111	511	=BIN2DEC(C8)
1111111111	-1	=BIN2DEC(C9)
1111111110	-2	=BIN2DEC(C10)
1111111101	-3	=BIN2DEC(C11)
1000000000	-512	=BIN2DEC(C12)
11111111111	Err:502	=BIN2DEC(C13)

What Does It Do ?

This function converts a binary number to decimal.
Negative numbers are represented using two's-complement notation.

Syntax

=BIN2DEC(BinaryNumber)
The binary number has a limit of ten characters.

Formatting

No special formatting is needed.

CEILING

Number	Raised Up	
2.1	3	=CEILING(C4,1)
1.5	2	=CEILING(C5,1)
1.9	2	=CEILING(C6,1)
20	30	=CEILING(C7,30)
25	30	=CEILING(C8,30)
40	60	=CEILING(C9,30)

What Does It Do ?

This function rounds a number up to the nearest multiple specified by the user.

Syntax

=CEILING(ValueToRound,MultipleToRoundUpTo)

The ValueToRound can be a cell address or a calculation.

Formatting

No special formatting is needed.

Example 1

The following table was used by a estate agent renting holiday apartments.
The properties being rented are only available on a weekly basis.
When the customer supplies the number of days required in the property the =CEILING()
function rounds it up by a multiple of 7 to calculate the number of full weeks to be billed.

	Days Required	Days To Be Billed	
Customer 1	3	7	=CEILING(D28,7)
Customer 2	4	7	=CEILING(D29,7)
Customer 3	10	14	=CEILING(D30,7)

Example 2

The following table was used by a builders merchant delivering products to a construction site.
The merchant needs to hire trucks to move each product.
Each product needs a particular type of truck of a fixed capacity.

Table 1 calculates the number of trucks required by dividing the Units To Be Moved by
the Capacity of the truck.
This results of the division are not whole numbers, and the builder cannot hire just part
of a truck.

Table 1

Item	Units To Be Moved	Truck Capacity	Trucks Needed	
Bricks	1000	300	3.33	=D45/E45
Wood	5000	600	8.33	=D46/E46
Cement	2000	350	5.71	=D47/E47

Table 2 shows how the =CEILING() function has been used to round up the result of
the division to a whole number, and thus given the exact amount of trucks needed.

Table 2

	A	B	C	D	E	F	G	H
53			Item	Units To Be Moved	Truck Capacity	Trucks Needed		
54			Bricks	1000	300	4	=CEILING(D54/E54,1)	
55			Wood	5000	600	9	=CEILING(D55/E55,1)	
56			Cement	2000	350	6	=CEILING(D56/E56,1)	
57								
58								
59		**Example 3**						
60		The following tables were used by a shopkeeper to calculate the selling price of an item.						
61		The shopkeeper buys products by the box.						
62		The cost of the item is calculated by dividing the Box Cost by the Box Quantity.						
63		The shopkeeper always wants the price to end in 99 pence.						
64								
65		Table 1 shows how just a normal division results in varying Item Costs.						
66								
67		Table 1						
68		Item	Box Qnty	Box Cost	Cost Per Item			
69		Plugs	11	£20	1.81818	=D69/C69		
70		Sockets	7	£18.25	2.60714	=D70/C70		
71		Junctions	5	£28.10	5.62000	=D71/C71		
72		Adapters	16	£28	1.75000	=D72/C72		
73								
74								
75		Table 2 shows how the =CEILING() function has been used to raise the Item Cost to						
76		always end in 99 pence.						
77								
78		Table 2						
79		Item	In Box	Box Cost	Cost Per Item	Raised Cost		
80		Plugs	11	£20	1.81818	1.99		
81		Sockets	7	£18.25	2.60714	2.99		
82		Junctions	5	£28.10	5.62000	5.99		
83		Adapters	16	£28	1.75000	1.99		
84						=INT(E83)+CEILING(MOD(E83,1),0.99)		
85								
86		**Explanation**						
87		=INT(E83)			Calculates the integer part of the price.			
88		=MOD(E83,1)			Calculates the decimal part of the price.			
89		=CEILING(MOD(E83),0.99)			Raises the decimal to 0.99			

CELL

This is the cell and contents to test.	17.50%
The cell address.	$D$3 =CELL("address",D3)
The column number.	4 =CELL("col",D3)
The row number.	3 =CELL("row",D3)
The actual contents of the cell.	0.18 =CELL("contents",D3)
The type of entry in the cell. Shown as **b** for blank, **l** for text, **v** for value.	v =CELL("type",D3)
The alignment of the cell. Shown as **'** for left, **^** for centre, **"** for right. Nothing is shown for numeric entries.	=CELL("prefix",D3)
The width of the cell.	12 =CELL("width",D3)
The number format fo the cell. (See the table shown below)	P2 =CELL("format",D3)
Formatted for braces ( ) on positive values. **1** for yes, **0** for no.	0 =CELL("parentheses",D3)
Formatted for coloured negatives. **1** for yes, **0** for no.	0 =CELL("color",D3)
The type of cell protection. **1** for a locked, **0** for unlocked.	1 =CELL("protect",D3)
The filename containing the cell.	'file:///var/www/apps/scribd/scribd/tmp/scratch6/9568157.xls'#$CELL =CELL("filename",D3)

What Does It Do ?

This function examines a cell and displays information about the contents, position and formatting.

Syntax

=CELL("TypeOfInfoRequired",CellToTest)

The TypeOfInfoRequired is a text entry which must be surrounded with quotes " ".

Formatting

No special formatting is needed.

Codes used to show the formatting of the cell.

Numeric Format	Code
General	G
0	F0
#,##0	,0
0.00	F2
#,##0.00	,2
$#,##0_);($#,##0)	C0
$#,##0_);[Red]($#,##0)	C0-
$#,##0.00_);($#,##0.00)	C2
$#,##0.00_);[Red]($#,##0.00)	C2-
0%	P0
0.00%	P2
0.00E+00	S2
# ?/? or # ??/??	G
m/d/yy or m/d/yy h:mm or mm/dd/yy.	D4
d-mmm-yy or dd-mmm-yy	D1
d-mmm or dd-mmm	D2
mmm-yy	D3
mm/dd	D5
h:mm AM/PM	D7
h:mm:ss AM/PM	D6
h:mm	D9
h:mm:ss	D8

Example

The following example uses the =CELL() function as part of a formula which extracts the filename.

The name of the current file is : #VALUE!

```
=MID(CELL("filename"),FIND("[",CELL("filename"))+1,FIND("]",CELL("filename"))-FIND("[",CELL("filename"))-1)
```

CHAR

ANSI Number	Character	
65	A	=CHAR(G4)
66	B	=CHAR(G5)
169	©	=CHAR(G6)

What Does It Do?

This function converts a normal number to the character it represent in the ANSI character set used by Windows.

Syntax

=CHAR(Number)

The Number must be between 1 and 255.

Formatting

The result will be a character with no special formatting.

Example

The following is a list of all 255 numbers and the characters they represent.
Note that most Windows based program may not display some of the special characters, these will be displayed as a small box.

1		26		51	3	76	L	101	e	126	~	151	–	176	∞	201	É	226	â	251	û
2		27		52	4	77	M	102	f	127	□	152	˜	177	±	202	Ê	227	ã	252	ü
3		28		53	5	78	N	103	g	128	Ä	153	ô	178	≤	203	À	228	‰	253	ý
4		29		54	6	79	O	104	h	129	Å	154	ö	179	≥	204	Ã	229	Â	254	þ
5		30		55	7	80	P	105	i	130	Ç	155	õ	180	¥	205	Í	230	Ê	255	ÿ
6		31		56	8	81	Q	106	j	131	É	156	ú	181	µ	206	Œ	231	ç		
7		32		57	9	82	R	107	k	132	Ñ	157	ù	182	∂	207	œ	232	è		
8		33	!	58	:	83	S	108	l	133	Ö	158	û	183	∑	208	–	233	é		
9		34	"	59	;	84	T	109	m	134	Ü	159	ü	184	∏	209	—	234	ê		
10		35	#	60	<	85	U	110	n	135	á	160	†	185	π	210	“	235	ë		
11		36	$	61	=	86	V	111	o	136	à	161	°	186	°	211	Ó	236	ì		
12		37	%	62	>	87	W	112	p	137	â	162	¢	187	»	212	‘	237	í		
13		38	&	63	?	88	X	113	q	138	ä	163	£	188	º	213	’	238	î		
14		39	'	64	@	89	Y	114	r	139	ã	164	§	189	Ω	214	÷	239	ï		
15		40	(	65	A	90	Z	115	s	140	å	165	•	190	æ	215	◊	240	ð		
16		41	)	66	B	91	[	116	t	141	ç	166	¶	191	¿	216	ÿ	241	ñ		
17		42	*	67	C	92	\	117	u	142	é	167	ß	192	¿	217	Ÿ	242	ò		
18		43	+	68	D	93	]	118	v	143	è	168	®	193	Á	218	⁄	243	ó		
19		44	,	69	E	94	^	119	w	144	ê	169	©	194	Â	219	€	244	ô		
20		45	-	70	F	95	_	120	x	145	ë	170	™	195	Ã	220	‹	245	õ		
21		46	.	71	G	96	`	121	y	146	í	171	«	196	ƒ	221	›	246	ö		
22		47	/	72	H	97	a	122	z	147	ì	172	¨	197	≈	222	fi	247	÷		
23		48	0	73	I	98	b	123	{	148	î	173	≠	198	Δ	223	fl	248	ø		
24		49	1	74	J	99	c	124	|	149	ï	174	®	199	Ç	224	à	249	ù		
25		50	2	75	K	100	d	125	}	150	ñ	175	Ø	200	»	225	á	250	ú		

Note

Number 32 does not show as it is the SPACEBAR character.

CHOOSE

Index Value	Result	
1	Alan	=CHOOSE(C4,"Alan","Bob","Carol")
3	Carol	=CHOOSE(C5,"Alan","Bob","Carol")
2	Bob	=CHOOSE(C6,"Alan","Bob","Carol")
3	18%	=CHOOSE(C7,10%,15%,18%)
1	10%	=CHOOSE(C8,10%,15%,18%)
2	15%	=CHOOSE(C9,10%,15%,18%)

What Does It Do?

This function picks from a list of options based upon an Index value given to by the user.

Syntax

=CHOOSE(UserValue, Item1, Item2, Item3 through to Item29)

Formatting

No special formatting is required.

Example

The following table was used to calculate the medals for athletes taking part in a race.
The Time for each athlete is entered.
The =RANK() function calculates the finishing position of each athlete.
The =CHOOSE() then allocates the correct medal.
The =IF() has been used to filter out any positions above 3, as this would cause
the error of #VALUE to appear, due to the fact the =CHOOSE() has only three items in it.

Name	Time	Position	Medal	
Alan	1:30	2	Silver	=IF(D30<=3,CHOOSE(D30,"Gold","Silver","Bronze"),"unplaced")
Bob	1:15	4	unplaced	=IF(D31<=3,CHOOSE(D31,"Gold","Silver","Bronze"),"unplaced")
Carol	2:45	1	Gold	=IF(D32<=3,CHOOSE(D32,"Gold","Silver","Bronze"),"unplaced")
David	1:05	5	unplaced	=IF(D33<=3,CHOOSE(D33,"Gold","Silver","Bronze"),"unplaced")
Eric	1:20	3	Bronze	=IF(D34<=3,CHOOSE(D34,"Gold","Silver","Bronze"),"unplaced")
		=RANK(C34,C30:C34)		

CLEAN

Dirty Text	Clean Text	
Hello	Hello	=CLEAN(C4)
Hello	Hello	=CLEAN(C5)
Hello	Hello	=CLEAN(C6)

What Does It Do?

This function removes any nonprintable characters from text.
These nonprinting characters are often found in data which has been imported from other systems such as database imports from mainframes.

Syntax

=CLEAN(TextToBeCleaned)

Formatting

No special formatting is needed. The result will show as normal text.

CODE

Letter	ANSI Code	
A	65	=CODE(C4)
B	66	=CODE(C5)
C	67	=CODE(C6)
a	97	=CODE(C7)
b	98	=CODE(C8)
c	99	=CODE(C9)
Alan	65	=CODE(C10)
Bob	66	=CODE(C11)
Carol	67	=CODE(C12)

What Does It Do?

This function shows the ANSI value of a single character, or the first character in a piece of text.
The ANSI character set is used by Windows to identify each keyboard character by using a unique number.
There are 255 characters in the ANSI set.

Syntax

=CODE(Text)

Formatting

No special formatting is needed, the result will be shown as a number between 1 and 255.

Example

See the example for FREQUENCY.

1		26		51	3	76	L	101	e	126	~	151	—	176	∞	201	Ë	226	â	251	û
2		27		52	4	77	M	102	f	127	□	152	˜	177	±	202	Ê	227	ã	252	ü
3		28		53	5	78	N	103	g	128	€	153	™	178	≤	203	À	228	‰	253	ý
4		29		54	6	79	O	104	h	129	Å	154	š	179	≥	204	Ã	229	Â	254	þ
5		30		55	7	80	P	105	i	130	,	155	›	180	¥	205	Í	230	Ê	255	ÿ
6		31		56	8	81	Q	106	j	131	ƒ	156	œ	181	µ	206	Œ	231	ç		
7		32		57	9	82	R	107	k	132	„	157	ù	182	∂	207	œ	232	è		
8		33	!	58	:	83	S	108	l	133	…	158	ž	183	∑	208	–	233	é		
9		34	"	59	;	84	T	109	m	134	†	159	Ÿ	184	∏	209	—	234	ê		
10		35	#	60	<	85	U	110	n	135	‡	160	†	185	π	210	“	235	ë		
11		36	$	61	=	86	V	111	o	136	ˆ	161	°	186	º	211	Ó	236	ì		
12		37	%	62	>	87	W	112	p	137	‰	162	¢	187	»	212	‘	237	í		
13		38	&	63	?	88	X	113	q	138	Š	163	£	188	º	213	’	238	î		
14		39	'	64	@	89	Y	114	r	139	‹	164	§	189	Ω	214	÷	239	ï		
15		40	(	65	A	90	Z	115	s	140	Œ	165	•	190	æ	215	◊	240	ð		
16		41	)	66	B	91	[	116	t	141	ç	166	¶	191	¿	216	ÿ	241	ñ		
17		42	*	67	C	92	\	117	u	142	Ž	167	ß	192	¿	217	Ÿ	242	ò		
18		43	+	68	D	93	]	118	v	143	è	168	®	193	Á	218	⁄	243	ó		
19		44	,	69	E	94	^	119	w	144	ê	169	©	194	Â	219	€	244	ô		
20		45	-	70	F	95	_	120	x	145	‘	170	™	195	Ã	220	‹	245	õ		
21		46	.	71	G	96	`	121	y	146	’	171	«	196	ƒ	221	›	246	ö		
22		47	/	72	H	97	a	122	z	147	“	172	¨	197	≈	222	fi	247	÷		
23		48	0	73	I	98	b	123	{	148	”	173	≠	198	Δ	223	fl	248	ø		
24		49	1	74	J	99	c	124	|	149	•	174	®	199	Ç	224	à	249	ù		
25		50	2	75	K	100	d	125	}	150	–	175	Ø	200	»	225	á	250	ú		

COMBIN

Pool Of Items	Items In A Group	Possible Groups	
4	2	6	=COMBIN(C4,D4)
4	3	4	=COMBIN(C5,D5)
26	2	325	=COMBIN(C6,D6)

What Does It Do ?

This function calculates the highest number of combinations available based upon a fixed number of items.

The internal order of the combination does not matter, so AB is the same as BA.

Syntax

=COMBIN(HowManyItems,GroupSize)

Formatting

No special formatting is required.

Example 1

This example calculates the possible number of pairs of letters available from the four characters ABCD.

Total Characters	Group Size	Combinations	
4	2	6	=COMBIN(C25,D25)

The proof !	The four letters :	ABCD
	Pair 1	AB
	Pair 2	AC
	Pair 3	AD
	Pair 4	BC
	Pair 5	BD
	Pair 6	CD

Example 2

A decorator is asked to design a colour scheme for a new office.

The decorator is given five colours to work with, but can only use three in any scheme.

How many colours schemes can be created ?

Available Colours	Colours Per Scheme	Totals Schemes	
5	3	10	=COMBIN(C41,D41)

The colours

Red
Green
Blue
Yellow
Black

Scheme 1	Scheme 2	Scheme 3	Scheme 4	Scheme 5
Red	Red	Red	Red	Red
Green	Green	Green	Blue	Blue
Blue	Yellow	Black	Yellow	Black

Scheme 6	Scheme 7	Scheme 8	Scheme 9	Scheme 10
Green	Green	Green	Blue	??????
Blue	Blue	Yellow	Yellow	
Yellow	Black	Black	Black	

CONCATENATE

Name 1	Name 2	Concatenated Text	
Alan	Jones	AlanJones	=CONCATENATE(C4,D4)
Bob	Williams	BobWilliams	=CONCATENATE(C5,D5)
Carol	Davies	CarolDavies	=CONCATENATE(C6,D6)
Alan	Jones	Alan Jones	=CONCATENATE(C7," ",D7)
Bob	Williams	Williams, Bob	=CONCATENATE(D8,", ",C8)
Carol	Davies	Davies, Carol	=CONCATENATE(D9,", ",C9)

What Does It Do?

This function joins separate pieces of text into one item.

Syntax

=CONCATENATE(Text1,Text2,Text3...Text30)

Up to thirty pieces of text can be joined.

Formatting

No special formatting is needed, the result will be shown as normal text.

Note

You can achieve the same result by using the **&** operator.

Name 1	Name 2	Concatenated Text	
Alan	Jones	AlanJones	=C25&D25
Bob	Williams	BobWilliams	=C26&D26
Carol	Davies	CarolDavies	=C27&D27
Alan	Jones	Alan Jones	=C28&" "&D28
Bob	Williams	Williams, Bob	=D29&", "&C29
Carol	Davies	Davies, Carol	=D30&", "&C30

CONVERT

Amount To Convert	Converting From	Converting To	Converted Amount	
1	in	cm	2.54	=CONVERT(C4,D4,E4)
1	ft	m	0.3	=CONVERT(C5,D5,E5)
1	yd	m	0.91	=CONVERT(C6,D6,E6)
1	yr	day	365.25	=CONVERT(C8,D8,E8)
1	day	hr	24	=CONVERT(C9,D9,E9)
1.5	hr	mn	90	=CONVERT(C10,D10,E10)
0.5	mn	sec	30	=CONVERT(C11,D11,E11)

What Does It Do ?

This function converts a value measure in one type of unit, to the same value expressed in a different type of unit, such as Inches to Centimetres.

Syntax

=CONVERT(AmountToConvert,UnitToConvertFrom,UnitToConvertTo)

Formatting

No special formatting is needed.

Example

The following table was used by an Import / Exporting company to convert the weight and size of packages from old style UK measuring system to European system.

	Pounds	Ounces	Kilograms
Weight	5	3	2.35

=CONVERT(D28,"lbm","kg")+CONVERT(E28,"ozm","kg")

	Feet	Inches	Metres
Height	12	6	3.81
Length	8	3	2.51
Width	5	2	1.57

=CONVERT(D34,"ft","m")+CONVERT(E34,"in","m")

Abbreviations

This is a list of all the possible abbreviations which can be used to denote measuring systems.

Weight & Mass	
Gram	g
Kilogram	kg
Slug	sg
Pound mass	lbm
U (atomic mass)	u
Ounce mass	ozm
Time	
Year	yr
Day	day
Hour	hr
Minute	mn
Second	sec

Distance	
Meter	m
Statute mile	mi
Nautical mile	Nmi
Inch	in
Foot	ft
Yard	yd
Angstrom	ang
Pica (1/72 in.)	Pica
Pressure	
Pascal	Pa
Atmosphere	atm
mm of Mercury	mmHg

	A	B	C	D	E	F	G	H
54								
55		**Temperature**			**Liquid**			
56		Degree Celsius	C		Teaspoon	tsp		
57		Degree Fahrenhei	F		Tablespoon	tbs		
58		Degree Kelvin	K		Fluid ounce	oz		
59					Cup	cup		
60		**Force**			Pint	pt		
61		Newton	N		Quart	qt		
62		Dyne	dyn		Gallon	gal		
63		Pound force	lbf		Liter	l		
64								
65		**Energy**			**Power**			
66		Joule	J		Horsepower	HP		
67		Erg	e		Watt	W		
68		Thermodynamic calorie	c					
69		IT calorie	cal		**Magnetism**			
70		Electron volt	eV		Tesla	T		
71		Horsepower-hour	HPh		Gauss	ga		
72		Watt-hour	Wh					
73		Foot-pound	flb					
74		BTU	BTU					
75								
76								
77		These characters can be used as a prefix to access further units of measure.						
78		Using "**c**" as a prefix to meters "**m**" will allow centimetres "**cm**" to be calculated.						
79								
80		Prefix	Multiplier	Abbreviation		Prefix	Multiplier	Abbreviation
81		exa	1.00E+18	E		deci	1.00E-01	d
82		peta	1.00E+15	P		centi	1.00E-02	c
83		tera	1.00E+12	T		milli	1.00E-03	m
84		giga	1.00E+09	G		micro	1.00E-06	u
85		mega	1.00E+06	M		nano	1.00E-09	n
86		kilo	1.00E+03	k		pico	1.00E-12	p
87		hecto	1.00E+02	h		femto	1.00E-15	f
88		dekao	1.00E+01	e		atto	1.00E-18	a

CORREL

Table 1

Month	Avg Temp	Air Cond Sales
Jan	20	100
Feb	30	200
Mar	30	300
Apr	40	200
May	50	400
Jun	50	400
	Correlation	0.864

=CORREL(D5:D10,E5:E10)

Table 2

Advertising Costs	Sales
£2,000	£20,000
£1,000	£30,000
£5,000	£20,000
£1,000	£40,000
£8,000	£40,000
£1,000	£20,000
Correlation	28%

=CORREL(G5:G10,H5:H10)

What Does It Do ?

This function examines two sets of data to determine the degree of relationship between the two sets.
The result will be a decimal between 0 and 1.
The larger the result, the greater the correlation.

In Table 1 the Monthly temperature is compared against the Sales of air conditioning units.
The correlation shows that there is an 0.864 realtionship between the data.

In Table 2 the Cost of advertising has been compared to Sales.
It can be formatted as percentage % to show a more meaning full result.
The correlation shows that there is an 28% realtionship between the data.

Syntax

=CORREL(Range1,Range2)

Formatting

The result will normally be shown in decimal format.

COUNT

Entries To Be Counted			Count	
10	20	30	3	=COUNT(C4:E4)
10	0	30	3	=COUNT(C5:E5)
10	-20	30	3	=COUNT(C6:E6)
10	1-Jan-88	30	3	=COUNT(C7:E7)
10	21:30	30	3	=COUNT(C8:E8)
10	0.81	30	3	=COUNT(C9:E9)
10		30	2	=COUNT(C10:E10)
10	Hello	30	2	=COUNT(C11:E11)
10	#DIV/0!	30	#DIV/0!	=COUNT(C12:E12)

What Does It Do ?

This function counts the number of numeric entries in a list.
It will ignore blanks, text and errors.

Syntax

=COUNT(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Example

The following table was used by a builders merchant to calculate the number of sales for various products in each month.

Item	Jan	Feb	Mar
Bricks	£1,000		
Wood		£5,000	
Glass	£2,000	£1,000	
Metal	£1,000		
Count	3	2	0

=COUNT(D29:D32)

COUNTA

Entries To Be Counted			Count	
10	20	30	3	=COUNTA(C4:E4)
10	0	30	3	=COUNTA(C5:E5)
10	-20	30	3	=COUNTA(C6:E6)
10	1-Jan-88	30	3	=COUNTA(C7:E7)
10	21:30	30	3	=COUNTA(C8:E8)
10	0.41	30	3	=COUNTA(C9:E9)
10		30	2	=COUNTA(C10:E10)
10	Hello	30	3	=COUNTA(C11:E11)
10	#DIV/0!	30	3	=COUNTA(C12:E12)

What Does It Do ?

This function counts the number of numeric or text entries in a list.
It will ignore blanks.

Syntax

=COUNTA(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Example

The following table was used by a school to keep track of the examinations taken by each pupil.
Each exam passed was graded as 1, 2 or 3.
A failure was entered as Fail.

The school needed to known how many pupils sat each exam.
The school also needed to know how many exams were taken by each pupil.

The =COUNTA() function has been used because of its ability to count text and numeric entries.

	Maths	English	Art	History	Exams Taken By Each Pupil
Alan	Fail		1		2
Bob	2	1	3		3
Carol		1	1	1	3
David	Fail		Fail		2
Elaine	1	3	2	Fail	4
					=COUNTA(D39:G39)

How many pupils sat each Exam.			
Maths	English	Art	History
4	3	5	2

=COUNTA(D35:D39)

COUNTBLANK

Range To Test
1
Hello
3
0
1-Jan-98
5

Blanks	
2	=COUNTBLANK(C4:C11)

What Does It Do ?

This function counts the number of blank cells in a range.

Syntax

=COUNTBLANK(RangeToTest)

Formatting

No special formatting is needed.

Example

The following table was used by a company which was balloting its workers on whether the company should have a no smoking policy.
Each of the departments in the various factories were questioned.
The response to the question could be Y or N.
As the results of the vote were collated they were entered in to the table.
The =COUNTBLANK() function has been used to calculate the number of departments which have no yet registered a vote.

	Admin	Accounts	Production	Personnel
Factory 1	Y	N		
Factory 2		Y	Y	N
Factory 3				
Factory 4	N		N	N
Factory 5	Y		Y	
Factory 6	Y	Y	Y	N
Factory 7		N	Y	
Factory 8	N	N	Y	Y
Factory 9			Y	
Factory 10	Y	N		Y

Votes not vet registered :	16	=COUNTBLANK(C32:F41)
Votes for Yes :	14	=COUNTIF(C32:F41,"Y")
Votes for No :	10	=COUNTIF(C32:F41,"N")

COUNTIF

Item	Date	Cost
Brakes	1-Jan-98	80
Tyres	10-May-98	25
Brakes	1-Feb-98	80
Service	1-Mar-98	150
Service	5-Jan-98	300
Window	1-Jun-98	50
Tyres	1-Apr-98	200
Tyres	1-Mar-98	100
Clutch	1-May-98	250

How many Brake Shoes Have been bought.	2	=COUNTIF(C4:C12,"Brakes")
How many Tyres have been bought.	3	=COUNTIF(C4:C12,"Tyres")
How many items cost £100 or above.	5	=COUNTIF(E4:E12,">=100")
Type the name of the item to count. service	2	=COUNTIF(C4:C12,E18)

What Does It Do ?

This function counts the number of items which match criteria set by the user.

Syntax

=COUNTIF(RangeOfThingsToBeCounted,CriteriaToBeMatched)

The criteria can be typed in any of the following ways.

To match a specific number type the number, such as =COUNTIF(A1:A5,**100**)

To match a piece of text type the text in quotes, such as =COUNTIF(A1:A5,**"Hello"**)

To match using operators surround the expression with quotes, such as =COUNTIF(A1:A5,**">100"**)

Formatting

No special formatting is needed.

DATE

Day	Month	Year	Date	
25	12	99	12/25/99	=DATE(E4,D4,C4)
25	12	99	25-Dec-99	=DATE(E5,D5,C5)
33	12	99	January 2, 2000	=DATE(E6,D6,C6)

What Does It Do?

This function creates a real date by using three normal numbers typed into separate cells.

Syntax

=DATE(year,month,day)

Formatting

The result will normally be displayed in the dd/mm/yy format.

By using the Format,Cells,Number,Date command the format can be changed.

DATEDIF

FirstDate	SecondDate	Interval	Difference	
1-Jan-60	10-May-70	days	#NAME?	=DATEDIF(C4,D4,"d")
1-Jan-60	10-May-70	months	#NAME?	=DATEDIF(C5,D5,"m")
1-Jan-60	10-May-70	years	#NAME?	=DATEDIF(C6,D6,"y")
1-Jan-60	10-May-70	yeardays	#NAME?	=DATEDIF(C7,D7,"yd")
1-Jan-60	10-May-70	yearmonths	#NAME?	=DATEDIF(C8,D8,"ym")
1-Jan-60	10-May-70	monthdays	#NAME?	=DATEDIF(C9,D9,"md")

What Does It Do?

This function calculates the difference between two dates.
It can show the result in weeks, months or years.

Syntax

=DATEDIF(FirstDate,SecondDate,"Interval")
FirstDate : This is the earliest of the two dates.
SecondDate : This is the most recent of the two dates.
"Interval" : This indicates what you want to calculate.
These are the available intervals.

"d"	Days between the two dates.
"m"	Months between the two dates.
"y"	Years between the two dates.
"yd"	Days between the dates, as if the dates were in the same year.
"ym"	Months between the dates, as if the dates were in the same year.
"md"	Days between the two dates, as if the dates were in the same month and year.

Formatting

No special formatting is needed.

Birth date :	1-Jan-60	
Years lived :	#NAME?	=DATEDIF(C8,TODAY(),"y")
and the month	#NAME?	=DATEDIF(C8,TODAY(),"ym")
and the days :	#NAME?	=DATEDIF(C8,TODAY(),"md")

You can put this all together in one calculation, which creates a text version.

#NAME?

```
="Age is "&DATEDIF(C8,TODAY(),"y")&" Years, "&DATEDIF(C8,TODAY(),"ym")&" Months and "&DATEDIF(C8,TODAY(),"md")&" Days"
```

DATEVALUE

Date	Date Value	
25-dec-99	36519	=DATEVALUE(C4)
25/12/99	Err:502	=DATEVALUE(C5)
25-dec-99	36519	=DATEVALUE(C6)
25/12/99	Err:502	=DATEVALUE(C7)

What Does It Do?

The function is used to convert a piece of text into a date which can be used in calculations. Dates expressed as text are often created when data is imported from other programs, such as exports from mainframe computers.

Syntax

=DATEVALUE(text)

Formatting

The result will normally be shown as a number which represents the date. This number can be formatted to any of the normal date formats by using Format,Cells,Number,Date.

Example

The example uses the =DATEVALUE and the =TODAY functions to calculate the number of days remaining on a property lease.

The =DATEVALUE function was used because the date has been entered in the cell as a piece of text, probably after being imported from an external program.

Property Ref.	Expiry Date	Days Until Expiry
BC100	25-dec-99	-3224
FG700	10-july/99	Err:502
TD200	13-sep-98	-3692
HJ900	30/5/2000	Err:502

=DATEVALUE(E32)-TODAY()

DAVERAGE

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	0	£0.00
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	0	£0.00
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

To calculate the Average cost of a particular Brand of bulb.

Brand	*These two cells are the **Criteria** range.*
sunbeam	

Type the brand name :

The Average cost of sunbeam is : £1.24 =DAVERAGE(B3:I19,F3,E23:E24)

What Does It Do ?

This function examines a list of information and produces and average.

Syntax

=DAVERAGE(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to be averaged, such as "Unit Cost" or F3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The average Unit Cost of a particular Product of a particular Brand.

Product	Brand
Bulb	Horizon

The average of Horizon Bulb is : £1.16 =DAVERAGE(B3:I19,F3,E49:F50)

	A	B	C	D	E	F	G	H	I	J
53										
54		This is the same calculation but using the actual name "Unit Cost" instead of the cell address.								
55										
56					£1.16	=DAVERAGE(B3:I19,"Unit Cost",E49:F50)				
57										
58		**The average Unit Cost of a Bulb equal to a particular Wattage.**								
59										
60					Product	Wattage				
61					Bulb	100				
62										
63		Average of Bulb 100 is :			£0.53	=DAVERAGE(B3:I19,"Unit Cost",E60:F61)				
64										
65		**The average Unit Cost of a Bulb less then a particular Wattage.**								
66										
67					Product	Wattage				
68					Bulb	<100				
69										
70		Average of Bulb <100 is :			£0.17	=DAVERAGE(B3:I19,"Unit Cost",E67:F68)				

DAY

Full Date	The Day	
25-Dec-98	25	=DAY(C4)
22-Oct-08	Sun 21	=DAY(C5)
22-Oct-08	22	=DAY(C6)

What Does It Do?

This function extracts the day of the month from a complete date.

Syntax

=DAY(value)

Formatting

Normally the result will be a number, but this can be formatted to show the actual day of the week by using Format,Cells,Number,Custom and using the code ddd or dddd.

Example

The =DAY function has been used to calculate the name of the day for your birthday.

Please enter your date of birth in the format dd/mm/yy :	3/25/1962	
You were born on :	Wednesday 24	=DAY(F21)

DAYS360

StartDate	EndDate	Days Between	* See the Note below.
1-Jan-98	5-Jan-98	4	=DAYS360(C4,D4,TRUE)
1-Jan-98	1-Feb-98	30	=DAYS360(C5,D5,TRUE)
1-Jan-98	31-Mar-98	89	=DAYS360(C6,D6,TRUE)
1-Jan-98	31-Dec-98	359	=DAYS360(C7,D7,TRUE)

What Does It Do?

Shows the number of days between two dates based on a 360-day year (twelve 30-day months).
Use this function if your accounting system is based on twelve 30-day months.

Syntax

=DAYS360(StartDate,EndDate,TRUE of FALSE)
TRUE : Use this for European accounting systems.
FALSE : Use this for USA accounting systems.

Formatting

The result will be shown as a number.

Note

The calculation does not include the last day. The result of using 1-Jan-98 and 5-Jan-98 will give a result of 4. To correct this add 1 to the result. =DAYS360(Start,End,TRUE)+1

DB

Purchase Price :	£5,000
Life in Years :	5
Salvage value :	£200

Year	Deprecation	
1	£2,375.00	=DB(E3,E5,E4,D8)
2	£1,246.88	=DB(E3,E5,E4,D9)
3	£654.61	=DB(E3,E5,E4,D10)
4	£343.67	=DB(E3,E5,E4,D11)
5	£180.43	=DB(E3,E5,E4,D12)

Total Depreciation : £4,800.58 ** See example 4 below.*

What Does It Do ?

This function calculates deprecation based upon a fixed percentage.
The first year is depreciated by the fixed percentage.
The second year uses the same percentage, but uses the original value of the item less the first years depreciation.
Any subsequent years use the same percentage, using the original value of the item less the depreciation of the previous years.
The percentage used in the depreciation is not set by the user, the function calculates the necessary percentage, which will be vary based upon the values inputted by the user.

An additional feature of this function is the ability to take into account when the item was originally purchased.
If the item was purchased part way through the financial year, the first years depreciation will be based on the remaining part of the year.

Syntax

=DB(PurchasePrice,SalvageValue,Life,PeriodToCalculate,FirstYearMonth)
The FirstYearMonth is the month in which the item was purchased during the first financial year. This is an optional value, if it not used the function will assume 12 as the value.

Formatting

No special formatting is needed.

Example 1

This example shows the percentage used in the depreciation.
Year 1 depreciation is based upon the original Purchase Price alone.
Year 2 depreciation is based upon the original Purchase Price minus Year 1 deprecation.
Year 3 deprecation is based upon original Purchase Price minus Year 1 + Year 2 deprecation.
The % Deprc has been calculated purely to demonstrate what % is being used.

Purchase Price :	£5,000
Salvage value :	£1,000
Life in Years :	5

Year	Deprecation	% Deprc
1	£1,375.00	27.50%
2	£996.88	27.50%
3	£722.73	27.50%
4	£523.98	27.50%
5	£379.89	27.50%

	A	B	C	D	E	F	G	H	I
57					=DB(E47,E48,E49,D56)				
58									
59			Total Depreciation :		£3,998.48				
60									
61									
62		**Example 2**							
63		This example is similar to the previous, with the exception of the deprecation being calculated							
64		on a monthly basis. This has been done by multiplying the years by 12.							
65									
66			Purchase Price :		£5,000				
67			Life in Years :		£5				
68			Salvage value :		100				
69									
70				Month	Deprecation				
71				56	£8.79				
72				57	£8.24				
73				58	£7.72				
74				59	£7.23				
75				60	£6.78				
76					=DB(E66,E68,E67*12,D75)				
77									
78									
79		**Example 3**							
80		This example shows how the length of the first years ownership has been taken into account.							
81									
82			Purchase Price :		£5,000				
83			Life in Years :		5				
84			Salvage value :		£1,000				
85		First Year Ownership In Months :			6				
86									
87				Year	Deprecation		% Deprc		
88				1	£687.50		13.75%		
89				2	£1,185.94		27.50%		
90				3	£859.80		27.50%		
91				4	£623.36		27.50%		
92				5	£451.93		27.50%		
93					=DB(E74,E76,E75,D84,E77)				
94									
95			Total Depreciation :		£3,808.54				
96									
97									
98		**Why Is The Answer Wrong ?**							
99		In all of the examples above the total depreceation may not be exactly the expected value.							
100		This is due to the way in which the percentage value for the depreceation has been calculated							
101		by the =DB() fumction.							
102		The percentage rate is calculated by Execl using the formula = 1 - ((salvage / cost) ^ (1 / life)).							
103		The result of this calculation is then rounded to three decimal places.							
104		Although this rounding may only make a minor change to the percentage rate, when applied							
105		to large values, the differnce is compounded resulting in what could be considered as							
106		approximate values for the the depreceation.							
107									
108		**Example 4**							
109		This example has been created with both the Excel calculated percentage and the 'real'							
110		percentage calculated manually.							
111		The Excel Deprecation uses the =DB() function.							
112		The Real Deprecation uses a manual calculation.							

	A	B	C	D	E	F	G	H	I
113									
114		This is the 'real' deprecation percentage, calculated manually :					27.522034%		
115						=1-((E117/E116)^(1/E118))			
116			Purchase Price :		£5,000	= 1 - ((salvage / cost) ^ (1 / life)).			
117			Salvage value :		£1,000				
118			Life in Years :		5				
119									
120				Year	Excel Deprecation	Real Depreciation		Excel % Deprc	
121				1	£1,375.0000	£1,376.1017		27.500%	
122				2	£996.8750	£997.3705		27.500%	
123				3	£722.7344	£722.8739		27.500%	
124				4	£523.9824	£523.9243		27.500%	
125				5	£379.8873	£379.7297		27.500%	
126									
127			Total Depreciation :		£3,998.48	£4,000.00			
128									
129				Error difference :		£1.52			

DCOUNT

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	1	£37.50
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	1	£3.75
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

Count the number of products of a particular Brand which have a Life Hours rating.

Brand	*These two cells are the **Criteria** range.*
Type the brand name : Horizon	

The COUNT value of Horizon is : 7 =DCOUNT(B3:I19,D3,E23:E24)

What Does It Do ?

This function examines a list of information and counts the values in a specified column.
It can only count values, the text items and blank cells are ignored.

Syntax

=DCOUNT(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to Count, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The count of a particular product, with a specific number of boxes in stock.

Product	Boxes In Stock
Bulb	5

	A	B	C	D	E	F	G	H	I	J
52										
53			The number of products is :		3	=DCOUNT(B3:I19,H3,E50:F51)				
54										
55		This is the same calculation but using the name "Boxes In Stock" instead of the cell address.								
56										
57					3	=DCOUNT(B3:I19,"Boxes In Stock",E50:F51)				
58										
59		**The count of the number of Bulb products equal to a particular Wattage.**								
60										
61					Product	Wattage				
62					Bulb	100				
63										
64			The count is :		2	=DCOUNT(B3:I19,"Boxes In Stock",E61:F62)				
65										
66		**The count of Bulb products between two Wattage values.**								
67										
68					Product	Wattage	Wattage			
69					Bulb	>=80	<=100			
70										
71			The count is :		4	=DCOUNT(B3:I19,"Boxes In Stock",E68:G69)				

DCOUNTA

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	1	£37.50
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	1	£3.75
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

Count the number of products of a particular Brand.

Brand	*These two cells are the **Criteria** range.*
Horizon	

Type the brand name :

The COUNT value of Horizon is : 8 =DCOUNTA(B3:I19,E3,E23:E24)

What Does It Do ?

This function examines a list of information and counts the non blank cells in a specified column.
It counts values and text items, but blank cells are ignored.

Syntax

=DCOUNTA(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to Count, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The count of a product with an unknown Life Hours value.

Product	Life Hours
Bulb	unknown

	A	B	C	D	E	F	G	H	I	J
53		The number of products is :			1	=DCOUNTA(B3:I19,D3,E50:F51)				
54										
55		This is the same calculation but using the name "Life Hours" instead of the cell address.								
56										
57					1	=DCOUNTA(B3:I19,"Life Hours",E50:F51)				
58										
59		**The count of the number of particular product of a specific brand.**								
60										
61					Product	Brand				
62					Bulb	Horizon				
63										
64			The count is :		5	=DCOUNTA(B3:I19,"Product",E61:F62)				
65										
66		**The count of particular products from specific brands.**								
67										
68					Product	Brand				
69					Spot	Horizon				
70					Neon	Sunbeam				
71										
72			The count is :		3	=DCOUNTA(B3:I19,"Product",E68:F70)				

DEC2BIN

Decimal Number	Binary Equivalent	
0	0	=DEC2BIN(C4)
1	1	=DEC2BIN(C5)
2	10	=DEC2BIN(C6)
3	11	=DEC2BIN(C7)
511	111111111	=DEC2BIN(C8)
512	Err:502	=DEC2BIN(C9)
-1	1111111111	=DEC2BIN(C10)
-2	1111111110	=DEC2BIN(C11)
-3	1111111101	=DEC2BIN(C12)
-511	1000000001	=DEC2BIN(C13)
-512	1000000000	=DEC2BIN(C14)

Decimal Number	Places To Pad	Binary Equivalent	
1	1	1	=DEC2BIN(C17,D17)
1	2	01	=DEC2BIN(C18,D18)
1	3	001	=DEC2BIN(C19,D19)
1	9	000000001	=DEC2BIN(C20,D20)
-1	1	1111111111	=DEC2BIN(C21,D21)

What Does It Do ?

This function converts a decimal number to its binary equivalent.
It can only cope with decimals ranging from -512 to 511.
The result can be padded with leading 0 zeros, although this is ignored for negatives.

Syntax

=DEC2BIN(DecimalNumber,PlacesToPad)
The PlacesToPad is optional.

Formatting

No special formatting is needed.

DEC2HEX

Decimal Number	Hexadecimal	
0	0	=DEC2HEX(C4)
1	1	=DEC2HEX(C5)
2	2	=DEC2HEX(C6)
3	3	=DEC2HEX(C7)
25	19	=DEC2HEX(C8)
26	1A	=DEC2HEX(C9)
27	1B	=DEC2HEX(C10)
28	1C	=DEC2HEX(C11)
-1	FFFFFFFFFF	=DEC2HEX(C12)
-2	FFFFFFFFFE	=DEC2HEX(C13)
-3	FFFFFFFFFD	=DEC2HEX(C14)
-2	FFFFFFFFFE	=DEC2HEX(C15)
-1	FFFFFFFFFF	=DEC2HEX(C16)
549,755,813,887	7FFFFFFFFF	=DEC2HEX(C17)
-549,755,813,888	8000000000	=DEC2HEX(C18)
549,755,813,888	8000000000	=DEC2HEX(C19)
-549,755,813,889	7FFFFFFFFF	=DEC2HEX(C20)

Decimal Number	Places To Pad	Hexadecimal	
1	1	1	=DEC2HEX(C23,D23)
1	2	01	=DEC2HEX(C24,D24)
26	3	01A	=DEC2HEX(C25,D25)
26	9	00000001A	=DEC2HEX(C26,D26)
-26	1	FFFFFFFFE6	=DEC2HEX(C27,D27)

What Does It Do ?

This function converts a decimal number to its hexadecimal equivalent.
It can only cope with decimals ranging from -549,755,813,888 to 549,755,813,887.
The result can be padded with leading 0 zeros, although this is ignored for negatives.

Syntax

=DEC2HEX(DecimalNumber,PlacesToPad)
The PlacesToPad is optional.

Formatting

No special formatting is needed.

DELTA

Number1	Number2	Delta	
10	20	0	=DELTA(C4,D4)
50	50	1	=DELTA(C5,D5)
17.5	17.5	1	=DELTA(C6,D6)
17.5	18	1	=DELTA(C7,D7)
17.50%	0.18	1	=DELTA(C8,D8)
Hello	Hello	Err:502	=DELTA(C9,D9)
		1	=DELTA(C10,D10)

What Does It Do ?

This function compares two values and tests whether they are exactly the same.
If the numbers are the same the result will be 1, otherwise the result is 0.
It only works with numbers, text values produce a result of #VALUE.
The formatting of the number is not significant, so numbers which appear rounded due to the removal of decimal places will still match correctly with non rounded values.

Syntax

=DELTA(FirstNumber,SecondNumber)

Formatting

No special formatting is needed.

Example

The following table is used to determine how may pairs of similar numbers are in a list.
The =DELTA() function tests each pair and then the =SUM() function totals them.

Number1	Number2	Delta	
10	20	0	=DELTA(C30,D30)
50	50	1	=DELTA(C31,D31)
30	30	1	=DELTA(C32,D32)
17.5	18	1	=DELTA(C33,D33)
12	8	0	=DELTA(C34,D34)
100	100	1	=DELTA(C35,D35)
150	125	0	=DELTA(C36,D36)
	Total Pairs	4	=SUM(E30:E36)

DGET

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	1	£37.50
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	1	£3.75
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

How many boxes of a particular item do we have in stock?

Product	Wattage	Life Hours	Brand
Bulb	100		Horizon

The number in stock is : 5 =DGET(B3:I19,H3,C23:F24)

What Does It Do ?

This function examines a list of information and produces one result.
If more than one record matches the criteria the error #NUM is shown.
If no records match the criteria the error #VALUE is shown.

Syntax

=DGET(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to Get, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.
The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record which needs to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Example 1

This example extracts information from just one record.

How many boxes of a particular item do we have in stock?

	A	B	C	D	E	F	G	H	I	J
52			**Product**	**Wattage**	**Life Hours**	**Brand**				
53			Bulb	100		Horizon				
54										
55					The number in stock is :	5	=DGET(B3:I19,H3,C51:F52)			
56										
57										
58		**Example 2**								
59		This example extracts information from multiple records and therefore shows the #NUM error.								
60										
61		How many boxes of a particular item do we have in stock?								
62										
63			**Product**	**Wattage**	**Life Hours**	**Brand**				
64			Bulb	100						
65										
66					The number in stock is :	Err:502	=DGET(B3:I19,H3,C63:F64)			
67										
68										
69		**Example 3**								
70		This example extracts information from no records and therefore shows the #VALUE error.								
71										
72		How many boxes of a particular item do we have in stock?								
73										
74			**Product**	**Wattage**	**Life Hours**	**Brand**				
75			Bulb	9999						
76										
77					The number in stock is :	#VALUE!	=DGET(B3:I19,H3,C64:F65)			
78										
79										
80		**Example 4**								
81		This example uses the =IF() function to display a message when an error occurs.								
82										
83		How many boxes of a particular item do we have in stock?								
84										
85			**Product**	**Wattage**	**Life Hours**	**Brand**				
86			Bulb	9999						
87										
88					The number in stock is :	#VALUE!	=DGET(B3:I19,H3,C85:F86)			
89										
90						Err:502				
91		=IF(ISERR(F88),CHOOSE(ERROR.TYPE(F88)/3,"No such product.","Duplicates products found."),"One product found.")								

DMAX

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	0	£0.00
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	0	£0.00
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

To calculate largest Value Of Stock of a particular Brand of bulb.

Brand	*These two cells are the **Criteria** range.*
Horizon	

Type the brand name :

The MAX value of Horizon is : £60.00 =DMAX(B3:I19,I3,E23:E24)

What Does It Do ?

This function examines a list of information and produces the largest value from a specified column.

Syntax

=DMAX(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name or cell, of the values to pick the Max from, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The largest Value Of Stock of a particular Product of a particular Brand.

Product	Brand
Bulb	sunbeam

The largest value is : £30.00 =DMAX(B3:I19,I3,E49:F50)

This is the same calculation but using the name "Value Of Stock" instead of the cell address.

£30.00 =DMAX(B3:I19,"Value Of Stock",E49:F50)

The largest Value Of Stock of a Bulb equal to a particular Wattage.

	A	B	C	D	E	F	G	H	I	J	K
59											
60					Product	Wattage					
61					Bulb	100					
62											
63		The largest Value Of Stock is :			£40.00	=DMAX(B3:I19,"Value Of Stock",E60:F61)					
64											
65		**The largest Value Of Stock of a Bulb less than a particular Wattage.**									
66											
67					Product	Wattage					
68					Bulb	<100					
69											
70		The largest Value Of Stock is :			£24.00	=DMAX(B3:I19,"Value Of Stock",E67:F68)					

DMIN

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	1	£37.50
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	1	£3.75
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

To calculate lowest Value Of Stock of a particular Brand of bulb.

Brand	*These two cells are the **Criteria** range.*
Horizon	

Type the brand name :

The MIN value of Horizon is : £10.00 =DMIN(B3:I19,I3,E23:E24)

What Does It Do ?

This function examines a list of information and produces smallest value from a specified column.

Syntax

=DMIN(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to pick the Min from, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The lowest Value Of Stock of a particular Product of a particular Brand.

Product	Brand
Bulb	sunbeam

The lowest value is : £3.75 =DMIN(B3:I19,I3,E49:F50)

This is the same calculation but using the name "Value Of Stock" instead of the cell address.

£3.75 =DMIN(B3:I19,"Value Of Stock",E49:F50)

The lowest Value Of Stock of a Bulb equal to a particular Wattage.

	A	B	C	D	E	F	G	H	I	J	K
59											
60					Product	Wattage					
61					Bulb	100					
62											
63		The lowest Value Of Stock is :			£12.50	=DMIN(B3:I19,"Value Of Stock",E60:F61)					
64											
65		**The lowest Value Of Stock of a Bulb between two Wattage values.**									
66											
67					Product	Wattage	Wattage				
68					Bulb	>=80	<=100				
69											
70		The lowest Value Of Stock is :			£12.00	=DMIN(B3:I19,"Value Of Stock",E67:G68)					

DOLLAR

Original Number	Converted To Text	
10	$10.00	=DOLLAR(C4)
10	$10	=DOLLAR(C5,0)
10	$10.0	=DOLLAR(C6,1)
10	$10.00	=DOLLAR(C7,2)
10.25	$10.25	=DOLLAR(C8)
10.25	$10	=DOLLAR(C9,0)
10.25	$10.3	=DOLLAR(C10,1)
10.25	$10.25	=DOLLAR(C11,2)

What Does It Do?

This function converts a number into a piece of text formatted as currency.

Syntax

=DOLLAR(Number,DecimalPlaces)

Number : This is the number which needs to be converted.

DecimalPlaces : This is the amount of decimal places needed in the converted number.

Formatting

No special formatting is needed.

The result will be shown as a text entry.

DSUM

*This is the **Database** range.*

Product	Wattage	Life Hours	Brand	Unit Cost	Box Quantity	Boxes In Stock	Value Of Stock
Bulb	200	3000	Horizon	£4.50	4	3	£54.00
Neon	100	2000	Horizon	£2.00	15	2	£60.00
Spot	60						£0.00
Other	10	8000	Sunbeam	£0.80	25	6	£120.00
Bulb	80	1000	Horizon	£0.20	40	3	£24.00
Spot	100	unknown	Horizon	£1.25	10	4	£50.00
Spot	200	3000	Horizon	£2.50	15	0	£0.00
Other	25	unknown	Sunbeam	£0.50	10	3	£15.00
Bulb	200	3000	Sunbeam	£5.00	3	2	£30.00
Neon	100	2000	Sunbeam	£1.80	20	5	£180.00
Bulb	100	unknown	Sunbeam	£0.25	10	5	£12.50
Bulb	10	800	Horizon	£0.20	25	2	£10.00
Bulb	60	1000	Sunbeam	£0.15	25	0	£0.00
Bulb	80	1000	Sunbeam	£0.20	30	2	£12.00
Bulb	100	2000	Horizon	£0.80	10	5	£40.00
Bulb	40	1000	Horizon	£0.10	20	5	£10.00

To calculate the total Value Of Stock of a particular Brand of bulb.

Brand	*These two cells are the **Criteria** range.*
Horizon	

Type the brand name :

The stock value of Horizon is : £248.00 =DSUM(B3:I19,I3,E23:E24)

What Does It Do ?

This function examines a list of information and produces the total.

Syntax

=DSUM(DatabaseRange,FieldName,CriteriaRange)

The **DatabaseRange** is the entire list of information you need to examine, including the field names at the top of the columns.

The **FieldName** is the name, or cell, of the values to be totalled, such as "Value Of Stock" or I3.

The **CriteriaRange** is made up of two types of information.

The first set of information is the name, or names, of the Fields(s) to be used as the basis for selecting the records, such as the category Brand or Wattage.

The second set of information is the actual record, or records, which are to be selected, such as Horizon as a brand name, or 100 as the wattage.

Formatting

No special formatting is needed.

Examples

The total Value Of Stock of a particular Product of a particular Brand.

Product	Brand
Bulb	sunbeam

Total stock value is : £54.50 =DSUM(B3:I19,I3,E49:F50)

	A	B	C	D	E	F	G	H	I	J
53										
54		This is the same calculation but using the name "Value Of Stock" instead of the cell address.								
55										
56					£54.50	=DSUM(B3:I19,"Value Of Stock",E49:F50)				
57										
58		**The total Value Of Stock of a Bulb equal to a particular Wattage.**								
59										
60					Product	Wattage				
61					Bulb	100				
62										
63		Total Value Of Stock is :			£52.50	=DSUM(B3:I19,"Value Of Stock",E60:F61)				
64										
65		**The total Value Of Stock of a Bulb less than a particular Wattage.**								
66										
67					Product	Wattage				
68					Bulb	<100				
69										
70		Total Value Of Stock is :			£56.00	=DSUM(B3:I19,"Value Of Stock",E67:F68)				

	A	B	C	D	E	F	G	H	I	J
1	Eastern data.									
2	Used by the example for the =INDIRECT() function.									
3										
4			Jan	Feb	Mar	Total				
5		Alan	1000	2000	3000	6000				
6		Bob	4000	5000	6000	15000				
7		Carol	7000	8000	9000	24000				
8		Total	12000	15000	18000	45000				

EDATE

Start Date	Plus Months	End Date	
1-Jan-98	3	1-Apr-98	=EDATE(C4,D4)
2-Jan-98	3	2-Apr-98	=EDATE(C5,D5)
2-Jan-98	-3	2-Oct-97	=EDATE(C6,D6)

What Does It Do?

This function is used to calculate a date which is a specific number of months in the past or in the future.

Syntax

=EDATE(StartDate,Months)

Formatting

The result will normally be expressed as a number, this can be formatted to represent a date by using the Format,Cells,Number,Date command.

Example

This example was used by a company hiring contract staff.
The company needed to know the end date of the employment.
The Start date is entered.
The contract Duration is entered as months.
The =EDATE() function has been used to calculate the end of the contract.

Start	Duration	End	
Tue 06-Jan-98	3	Mon 06-Apr-98	=EDATE(C27,D27)
Mon 12-Jan-98	3	Sun 12-Apr-98	=EDATE(C28,D28)
Fri 09-Jan-98	4	Sat 09-May-98	=EDATE(C29,D29)
Fri 09-Jan-98	3	Thu 09-Apr-98	=EDATE(C30,D30)
Mon 19-Jan-98	3	Sun 19-Apr-98	=EDATE(C31,D31)
Mon 26-Jan-98	3	Sun 26-Apr-98	=EDATE(C32,D32)
Mon 12-Jan-98	3	Sun 12-Apr-98	=EDATE(C33,D33)

The company decide not to end contracts on Saturday or Sunday.
The =WEEKDAY() function has been used to identify the actaul weekday number of the end date.
If the week day number is 6 or 7, (Sat or Sun), then 5 is subtracted from the =EDATE() to ensure the end of contract falls on a Friday.

Start	Duration	End
Tue 06-Jan-98	3	Mon 06-Apr-98
Mon 12-Jan-98	3	Fri 10-Apr-98
Fri 09-Jan-98	4	Fri 08-May-98
Fri 09-Jan-98	3	Thu 09-Apr-98
Mon 19-Jan-98	3	Fri 17-Apr-98
Mon 26-Jan-98	3	Fri 24-Apr-98
Mon 12-Jan-98	3	Fri 10-Apr-98

```
=EDATE(C48,D48)-IF(WEEKDAY(EDATE(C48,D48),2)>5,WEEKDAY(EDATE(C48,D48),2)-5,0)
```

EOMONTH

StartDate	Plus Months	End Of Month	
5-Jan-98	2	35885	=EOMONTH(C4,D4)
5-Jan-98	2	31-Mar-98	=EOMONTH(C5,D5)
5-Jan-98	-2	30-Nov-97	=EOMONTH(C6,D6)

What Does It Do?

This function will show the last day of the month which is a specified number of months before or after a given date.

Syntax

=EOMONTH(StartDate,Months)

Formatting

The result will normally be expressed as a number, this can be formatted to represent a date by using the Format,Cells,Number,Date command.

ERROR.TYPE

Data		The Error	Error Type	
10	0	#DIV/0!	532	=ERROR.TYPE(E4)
10	3	#NAME?	525	=ERROR.TYPE(E5)
10	3	#REF!	524	=ERROR.TYPE(E6)
10:00	13:00	21:00	#N/A	=ERROR.TYPE(E7)

What Does It Do?

This function will show a number which corresponds to an error produced by a formula.

Syntax

=ERROR.TYPE(Error)

Error is the cell reference where the error occurred.

Formatting

The result will be formatted as a normal number.

Example

See Example 4 in the =DGET() function.

EVEN

Original Value	Evenly Rounded	
1	2	=EVEN(C4)
1.2	2	=EVEN(C5)
2.3	4	=EVEN(C6)
25	26	=EVEN(C7)

What Does It Do ?

This function round a number up the nearest even whole number.

Syntax

=EVEN(Number)

Formatting

No special formatting is needed.

Example

The following table is used by a garage which repairs cars.
The garage is repairing a fleet of cars from three manufactures.
Each manufacturer uses a different type of windscreen wiper which are only supplied in pairs.
Table 1 was used to enter the number of wipers required for each type of car and then show how many pairs need to be ordered.

Table 1

Car	Wipers To Order	Pairs to Order	
Vauxhall	5	3	=EVEN(D28)/2
Ford	9	5	=EVEN(D29)/2
Peugeot	7	4	=EVEN(D30)/2

EXACT

Text1	Text2	Result	
Hello	Hello	TRUE	=EXACT(C4,D4)
Hello	hello	FALSE	=EXACT(C5,D5)
Hello	Goodbye	FALSE	=EXACT(C6,D6)

What Does It Do?

This function compares two items of text and determine whether they are exactly the same.
The case of the characters is taken into account, only words which are spelt the same and
which have upper and lower case characters in the same position will be considered as equal.

Syntax

=EXACT(Text1,Text2)

Only two items of text can be compared.

Formatting

If the two items of text are exactly the same the result of TRUE will be shown.
If there is any difference in the two items of text the result of FALSE will be shown.

Example

Here is a simple password checking formula.
You need to guess the correct password.
The password is the name of a colour, either red blue or green.
The case of the password is important.
The =EXACT() function is used to check your guess.

Guess the password :	red
Is it correct :	No

(To stop you from cheating, the correct password has been entered as a series of =CHAR()
functions, which use the ANSI number of the characters rather than the character itself!)
Its still very easy though.

FACT

Number	Factorial	
3	6	=FACT(C4)
3.5	6	=FACT(C5)
5	120	=FACT(C6)
10	3,628,800	=FACT(C7)
20	2,432,902,008,176,640,000	=FACT(C8)

What Does It Do ?

This function calculates the factorial of a number.
The factorial is calculated as 1*2*3*4..etc.
The factorial of 5 is calculated as 1*2*3*4*5, which results in 120.
Decimal fractions of the number are ignored.

Syntax

=FACT(Number)

Formatting.

No special formatting is needed.

FIND

Text	Letter To Find	Position Of Letter	
Hello	e	2	=FIND(D4,C4)
Hello	H	1	=FIND(D5,C5)
Hello	o	5	=FIND(D6,C6)
Alan Williams	a	3	=FIND(D7,C7)
Alan Williams	a	11	=FIND(D8,C8,6)
Alan Williams	T	#VALUE!	=FIND(D9,C9)

What Does It Do?

This function looks for a specified letter inside another piece of text.
When the letter is found the position is shown as a number.
If the text contains more than one reference to the letter, the first occurrence is used.
An additional option can be used to start the search at a specific point in the text, thus enabling the search to find duplicate occurrences of the letter.
If the letter is not found in the text, the result #VALUE is shown.

Syntax

=FIND(LetterToLookFor,TextToLookInside,StartPosition)
LetterToLookFor : This needs to be a single character.
TextToLookInside : This is the piece of text to be searched through.
StartPosition : This is optional, it specifies at which point in the text the search should begin.

Formatting

No special formatting is needed, the result will be shown as a number.

FIXED

Original Number	Converted To Text	
10	10.00	=FIXED(C4)
10	10	=FIXED(C5,0)
10	10.0	=FIXED(C6,1)
10	10.00	=FIXED(C7,2)
10.25	10.25	=FIXED(C8)
10.25	10	=FIXED(C9,0)
10.25	10.3	=FIXED(C10,1)
10.25	10.25	=FIXED(C11,2)
1000	1,000.00	=FIXED(C12)
1000.23	1,000	=FIXED(C13,0)
1000.23	1000	=FIXED(C14,0,TRUE)

What Does It Do ?

This function converts a numeric value to text.
During the conversion the value can be rounded to a specific number of decimal places, and commas can be inserted at the 1,000's.

Syntax

=FIXED(NumberToConvert,DecimalPlaces,Commas)
If DecimalPlaces places is not specified the function will assume 2.
The Commas option can be TRUE for commas or FALSE for no commas.
If the Commas is not specified the function will assume TRUE.

Formatting

No special formatting is needed.
Note that any further formatting with the Format, Cells, Number command will not have any effect.

FLOOR

Number	Rounded Down	
1.5	1	=FLOOR(C4,1)
2.3	2	=FLOOR(C5,1)
2.9	2	=FLOOR(C6,1)
123	100	=FLOOR(C7,50)
145	100	=FLOOR(C8,50)
175	150	=FLOOR(C9,50)

What Does It Do ?

This function rounds a value down to the nearest multiple specified by the user.

Syntax

=FLOOR(NumberToRound,SignificantValue)

Formatting

No special formatting is needed.

Example

The following table was used to calculate commission for members of a sales team.
Commission is only paid for every £1000 of sales.
The =FLOOR() function has been used to round down the Actual Sales to the nearest 1000, which is then used as the basis for Commission.

Name	Actual Sales	Relevant Sales	Commission
Alan	£23,500	£23,000	£230
Bob	£56,890	£56,000	£560
Carol	£18,125	£18,000	£180

=FLOOR(D29,1000)

FORECAST

Month	Sales
1	£1,000
2	£2,000
3	£2,500
4	£3,500
5	£3,800
6	£4,000

Type the month number to predict : 12

The Forecast sales figure is : £7,997 =FORECAST(E11,F4:F9,E4:E9)

What Does It Do ?

This function uses two sets of values to predict a single value.
The predicted value is based on the relationship between the two original sets of values.
If the values are sales figures for months 1 to 6, (Jan to Jun), you can use the function
to predict what the sales figure will be in any other month.
The way in which the prediction is calculated is based upon the assumption of a Linear Trend.

Syntax

=FORECAST(ItemToForeCast,RangeY,RangeX)
ItemToForecast is the point in the future, (or past), for which you need the forecast.
RangeY is the list of values which contain the historical data to be used as the basis
of the forecast, such as Sales figures.
RangeX is the intervals used when recording the historical data, such as Month number.

Formatting

No special formatting is needed.

Example

The following table was used by a company considering expansion of their sales team.
The Size and Performance of the previous teams over a period of three years were entered.
The size of the New Sales team is entered.
The =FORECAST() function is used to calculate the predicted performance for the new sales
team based upon a linear trend.

Year	Size Of Sales Team	Known Performance
1996	10	£5,000
1997	20	£8,000
1998	30	£8,500

Size Of The New Sales Team : 40

Estimated Forecast Of Performance : £10,667 =FORECAST(E43,E39:E41,D39:D41)

FREQUENCY

	Jan	Feb	Mar
North	£5,000	£6,000	£4,500
South	£5,800	£7,000	£3,000
East	£3,500	£2,000	£10,000
West	£12,000	£4,000	£6,000

Sales £4,000 and below.	£4,000	4
Sales above £4,000 up to £6,000	£6,000	5
Sales above £6,000	£999,999	3

{=FREQUENCY(D4:F7,E9:E11)}
{=FREQUENCY(D4:F7,E9:E11)}
{=FREQUENCY(D4:F7,E9:E11)}

What Does It Do ?

This function compares a range of data against a list of intervals.
The result shows how many items in the range of data fall between the intervals.
The function is entered in the cells as an array, that is why it is enclosed in { } braces.

Syntax

=FREQUENCY(RangeOfData,ListOfIntervals)

Formatting

No special formatting is needed.

Example 1

The following tables were used to record the weight of a group of children.
The =FREQUENCY() function was then used to calculate the number of children whose weights fell between specified intervals.

	Weight Kg
Child 1	20.47
Child 2	22.83
Child 3	15.74
Child 4	10.80
Child 5	8.28
Child 6	20.66
Child 7	17.36
Child 8	16.67
Child 9	18.01

Kg Weight Intervals
15
20
100

Number Of Children:	
Between 0 - 15 Kg	2
Above 15 but less than or equal to 20 Kg	4
Above 20 Kg	3

{=FREQUENCY(C30:C38,C41:C43)}
{=FREQUENCY(C30:C38,C41:C43)}
{=FREQUENCY(C30:C38,C41:C43)}

Example 2

This example uses characters instead of values.
A restaurant has asked 40 customers for their rating of the food in the restaurant.
The ratings were entered into a table as a single letter, E, V, A, P or D.
The manager now wants to calculate how many responses fell into each category.
Unfortunately, the =FREQUENCY() function ignores text entries, so how can the frequency of text be calculated?

The answer is to use the =CODE() and =UPPER() functions.
The =UPPER() forces all the text entries to be considered as capital letters.
The =CODE() function calculates the unique ANSI code for each character.
As this code is a numeric value, the =FREQUENCY() function can then be used!

	Rating	Frequency	
Excellent	E	9	{=FREQUENCY(CODE(UPPER(B67:I71)),CODE(UPPER(C60:C64)))}
Very Good	V	9	{=FREQUENCY(CODE(UPPER(B67:I71)),CODE(UPPER(C60:C64)))}
Average	A	6	{=FREQUENCY(CODE(UPPER(B67:I71)),CODE(UPPER(C60:C64)))}
Poor	P	8	{=FREQUENCY(CODE(UPPER(B67:I71)),CODE(UPPER(C60:C64)))}
Disgusting	D	8	{=FREQUENCY(CODE(UPPER(B67:I71)),CODE(UPPER(C60:C64)))}

Customer Ratings							
V	D	V	A	p	A	D	D
V	P	a	D	A	P	V	d
A	V	E	P	p	E	D	A
A	E	d	V	D	P	a	E
V	e	P	P	A	V	E	D

FREQUENCY 2

This example shows how the =FREQUENCY() function has been used to calculate how often certain numbers appear in the Lottery results.

Table 1 is a record of all the results from the past seven weeks.

Table 1

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
1st Number	3	36	5	3	2	41	45
2nd Number	6	3	19	37	23	15	4
3rd Number	15	44	35	20	47	29	44
4th Number	32	15	32	46	6	45	23
5th Number	37	31	13	22	49	13	43
6th Number	5	22	30	8	49	11	46
Bonus Ball	17	13	15	25	18	17	1

Table 2 is the list of possible number from 1 to 49, and how many appearances each number has made during the past seven weeks.

Table 2

Lottery Number	How Many Appearances
1	1
2	1
3	3
4	1
5	2
6	2
7	0
8	1
9	0
10	0
11	1
12	0
13	3
14	0
15	4
16	0
17	2
18	1
19	1
20	1
21	0
22	2
23	2
24	0
25	1
26	0
27	0
28	0
29	1
30	1
31	1

{=FREQUENCY(C10:I16,B24:B72)}
{=FREQUENCY(C10:I16,B24:B72)}
{=FREQUENCY(C10:I16,B24:B72)}
{=FREQUENCY(C10:I16,B24:B72)}

Special tip!
To count how many unique numbers in a range use the following formula. It has to be entered, as an array, so press Ctrl+Shift+Enter rather than, just Enter alone.

Unique values. 31

=SUM(1/COUNTIF(C10:I16,C10:I16))

	A	B	C	D	E	F	G	H	I
55		32	2						
56		33	0						
57		34	0						
58		35	1						
59		36	1						
60		37	2						
61		38	0						
62		39	0						
63		40	0						
64		41	1						
65		42	0						
66		43	1						
67		44	2						
68		45	2						
69		46	2						
70		47	1						
71		48	0						
72		49	2						

GCD

Numbers		Greatest Divisor	
6	15	3	=GCD(C4,D4)
28	49	7	=GCD(C5,D5)
5	99	1	=GCD(C6,D6)

Numbers			Greatest Divisor	
18	72	96	6	=GCD(C9,D9,E9)
300	500	200	100	=GCD(C10,D10,E10)
2.5	4	6	0.5	=GCD(C11,D11,E11)

What Does It Do ?

This function calculates the largest number which can be used to divided all the values specified.
The result is always a whole number.
Where there is no common divisor the value of 1 is used.
Decimal fractions are ignored.

Syntax

=GCD(Number1,Number2,Number3... through to Number29)

Formatting

No special formatting is needed.

GESTEP

Number1	Number2	GESTEP	
10	20	0	=GESTEP(C4,D4)
50	20	1	=GESTEP(C5,D5)
99	100	0	=GESTEP(C6,D6)
100	100	1	=GESTEP(C7,D7)
101	100	1	=GESTEP(C8,D8)
2		1	=GESTEP(C9,D9)
	2	0	=GESTEP(C10,D10)

What Does It Do ?

This function test a number to see if it is greater than or equal to another number.
If the number is greater than or equal, the result of 1 will be shown, otherwise 0 is shown.

Syntax

=GESTEP(NumberToTest,NumberToTestAgainst)

Formatting

No special formatting is needed.

Example

The following table was used to calculate how many sales staff achieved their targets.
The =GESTEP() function compares the Sales with Target, and the results are totalled.

Name	Sales	Target	GESTEP	
Alan	£3,000	£4,000	0	=GESTEP(D27,E27)
Bob	£5,000	£4,000	1	=GESTEP(D28,E28)
Carol	£1,000	£2,000	0	=GESTEP(D29,E29)
David	£2,000	£2,000	1	=GESTEP(D30,E30)
Eric	£8,000	£7,000	1	=GESTEP(D31,E31)
	Targets Achieved		3	=SUM(F27:F31)

HEX2DEC

Hexadecimal	Decimal Number	
0	0	=HEX2DEC(C4)
1	1	=HEX2DEC(C5)
2	2	=HEX2DEC(C6)
3	3	=HEX2DEC(C7)
1A	26	=HEX2DEC(C8)
1B	27	=HEX2DEC(C9)
7FFFFFFFFF	549,755,813,887	=HEX2DEC(C10)
8000000000	549,755,813,888	=HEX2DEC(C11)
FFFFFFFFFF	-1	=HEX2DEC(C12)
FFFFFFFFFE	-2	=HEX2DEC(C13)
FFFFFFFFFD	-3	=HEX2DEC(C14)

What Does It Do ?

This function converts a hexadecimal number to its decimal equivalent.

Syntax

=HEX2DEC(HexaDecimalNumber)

Formatting

No special formatting is needed.

Example

The following table was used to add two hexadecimal values together.

	Hexadecimal	
Value 1	F	
Value 2	1A	
Result	29	=DEC2HEX(HEX2DEC(C29)+HEX2DEC(C30))

HLOOKUP

	Jan	Feb	Mar	
3	Jan	Feb	Mar	*row 1*
4	10	80	97	*row 2*
5	20	90	69	*row 3*
6	30	100	45	*row 4*
7	40	110	51	*row 5*
8	50	120	77	*row 6*

The row numbers are not needed. they are part of the illustration.

Type a month to look for : Feb

Which row needs to be picked out : 4

The result is : 100

=HLOOKUP(F10,D3:F10,F11,FALSE)

What Does It Do ?

This function scans across the column headings at the top of a table to find a specified item.
When the item is found, it then scans down the column to pick a cell entry.

Syntax

=HLOOKUP(ItemToFind,RangeToLookIn,RowToPickFrom,SortedOrUnsorted)

The ItemToFind is a single item specified by the user.
The RangeToLookIn is the range of data with the column headings at the top.
The RowToPickFrom is how far down the column the function should look to pick from.
The Sorted/Unsorted is whether the column headings are sorted. TRUE for yes, FALSE for no.

Formatting

No special formatting is needed.

Example 1

This table is used to find a value based on a specified month and name.
The =HLOOKUP() is used to scan across to find the month.
The problem arises when we need to scan down to find the row adjacent to the name.
To solve the problem the =MATCH() function is used.

The =MATCH() looks through the list of names to find the name we require. It then calculates the position of the name in the list. Unfortunately, because the list of names is not as deep as the lookup range, the =MATCH() number is 1 less than we require, so and extra 1 is added to compensate.

The =HLOOKUP() now uses this =MATCH() number to look down the month column and picks out the correct cell entry.

The =HLOOKUP() uses FALSE at the end of the function to indicate to Excel that the column headings are not sorted, even though to us the order of Jan,Feb,Mar is correct.
If they were sorted alphabetically they would have read as **F**eb,**J**an,**M**ar.

	Jan	Feb	Mar
Bob	10	80	97
Eric	20	90	69
Alan	30	100	45
Carol	40	110	51
David	50	120	77

Type a month to look for :	feb
Type a name to look for :	alan
The result is :	100

=HLOOKUP(F54,D47:F54,MATCH(F55,C48:C52,0)+1,FALSE)

Example 2

This example shows how the =HLOOKUP() is used to pick the cost of a spare part for different makes of cars.

The =HLOOKUP() scans the column headings for the make of car specified in column B.

When the make is found, the =HLOOKUP() then looks down the column to the row specified by the =MATCH() function, which scans the list of spares for the item specified in column C.

The function uses the absolute ranges indicated by the dollar symbol $. This ensures that when the formula is copied to more cells, the ranges for =HLOOKUP() and =MATCH() do not change.

Maker	Spare	Cost
Vauxhall	Ignition	£50
VW	GearBox	£600
Ford	Engine	£1,200
VW	Steering	£275
Ford	Ignition	£70
Ford	CYHead	£290
Vauxhall	GearBox	£500
Ford	Engine	£1,200

	Vauxhall	Ford	VW
GearBox	500	450	600
Engine	1000	1200	800
Steering	250	350	275
Ignition	50	70	45
CYHead	300	290	310

```
=HLOOKUP(B79,G72:I77,MATCH(C79,F73:F77,0)+1,FALSE)
```

Example 3

In the following example a builders merchant is offering discount on large orders.
The Unit Cost Table holds the cost of 1 unit of Brick, Wood and Glass.
The Discount Table holds the various discounts for different quantities of each product.
The Orders Table is used to enter the orders and calculate the Total.

All the calculations take place in the Orders Table.
The name of the Item is typed in column C.

The Unit Cost of the item is then looked up in the Unit Cost Table.
The FALSE option has been used at the end of the function to indicate that the product names across the top of the Unit Cost Table are not sorted.
Using the FALSE option forces the function to search for an exact match. If a match is not found, the function will produce an error.
=HLOOKUP(C127,E111:G112,2,FALSE)

The discount is then looked up in the Discount Table
If the Quantity Ordered matches a value at the top of the Discount Table the =HLOOKUP will look down the column to find the correct discount.
The TRUE option has been used at the end of the function to indicate that the values across the top of the Discount Table are sorted.
Using TRUE will allow the function to make an approximate match. If the Quantity Ordered does not match a value at the top of the Discount Table, the next lowest value is used.
Trying to match an order of 125 will drop down to 100, and the discount from the 100 column is used.
=HLOOKUP(D127,E115:G118,MATCH(C127,D116:D118,0)+1,TRUE)

Unit Cost Table

Brick	Wood	Glass
£2	£1	£3

Discount Table

	1	100	300
Brick	0%	6%	8%
Wood	0%	3%	5%
Glass	0%	12%	15%

Orders Table

Item	Units	Unit Cost	Discount	Total
Brick	100	£2	6%	£188
Wood	200	£1	3%	£194
Glass	150	£3	12%	£396
Brick	225	£2	6%	£423
Wood	50	£1	0%	£50
Glass	500	£3	15%	£1,275

Unit Cost =HLOOKUP(C127,E111:G112,2,FALSE)

Discount =HLOOKUP(D127,E115:G118,MATCH(C127,D116:D118,0)+1,TRUE)

HOUR

Number	Hour	
21:15	21	=HOUR(C4)
0.25	6	=HOUR(C5)

What Does It Do?

The function will show the hour of the day based upon a time or a number.

Syntax

=HOUR(Number)

Formatting

The result will be shown as a normal number between 0 and 23.

IF

Name	Sales	Target	Result	
Alan	1000	5000	Not Achieved	=IF(C4>=D4,"Achieved","Not Achieved")
Bob	6000	5000	Achieved	=IF(C5>=D5,"Achieved","Not Achieved")
Carol	2000	4000	Not Achieved	=IF(C6>=D6,"Achieved","Not Achieved")

What Does It Do?

This function tests a condition.
If the condition is met it is considered to be TRUE.
If the condition is not met it is considered as FALSE.
Depending upon the result, one of two actions will be carried out.

Syntax

=IF(Condition,ActionIfTrue,ActionIfFalse)
The Condition is usually a test of two cells, such as A1=A2.
The ActionIfTrue and ActionIfFalse can be numbers, text or calculations.

Formatting

No special formatting is required.

Example 1

The following table shows the Sales figures and Targets for sales reps.
Each has their own target which they must reach.
The =IF() function is used to compare the Sales with the Target.
If the Sales are greater than or equal to the Target the result of Achieved is shown.
If the Sales do not reach the target the result of Not Achieved is shown.
Note that the text used in the =IF() function needs to be placed in double quotes "Achieved".

Name	Sales	Target	Result	
Alan	1000	5000	Not Achieved	=IF(C31>=D31,"Achieved","Not Achieved")
Bob	6000	5000	Achieved	=IF(C32>=D32,"Achieved","Not Achieved")
Carol	2000	4000	Not Achieved	=IF(C33>=D33,"Achieved","Not Achieved")

Example 2

The following table is similar to that in Example 1.
This time the Commission to be paid to the sales rep is calculated.
If the Sales are greater than or equal to the Target, the Commission is 10% of Sales.
If the Sales do not reach Target, the Commission is only 5% of Sales.

Name	Sales	Target	Commission	
Alan	1000	5000	50	=IF(C43>=D43,C43*10%,C43*5%)
Bob	6000	5000	600	=IF(C44>=D44,C44*10%,C44*5%)
Carol	2000	4000	100	=IF(C45>=D45,C45*10%,C45*5%)

Example 3

This example uses the =AND() within the =IF() function.
A builders merchant gives 10% discount on certain product lines.
The discount is only given on products which are on Special Offer, when the Order Value is £1000 or above.
The =AND() function is used with the =IF() to check that the product is on offer **and** that the value of the order is above £1000.

Product	Special Offer	Order Value	Discount	Total
Wood	Yes	£2,000	£200	£1,800
Glass	No	£2,000	£-	£2,000
Cement	Yes	£500	£-	£500

	A	B	C	D	E	F	G	H	I	J
61		Turf	Yes	£3,000	£300	£2,700				
62					=IF(AND(C61="Yes",D61>=1000),D61*10%,0)					

INDEX

Holiday booking price list.				
	People			
Weeks	1	2	3	4
1	£500	£300	£250	£200
2	£600	£400	£300	£250
3	£700	£500	£350	£300

How many weeks required :	2
How many people in the party :	4
Cost per person is :	250

=INDEX(D7:G9,G11,G12)

What Does It Do ?

This function picks a value from a range of data by looking down a specified number of rows and then across a specified number of columns.
It can be used with a single block of data, or non-continuos blocks.

Syntax

There are various forms of syntax for this function.

Syntax 1

=INDEX(RangeToLookIn,Coordinate)
This is used when the RangeToLookIn is either a single column or row.
The Co-ordinate indicates how far down or across to look when picking the data from the range.
Both of the examples below use the same syntax, but the Co-ordinate refers to a row when the range is vertical and a column when the range is horizontal.

Colours
Red
Green
Blue

Type either 1, 2 or 3 :	2
The colour is :	Green

=INDEX(D32:D34,D36)

Size	Large	Medium	Small

Type either 1, 2 or 3 :	2
The size is :	Err:504

=INDEX(G34:I34,H36)

Syntax 2

=INDEX(RangeToLookIn,RowCoordinate,ColumnColumnCordinate)

This syntax is used when the range is made up of rows and columns.

Country	Currency	Population	Capitol
England	Sterling	50 M	London
France	Franc	40 M	Paris
Germany	DM	60 M	Bonn
Spain	Peseta	30 M	Barcelona

Type 1,2,3 or 4 for the country :	2
Type 1,2 or 3 for statistics :	3
The result is :	Paris

=INDEX(D45:F48,F50,F51)

Syntax 3

=INDEX(NamedRangeToLookIn,RowCoordinate,ColumnColumnCordinate,AreaToPickFrom)

Using this syntax the range to look in can be made up of multiple areas.

The easiest way to refer to these areas is to select them and give them a single name.

The AreaToPickFrom indicates which of the multiple areas should be used.

In the following example the figures for North and South have been named as one range called NorthAndSouth.

NORTH	Qtr1	Qtr2	Qtr3	Qtr4
Bricks	£1,000	£2,000	£3,000	£4,000
Wood	£5,000	£6,000	£7,000	£8,000
Glass	£9,000	£10,000	£11,000	£12,000

SOUTH	Qtr1	Qtr2	Qtr3	Qtr4
Bricks	£1,500	£2,500	£3,500	£4,500
Wood	£5,500	£6,500	£7,500	£8,500
Glass	£9,500	£10,500	£11,500	£12,500

Type 1, 2 or 3 for the product :	1
Type 1, 2, 3 or 4 for the Qtr :	3
Type 1 for North or 2 for South :	2
The result is :	Err:504

=INDEX(NorthAndSouth,F76,F77,F78)

	A	B	C	D	E	F	G	H	I
83									
84		**Example**							
85		This is an extended version of the previous example.							
86		It allows the names of products and the quarters to be entered.							
87		The =MATCH() function is used to find the row and column positions of the names entered.							
88		These positions are then used by the =INDEX() function to look for the data.							
89									
90			EAST	Qtr1	Qtr2	Qtr3	Qtr4		
91			Bricks	£1,000	£2,000	£3,000	£4,000		
92			Wood	£5,000	£6,000	£7,000	£8,000		
93			Glass	£9,000	£10,000	£11,000	£12,000		
94									
95			WEST	Qtr1	Qtr2	Qtr3	Qtr4		
96			Bricks	£1,500	£2,500	£3,500	£4,500		
97			Wood	£5,500	£6,500	£7,500	£8,500		
98			Glass	£9,500	£10,500	£11,500	£12,500		
99									
100			Type 1, 2 or 3 for the product :			wood			
101			Type 1, 2, 3 or 4 for the Qtr :			qtr2			
102			Type 1 for North or 2 for South :			west			
103									
104				The result is :		Err:504			
105									
106		=INDEX(EastAndWest,MATCH(F100,C91:C93,0),MATCH(F101,D90:G90,0),IF(F102=C90,1,IF(F102=C95,2)))							

INDIRECT

	Jan	Feb	Mar
North	10	20	30
South	40	50	60
East	70	80	90
West	100	110	120

Type address of any of the cells in the above table, such as G6 : G6

The value in the cell you typed is : 80 =INDIRECT(H9)

What Does It Do ?

This function converts a plain piece of text which looks like a cell address into a usable cell reference.
The address can be either on the same worksheet or on a different worksheet.

Syntax

=INDIRECT(Text)

Formatting

No special formatting is needed.

Example 1

This example shows how data can be picked form other worksheets by using the worksheet name and a cell address.
The example uses three other worksheets named NORTH, SOUTH and EAST.
The data on these three sheets is laid out in the same cells on each sheet.

When a reference to a sheet is made the exclamation symbol ! needs to be placed between the sheet name and cell address acting as punctuation.

Type the name of the **sheet**, such as North : North
Type the **cell** to pick data from, such as C8 : C8
The contents of the cell C8 on North is : 120 =INDIRECT(G33&"!"&G34)

The =INDIRECT() created a reference to =NORTH!C8

Example 2

This example uses the same data as above, but this time the =SUM() function is used to calculate a total from a range of cells.

Type the name of the **sheet**, such as South : South
Type the **start** cell of the range, such as C5 : C5
Type the **end** cell of the range, such as C7 : C7
The sum of the range C5:C7 on South is : 1200

=SUM(INDIRECT(G44&"!"&G45&":"&G46))

The =INDIRECT() created a reference to =SUM(SOUTH!C5:C7)

INFO

	System Information	
Current directory	Err:504	=INFO("directory")
Available bytes of memory	Err:504	=INFO("memavail")
Memory in use	Err:504	=INFO("memused")
Total bytes of memory	Err:504	=INFO("totmem")
Number of active worksheets	1	=INFO("numfile")
Cell currently in the top left of the window	Err:504	=INFO("origin")
Operating system	Windows (32-bit) NT 5.01	=INFO("osversion")
Recalculation mode	Automatic	=INFO("recalc")
Excel version	680m14(Build:9134)	=INFO("release")
Name of system. (PC or Mac)	LINUX	=INFO("system")

What Does It Do?

This function provides information about the operating environment of the computer.

Syntax

=INFO(text)

text : This is the name of the item you require information about.

Formatting

The results will be shown as text or a number depending upon what was requested.

INT

Number	Integer	
1.5	1	=INT(C4)
2.3	2	=INT(C5)
10.75	10	=INT(C6)
-1.48	-2	=INT(C7)

What Does It Do ?

This function rounds a number down to the nearest whole number.

Syntax

=INT(Number)

Formatting

No special formatting is needed.

Example

The following table was used by a school to calculate the age a child when the school year started.
A child can only be admitted to school if they are over 8 years old.
The Birth Date and the Term Start date are entered and the age calculated.
Table 1 shows the age of the child with decimal places

Table 1

Birth Date	Term Start	Age	
1-Jan-80	1-Sep-88	8.67	=(D27-C27)/365.25
5-Feb-81	1-Sep-88	7.57	
20-Oct-79	1-Sep-88	8.87	
1-Mar-81	1-Sep-88	7.5	

Table 2 shows the age of the child with the Age formatted with no decimal places.
This has the effect of increasing the child age.

Table 2

Birth Date	Term Start	Age	
1-Jan-80	1-Sep-88	9	=(D38-C38)/365.25
5-Feb-81	1-Sep-88	8	
20-Oct-79	1-Sep-88	9	
1-Mar-81	1-Sep-88	8	

Table 3 shows the age of the child with the Age calculated using the =INT() function to remove the decimal part of the number to give the correct age.

Table 3

Birth Date	Term Start	Age	
1-Jan-80	1-Sep-88	8	=INT((D49-C49)/365.25)
5-Feb-81	1-Sep-88	7	
20-Oct-79	1-Sep-88	8	
1-Mar-81	1-Sep-88	7	

Note

The age is calculated by subtracting the Birth Date from the Term Start to find the age of the child in days.
The number of days is then divided by 365.25
The reason for using 365.25 is to take account of the leap years.

ISBLANK

Data	Is The Cell Blank	
1	FALSE	=ISBLANK(C4)
Hello	FALSE	=ISBLANK(C5)
	TRUE	=ISBLANK(C6)
25-Dec-98	FALSE	=ISBLANK(C7)

What Does It Do?

This function will determine if there is an entry in a particular cell.
It can be used when a spreadsheet has blank cells which may cause errors, but which will be filled later as the data is received by the user.
Usually the function is used in conjunction with the =IF() function which can test the result of the =ISBLANK()

Syntax

=ISBLANK(CellToTest)

Formatting

Used by itself the result will be shown as TRUE or FALSE.

Example

The following example shows a list of cheques received by a company.
When the cheque is cleared the date is entered.
Until the Cleared date is entered the Cleared column is blank.
While the Cleared column is blank the cheque will still be Outstanding.
When the Cleared date is entered the cheque will be shown as Banked.
The =ISBLANK() function is used to determine whether the Cleared column is empty or not.

Cheques Received		Date		Date		
Num	From	Received	Amount	Cleared	Banked	Outstanding
chq1	ABC Ltd	1-Jan-98	£100	2-Jan-98	100	0
chq2	CJ Design	1-Jan-98	£200	7-Jan-98	200	0
chq3	J Smith	2-Jan-98	£50		0	50
chq4	Travel Co.	3-Jan-98	£1,000		0	1000
chq5	J Smith	4-Jan-98	£250	6-Jan-98	250	0
			=IF(ISBLANK(F36),0,E36)			
				=IF(ISBLANK(F36),E36,0)		
				Totals	550	1050

ISERR

Cell to test	Result	
3	FALSE	=ISERR(D4)
#DIV/0!	TRUE	=ISERR(D5)
#NAME?	TRUE	=ISERR(D6)
#REF!	TRUE	=ISERR(D7)
Err:502	TRUE	=ISERR(D8)
Err:502	TRUE	=ISERR(D9)
#N/A	FALSE	=ISERR(D10)

What Does It Do ?

This function tests a cell and shows TRUE if there is an error value in the cell.
It will show FALSE if the contents of the cell calculate without an error, or if the error is the #NA message.

Syntax

=ISERR(CellToTest)
The CellToTest can be a cell reference or a calculation.

Formatting

No special formatting is needed.

Example

The following tables were used by a publican to calculate the cost of a single bottle of champagne, by dividing the cost of the crate by the quantity of bottles in the crate.

Table 1 shows what happens when the value zero 0 is entered as the number of bottles.
The #DIV/0 indicates that an attempt was made to divide by zero 0, which Excel does not do.

Table 1

Cost Of Crate :	£24	
Bottles In Crate :	0	
Cost of single bottle :	#DIV/0!	=E32/E33

Table 2 shows how this error can be trapped by using the =ISERR() function.

Table 2

Cost Of Crate :	£24	
Bottles In Crate :	0	
Cost of single bottle :	Try again!	=IF(ISERR(E40/E41),"Try again!",E40/E41)

ISERROR

Cell to test	Result	
3	FALSE	=ISERROR(D4)
#DIV/0!	TRUE	=ISERROR(D5)
#NAME?	TRUE	=ISERROR(D6)
#REF!	TRUE	=ISERROR(D7)
Err:502	TRUE	=ISERROR(D8)
Err:502	TRUE	=ISERROR(D9)
#N/A	TRUE	=ISERROR(D10)

What Does It Do ?

This function tests a cell or calculation to determine whether an error has been generated. It will show TRUE for any type of error and FALSE if no error is found.

Syntax

=ISERROR(CellToTest)

The CellToTest can be a cell reference or a formula.

Formatting

No special formatting is needed.

Example

The following tables was used to calculate the difference between two dates.

Table 1 shows an error due to the fact that the first entry was entered using an inappropriate date format.

Table 1

Start date :	Jan 01 98
End date :	5-Jan-98
Difference :	3-Jan-00

=D31-D30

Table 2 shows how the =ISERROR() function has been used to trap the error and inform the user that there has been an error in the data entry.

Table 2

Start date :	Jan 01 98
End date :	5-Jan-98
Difference :	3-Jan-00

=IF(ISERROR(D40-D39),"Error in data entry",D40-D39)

ISEVEN

Number	Is it Even	
1	0	=ISEVEN(C4)
2	1	=ISEVEN(C5)
2.5	1	=ISEVEN(C6)
2.6	1	=ISEVEN(C7)
3.5	0	=ISEVEN(C8)
3.6	0	=ISEVEN(C9)
Hello	1	=ISEVEN(C10)
1-Feb-98	0	=ISEVEN(C11)
1-Feb-96	1	=ISEVEN(C12)

What Does It Do ?

This function tests a number to determine whether it is even.
An even number is shown as TRUE an odd number is shown as FALSE.
Note that decimal fractions are ignored.
Note that dates can be even or odd.
Note that text entries result in the #VALUE! error.

Syntax

=ISEVEN(CellToTest)

Formatting

No special formatting is required.

ISLOGICAL

Cell To Test	Result	
FALSE	TRUE	=ISLOGICAL(D4)
TRUE	TRUE	=ISLOGICAL(D5)
	FALSE	=ISLOGICAL(D6)
20	FALSE	=ISLOGICAL(D7)
1-Jan-98	FALSE	=ISLOGICAL(D8)
Hello	FALSE	=ISLOGICAL(D9)
#DIV/0!	FALSE	=ISLOGICAL(D10)

What Does It Do ?

This function tests a cell to determine whether the cell contents are logical.
The logical values can only be TRUE or FALSE.
If the cell does contain a logical value, the result TRUE is shown.
If the cell does not contain a logical value, the result FALSE is shown.

Syntax

=ISLOGICAL(CellToTest)

Formatting

No special formatting is needed.

ISNA

Number	Result	
1	FALSE	=ISNA(C4)
Hello	FALSE	=ISNA(C5)
	FALSE	=ISNA(C6)
1-Jan-98	FALSE	=ISNA(C7)
#N/A	TRUE	=ISNA(C8)

What Does It Do?

This function tests a cell to determine whether it contains the Not Available error #N/A.
The #N/A is generated when a function cannot work properly because of missing data.
The #N/A can also be typed in to a cell by the user to indicate the cell is currently empty,
but will be used for data entry in the future.
The function is normally used with other functions such as the =IF() function.

Syntax

=ISNA(CellToTest)

Formatting

No special formatting is needed.

ISNONTEXT

Item To Test	Is It A Number?	
10	TRUE	=ISNONTEXT(C4)
Hello	FALSE	=ISNONTEXT(C5)
	TRUE	=ISNONTEXT(C6)
1-Jan-98	TRUE	=ISNONTEXT(C7)
1OO	FALSE	=ISNONTEXT(C8)

What Does It Do?

This functions tests an entry to determine whether it is a number, rather than text. It would be used to ensure that only numeric entries are used in calculations, rather than text which looks like a number, such as typing the letter O instead of zero 0. The function is normally used with other function such as the =IF() function.

Syntax

=ISNONTEXT(CellToTest)

Formatting

No special formatting.

Examples

The following table is used by an electrical retailer to calculate the selling price of an item based on the buying price and the shop mark-up.

Table 1 shows the #VALUE! error generated when a number, 300, is entered using the letter O instead of the zero 0.

Table 1

Item	Buying Price	Mark-up	Profit	
Radio	400	150%	600	
TV	800	200%	1600	
Video	**3OO**	150%	0	=D32*E32

Table 2 shows how the error is trapped using the =ISNONTEXT function and the =IF() function in the calculation.

Table 2

Item	Buying Price	Mark-up	Profit
Radio	400	150%	600
TV	800	200%	1600
Video	**3OO**	150%	Retype the Price

=IF(ISNONTEXT(D40),D40*E40,"Retype the Price")

ISNUMBER

Cell Entry	Result	
1	TRUE	=ISNUMBER(D4)
1-Jan-98	TRUE	=ISNUMBER(D5)
	FALSE	=ISNUMBER(D6)
#DIV/0!	FALSE	=ISNUMBER(D7)
Hello	FALSE	=ISNUMBER(D8)

What Does It Do ?

This function examines a cell or calculation to determine whether it is a numeric value.
If the cell or calculation is a numeric value the result TRUE is shown.
If the cell or calculation is not numeric, or is blank, the result FALSE is shown.

Syntax

=ISNUMBER(CellToTest)
The cell to test can be a cell reference or a calculation.

Formatting

No special formatting is needed.

Example

The following table was used by a personnel department to lookup the salary of an employee.
The employee can be entered as a Name or as a Numeric value.
The =ISNUMBER() function has been used to identify the type of entry made, and then
the =IF() decides which VLOOKUP to perform.

ID No.	Name	Salary
1	Alan	£10,000
2	Eric	£12,000
3	Carol	£8,000
4	Bob	£15,000
5	David	£12,000

Type Employee Name or ID :	eric
The Salary is :	£12,000

=IF(ISNUMBER(E35),VLOOKUP(E35,C29:E33,3,FALSE),VLOOKUP(E35,D29:E33,2,FALSE))

ISODD

Number	Is it Odd	
1	1	=ISODD(C4)
2	0	=ISODD(C5)
2.5	0	=ISODD(C6)
2.6	0	=ISODD(C7)
3.5	1	=ISODD(C8)
3.6	1	=ISODD(C9)
Hello	0	=ISODD(C10)
1-Feb-98	1	=ISODD(C11)
1-Feb-96	0	=ISODD(C12)

What Does It Do ?

This function tests a number to determine whether it is odd.
An odd number is shown as TRUE an even number is shown as FALSE.
Note that decimal fractions are ignored.
Note that dates can be odd or even.
Note that text entries result in the #VALUE! error.

Syntax

=ISODD(CellToTest)

Formatting

No special formatting is required.

ISREF

TRUE	=ISREF(A1)
FALSE	=ISREF(B99)
Err:508	=ISREF(Hello)
FALSE	=ISREF(10)
FALSE	=ISREF(NOW())
FALSE	=ISREF("A1")
Err:508	=ISREF(XX99)

What Does It Do ?

This function shows TRUE if given a cell address, or FALSE for any other type of value.
Its a bit of an odd one, and is normally used in macros rather than on the worksheet.

Syntax

=ISREF(ValueToTest)

The ValueToTest can be any type of data, but when used on the worksheet, it cannot be a reference to the contents of another cell, as the reference will itself be evaluated by the function.

Formatting

No special formatting is needed.

ISTEXT

Cell To Test	Result	
Hello	TRUE	=ISTEXT(D4)
1	FALSE	=ISTEXT(D5)
25-Dec-98	FALSE	=ISTEXT(D6)
	FALSE	=ISTEXT(D7)

What Does It Do ?

This functions tests an entry to determine whether it is text.
If the entry is text is shows TRUE.
If the entry is any other type it shows FALSE.

Syntax

=ISTEXT(CellToTest)

Formatting

No special formatting is needed.

Example

The following table was used by a personnel department to lookup the salary of an employee.
The employee can be entered as a Name or as a Numeric value.
The =ISTEXT() function has been used to identify the type of entry made, and then
the =IF() decides which VLOOKUP to perform.

ID No.	Name	Salary
1	Alan	£10,000
2	Eric	£12,000
3	Carol	£8,000
4	Bob	£15,000
5	David	£12,000

Type Employee Name or ID :	3
The Salary is :	£8,000

=IF(ISTEXT(E33),VLOOKUP(E33,D27:E31,2,FALSE),VLOOKUP(E33,C27:E31,3,FALSE))

LARGE

Values	Highest Value	800
120	2nd Highest Value	250
800	3rd Highest Value	120
100	4th Highest Value	120
120	5th Highest Value	100
250		

=LARGE(C4:C8,1)
=LARGE(C4:C8,2)
=LARGE(C4:C8,3)
=LARGE(C4:C8,4)
=LARGE(C4:C8,5)

What Does It Do ?

This function examines a list of values and picks the value at a user specified position in the list.

Syntax

=LARGE(ListOfNumbersToExamine,PositionToPickFrom)

Formatting

No special formatting is needed.

Example

The following table was used to calculate the top 3 sales figures between Jan, Feb and Mar.

Sales	Jan	Feb	Mar
North	£5,000	£6,000	£4,500
South	£5,800	£7,000	£3,000
East	£3,500	£2,000	£10,000
West	£12,000	£4,000	£6,000

Highest Value	£12,000	=LARGE(D24:F27,1)
2nd Highest Value	£10,000	=LARGE(D24:F27,2)
3rd Highest Value	£7,000	=LARGE(D24:F27,3)

Note

Another way to find the Highest and Lowest values would have been to use the =MAX() and =MIN() functions.

Highest	£12,000	=MAX(D24:F27)
Lowest	£2,000	=MIN(D24:F27)

LCM

Numbers		Least Common Multiple	
6	20	60	=LCM(C4,D4)
12	18	36	=LCM(C5,D5)
34	96	1632	=LCM(C6,D6)

What Does It Do ?

This function calculate the Least Common Multiple, which is the smallest number that can be divided by each of the given numbers.

Syntax

=LCM(Number1,Number2,Number3... through to Number29)

Formatting

No special formatting is needed.

LEFT

Text	Number Of Characters Required	Left String	
Alan Jones	1	A	=LEFT(C4,D4)
Alan Jones	2	Al	=LEFT(C5,D5)
Alan Jones	3	Ala	=LEFT(C6,D6)
Cardiff	6	Cardif	=LEFT(C7,D7)
ABC123	4	ABC1	=LEFT(C8,D8)

What Does It Do ?

This function displays a specified number of characters from the left hand side of a piece of text.

Syntax

=LEFT(OriginalText,NumberOfCharactersRequired)

Formatting

No special formatting is needed.

Example

The following table was used to extract the first name of a person from their full name.
The =FIND() function was used to locate position of the space between the first and second name.
The length of the first name is therefore the position of the space minus one character.
The =LEFT() function can now extract the first name based on the position of the space.

Full Name	First Name	
Alan Jones	Alan	=LEFT(C27,FIND(" ",C27)-1)
Bob Smith	Bob	=LEFT(C28,FIND(" ",C28)-1)
Carol Williams	Carol	=LEFT(C29,FIND(" ",C29)-1)

LEN

Text	Length	
Alan Jones	10	=LEN(C4)
Bob Smith	9	=LEN(C5)
Carol Williams	14	=LEN(C6)
Cardiff	7	=LEN(C7)
ABC123	6	=LEN(C8)

What Does It Do ?

This function counts the number of characters, including spaces and numbers, in a piece of text.

Syntax

=LEN(Text)

Formatting

No Special formatting is needed.

Example

This example shows how the =LEN() function is used in a formula which extracts the second name from a text entry containing both first and second names.

Original Text		
Carol Williams	6	=FIND(" ",C24)
		This is the position of the space.
Carol Williams	8	=LEN(C24)-FIND(" ",C24)
		This is the length of the second name.
		Calculated by taking the overall length of the complete name and subtracting the position of the space.
		=RIGHT(C24,LEN(C24)-FIND(" ",C24))
		This is just the second name.
		Calculated by using the =RIGHT() function to extract the rightmost characters up to the length of the second name.

LOOKUP (Array)

Name	Jan	Feb	Mar
Alan	10	80	**97**
Bob	20	90	**69**
Carol	30	100	**45**
David	40	110	**51**
Eric	50	120	**77**
Francis	60	130	**28**
Gail	70	140	**73**

Type a Name in this cell : Eric

The **March** value for this person is : 77 =LOOKUP(F12,D4:G10)

What Does It Do ?

This function looks for a piece of information in a list, and then picks an item from the last cell in the adjacent row or column.

It always picks the data from the end of the row or column, so it is no good if you need to pick data from part way across a list, (use VLOOKUP or HLOOKUP).

The way in which the function decides whether to pick from the row or column is based on the size of the table.

If the table has more **rows** than columns : the function will look **down** the left most column trying to find a match for the piece of information you asked it to look for. When a match is found, the function will look across to the right most column to pick the last entry on the row.

If the table has the **same** amount of rows and columns : the function will look **down** the left most column and work in just the same way as if the table had more rows than columns, as in the description above.

If the table has more **columns** than rows : the function will look **across** the top row trying to find a match for the piece of information you have asked it to look for. When a match is found, the function will then look down to the bottom cell of the column to pick the last entry of the column.

Syntax

=LOOKUP(WhatToLookFor,RangeToLookIn)

The WhatToLookFor should be a single item.

The RangeToLook in can be either horizontal or vertical.

Be careful not to include unnecessary heading in the range as these will cause errors.

Example 1

In this table there are more rows than columns, so the column heading of Jan is not included in the lookup

Example 2

In this table there are more columns than rows, so the row heading of Jan is not included in the lookup range.

	A	B	C	D	E	F	G	H	I	J
56		range.					Alan	Bob	Carol	David
57			Jan			Jan	100	100	100	100
58		Alan	100							
59		Bob	100							
60		Carol	100							
61		David	100							
62		Eric	100							
63		Fred	100							
64										
65		**Formatting**								
66		No special formatting is needed.								
67										
68		**Problems**								
69		The list of information to be looked through must be sorted in ascending order, otherwise errors								
70		will occur, either as #N/A or incorrect results.								
71										
72		Table 1 shows the Name column sorted alphabetically, the results of using =LOOKUP() will								
73		be correct.								
74										
75		Table 2 shows the same data, but not sorted. Sometimes the results will be correct, but other								
76		times the result will be an #N/A error or incorrect figure.								
77										
78		**Table 1**					**Table 2**			
79		Name	Jan	Feb	Mar		Name	Jan	Feb	Mar
80		**Alan**	10	80	**97**		**David**	40	110	**51**
81		**Bob**	20	90	**69**		**Eric**	50	120	**77**
82		**Carol**	30	100	**45**		**Alan**	10	80	**97**
83		**David**	40	110	**51**		**Bob**	20	90	**69**
84		**Eric**	50	120	**77**		**Carol**	30	100	**45**
85		**Francis**	60	130	**28**		**Francis**	60	130	**28**
86		**Gail**	70	140	**73**		**Gail**	70	140	**73**
87										
88		Name :	Eric				Name :	Eric		
89										
90		Value :	77				Value :	77		
91			=LOOKUP(C88,B80:E86)					=LOOKUP(H88,G80:J86)		

LOOKUP (Vector)

Name	Jan	Feb	Mar
Alan	10	**80**	97
Bob	20	**90**	69
Carol	30	**100**	45
David	40	**110**	51
Eric	50	**120**	77
Francis	60	**130**	28
Gail	70	**140**	73

Type a Name in this cell : Eric

The **Feb** value for this person is : Err:504 =LOOKUP(F12,D4:G10,F4:F10)

What Does It Do ?

This function looks for a piece of information in a list, and then picks an item from a second range of cells.

Syntax

=LOOKUP(WhatToLookFor,RangeToLookIn,RangeToPickFrom)

The WhatToLookFor should be a single item.

The RangeToLook in can be either horizontal or vertical.

The RangeToPickFrom must have the same number of cells in it as the RangeToLookin.

Be careful not to include unnecessary heading in the ranges as these will cause errors.

Formatting

No special formatting is needed.

Example

The following example shows how the =LOOKUP() function was used to match a name typed in cell G41 against the list of names in C38:C43. When a match is found the =LOOKUP() then picks from the second range E38:J38.

If the name Carol is used, the match is made in the third cell of the list of names, and then the function picks the third cell from the list of values.

RangeToLookIn
Alan
Bob
Carol
David
Eric
Fred

RangeToPickFrom					
5	10	15	20	25	30

Type a name : Carol

Value : 15

=LOOKUP(G41,C38:C43,E38:J38)

Problems

The list of information to be looked through must be sorted in ascending order, otherwise errors will occur, either as #N/A or incorrect results.

LOWER

Upper Case Text	Lower Case	
ALAN JONES	alan jones	=LOWER(C4)
BOB SMITH	bob smith	=LOWER(C5)
CAROL WILLIAMS	carol williams	=LOWER(C6)
CARDIFF	cardiff	=LOWER(C7)
ABC123	abc123	=LOWER(C8)

What Does It Do ?

This function converts all characters in a piece of text to lower case.

Syntax

=LOWER(TextToConvert)

Formatting

No special formatting is needed.

MATCH

	Names	Values
	Bob	250
	Alan	600
	David	1000
	Carol	4000

Type a name to look for : Alan

The position of Alan is : 2

=MATCH(E9,E4:E7,0)

Type a value : 1000

Value position : 3

=MATCH(I9,I4:I7,1)

What Does It Do ?

This function looks for an item in a list and shows its position.
It can be used with text and numbers.
It can look for an exact match or an approximate match.

Syntax

=MATCH(WhatToLookFor,WhereToLook,TypeOfMatch)
The TypeOfMatch either 0, 1 or -1.

Using 0 will look for an exact match. If no match is found the #NA error will be shown.

Using 1 will look for an exact match, or the next lowest number if no exact match exists.
If there is no match or next lowest number the error #NA is shown.
The list of values being examined must be sorted for this to work correctly.

Using -1 will look for an exact match, or the next highest number if no exact match exists.
If there is no exact match or next highest number the error #NA is shown.
The list must be sorted for this to work properly.

Examples 1

Using the 0 option suitable for an exact match.
The **Ascending** list gives the exact match.
The **Descending** list gives the exact match.
The **Wrong Value** list cannot find an exact match, so the #NA is shown.

Ascending	Descending	Wrong Value
10	40	10
20	30	20
30	20	30
40	10	40
20	20	25
2	3	#N/A

=MATCH(G45,G40:G43,0)

	A	B	C	D	E	F	G	H	I
49									
50		**Example 2**							
51		Using the 1 option suitable for a ascending list to find an exact or next lowest match.							
52		The **Ascending** list gives the exact match.							
53		The **Descending** list gives the #NA error.							
54		The **Wrong Value** list finds the **next lowest** number..							
55									
56			Ascending		Descending		Wrong Value		
57			10		40		10		
58			20		30		20		
59			30		20		30		
60			40		10		40		
61									
62			20		20		25		
63			2		#N/A		2		
64							=MATCH(G62,G57:G60,1)		
65									
66									
67		**Example 3**							
68		Using the -1 option suitable for a descending list to find an exact or next highest match.							
69		The **Ascending** list gives the #NA error.							
70		The **Descending** list gives the exact match.							
71		The **Wrong Value** list finds the **next highest** number.							
72									
73			Ascending		Descending		Wrong Value		
74			10		40		40		
75			20		30		30		
76			30		20		20		
77			40		10		10		
78									
79			20		20		25		
80			2		3		2		
81							=MATCH(G79,G74:G77,-1)		
82									

Example 4

The tables below were used to by a bus company taking booking for bus tours.
They need to allocate a bus with enough seats for the all the passengers.
The list of bus sizes has been entered in a list.
The number of passengers on the tour is then entered.
The =MATCH() function looks down the list to find the bus with enough seats.
If the number of passengers is not an exact match, the next biggest bus will be picked.
After the =MATCH() function has found the bus, the =INDEX() function has been used to look down the list again and pick out the actual bus size required.

	Bus Size
Bus 1	54
Bus 2	50
Bus 3	22
Bus 4	15
Bus 5	6

Passengers on the tour :	23
Bus size needed :	50

```
=INDEX(D95:D99,MATCH(H94,D95:D99,-1),0)
```

Example 5

The tables below were used by a school to calculate the exam grades for pupils.
The list of grade breakpoints was entered in a list.
The pupils scores were entered in another list.
The pupils scores are compared against the breakpoints.
If an exact match is not found, the next lowest breakpoint is used.
The =INDEX() function then looks down the Grade list to find the grade.

Exam Score	Grade
0	Fail
50	Pass
90	Merit
95	Distinction

	Pupil Score	Grade
Alan	60	Pass
Bob	6	Fail
Carol	97	Distinction
David	89	Pass

```
=INDEX(D111:D114,MATCH(G114,C111:C114,1),0)
```

MAX

	Values					Maximum	
	120	800	100	120	250	800	=MAX(C4:G4)
	Dates					Maximum	
	1-Jan-98	25-Dec-98	31-Mar-98	27-Dec-98	4-Jul-98	27-Dec-98	=MAX(C7:G7)

What Does It Do ?

This function picks the highest value from a list of data.

Syntax

=MAX(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Example

In the following example the =MAX() function has been used to find the highest value for each region, month and overall.

Sales	Jan	Feb	Mar		Region Max	
North	£5,000	£6,000	£4,500		£6,000	=MAX(C23:E23)
South	£5,800	£7,000	£3,000		£7,000	
East	£3,500	£2,000	£10,000		£10,000	
West	£12,000	£4,000	£6,000		£12,000	
Month Max	£12,000	£7,000	£10,000			
		=MAX(E23:E26)				
Overall Max	£12,000					
=MAX(C23:E26)						

MEDIAN

	Value1	Value2	Value3	Value4	Value5	Median	
4	20	50	10	30	40	30	=MEDIAN(C4:G4)
6	2000	1000	10	20	8000	1000	=MEDIAN(C6:G6)
8	10	20	40	40	40	40	=MEDIAN(C8:G8)

	Value1	Value2	Value3	Value4		Median	
11	20	40	30	10		25	=MEDIAN(C11:F11)
13	20	20	40	20		20	=MEDIAN(C13:F13)

What Does It Do ?

This function finds the median value of a group of values.
The median is not the average, it is the half way point where half the numbers in the group are larger than it and half the numbers are less than it.
If there is no exact median number in the group, the two nearest the half way point are added and their average is used as the median.

Syntax

=MEDIAN(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

MID

Text	Start Position	How Many Characters	Mid String	
ABCDEDF	1	3	ABC	=MID(C4,D4,E4)
ABCDEDF	2	3	BCD	=MID(C5,D5,E5)
ABCDEDF	5	2	ED	=MID(C6,D6,E6)

ABC-100-DEF	100	=MID(C8,5,3)
ABC-200-DEF	200	=MID(C9,5,3)
ABC-300-DEF	300	=MID(C10,5,3)

Item Size: Large	Large	=MID(C12,12,99)
Item Size: Medium	Medium	=MID(C13,12,99)
Item Size: Small	Small	=MID(C14,12,99)

What Does It Do ?

This function picks out a piece of text from the middle of a text entry.
The function needs to know at what point it should start, and how many characters to pick.
If the number of characters to pick exceeds what is available, only the available characters will be picked.

Syntax

=MID(OriginalText,PositionToStartPicking,NumberOfCharactersToPick)

Formatting

No special formatting is needed.

Example 1

The following table uses the =MID() function to extract a post code from a branch ID used by a company.
It is assumed that all branch ID's follow the same format with the letters identifying the postal region being in the 5th and 6th positions.

Branch ID	Postal Region	
DRS-CF-476	CF	=MID(C35,5,2)
DRS-WA-842	WA	=MID(C36,5,2)
HLT-NP-190	NP	=MID(C37,5,2)

Example 2

This example shows how to extract an item which is of variable length, which is inside a piece of text which has no standard format, other than the required text is always between two slash / symbols.

Full Branch Code	Postal Region
DRS/STC/872	STC
HDRS/FC/111	FC
S/NORTH/874	NORTH
HQ/K/875	K
SPECIAL/UK & FR/876	UK & FR

=MID(C50,FIND("/",C50)+1,FIND("/",C50,FIND("/",C50)+1)-FIND("/",C50)-1)

Find the first /, plus 1 for the Start of the code.
Find the second /, occurring after the first /

	A	B	C	D	E	F	G	H
55			Calculate the length of the text to extract, by subtracting the position					
56			of the first / from the position of the second /					

MIN

Values					Minimum	
120	800	100	120	250	100	=MIN(C4:G4)
Dates					Maximum	
1-Jan-98	25-Dec-98	31-Mar-98	27-Dec-98	4-Jul-98	1-Jan-98	=MIN(C7:G7)

What Does It Do ?

This function picks the lowest value from a list of data.

Syntax

=MIN(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Example

In the following example the =MIN() function has been used to find the lowest value for each region, month and overall.

Sales	Jan	Feb	Mar		Region Min	
North	£5,000	£6,000	£4,500		£4,500	=MIN(C23:E23)
South	£5,800	£7,000	£3,000		£3,000	
East	£3,500	£2,000	£10,000		£2,000	
West	£12,000	£4,000	£6,000		£4,000	
Month MIN	£3,500	£2,000	£3,000			
		=MIN(E23:E26)				
Overall MIN	£2,000					
=MIN(C23:E26)						

MINUTE

Number	Minute	
###	45	=MINUTE(D4)
9:15:00 PM	15	=MINUTE(D5)
0.02	28	=MINUTE(D6)
0.52	28	=MINUTE(D7)
1.52	28	=MINUTE(D8)

What Does It Do?

The function will show the minute of the hour based upon a time or a number.
Only the fraction part of the number is used as it is this which relates to time of day.

Syntax

=MINUTE(Number)

Formatting

The result will be shown as a normal number between 0 and 59.

Example

The =REPT() function has been used to make a digital display for the current time.
The time functions of =HOUR(), =MINUTE() and =SECOND() have been used in conjunction
with the =NOW() as the basis for the number of repeats.
To update the clock press the function key F9.

Clock	
Hour	||||||||||||||||| 17
Minute	||||||||||||||||||||||||||||||||||||||||||||||||||||| 45
Second	||||||||||||||||||||||||| 26

```
=REPT("|",HOUR(NOW()))&" "&TEXT(HOUR(NOW()),"00")
=REPT("|",MINUTE(NOW()))&" "&TEXT(MINUTE(NOW()),"00")
=REPT("|",SECOND(NOW()))&" "&TEXT(SECOND(NOW()),"00")
```

Related Information

To convert a time in hh:mm format to decimal format.

Enter a time in hh:mm format :	2:45	
The same time converted to a decimal :	2.75	=F38*24
To extract the hours as a decimal :	2	=INT(F38*24)
To extract the minutes as a decimal :	0.75	=MOD(F38*24,1)

To convert a time in decimal format to hh:mm format.

Enter a time in decimal format :	3.75	
The same time converted to hh:mm format is :	3:45	=F49/24
To extract the hours in hh:mm format :	3:00	=INT(F49)/24
To extract the minutes in hh:mm format :	0:45	=MOD(F49,1)/24

The three formula above have also been formatted as **hh:mm** using
the **Format**, **Cells**, **Number**, **Time** command.

MMULT

What Does It Do ?

This function multiplies one range of values with another range of values.
The ranges do not have to be of equal size.
The dimensions of the result range is in direct proportion to dimensions of the two input ranges.
It is an Array function and must be entered using the Ctrl+Shift+Enter combination.

Syntax

=MMULT(Range1,Range2)

Formatting

No special formatting is needed.

Example

The following tables were used by a company producing boxes of chocolates.
The types of chocolate produced were Milk, Dark and White.
The company boxed the chocolates in three differing mixtures of Milk, Dark and White.
In the run up to Christmas customers ordered various quantities of each box.
The chocolate company now needed to know what quantity of each type of chocolate to produce.
The =MMULT() function was used to multiply the contents of boxes by the customer orders.
The result of the =MMULT() is the total number of each type of chocolate to produce.

	Chocolates in the box		
Size	Milk	Dark	White
Giant	50	50	50
Standard	30	20	10
Economy	20	5	5

Customers Orders		
Giant	Standard	Economy
300	400	500

Quantity To Produce		
Milk	Dark	White
37,000	25,500	21,500

{=MMULT(C32:E32,C26:E28)}
In all three cells

How It Was Done

Cells C36 to E36 were selected.
The formula =MMULT(C32:E32,C26:E28) was typed, (but not yet entered).
The keys Ctrl+Shift+Enter were pressed to confirm the entry as an array.
The formula then showed the correct result.

Getting The Dimensions Correct

The dimensions of the Result range are directly related to the two input ranges.
The number of rows in the Result should be equal to the rows in Range1.
The number of columns in the Result should be equal to the columns in Range2.

Example 2

The following tables were used by the chocolate company to calculate the amount of ingredients needed to produce batches of chocolate.

The company has four factories, each of which has to order enough Butter, Eggs and Sugar to ensure they can meet production targets.

Range 1 contains the planned production of Milk and Dark chocolate for each factory.

Range 2 contains the amount Butter, Eggs and Sugar needed to make 1 unit of Milk or Plain.

The Result range shows the quantities of each ingredient that will have to be ordered to meet the production target.

Note the depth of the Result is the same as the depth of Range 1, and the width of the Result is the same as the width of Range 2.

Range 1

Production	Milk	Dark
Factory 1	20	0
Factory 2	20	1
Factory 3	10	5
Factory 4	20	10

Range 2

Ingredients	Butter	Eggs	Sugar
Milk	1	3	10
Dark	2	2	5

Result

Ingredients To Order	Butter	Eggs	Sugar
Factory 1	20	60	200
Factory 2	22	62	205
Factory 3	20	40	125
Factory 4	40	80	250

{=MMULT(C69:D72,G69:I70)}

In all cells

Hint

To get a feel for how the =MMULT() function operates, set all values in Range1 and Range2 to zero 0, then change a single value in each.

MOD

Number	Divisor	Remainder	
12	5	2	=MOD(C4,D4)
20	7	6	=MOD(C5,D5)
18	3	0	=MOD(C6,D6)
9	2	1	=MOD(C7,D7)
24	7	3	=MOD(C8,D8)

What Does It Do ?

This function calculates the remainder after a number has been divided by another number.

Syntax

=MOD(Number,Divisor)

Formatting

No special formatting is needed.

MODE

Value1	Value2	Value3	Value4	Value5	Mode	
20	50	10	10	40	10	=MODE(C4:G4)
40	20	40	10	40	40	=MODE(C6:G6)
10	10	99	20	20	10	=MODE(C8:G8)
20	20	99	10	10	10	=MODE(C9:G9)
10	20	20	99	10	10	=MODE(C10:G10)
10	20	30	40	50	#VALUE!	=MODE(C12:G12)

What Does It Do ?

This function displays the most frequently occurring number in a group of numbers.
For it to work correctly there must be at least two numbers which are the same.
If all the values in the group are unique the function shows the error #N/A.
When there is more than one set of duplicates, the number closest to the beginning
of the group will be used. (Which is not really an accurate answer!)

Syntax

=MODE(Range1,Range2,Range3... through to Range30)

Formatting

No special formatting is needed.

Example

The following table shows garments sold in a clothes shop.
The shopkeeper wants to keep track of the most commonly sold size.
The =MODE() function has been used to calulate this.

Order	Garmet	Size
001	Blouse	10
002	Skirt	10
003	Shirt	8
004	Blouse	10
005	Skirt	12
006	Dress	8
007	Shirt	10
008	Blouse	10
009	Dress	8
010	Shirt	10
011	Dress	12
012	Skirt	12
013	Skirt	10
014	Shirt	10
015	Dress	8
016	Shirt	10
017	Blouse	10
018	Blouse	8
019	Dress	10
020	Skirt	8

Most frequently ordered size :	10	
		=MODE(D33:D52)
Count of size 8 :	6	
		=COUNTIF(D33:D52,"8")
Count of size 10 :	11	
		=COUNTIF(D33:D52,"10")
Count of size 12 :	3	
		=COUNTIF(D33:D52,"12")

Note

If the =AVERAGE() function had been used the answer would have been : 9.7

	A	B	C	D	E	F	G	H	I	J
56		This figure is of no benefit to the shopkeeper as there are no garmets of this size!								

MONTH

Original Date	Month	
1-Jan-98	1	=MONTH(C4)
1-Jan-98	December	=MONTH(C5)

What Does It Do?

This function extracts the month from a complete date.

Syntax

=MONTH(Date)

Formatting

Normally the result will be a number, but this can be formatted to show the actual month by using Format,Cells,Number,Custom and using the code mmm or mmmm.

Example

The =MONTH function has been used to calculate the name of the month for your birthday.

Please enter your date of birth in the format dd/mm/yy	3/25/1962	
You were born in	January	=MONTH(F20)

MROUND

Number	Multiple	Rounded Value	
110	50	100	=MROUND(C4,D4)
120	50	100	=MROUND(C5,D5)
150	50	150	=MROUND(C6,D6)
160	50	150	=MROUND(C7,D7)
170	50	150	=MROUND(C8,D8)

What Does It Do ?

This function rounds a number up or down to the nearest multiple specified by the user.

Syntax

=MROUND(NumberToRound,MultipleToUse)

Formatting

No special formatting is needed.

N

Original	Converted	
1	1	=N(C4)
3 1/2	3.5	=N(C5)
3.5	3.5	=N(C6)
3.50%	0.04	=N(C7)
25-Dec-98	36154	=N(C8)
TRUE	1	=N(C9)
FALSE	0	=N(C10)
Hello	0	=N(C11)
	0	=N(C12)

What Does It Do ?

This function converts a numeric entry to its mathematical value.
Anything which will not convert is shown as 0 zero.
Excel does not really need this function, due to the fact that Excel calculates in this way naturally. The function is included for compatibility with other spreadsheet programs.

Syntax

=N(NumericEntry)

Formatting

No special formatting is needed.

NA

	#N/A	=NA()

Value	Test	
10	11	=IF(ISBLANK(C6),NA(),C6+1)
	#N/A	=IF(ISBLANK(C7),NA(),C7+1)
30	31	=IF(ISBLANK(C8),NA(),C8+1)

	Sales	
North	100	
South	#N/A	=NA()
East	#N/A	=NA()
West	200	
Total	#N/A	=SUM(D11:D14)

What Does It Do ?

This function is a place marker used to indicate that required information is Not Available.
It can be type directly in to a cell as =NA() or it can be used as part of a calculation.
When the =NA() is used, any calculations which depend upon the cell will also show #NA.
It is used to indicate that all the data has not yet been entered in to the spreadsheet.

Syntax

=NA()

Formatting

No special formatting is required.

Example

The following table was used by a company to calculate the monthly Wage of an employee.
The Salary and Tax percentage are entered.
The Tax is then deducted from the Salary to calculate the Wage.

Table 1 shows that when the Tax is not entered, the Wage is still calculated.
On a large spreadsheet this may go unnoticed and the wrong Wage paid.

Table 1

	Salary	Tax %	Pay	
Alan	1000	25%	750	=C39-C39*D39
Bob	1000		1000	=C40-C40*D40
Carol	1000	20%	800	=C41-C41*D41

Table 2 shows how the =NA() has been inserted in the unknown Tax to act as a reminder that the Tax still needs to be entered.

Table 2

	Salary	Tax %	Pay	
Alan	1000	25%	750	=C49-C49*D49
Bob	1000	#N/A	#N/A	=C50-C50*D50
Carol	1000	20%	800	=C51-C51*D51

NETWORKDAYS

Start Date	End Date	Work Days	
1-Mar-98	7-Mar-98	5	=NETWORKDAYS(C4,D4)
25-Apr-98	30-Jul-98	69	=NETWORKDAYS(C5,D5)
24-Dec-98	5-Jan-99	9	=NETWORKDAYS(C6,D6)

What Does It Do?

This function will calculate the number of working days between two dates.
It will exclude weekends and any holidays.

Syntax

=NETWORKDAYS(StartDate,EndDate,Holidays)

Holidays : This is a list of dates which will be excluded from the calculation, such as Xmas and Bank holidays.

Formatting

The result will be shown as a number.

Note

The calculation does not include the last day. The result of using 1-Jan-98 and 5-Jan-98 will give a result of 4. To correct this add 1 to the result. =NETWORKDAYS(Start,End,Holidays)+1

Example

The following example shows how a list of Holidays can be created.

Start Date	End Date	Work Days	
Mon 02-Mar-98	Fri 06-Mar-98	5	=NETWORKDAYS(B28,C28,C33:C37)
Mon 02-Mar-98	Fri 13-Mar-98	10	=NETWORKDAYS(B29,C29,C33:C37)
Mon 27-Apr-98	Fri 01-May-98	4	=NETWORKDAYS(B30,C30,C33:C37)

	Holidays
Bank Holiday	1-May-98
Xmas	25-Dec-98
New Year	1-Jan-97
New Year	1-Jan-98
New Year	1-Jan-99

	A	B	C	D	E	F	G	H	I	J
1	Northern data.									
2	Used by the example for the =INDIRECT() function.									
3		Alan								
4			Jan	Feb	Mar	Total				
5		Alan	10	20	30	60				
6		Bob	40	50	60	150				
7		Carol	70	80	90	240				
8		Total	120	150	180	450				
9										

NOT

Cells To Test		Result	
10	20	TRUE	=NOT(C4>D4)
10	20	TRUE	=NOT(C5=D5)
10	20	FALSE	=NOT(C6<D6)
1-Jan-98	1-Feb-98	TRUE	=NOT(C7>D7)
Hello	Goodbye	TRUE	=NOT(C8=D8)
Hello	Hello	FALSE	=NOT(C9=D9)

What Does It Do ?

This function performs a test to see if the test fails. (A type of reverse logic).
If the test fails, the result is TRUE.
If the test is met, then the result is FALSE.

Syntax

=NOT(TestToPerform)
The TestToPerform can be reference to cells or another calculation.

Formatting

No special formatting is needed.

Example

The following table was used by a library to track books borrowed.
The date the book was Taken out is entered.
The period of the Loan is entered.
The date the book was returned is entered.
The =NOT() function has been used to calculate whether the book was returned within the correct time, by adding the Loan value to the Taken date.
If the book was not returned on time the result Overdue is shown, otherwise OK is shown.

Taken	Loan	Returned	Status	
1-Jan-98	14	5-Jan-98	OK	=IF(NOT(D33<=B33+C33),"Overdue","OK")
1-Jan-98	14	15-Jan-98	OK	=IF(NOT(D34<=B34+C34),"Overdue","OK")
1-Jan-98	14	20-Jan-98	Overdue	=IF(NOT(D35<=B35+C35),"Overdue","OK")

NOW

The current Date and Time	
10/22/2008 17:45	=NOW()
10/22/08 05:45 PM	=NOW()

What Does It Do?

This function shows the current date and time. The result will be updated each time the worksheet is opened and every time an entry is made anywhere on the worksheet.

Syntax

=NOW()

Formatting

The result will be shown as a date and time. If it is formatted to show as a number the integer part is used for the date and the decimal portion represent the time.

	A	B	C	D	E	F	G	H	I
1	**ODD**								
2									
3			Number	Rounded To Next Odd					
4			2	3	=ODD(C4)				
5			2.4	3	=ODD(C5)				
6			2.9	3	=ODD(C6)				
7			3	3	=ODD(C7)				
8			3.4	5	=ODD(C8)				
9			3.9	5	=ODD(C9)				
10									
11		**What Does It Do ?**							
12		This function rounds a number up to the next highest whole odd number.							
13									
14		**Syntax**							
15		=ODD(NumberToBeRounded)							
16									
17		**Formatting**							
18		No special formatting is needed.							

OR

Order No.	Cost	Payment Type	Handling Charge	
AB001	1000	Cash	£-	=IF(OR(E4="Visa",E4="Delta"),5,0)
AB002	1000	Visa	£5	=IF(OR(E5="Visa",E5="Delta"),5,0)
AB003	2000	Cheque	£-	=IF(OR(E6="Visa",E6="Delta"),5,0)
AB004	5000	Delta	£5	=IF(OR(E7="Visa",E7="Delta"),5,0)

What Does It Do?

This function tests two or more conditions to see if any of them are true.
It can be used to test that at least one of a series of numbers meets certain conditions.
Normally the OR() function would be used in conjunction with a function such as =IF().

Syntax

=OR(Test1,Test2)

Note that there can be up to 30 possible tests.

Formatting

When used by itself it will show TRUE or FALSE.

Example

The following table shows a list of orders taken by a company.
A handling charge of £5 is made on all orders paid by Visa or Delta cards.
The =OR() function has been used to determine whether the charge needs to be applied.

Order No.	Cost	Payment Type	Handling Charge	
AB001	1000	Cash	£-	=IF(OR(E27="Visa",E27="Delta"),5,0)
AB002	1000	Visa	£5	
AB003	2000	Cheque	£-	
AB004	5000	Delta	£5	

	A	B	C	D	E	F	G	H	I
1	**Ordering Stock**								
2									
3	This is an example of a spreadsheet to calculate the best time interval to order stock.								
4									
5	**Scenario**								
6		A garage fits exhaust systems.							
7		The manager orders the exhausts on a regular basis.							
8		Each time an order is made for new stock, there is a fixed administrative cost.							
9		The exhausts are kept in stock until needed.							
10		Keeping the exhausts in stock incurs a cost due to capital tied up and warehouse costs.							
11		The supplier of the Exhausts gives a discount on large orders.							
12									
13	**Objective**								
14	Find the time interval to order stock which will result in the lowest Admin and Warehouse costs.								
15									

	A	B	C	D	E	F	G	H	I
16	**Input Data**								
17					Cost of a single Exhaust system :		£75		
18			Cost of keeping Exhaust in stock. (As a % of the stock value) :				12%		
19					Quantity of Exhausts used per day :		10		
20			Admin cost each time new Exhausts are ordered :				£25		
21			Average quantity of Exhausts in stock (As % of ordered quantity) :				0.5		
22			Ordering Intervals to evaluate. (Expressed in Days) :				2		
23									
24			Suppliers first Price Break and Discount% offered :				200	1%	
25			Suppliers second Price Break and Discount% offered :				750	5%	
26									
27	**Output**								
28	Ordering Interval In Days	Quantity Per Order	Order Value	Order Discount	Orders Per Year	Annual Admin Cost	Annual Ware house Costs	Annual Total	The Best Ordering Interval
29	1	10	£750	£-	365	£9,125	£45	£9,170	-
30	2	20	£1,500	£-	183	£4,575	£90	£4,665	-
31	4	40	£3,000	£-	92	£2,300	£180	£2,480	-
32	6	60	£4,500	£-	61	£1,525	£270	£1,795	-
33	8	80	£6,000	£-	46	£1,150	£360	£1,510	-
34	10	100	£7,500	£-	37	£925	£450	£1,375	-
35	12	120	£9,000	£-	31	£775	£540	£1,315	-
36	14	140	£10,500	£-	27	£675	£630	£1,305	-
37	16	160	£12,000	£-	23	£575	£720	£1,295	-
38	18	180	£13,500	£-	21	£525	£810	£1,335	-
39	20	200	£15,000	£150	19	£475	£900	£1,225	**Best**
40	22	220	£16,500	£165	17	£425	£990	£1,250	-
41	24	240	£18,000	£180	16	£400	£1,080	£1,300	-
42	26	260	£19,500	£195	15	£375	£1,170	£1,350	-
43	28	280	£21,000	£210	14	£350	£1,260	£1,400	-
44	30	300	£22,500	£225	13	£325	£1,350	£1,450	-
45	32	320	£24,000	£240	12	£300	£1,440	£1,500	-
46	34	340	£25,500	£255	11	£275	£1,530	£1,550	-
47	36	360	£27,000	£270	11	£275	£1,620	£1,625	-
48	38	380	£28,500	£285	10	£250	£1,710	£1,675	-
49	40	400	£30,000	£300	10	£250	£1,800	£1,750	-
50	42	420	£31,500	£315	9	£225	£1,890	£1,800	-
51	44	440	£33,000	£330	9	£225	£1,980	£1,875	-
52	46	460	£34,500	£345	8	£200	£2,070	£1,925	-
53	48	480	£36,000	£360	8	£200	£2,160	£2,000	-
54	50	500	£37,500	£375	8	£200	£2,250	£2,075	-
55	52	520	£39,000	£390	8	£200	£2,340	£2,150	-
56	54	540	£40,500	£405	7	£175	£2,430	£2,200	-
57	56	560	£42,000	£420	7	£175	£2,520	£2,275	-
58	58	580	£43,500	£435	7	£175	£2,610	£2,350	-
59	60	600	£45,000	£450	7	£175	£2,700	£2,425	-
60									
61	**Things To Try**								
62		Change the Discount % to 0% and 0%.							
63		Change the Ordering Interval to 1 or 30.							
64		Change the Cost of the Exhaust making it cheaper or more expensive.							
65		Change the Quantity used per day to a larger or smaller number.							
66									

	A	B	C	D	E	F	G	H	I
67	**Explanation**								
68	Column A	**Ordering Interval In Days**							
69		The first of these cells has the value 1 entered in it.							
70		This is the smallest ordering period, which would require stock to be ordered every day.							
71		The second cell picks the ordering interval from the Input Data table.							
72		The third and subsequent cells add the ordering interval to the previous cell to create							
73		a list of values of the same interval.							
74									
75	Column B	**Quantity Per Order**							
76		This is the number of Exhausts which will need to be ordered.							
77									
78		Calculation :	OrderingInterval * QuantityUsedPerDay						
79									
80	Column C	**Order Value**							
81		This is the value of the Order before any discount.							
82									
83		Calculation :	QuantityOrdered * CostOfExhaust						
84									
85	Column D	**Order Discount**							
86		The discount which can be subtracted from the order value.							
87		The discount is only given on orders which are equal to or greater than the							
88		Price Break values set by the supplier.							
89									
90		Calculation :	OrderValue * SupplierDiscount						
91			The supplier discount is calculated using the =IF() and the =AND() functions.						
92									
93			If the OrderQuantity is equal to or above the first Price Break, but below						
94			the second Price Break, then the first Price Break discount is used.						
95			=C29***IF(AND(B29>=$G$24,B29<$G$25),$H$24**,IF(B29>=$G$25,$H$25,0))						
96									
97			If the OrderQuantity is equal to or above the second Price Break,						
98			the second Price Break discount is used.						
99			=C29*IF(AND(B29>=$G$24,B29<$G$25),$H$24,**IF(B29>=$G$25,$H$25**,0))						
100									
101			If the OrderQuantity does not qualify for a discount, zero discount is used.						
102			=C29*IF(AND(B29>=$G$24,B29<$G$25),$H$24,IF(B29>=$G$25,$H$25,**0**))						
103									
104	Column E	**Orders Per Year**							
105		This is how many orders will need to be made based upon the ordering interval.							
106		With an interval of 1, there will have to be 365 orders.							
107									
108		Calculation :	365/OrderingInterval						
109			This calculation may give results which are decimal, such as 2.3						
110			This decimal will cause problems, due to the fact that the number of						
111			orders must always be a whole number.						
112			The =CEILING() function has been used to 'round up' any decimals to						
113			the next highest whole number.						
114			=CEILING(365/A29,1)						
115									

	A	B	C	D	E	F	G	H	I
116	Column F	**Annual Admin Costs**							
117		This is the administration costs involved in making the orders.							
118									
119		Calculation :	OrdersPerYear * AdminCost						
120			=E29*$G$20						
121									
122	Column G	**Annual Warehouse Costs**							
123		This is the cost of keeping the stock in the warehouse.							
124		It is based on the managers knowledge that on average the stock level is 50% of the							
125		quantity ordered.							
126									
127		Calculation :	QuantityOrdered * AverageStockLevel) * ExhaustCost * WarehousingCost						
128			=(B29*$G$21)*$G$17*$G$18						
129									
130	Column H	**Annual Total**							
131		This is the full yearly cost of ordering the Exhausts, based upon how frequently the							
132		orders are made.							
133		It does not take in to account the actual costs of the Exhausts, as the manager only							
134		wants to know what the lowest values for the overheads associated with ordering and							
135		storing the exhaust systems.							
136		However, the Discount figure is taken into account as this can be used to offset some							
137		of the overheads.							
138									
139		Calculation :	AnnualAdminCosts + AnnualWarehouseCosts - OrderDiscount						
140			=F29+G29-D29						
141									
142	Column I	**The Best Ordering Interval**							
143		This shows the Best ordering interval, giving the lowest annual overheads.							
144		It compares the value in column H against the minimum value for all of column H.							
145		If the two values match the word Best is shown, otherwise a dash is shown.							
146			=IF(H29=MIN($H$29:$H$59),"Best","-")						

	A	B	C	D	E	F	G	H	I	J
1										
2		Box size	Sample	Packer1	Packer2	Packer3	Packer4			
3		Small	1	10	10	10	10			
4		Medium	1	20	20	20	21			
5		Large	1	30	28	35	30			
6		Small	2	11	9	10	10			
7		Medium	2	21	20	0	20			
8		Large	2	31	28	30	30			
9		Small	3	8	10	12	10			
10		Medium	3	22	20	20	19			
11		Large	3	32	28	30	30			
12										
13		Box size	Sample	Packer1	Packer2	Packer3	Packer4			

PERMUT

Pool Of Items	Items In A Group	Permutations	
4	2	12	=PERMUT(C4,D4)
4	3	24	=PERMUT(C5,D5)
10	4	5040	=PERMUT(C6,D6)
26	6	165,765,600	=PERMUT(C7,D7)

What Does It Do ?

This function calculates the maximum number of permutations given a fixed number of items.
The internal order is significant, so AB and BA will be considered as two possible permutations.
It could be used to calculate the possible number of 4 digit passwords from the digits 0 to 9.

Syntax

=PERMUT(PoolToPickFrom,ItemsInAGroup)

Formatting

No special formatting is needed.

Example

The following table was used to calculate the total number of 8 letter passwords which can be created by using all 26 letters of the alphabet.

Letter In Alphabet	26
Password Size	8
Permutations	62,990,928,000

In the case of a two letter password made from the letter A, B, C and D, the following twelve permutations would be possible.

ABCD

Password 1	**AB**	Password 7	**BA**
Password 2	**AC**	Password 8	**CA**
Password 3	**AD**	Password 9	**DA**
Password 4	**BC**	Password 10	**CB**
Password 5	**BD**	Password 11	**DB**
Password 6	**CD**	Password 12	**DC**

PI

π
3.14159265358979

=PI()

What Does It Do ?

This function is equal to the value of Pi.
It is correct to 15 decimal places.
It does not need any input, it is a self contained function.

Syntax

=PI()

Formatting

No special formatting is needed.

Example

To calculate the area of a circle.

Radius	Area	
5	78.54	=PI()*(C21^2)
25	1963.50	

POWER

Number	Power	Result	
3	2	9	=POWER(C4,D4)
3	4	81	=POWER(C5,D5)
5	2	25	=POWER(C6,D6)
5	4	625	=POWER(C7,D7)

What Does It Do ?

This function raises a number to a user specified power.
It is the same as using the ^ operator, such as 3^4, which result is 81.
Both the POWER() function and the ^ operator are the same as using 3*3*3*3.

Syntax

=POWER(NumberToBeRaised,Power)

Formatting

No special formatting is needed.

Example

To calculate the area of a circle.

Radius	Area	
5	78.54	=PI()*POWER(C22,2)
25	1963.50	

PRODUCT

Numbers		Product	
2	3	6	=PRODUCT(C4,D4)
5	10	50	=PRODUCT(C5:D5)
3	7	210	=PRODUCT(C6:D6,10)
		6300	=PRODUCT(C4:D6)

What Does It Do ?

This function multiples a group of numbers together.
It is the same as using 2*3*5*10*3*7, which results in 6300.

Syntax

=PRODUCT(Number1,Number2,Number3... through to Number30)
or
=PRODUCT(RangeOfNumbers)
or
=PRODUCT(Number1,Range,Number2...)

Formatting

No special formatting is needed.

PROPER

Original Text	Proper	
alan jones	Alan Jones	=PROPER(C4)
bob smith	Bob Smith	=PROPER(C5)
caRol wILLIAMS	Carol Williams	=PROPER(C6)
cardiff	Cardiff	=PROPER(C7)
ABC123	Abc123	=PROPER(C8)

What Does It Do ?

This function converts the first letter of each word to uppercase, and all subsequent letters are converted to lower case.

Syntax

=PROPER(TextToConvert)

Formatting

No special formatting is needed.

QUARTILE

Values		Quarter No.	Quartile	
1		0	1	=QUARTILE(C4:C8,E4)
25		1	25	=QUARTILE(C4:C8,E5)
50		2	50	=QUARTILE(C4:C8,E6)
75		3	75	=QUARTILE(C4:C8,E7)
100		4	100	=QUARTILE(C4:C8,E8)

Values				Quarter No.	Quartile	
817	104	640	767	0	104	=QUARTILE(C12:F16,H12)
748	756	369	703	1	285.75	=QUARTILE(C12:F16,H13)
372	993	294	261	2	489	=QUARTILE(C12:F16,H14)
487	384	185	491	3	750	=QUARTILE(C12:F16,H15)
140	607	894	182	4	993	=QUARTILE(C12:F16,H16)

What Does It Do ?

This function examines a group of values and then shows the values which are of the upper limits of the 1st, 2nd, 3rd and 4th quarters of the data.

The Quartile of 0 (zero) is actually lowest value, which can be obtained using the =MIN() function.

The Quartile of 4 is actually highest value, which can be obtained using the =MAX() function.

Syntax

=QUARTILE(RangeToBeExamined,QuartileValue)

The QuartileValue can only be 0,1,2,3 or 4.

Formatting

No special formatting is needed.

QUOTIENT

Number	Divisor	Result	
12	5	2	=QUOTIENT(C4,D4)
20	3	6	=QUOTIENT(C5,D5)
46	15	3	=QUOTIENT(C6,D6)

What Does It Do ?

This function calculates the number of times a number can be divided by another number.
It ignores any remainder, only showing the whole number.

Syntax

=QUOTIENT(NumberToBeDivided,Divisor)

Formatting

No special formatting is needed.

Example

The following example was used by a drinks merchant to calculate the number of crates which could be packed using bottles in stock.
The merchant can only sell full crates.

Table 1 calculates the crates by simple division. This however shows decimal fractions which are not needed.

Table 1

Item	Bottles To Pack	Bottles Per Crate	Crates Needed	
Wine	126	12	10.5	=D28/E28
Champagne	200	8	25	
Rum	15	4	3.75	
Beer	250	20	12.5	

Table 2 uses the =QUOTIENT() function to remove the decimal fraction to give the correct result.

Table 2

Item	Bottles To Pack	Bottles Per Crate	Crates Needed	
Wine	126	12	10	=QUOTIENT(D39,E39)
Champagne	200	8	25	
Rum	15	6	2	
Beer	250	20	12	

RAND

Random greater than or equal to 0 but less than 1.

0.2 =RAND()

Random greater than or equal to 0 but less than 10

3.44 =RAND()*10

Random between 5 and 10.

9.17 =RAND()*(10-5)+5

What Does It Do ?

This function creates a random number >=0 but <1.
The number will change each time the worksheet recalculates, or when F9 is pressed.

Syntax

=RAND()

Formatting

No special formatting is needed.

Examples

The following examples show how the =RAND() function has been used to randomly sort list of information.

A list of cards has been entered in column C, and =RAND() in column D.
By clicking inside the random numbers and then using Data, Sort or the Sort button the cards will be shuffled.

The same technique has been used to generate a list of six winning lottery numbers.

Card	Random	Lottery	Random
Clubs 8	0.56	29	0.99
Clubs 6	0.39	34	0.35
Diamond 9	0.59	30	0.39
Spades 13	0.73	41	0.97
Clubs 9	0.29	40	0.25
Diamond 7	0.95	37	0.09
Diamond 4	0.81	26	0.42
Clubs 10	0.47	32	0.71
Spades 3	0.75	21	0.17
Hearts 6	0.96	19	0.46
Hearts 4	0.55	7	0.9
Diamond 8	0.89	10	0.55
Hearts 11	0.43	16	0.45
Clubs 3	0.86	8	0.9
Clubs 13	0.7	48	0.4
Spades 5	0.98	43	0.18
Diamond 3	0.42	44	0.86
Spades 2	0.53	4	0.8
Diamond 6	0.28	3	0.79
Clubs 5	0.19	45	0.24
Spades 1	0.15	47	0.06
Clubs 12	0.55	49	0.11
Hearts 10	0.48	35	0.61

	A	B	C	D	E	F	G	H	I
56			Hearts 13	0.83		27	0.73		
57			Spades 7	0.79		1	0.96		
58			Spades 6	0.01		13	0.26		
59			Diamond 12	0.56		31	0.6		
60			Hearts 3	0.98		5	0.96		
61			Hearts 5	0.9		18	0.46		
62			Hearts 8	0.7		39	0.94		
63			Hearts 1	0.45		23	0.79		
64			Diamond 13	0.46		12	0.45		
65			Hearts 9	0.08		11	0.29		
66			Clubs 4	0.04		20	0.18		
67			Diamond 5	0.19		33	0.42		
68			Spades 4	0.38		42	0.55		
69			Clubs 1	0.99		24	0.27		
70			Spades 8	1		2	0.85		
71			Hearts 7	0.85		14	0.26		
72			Diamond 1	0.73		25	0.44		
73			Clubs 2	0.96		9	0.31		
74			Hearts 2	0.4		38	0.16		
75			Diamond 11	0.62		15	0.99		
76			Clubs 7	0.39		28	0.75		
77			Spades 12	0.26		17	0.06		
78			Spades 10	0.32		6	0.39		
79			Clubs 11	0.37		22	0.93		
80			Diamond 2	0.68		46	0.91		
81			Diamond 10	0.85		36	0.19		
82			Spades 9	0.65					
83			Spades 11	0.86					
84			Hearts 12	1					

RANDBETWEEN

Low	High	Random	
5	10	9	=RANDBETWEEN(C4,D4)
1	49	8	=RANDBETWEEN(C5,D5)

What Does It Do ?

This function produces a random whole number between two specified numbers.
The random number will change each time the spreadsheet is recalculated or F9 is pressed.

Syntax

=RANDOMBETWEEN(LowLimit,HighLimit)

Formatting

No special formatting is needed.

Example

The following table shows how the =RANDBETWEEN() has been used to generate six numbers to use for the National Lottery.
Note that the function does not check to ensure all numbers are unique, the same number could be generated twice or more.

Lottery Numbers		The Winning Ticket!		
1	49	12	=RANDBETWEEN($C$24,$D$24)	Number 1
		41	=RANDBETWEEN($C$24,$D$24)	Number 2
Press function Key		38	=RANDBETWEEN($C$24,$D$24)	Number 3
F9 to recalculate.		48	=RANDBETWEEN($C$24,$D$24)	Number 4
		39	=RANDBETWEEN($C$24,$D$24)	Number 5
		2	=RANDBETWEEN($C$24,$D$24)	Number 6
		29	=RANDBETWEEN($C$24,$D$24)	Bonus ball
		All OK		

{=IF(SUM(1/COUNTIF(E24:E30,E24:E30))<>7,"Duplicates! Spin again","All OK")}

This formula is used to determine whether all the numbers are different.
It is entered as an array using Ctrl+Shift+Enter.

RANK

Values	Ranking Position High to Low	
7	4	=RANK(C4,C4:C8)
4	5	=RANK(C5,C4:C8)
25	1	=RANK(C6,C4:C8)
8	3	=RANK(C7,C4:C8)
16	2	=RANK(C8,C4:C8)

Values	Ranking Position Low to High	
7	2	=RANK(C11,C11:C15,1)
4	1	=RANK(C12,C11:C15,1)
25	5	=RANK(C13,C11:C15,1)
8	3	=RANK(C14,C11:C15,1)
16	4	=RANK(C15,C11:C15,1)

Values	Ranking Position High to Low	
10	5	=RANK(C18,C18:C22)
30	2	=RANK(C19,C18:C22)
20	4	=RANK(C20,C18:C22)
30	2	=RANK(C21,C18:C22)
40	1	=RANK(C22,C18:C22)

What Does It Do ?

This function calculates the position of a value in a list relative to the other values in the list.
A typical usage would be to rank the times of athletes in a race to find the winner.
The ranking can be done on an ascending (low to high) or descending (high to low) basis.
If there are duplicate values in the list, they will be assigned the same rank. Subsequent ranks would not follow on sequentially, but would take into account the fact that there were duplicates.
If the numbers 30, 20, 20 and 10 were ranked, 30 is ranked as 1, both 20's are ranked as 2, and the 10 would be ranked as 4.

Value	Rank	
30	1	=RANK(B34,B34:B37)
20	2	=RANK(B35,B34:B37)
20	2	=RANK(B36,B34:B37)
10	4	=RANK(B37,B34:B37)

Syntax

=RANK(NumberToRank,ListOfNumbers,RankOrder)
The RankOrder can be 0 zero or 1.
Using 0 will rank larger numbers at the top. (This is optional, leaving it out has the same effect).
Using 1 will rank small numbers at the top.

Formatting

No special formatting is needed.

Example

The following table was used to record the times for athletes competing in a race.
The =RANK() function was then used to find their race positions based upon the finishing times.

Athlete	Time	Race Position

	A	B	C	D	E	F	G	H	I
53		John	1:30	4	=RANK(C53,C53:C58,1)				
54		Alan	1:45	6	=RANK(C54,C53:C58,1)				
55		David	1:02	1	=RANK(C55,C53:C58,1)				
56		Brian	1:36	5	=RANK(C56,C53:C58,1)				
57		Sue	1:27	3	=RANK(C57,C53:C58,1)				
58		Alex	1:03	2	=RANK(C58,C53:C58,1)				

REPLACE

Original Text	Start Position	Characters To Replace	New Character	Modified Text	
ABCDEFGH	2	1	x	AxCDEFGH	=REPLACE(C4,D4,E4,F4)
ABCDEFGH	2	5	x	AxGH	=REPLACE(C5,D5,E5,F5)
ABCDEFGH	2	1	hello	AhelloCDEFGH	=REPLACE(C6,D6,E6,F6)
ABCDEFGH	2	5	hello	AhelloGH	=REPLACE(C7,D7,E7,F7)

What Does It Do ?

This function replaces a portion of text with a new piece of text.
You need to specify where the replacement should start, how many characters to remove and what the new replacement text should be.

Syntax

=REPLACE(OriginalText,StartPosition,NumberOfCharactersToReplace,NewText)

Formatting

No special formatting is needed.

REPT

Text To Repeat	Number Of Repeats	Repeated Text	
A	3	AAA	=REPT(C4,D4)
AB	3	ABABAB	=REPT(C5,D5)
-	10	----------	=REPT(C6,D6)
|	10	||||||||||	=REPT(C7,D7)

What Does It Do ?

This function repeats a piece of text a specified number of times.
You need to specify the text to be repeated and how many times to repeat it.

Syntax

=REPT(TextToRepeat,Repetitions)
The maximum number of repetitions is 200.

Formatting

No special formatting is needed.

Example 1

The following table was used to display a simple histogram of sales figures.
The =REPT() function uses the value of Sales, but this is divided by 100 to scale down the number of repetitions to below the maximum of 200.

Month	Sales	
Jan	£1,000	||||||||||||||||||
Feb	£5,000	|||||||||||||||||| ... (50 pipes)
Mar	£3,000	|||||||||||||||||| ... (30 pipes)
Apr	£2,000	|||||||||||||||||| ... (20 pipes)
		=REPT("||",D29/100)

Example 2

The =REPT() function has been used to make a digital display for the current time.
The time functions of =HOUR(), =MINUTE() and =SECOND() have been used in conjunction with the =NOW() as the basis for the number of repeats.
To update the clock press the function key F9.

Clock		
Hour	|||||||||||||||| 17	
Minute	|||||||||||||||||||||||||||||||||||||||||| 45	
Second	|||||||||||||||||||||||| 27	

=REPT("|",HOUR(NOW()))&" "&TEXT(HOUR(NOW()),"00")
=REPT("|",MINUTE(NOW()))&" "&TEXT(MINUTE(NOW()),"00")
=REPT("|",SECOND(NOW()))&" "&TEXT(SECOND(NOW()),"00")

RIGHT

Original Text	Number Of Characters Required	Right String	
Alan Jones	1	s	=RIGHT(C4,D4)
Alan Jones	2	es	=RIGHT(C5,D5)
Alan Jones	3	nes	=RIGHT(C6,D6)
Cardiff	6	ardiff	=RIGHT(C7,D7)
ABC123	4	C123	=RIGHT(C8,D8)

What Does It Do ?

This function displays a specified number of characters from the right hand side of a piece of text.

Syntax

=RIGHT(OriginalText,NumberOfCharactersRequired)

Formatting

No special formatting is needed.

Example

The following table was used to extract the second name of a person from their full name. The =FIND() function locates the position of the space between the first and second name. The length of the second name is calculated by subtracting the position of the space from the overall length of the full name. The =RIGHT() function can then extract the second name.

Full Name	Second Name	
Alan Jones	Jones	=RIGHT(C28,LEN(C28)-FIND(" ",C28))
Bob Smith	Smith	=RIGHT(C29,LEN(C29)-FIND(" ",C29))
Carol Williams	Williams	=RIGHT(C30,LEN(C30)-FIND(" ",C30))

ROMAN

Number	Roman	
1	I	=ROMAN(C4)
2	II	=ROMAN(C5)
3	III	=ROMAN(C6)
5	V	=ROMAN(C7)
10	X	=ROMAN(C8)
1998	MCMXCVIII	=ROMAN(C9)
1998	MCMXCVIII	=ROMAN(C10,0)
1998	MLMVLIII	=ROMAN(C11,1)
1998	MXMVIII	=ROMAN(C12,2)
1998	MVMIII	=ROMAN(C13,3)
1998	MVMIII	=ROMAN(C14,4)
1998	MLMVLIII	=ROMAN(C15,TRUE)
1998	MCMXCVIII	=ROMAN(C16,FALSE)

What Does It Do ?

This function produces a number shown as Roman numerals in various formats.

Syntax

=ROMAN(NormalNumber,RomanNumberFormat)

The RomanNumberFormat can be any of the following.

0 is Classic. This is used if no format is specified.

1 is more Concise.

2 is even more Concise.

3 is even more Concise still.

4 is Simplified.

TRUE is Classic

FALSE is Simplified

Formatting

No special formatting is needed.

Note

There is no function to do the opposite calculation of Roman to normal.

ROUND

Number	Places To Round	Rounded Number	
1.48	0	1	=ROUND(C4,D4)
1.48	1	1.5	=ROUND(C5,D5)
1.48	2	1.48	=ROUND(C6,D6)
13643.48	-1	13640	=ROUND(C7,D7)
13643.48	-2	13600	=ROUND(C8,D8)
13643.48	-3	14000	=ROUND(C9,D9)

What Does It Do ?

This function rounds a number to a specified amount od decimal places.
If 0 is used the number is rounded to the nearest whole number.
If a negative amount of rounding is used the figures to the left of the decimal point are rounded.

Syntax

=ROUND(NumberToRound,DecimalPlacesToUse)

Formatting

No special formatting is needed.

ROUNDDOWN

Number	Places To Round	Rounded Down	
1.48	0	1	=ROUNDDOWN(C4,D4)
1.48	1	1.4	=ROUNDDOWN(C5,D5)
1.48	2	1.47	=ROUNDDOWN(C6,D6)
13643.48	-1	13640	=ROUNDDOWN(C7,D7)
13643.48	-2	13600	=ROUNDDOWN(C8,D8)
13643.48	-3	13000	=ROUNDDOWN(C9,D9)

What Does It Do ?

This function rounds a number down to a specified amount of decimal places.
If 0 is used the number is rounded down to the nearest whole number.
If a negative amount of rounding is used the figures to the left of the decimal point are rounded.

Syntax

=ROUNDDOWN(NumberToRound,DecimalPlacesToUse)

Formatting

No special formatting is needed.

ROUNDUP

Number	Places To Round	Rounded Up	
1.48	0	2	=ROUNDUP(C4,D4)
1.48	1	1.5	=ROUNDUP(C5,D5)
1.48	2	1.48	=ROUNDUP(C6,D6)
13643.48	-1	13650	=ROUNDUP(C7,D7)
13643.48	-2	13700	=ROUNDUP(C8,D8)
13643.48	-3	14000	=ROUNDUP(C9,D9)

What Does It Do ?

This function rounds a number up to a specified amount of decimal places.
If 0 is used the number is rounded up to the nearest whole number.
If a negative amount of rounding is used the figures to the left of the decimal point are rounded.

Syntax

=ROUNDUPNumberToRound,DecimalPlacesToUse)

Formatting

No special formatting is needed.

SECOND

Number	Second	
22/Oct/08 17:45:27	27	=SECOND(C4)
12:00:00 PM	0	=SECOND(C5)
0.50	0	=SECOND(C6)
0.51	24	=SECOND(C7)
1.51	24	=SECOND(C8)

What Does It Do?

The function will show the second of the minute based upon a time or a number.
Only the fraction part of the number is used as it is this which relates to time of day.

Syntax

=SECOND(Number)

Formatting

The result will be shown as a normal number between 0 and 59.

Example

The following table was used by a telephone compnay to calculate the cost of a call.
The telephone company only deals in seconds which are a multiple of 5.
The seconds in a call are rounded up to the nearest multiple of 5 before the bill is calculated.
The Duration of the call is entered.
The =MINUTES() function calculates the total number of minutes.
The =SECOND() function calculates the total number of seconds.
The =CEILING() function rounds the seconds up to the nearest muliple of 5.
The Cost of the call is then calculated.

Cost Per Second :	£0.01

	Billed Duration		
Duration	Minutes	Seconds	Cost
0:01:08	1	10	£0.70
0:02:03	2	5	£1.25
0:01:47	1	50	£1.10

=CEILING(SECOND(C36),5)

SIGN

Value	Positive or Negative	
10	1	=SIGN(C4)
20	1	=SIGN(C5)
0	0	=SIGN(C6)
-10	-1	=SIGN(C7)
-20	-1	=SIGN(C8)

What Does It Do ?

This function tests a value to determine whether it is positive or negative.
If the value is positive the result is 1.
If the value is negative the result is -1.
If the value is zero 0 the result is 0.

Syntax

=SIGN(CellToTest)
The CellToTest can be a cell or a calculation.

Formatting

No special formatting is needed.

SLN

Cost	£12,000	
Salvage	£2,000	
Life	4	
Straight Line Depreciation	£2,500	=SLN(F3,F4,F5)

Purchase Value Of A New Car	£20,000	
Second Hand Value	£8,000	
Number Of Years Ownership	6	
Annual Straight Line Depreciation	£2,000	=SLN(F9,F10,F11)

What Does It Do ?

This function calculates the Straight Line Depreciation of an item.
(Also known as Fixed Instalment method).
The Straight Line Depreciation is how much the value of an item reduced during a specific period of time. The result is a uniform depreciation value.

An example would be if you bought a new car for £20,000, then kept it for 6 years.
At the end of your ownership you sell the car for £8,000.
The difference between the original and the trade in price is £20,000 - £8,000 which is £12,000.
Because you owned the car for 6 years, the SLN is calculated as £12,000 / 6 which is £2,000.

Syntax

=SLN(OriginalCost,SellingPrice,LengthOfOwnership)
The LengthOfOwnership can be any time period, days, months or years.
However, the SLN which is calculated will, be for that time, specifying 2 years ownership as 24 months will give an SLN per month.

Formatting

No special formatting is needed.

SMALL

Values
120
800
100
120
250

Lowest Value	100	=SMALL(C4:C8,1)
2nd Lowest Value	120	=SMALL(C4:C8,2)
3rd Lowest Value	120	=SMALL(C4:C8,3)
4th Lowest Value	250	=SMALL(C4:C8,4)
5th Lowest Value	800	=SMALL(C4:C8,5)

What Does It Do ?

This function examines a list of values and picks the value at a user specified position in the list.

Syntax

=SMALL(ListOfNumbersToExamine,PositionToPickFrom)

Formatting

No special formatting is needed.

Example

The following table was used to calculate the bottom 3 sales figures between Jan, Feb and Mar.

Sales	Jan	Feb	Mar
North	£5,000	£6,000	£4,500
South	£5,800	£7,000	£3,000
East	£3,500	£2,000	£10,000
West	£12,000	£4,000	£6,000

Lowest Value	£2,000	=SMALL(D24:F27,1)
2nd Lowest Value	£3,000	=SMALL(D24:F27,2)
3rd Lowest Value	£3,500	=SMALL(D24:F27,3)

Note

Another way to find the Highest and Lowest values would have been to use the =MAX() and =MIN() functions.

Highest	£12,000	=MAX(D24:F27)
Lowest	£2,000	=MIN(D24:F27)

	A	B	C	D	E	F	G	H	I	J
1	Southern data.									
2	Used by the example for the =INDIRECT() function.									
3										
4			Jan	Feb	Mar	Total				
5		Alan	100	200	300	600				
6		Bob	400	500	600	1500				
7		Carol	700	800	900	2400				
8		Total	1200	1500	1800	4500				

STDEV

Values	Values	Values
10	10	10
10	10	11
9	11	9
10	10	12
0.5	0.5	1.29
=STDEV(C4:C7)	=STDEV(E4:E7)	=STDEV(G4:G7)

What Does It Do ?

This function calculates the sample population standard deviation of a list of values.
A sample population is used when the list of values represents a sample of a population.

Syntax

=STDEV(Range1,Range2,Range3 through to Range30)

Formatting

No special formatting is needed.

Example

The table below was used by a company interested in buying a new machine to pack washing powder.
Three machines were short listed and allow to run for a day.
At the end of the day four boxes of soap powder were picked at random from the production of each machine.
The boxes were weighed and the =STDEV() function used as these boxes only represented a sample of the complete days production.
The machine with the smallest deviation was the most consistent.

Soap Powder Box Filling Machine Test Results						
	Test 1	Test 2	Test 3	Test 4	Variance	
Machine 1	1.4	1.5	1.6	1.5	0.0816	=STDEV(D34:G34)
Machine 2	1.5	1.5	1.4	1.5	0.0500	=STDEV(D35:G35)
Machine 3	1.5	1.6	1.7	1.8	0.1291	=STDEV(D36:G36)
			The smallest deviation is :		0.0500	=MIN(H34:H36)
			The machine with the smallest deviation is :		Machine 2	
			=INDEX(C34:C36,MATCH(MIN(H34:H36),H34:H36,0))			

Explanation of formula:

This finds the lowest value.	=MIN(H34:H36)
This finds the position of the lowest value.	=MATCH(MIN(H34:H36),H34:H36,0)
This looks down the Machine column to find the machine name.	=INDEX(C34:C36,MATCH(MIN(H34:H36),H34:H36,0))

STDEVP

Values	Values	Values
10	10	10
10	10	11
9	11	9
10	10	12
0.43	0.43	1.12

=STDEVP(C4:C7) =STDEVP(E4:E7) =STDEVP(G4:G7)

What Does It Do ?

This function calculates the standard deviation of a list of values.
The result is calculated on the basis that the values represent the entire population.

Syntax

=STDEVP(Range1,Range2,Range3 through to Range30)

Formatting

No special formatting is needed.

Example

The table below was used by a company interested in buying a new machine to pack washing powder.
A trial run of just four boxes per machine were produced.
The boxes were weighed and the =STDEVP() function used as these boxes represented the entire test run.
The machine with the smallest variance was the most consistent. ??????????????????

Soap Powder Box Filling Machine Test Results						
	Test 1	Test 2	Test 3	Test 4	Variance	
Machine 1	1.4	1.5	1.6	1.5	0.0707	=STDEVP(D32:G32)
Machine 2	1.5	1.5	1.4	1.5	0.0433	=STDEVP(D33:G33)
Machine 3	1.5	1.6	1.7	1.8	0.1118	=STDEVP(D34:G34)

The smallest variance is : 0.0433 =MIN(H32:H34)

The machine with the smallest variance is : Machine 2
=INDEX(C32:C34,MATCH(MIN(H32:H34),H32:H34,0))

Explanation of formula:

This finds the lowest value.	=(MIN(H32:H34)
This finds the position of the lowest value.	=MATCH(MIN(H32:H34),H32:H34,0)
This looks down the Machine column to find the machine name.	=INDEX(C32:C34,MATCH(MIN(H32:H34),H32:H34,0))

SUBSTITUTE

Original Text	Old Text To Remove	New Text To Insert	Updated Text
ABCDEF	CD	hello	ABhelloEF
ABCDABCD	CD	hello	ABhelloABhello
Northern Region	Region	Area	Northern Area
Sand and Cement	and	&	S& & Cement

=SUBSTITUTE(B4,C4,D4)
=SUBSTITUTE(B5,C5,D5)
=SUBSTITUTE(B6,C6,D6)
=SUBSTITUTE(B7,C7,D7)

Original Text	Old Text To Remove	New Text To Insert	Instance To Be Replaced	Updated Text
ABCABCABC	ABC	hello	3	ABCABChello
Sand and Cement	and	&	2	Sand & Cement

=SUBSTITUTE(B10,C10,D10,E10)
=SUBSTITUTE(B11,C11,D11,E11)

What Does It Do ?

This function replaces a specified piece of text with a different piece of text.
It can either replace all occurrences of the text, or a specific instance.
The function is case sensitive.

Syntax

=SUBSTITUTE(OriginalText,TextToRemove,TextToInsert,InstanceToUse)
The InstanceToUse is optional, if it is omitted all instances will be substituted.

Formatting

No special formatting is needed.

Note

To cope with upper or lower case in the substitution you can use other text functions
such as =UPPER(), =LOWER() or =PROPER() to ensure that the substitution will take place.

Table 1 shows how differing text cases alter the result of the substitution.

Table 1

Original Text	Old Text To Remove	New Text To Insert	Updated Text
Northern Region	Region	Area	Northern Area
Northern **region**	Region	Area	Northern region
Northern Region	**region**	Area	Northern Region
Northern Region	Region	**area**	Northern area
Northern Region	**region**	**area**	Northern Region

=SUBSTITUTE(B39,C39,D39)

Table 2 shows how the =PROPER() function has been used to take account of the mixed cases.

Table 2

Original Text	Old Text To Remove	New Text To Insert	Updated Text
Northern Region	Region	Area	Northern Area
Northern **region**	Region	Area	Northern Area
Northern Region	**region**	Area	Northern Area
Northern Region	Region	**area**	Northern Area
Northern Region	**region**	**area**	Northern Area

=SUBSTITUTE(PROPER(B50),PROPER(C50),PROPER(D50))

SUM

Horizontal				
100	200	300	600	=SUM(C4:E4)

Vertical	
100	
200	
300	
600	=SUM(C7:C9)

Single Cells				
100		300	600	=SUM(C13,D14,E13)
	200			

Multiple Ranges			
100		400	
200		500	
3000		600	
		4800	=SUM(C17:C19,E17:E19)

Functions			
100		400	
200		500	
300		600	
		800	=SUM(AVERAGE(C23:C25),MAX(E23:E25))

What Does It Do ?

This function creates a total from a list of numbers.
It can be used either horizontally or vertically.
The numbers can be in single cells, ranges are from other functions.

Syntax

=SUM(Range1,Range2,Range3... through to Range30).

Formatting

No special formatting is needed.

Note

Many people use the =SUM() function incorrectly.

This example shows how the SUM has been combined with plus + symbols.
The formula is actually doing more work than needed.
It should have been entered as either =C48+C49+C50 or =SUM(C48:C50).

100		
200		
300		
600	=SUM(C48+C49+C50)	*Wrong!*
	=SUM(C48:C50)	Correct
	=C48+C49+C50	Correct

SUM (Running Total)

Using =SUM() For A Running Total

Month	Sales	Running Total	
Jan	10	10	=SUM($D$7:D7)
Feb	50	60	=SUM($D$7:D8)
Mar	30	90	=SUM($D$7:D9)
Apr	20	110	=SUM($D$7:D10)
May		110	=SUM($D$7:D11)
Jun		110	=SUM($D$7:D12)
Jul		110	=SUM($D$7:D13)
Aug		110	=SUM($D$7:D14)
Sep		110	=SUM($D$7:D15)
Oct		110	=SUM($D$7:D16)
Nov		110	=SUM($D$7:D17)
Dec		110	=SUM($D$7:D18)

Type the formula =SUM($D$7:D7) in cell E7 and then copy down the table.
It works because the first reference uses dollar symbols $ to keep $D$7 static as the formula is copied down. Each occurrence of the =SUM() then adds all the numbers from the first cell down.

The function can be tidied up to show 0 zero when there is no adjacent value by using the =IF() function.

Month	Sales	Running Total	
Jan	10	10	=SUM(IF(D7,$D$7:D7,0))
Feb	50	60	=SUM(IF(D8,$D$7:D8,0))
Mar	30	90	=SUM(IF(D9,$D$7:D9,0))
Apr	20	110	=SUM(IF(D10,$D$7:D10,0))
May		0	=SUM(IF(D11,$D$7:D11,0))
Jun		0	=SUM(IF(D12,$D$7:D12,0))
Jul		0	The =SUM() only takes place when
Aug		0	there is data in column D.
Sep		0	Otherwise the value 0 zero is entered.
Oct		0	
Nov		0	
Dec		0	

SUM and the =OFFSET function

Sometimes it is necessary to base a calculation on a set of cells in different locations.
An example would be when a total is required from certain months of the year, such as
the last 3 months in relation to the current date.

One solution would be to retype the calculation each time new data is entered, but this
would be time consuming and open to human error.

A better way is to indicate the start and end point of the range to be calculated by
using the =OFFSET() function.

The =OFFSET() picks out a cell a certain number of cells away from another cell.
By giving the =OFFSET() the address of the first cell in the range which needs to
be totalled, we can then indicate how far away the end cell should be and the =OFFSET()
will give us the address of cell which will be the end of the range to be totalled.

The =OFFSET() needs to know three things;

1. A cell address to use as the fixed point from where it should base the offset.
2. How many rows it should look up or down from the starting point.
3. How many columns it should look left or right from the starting point.

Total	Jan	Feb	Mar	Apr	May
Err:508	**10**	400	500	600	700

=SUM(E24:OFFSET(E24,0,0))
This example uses E24 as the starting point and offsets no rows or columns which
results in the range being summed as E24:E24.

Err:508	**10**	**400**	500	600	700

=SUM(E29:OFFSET(E29,0,1))
This example uses E29 as the starting point and offsets 1 col to pick out
cell F29 resulting in a the range E29:F29 being summed.

Err:508	**10**	**400**	**500**	600	700

=SUM(E34:OFFSET(E34,0,2))
This example uses E34 as the starting point and offsets 2 cols to pick out
cell G34 resulting in a the range E34:G34 being summed.

Using =OFFSET() Twice In A Formula

The following examples use =OFFSET() to pick both the start and end of the range
which needs to be totalled.

Total	Jan	Feb	Mar	Apr	May
Err:508	10	**400**	500	600	700

=SUM(OFFSET(E45,0,1):OFFSET(E45,0,1))
The cell E45 has been used as the starting point for both offsets and each has
been offset by just 1 column. The result is that just cell F45 is used as the
range F45:F45 for the sum function to calculate.

Err:508	10	**400**	**500**	600	700

=SUM(OFFSET(E51,0,1):OFFSET(E51,0,2))
The cell E51 has been used as the starting point of both offsets, the first offset is
offset by 1 column, the second by 2 columns. The result is the range F51:G51 which
is then totalled.

	A	B	C	D	E	F	G	H	I	J
56										
57			Err:508		10	**400**	**500**	**600**	700	
58			=SUM(OFFSET(E57,0,1):OFFSET(E57,0,3))							
59			The cell E57 has been used as the starting point for both offsets, the first offset is							
60			offset by 1 column, the second by 3 columns. The result is the range F57:H57 which							
61			is then totalled.							
62										
63										
64			**Example**							
65										
66			The following table shows five months of data.							
67			To calculate the total of a specific group of months the =OFFSET() function has been used.							
68			The Start and End dates entered in cells F71 and F72 are used as the offset to produce							
69			a range which can be totalled.							
70										
71				Type in the Start month.		**Feb-98**				
72				Type in the End month.		**Mar-98**				
73										
74			Total		Jan-98	Feb-98	Mar-98	Apr-98	May-98	
75			Err:508		10	400	500	600	700	
76										
77			Err:508		15	20	1000	2000	3000	
78										
79			Err:508		5	3	10	800	900	
80			=SUM(OFFSET(D79,0,MONTH(F71)):OFFSET(D79,0,MONTH(F72)))							
81										
82			**Explanation**							
83			The following formula represent a breakdown of what the =OFFSET function does.							
84			The formula displayed below are only dummies, but they will update as you enter							
85			dates into cells F71 and F72.							
86										
87			*Formula 1*	=SUM(OFFSET(D79,0,MONTH(F71)) : OFFSET(D79,0,MONTH(F72)))						
88				This is the actual formula entered by the user.						
89										
90			*Formula 2*	=SUM(OFFSET(D79,0,MONTH(2)) : OFFSET(D79,0,MONTH(3)))						
91				This shows how the =MONTH function calculates the month number.						
92				In this example the values of the months are 2 and 3 for Feb and Mar.						
93				These values are the 'offsets' relative to cell D79.						
94										
95			*Formula 3*	=SUM(OFFSET(D79,0,2) : OFFSET(D79,0,3))						
96				This shows where the month numbers are used in the =OFFSET function.						
97										
98			*Formula 4*	=SUM(F79:G79)						
99				This shows how the =OFFSET eventually equates to cell addresses						
100				to be used as a range for the =SUM function.						

SUMIF

Item	Date	Cost
Brakes	1-Jan-98	80
Tyres	10-May-98	25
Brakes	1-Feb-98	80
Service	1-Mar-98	150
Service	5-Jan-98	300
Window	1-Jun-98	50
Tyres	1-Apr-98	200
Tyres	1-Mar-98	100
Clutch	1-May-98	250

Total cost of all Brakes bought.	160	=SUMIF(C4:C12,"Brakes",E4:E12)
Total cost of all Tyres bought.	325	=SUMIF(C4:C12,"Tyres",E4:E12)
Total of items costing £100 or above.	1000	=SUMIF(E4:E12,">=100")
Total of item typed in following cell. service	450	=SUMIF(C4:C12,E18,E4:E12)

What Does It Do ?

This function adds the value of items which match criteria set by the user.

Syntax

=SUMIF(RangeOfThingsToBeExamined,CriteriaToBeMatched,RangeOfValuesToTotal)

=SUMIF(C4:C12,"Brakes",E4:E12)	This examines the names of products in C4:C12.
	It then identifies the entries for Brakes.
	It then totals the respective figures in E4:E12
=SUMIF(E4:E12,">=100")	This examines the values in E4:E12.
	If the value is >=100 the value is added to the total.

Formatting

No special formatting is needed.

SUMPRODUCT

Item	Sold	price
Tyres	5	100
Filters	2	10
Bulbs	3	2

Total Sales Value :	526	=SUMPRODUCT(D4:D6,E4:E6)

What Does It Do ?

This function uses at least two columns of values.
The values in the first column are multipled with the corresponding value in the second column.
The total of all the values is the result of the calculation.

Syntax

=SUMPRODUCT(Range1, Range, Range3 through to Range30)

Formatting

No special formatting is needed.

Example

The following table was used by a drinks merchant to keep track of stock.
The merchant needed to know the total purchase value of the stock, and the potential value of the stock when it is sold, takinging into account the markup percentage.

The =SUMPRODUCT() function is used to multiply the Cases In Stock with the Case Price to calculate what the merchant spent in buying the stock.

The =SUMPRODUCT() function is used to multiply the Cases In Stock with the Bottles In Case and the Bottle Setting Price, to calculate the potential value of the stock if it is all sold.

Product	Cases In Stock	Case Price	Bottles In Case	Bottle Cost	Markup	Bottle Selling Price
Red Wine	10	£120	10	£12.00	25%	£15.00
White Wine	8	£130	10	£13.00	25%	£16.25
Champagne	5	£200	6	£33.33	80%	£60.00
Beer	50	£24	12	£2.00	20%	£2.40
Lager	100	£30	12	£2.50	25%	£3.13
				=D39/E39		=F39+F39*G39

Total Value Of Stock :	£7,440	=SUMPRODUCT(C35:C39,D35:D39)
Total Selling Price Of Stock :	£9,790	=SUMPRODUCT(C35:C39,E35:E39,H35:H39)
Profit :	£2,350	=E44-E43

SYD

Purchase Value Of A New Car	£20,000	
Second Hand Value	£8,000	
Number Of Years Ownership	6	
Deprecation in year 1	£3,429	=SYD(F3,F4,F5,1)
Deprecation in year 2	£2,857	=SYD(F3,F4,F5,2)
Deprecation in year 3	£2,286	=SYD(F3,F4,F5,3)
Deprecation in year 4	£1,714	=SYD(F3,F4,F5,4)
Deprecation in year 5	£1,143	=SYD(F3,F4,F5,5)
Deprecation in year 6	£571	=SYD(F3,F4,F5,6)
Total Depreciation :	£12,000	=SUM(F7:F12)

What Does It Do ?

This function calculates the depreciation of an item throughout its life, using the sum of the years digits.

The depreciation is greatest in the earlier part of the items life.

What is the Sum Of The Years Digits ?

The sum of the years digits adds together the each of the years of the life.

A life of 3 years has a sum of 1+2+3 equalling 6.

Each of the years is then calculated as a percentage of the sum of the years.

Year 3 is 50% of 6, year 2 is 33% of 6, year 1 is 17% 6.

The total depreciation of the item is then allocated on the basis of these percentages.

A depreciation of £9000 is allocated as 50% being £4500, 33% being £3000, 17% being £1500.

		£9,000
1	17%	£1,500
2	33%	£3,000
3	50%	£4,500

As the greater part of the depreciation is allocated to the earliest years the values are inverted, year 1 is $4500, year 2 is £3000 and year 1 is £1500.

Example 1

Purchase Price Of A Car :	£10,000		
Salvage Value :	£1,000		
Expected Life in Years :	3		
			As % Of Total Depreciation
Depreciation in Year 1 :	£4,500	===>	0.5
Depreciation in Year 2 :	£3,000	===>	0.33
Depreciation in Year 3 :	£1,500	===>	0.17
	=SYD(E39,E40,E41,3)		

1. Add together the digits of the Life to get the SumOfTheYearsDigits, 1+2+3=6.
2. Subtract the Salvage from the Purchase Price to get Total Deprectation, £10000-£1000=£9000.
3. Divide the Total Deprectation by the SumOfTheYearsDigits, £9000/6=£1500.
4. Invert the year digits, 1,2,3 becomes 3,2,1.
5. Multiply 3,2,1 by £1500 to get £4500, £3000, £1500, these values are the depreciation values for each of the three years in the life of the item.

Example 2

	A	B	C	D	E	F	G	H	I
56		The same example using 4 years.							
57									
58			Purchase Price Of A Car :		£10,000				
59			Salvage Value :		£1,000				
60			Expected Life in Years :		4				
61								As % Of Total Depriciation	
62			Depreciation in Year 1 :		£3,600			0.4	
63			Depreciation in Year 2 :		£2,700			0.3	
64			Depreciation in Year 3 :		£1,800			0.2	
65			Depreciation in Year 4 :		£900			0.1	
66			Total Depreciation :		£9,000			100%	
67									
68		**Example 3**							
69		This example will adjust itself to accommodate any number of years between 1 and 10.							
70									
71			Purchase Price Of A Car :		£10,000				
72			Salvage Value :		£1,000				
73			Expected Life in Years (1 to 10) :		7				
74								As % Of Total Depriciation	
75			Year	1	£2,250			25%	
76			Year	2	£1,929			21%	
77			Year	3	£1,607			18%	
78			Year	4	£1,286			14%	
79			Year	5	£964			11%	
80			Year	6	£643			7%	
81			Year	7	£321			4%	
82			Year						
83			Year						
84			Year						
85					£9,000			100%	
86									
87		**Syntax**							
88		=SYD(OriginalCost,SalvageValue,Life,PeriodToCalculate)							
89									
90		**Formatting**							
91		No special formatting is needed.							

	A	B	C	D	E	F	G	H	I	J
1	**T**									
2										
3				Cell To Test	Result					
4				Hello	Hello	=T(D4)				
5				10		=T(D5)				
6				1-Jan-98		=T(D6)				
7						=T(D7)				
8										
9		**What Does It Do ?**								
10		This function examines an entry to determine whether it is text or not.								
11		If the value is text, then the text is the result of the function								
12		If the value is not text, the result is a blank.								
13		The function is not specifically needed by Excel, but is included for compatibility with								
14		other spreadsheet programs.								
15										
16		**Syntax**								
17		=T(CellToTest)								
18										
19		**Formatting**								
20		No special formatting is needed.								

TEXT

Original Number	Converted To Text	
10	10.00	=TEXT(C4,"0.00")
10	£10.00	=TEXT(C5,"£0.00")
10	10	=TEXT(C6,"0")
10	£10	=TEXT(C7,"£0")
10.25	10.3	=TEXT(C8,"0.0")
10.25	£10.3	=TEXT(C9,"£0.0")

What Does It Do ?

This function converts a number to a piece of text.
The formatting for the text needs to be specified in the function.

Syntax

=TEXT(NumberToConvert,FormatForConversion)

Formatting

No special formatting is required.

TIME

Hour	Minute	Second	Time	
14	30	59	14:30:59	=TIME(C4,D4,E4)
14	30	59	2:30:59 PM	=TIME(C5,D5,E5)
14	30	59	0.60485	=TIME(C6,D6,E6)

What Does It Do?

This function will convert three separate numbers to an actual time.

Syntax

=TIME(Hour,Minute,Second)

Formatting

The result will be shown as a time which can be formatted either as 12 or 24 hour style.
If a normal number format is applied a decimal fraction is shown which represents the time as a fraction of the day.

TIMEVALUE

Text	Time	
14:30:59	0.6	=TIMEVALUE(C4)
14:30:59	14:30:59	=TIMEVALUE(C5)
14:30:59	2:30:59 PM	=TIMEVALUE(C6)

What Does It Do?

This function will show an actual time based on a piece of text which looks like a time. It is useful when data is imported from other applications, such as from mainframe computers, which convert all values to text.

Syntax

=TIMEVALUE(Text)

Formatting

The result will be shown as a number representing the time a fraction of the day.

Formatting can be applied for either the 12 or 24 hour clock system.

TODAY

Today Is	
22-Oct-08	=TODAY()

What Does It Do?

Use this to show the current date.

Syntax

=TODAY()

Formatting

The result will normally be displayed using the DD-MMM-YY format.

Example

The following example shows how the Today function is used to calculate the number of days since a particular day.

Date	Days Since	
1-Jan-97	10/21/11	=TODAY()-C20
10-Aug-97	03/14/11	=TODAY()-C21

Note that the result is actually the number of days before todays date. To calculate a result which includes the current date an extra 1 will need to be added.

Date	Days Since	
1-Jan-97	4313	=TODAY()-C28+1
10-Aug-97	4092	=TODAY()-C29+1

Example

The following example shows the number of days from today until the year 2000.

Year 2000	Days Until	
01-Jan-2000	03/10/91	=C36-TODAY()

TRANSPOSE

	Jan	Feb
Alan	10	30
Bob	40	50
Carol	70	80
Total	120	160

	Alan	Bob	Carol	Total
Jan	10	40	70	120
Feb	30	50	80	160

{=TRANSPOSE(C3:E7)}

As an array formula in all these cells

What Does It Do ?

This function copies data from a range, and places in it in a new range, turning it so that the data originally in columns is now in rows, and the data originally in rows is in columns.

The transpose range must be the same size as the original range.

The function needs to be entered as an array formula.

To enter an array formula you must first highlight all the cells where the formula is required.

Next type the formula, such as =TRANSPOSE(A1:A5).

Finally press Ctrl+Shift+Enter to confirm it.

If changes need to be made to the formula, the entire array has to be highlighted, the edits can then be made and the Ctrl+Shift+Enter used to confirm it.

Syntax

=TRANSPOSE(Range)

Formatting

No special formatting is needed.

TREND WHAT IS CONST b ?

Historical Data		Predicted Values		
Month	Sales	Month	Sales	
1	£1,000	7	£4,940	{=TREND(C8:C13,B8:B13,E8:E13)}
2	£2,000	8	£5,551	{=TREND(C5:C10,B5:B10,E5:E10)}
3	£2,500	9	£6,163	{=TREND(C5:C10,B5:B10,E5:E10)}
4	£3,500	10	£6,774	{=TREND(C5:C10,B5:B10,E5:E10)}
5	£3,800	11	£7,386	{=TREND(C5:C10,B5:B10,E5:E10)}
6	£4,000	12	£7,997	{=TREND(C5:C10,B5:B10,E5:E10)}

What Does It Do ?

This function predicts values based upon three sets of related values.
The prediction is based upon the Linear Trend of the original values.
The function is an array function and must be entered using Ctrl+Shift+Enter.

Syntax

=TREND(KnownYs,KnownXs,RequiredXs,Constant)
The KnownYs is the range of values, such as Sales Figures.
The KnownXs is the intervals used when collecting the data, such as Months.
The RequiredXs is the range for which you want to make the prediction, such as Months.

Formatting

No special formatting is needed.

Example

The following tables were used by a company to predict when they would start to make a profit.
Their bank manager had told the company that unless they could show a profit by the end of the next year, the bank would no longer provide an overdraft facility.
To prove to the bank that, based upon the past years performance, the company would start to make a profit at the end of the next year, the =TREND() function was used.
The historical data for the past year was entered, months 1 to 12.
The months to predict were entered, 13 to 24.
The =TREND() function shows that it will be month 22 before the company make a profit.

Historical Data		Predicted Values		
Month	Profit	Month	Profit	
1	-£5,000	13	-£2,226	{=TREND(C41:C52,B41:B52,E41:E52)}
2	-£4,800	14	-£1,968	*The*
3	-£4,600	15	-£1,709	*same*
4	-£4,750	16	-£1,451	*function*
5	-£4,800	17	-£1,193	*used*
6	-£4,500	18	-£935	*in*
7	-£4,000	19	-£676	*all*
8	-£3,800	20	-£418	*cells*
9	-£3,300	21	-£160	*as*
10	-£2,000	22	£98	*an*
11	-£2,500	23	£356	*array*
12	-£2,800	24	£615	*formula*

How To Enter An Array Formula

Select all the cells where the array is required, such as F41 to F52.

	A	B	C	D	E	F	G	H	I	J
56		Type the formula such as =TREND(C41:C52,B41:B52,E41:E52), but do not press Enter.								
57		Hold the Ctrl+Shift keys down.								
58		Press Enter to enter the formula as an array.								

TRIM

Original Text	Trimmed Text	
ABCD	ABCD	=TRIM(C4)
A B C D	A B C D	=TRIM(C5)
Alan Jones	Alan Jones	=TRIM(C6)
ABCD	ABCD	=TRIM(C7)

What Does It Do ?

This function removes unwanted spaces from a piece of text.
The spaces before and after the text will be removed completely.
Multiple spaces within the text will be trimmed to a single space

Syntax

=TRIM(TextToTrim)

Formatting

No special formatting is needed.

TRUNC

Number	Precision For Truncation	Truncated Number	
1.48	0	1	=TRUNC(C4,D4)
1.48	1	1.4	=TRUNC(C5,D5)
1.48	2	1.47	=TRUNC(C6,D6)
-1.48	1	-1.4	=TRUNC(C7,D7)
-1.48	2	-1.47	=TRUNC(C8,D8)
13643.48	-1	13640	=TRUNC(C9,D9)
13643.48	-2	13600	=TRUNC(C10,D10)
13643.48	-3	13000	=TRUNC(C11,D11)

What Does It Do ?

This function removes the decimal part of a number, it does not actually round the number.

Syntax

=TRUNC(NumberToTuncate,Precision)

Formatting

No special formatting is needed.

UPPER

Original Text	Upper Case	
alan jones	ALAN JONES	=UPPER(C4)
bob smith	BOB SMITH	=UPPER(C5)
carOl wiLLiamS	CAROL WILLIAMS	=UPPER(C6)
cardiff	CARDIFF	=UPPER(C7)
abc123	ABC123	=UPPER(C8)

What Does It Do ?

This function converts all characters in a piece of text to upper case.

Syntax

=UPPER(TextToConvert)

Formatting

No special formatting is needed.

Example

See the example for FREQUENCY.

VALUE

Text Containing A Number	Value	
Annual turnover was £5000	Err:502	=VALUE(MID(C4,SEARCH("£",C4),99))
There was a 2% increase in sales.	#VALUE!	
There was a 50% increase in sales.	#VALUE!	
A 100% increase was achieved.	#VALUE!	
Only a 2% increase in sales.	#VALUE!	
Approx 50% increase in sales.	#VALUE!	
There was a 100% increase in sales.	#VALUE!	* *See explanation below.*

=**VALUE**(MID(SUBSTITUTE(C11," "," "),SEARCH("???%",SUBSTITUTE(C11," "," ")),4))

The winning time was 1:30 seconds.	#VALUE!	=VALUE(MID(C14,SEARCH("??:??",C14),5))
The winning time was 1:30 seconds.	#VALUE!	=VALUE(MID(C15,SEARCH("??:??",C15),5))
The winning time was 10:30 seconds.	#VALUE!	=VALUE(MID(C16,SEARCH("??:??",C16),5))
The winning time was 0:30 seconds.	#VALUE!	=VALUE(MID(C17,SEARCH("??:??",C17),5))

What Does It Do ?

This function converts a piece of text which resembles a number into an actual value.
If the number in the middle of a long piece of text it will have to be extracted using other
text functions such as =SEARCH(), =MID(), =FIND(), =SUBSTITUTE, =LEFT() or =RIGHT().

Syntax

=VALUE(TextToConvert)

Formatting

No special formatting is needed.
The result will be shown as a value, based upon the original text.
If the £ sign is included in the text it will be ignored.
If the % sign is included in the text, the result will be a decimal fraction which can then
be formatted as a percentage.
If the original text format appears as a time hh:mm the result will be a time.
The same will be true for other recognised formats.

Explanation of formula shown above.

To extract the values from the following text is complicated!
The actual percentage value is of variable length, it can be either one, two or three digits long.
The only way to identify the value is the fact it always ends with the % sign.
There is no way to identify the beginning of the value, other than it is preceded by a space.
The main problem is calculating the length of the value to extract.
If the extraction assumes the maximum length of three digits and the % sign, errors will occur
when the percentage is only one digit long, as alphabetic characters will be included.
To get around the problem the =SUBSTITUTE() function was used to increase the size of the
spaces in the text.
Now when the extraction takes place any unnecessary characters will be spaces which are
ignored by the =VALUE() function.

There was a 2% increase in sales.	#VALUE!
There was a 50% increase in sales.	#VALUE!
There was a 100% increase in sales.	#VALUE!

=VALUE(MID(SUBSTITUTE(C52," "," "),SEARCH("???%",SUBSTITUTE(C52," "," ")),4))

VAR

Values	Values	Values
10	10	10
10	10	11
9	11	9
10	10	12
0.25	0.25	1.67
=VAR(C4:C7)	=VAR(E4:E7)	=VAR(G4:G7)

What Does It Do ?

This function calculates the sample population variance of a list of values.
A sample population is used when the list of values represents a sample of a population.

Syntax

=VAR(Range1,Range2,Range3 through to Range30)

Formatting

No special formatting is needed.

Example

The table below was used by a company interested in buying a new machine to pack washing powder.
Three machines were short listed and allow to run for a day.
At the end of the day four boxes of soap powder were picked at random from the production of each machine.
The boxes were weighed and the =VAR() function used as these boxes only represented a sample of the complete days production.
The machine with the smallest variance was the most consistent.

Soap Powder Box Filling Machine Test Results					
	Test 1	Test 2	Test 3	Test 4	Variance	
Machine 1	1.4	1.5	1.6	1.5	0.0067	=VAR(D34:G34)
Machine 2	1.5	1.5	1.4	1.5	0.0025	=VAR(D35:G35)
Machine 3	1.5	1.6	1.7	1.8	0.0167	=VAR(D36:G36)
			The smallest variance is :		0.0025	=MIN(H34:H36)
	The machine with the smallest variance is :				Machine 2	
	=INDEX(C34:C36,MATCH(MIN(H34:H36),H34:H36,0))					

Explanation of formula:

This finds the lowest value.	=MIN(H34:H36)
This finds the position of the lowest value.	=MATCH(MIN(H34:H36),H34:H36,0)
This looks down the Machine column to find the machine name.	=INDEX(C34:C36,MATCH(MIN(H34:H36),H34:H36,0))

VARP

Values	Values	Values
10	10	10
10	10	11
9	11	9
10	10	12
0.19	0.19	1.25
=VARP(C4:C7)	=VARP(E4:E7)	=VARP(G4:G7)

What Does It Do ?

This function calculates the variance of a list of values.
The variance is calculated on the basis that the values represent the entire population.

Syntax

=VARP(Range1,Range2,Range3 through to Range30)

Formatting

No special formatting is needed.

Example

The table below was used by a company interested in buying a new machine to pack washing powder.
A trial run a just four boxes per machine were produced.
The boxes were weighed and the =VARP() function used as these boxes represented the entire test run.
The machine with the smallest variance was the most consistent.

Soap Powder Box Filling Machine Test Results					
	Test 1	Test 2	Test 3	Test 4	Variance
Machine 1	1.4	1.5	1.6	1.5	0.0050
Machine 2	1.5	1.5	1.4	1.5	0.0019
Machine 3	1.5	1.6	1.7	1.8	0.0125

The smallest variance is :	0.0019	=MIN(H32:H34)
The machine with the smallest variance is :	Machine 2	

=INDEX(C32:C34,MATCH(MIN(H32:H34),H32:H34,0))

Explanation of formula:

This finds the lowest value.	=(MIN(H32:H34)
This finds the position of the lowest value.	=MATCH(MIN(H32:H34),H32:H34,0)
This looks down the Machine column to find the machine name.	=INDEX(C32:C34,MATCH(MIN(H32:H34),H32:H34,0))

VLOOKUP

The column numbers are not needed.
they are part of the illustration.

	col 1	col 2	col 3	col 4	col 5	col 6
Jan	10	20	30	40	50
Feb	80	90	100	110	120
Mar	97	69	45	51	77

Type a month to look for : Feb

Which column needs to be picked out : 4

The result is : 100

=VLOOKUP(G11,C6:H8,G12,FALSE)

What Does It Do ?

This function scans down the row headings at the side of a table to find a specified item.
When the item is found, it then scans across to pick a cell entry.

Syntax

=VLOOKUP(ItemToFind,RangeToLookIn,ColumnToPickFrom,SortedOrUnsorted)

The ItemToFind is a single item specified by the user.
The RangeToLookIn is the range of data with the row headings at the left hand side.
The ColumnToPickFrom is how far across the table the function should look to pick from.
The Sorted/Unsorted is whether the column headings are sorted. TRUE for yes, FALSE for no.

Formatting

No special formatting is needed.

	A	B	C	D	E	F	G	H	I	J
31		**Example 1**								
32		This table is used to find a value based on a specified name and month.								
33		The =VLOOKUP() is used to scan down to find the name.								
34		The problem arises when we need to scan across to find the month column.								
35		To solve the problem the =MATCH() function is used.								
36										
37		The =MATCH() looks through the list of names to find the month we require. It then calculates								
38		the position of the month in the list. Unfortunately, because the list of months is not as wide								
39		as the lookup range, the =MATCH() number is 1 less than we require, so and extra 1 is								
40		added to compensate.								
41										
42		The =VLOOKUP() now uses this =MATCH() number to look across the columns and								
43		picks out the correct cell entry.								
44										
45		The =VLOOKUP() uses FALSE at the end of the function to indicate to Excel that the								
46		row headings are not sorted.								
47										
48										
49				Jan	Feb	Mar				
50			Bob	10	80	97				
51			Eric	20	90	69				
52			Alan	30	100	45				
53			Carol	40	110	51				
54			David	50	120	77				
55										
56			Type a name to look for :			eric				
57			Type a month to look for :			mar				
58										
59				The result is :		69				
60			=VLOOKUP(F56,C50:F54,MATCH(F57,D49:F49,0)+1,FALSE)							
61										

Example 2

This example shows how the =VLOOKUP() is used to pick the cost of a spare part for different makes of cars.

The =VLOOKUP() scans down row headings in column F for the spare part entered in column C. When the make is found, the =VLOOKUP() then scans across to find the price, using the result of the =MATCH() function to find the position of the make of car.

The functions use the absolute ranges indicated by the dollar symbol . This ensures that when the formula is copied to more cells, the ranges for =VLOOKUP() and =MATCH() do not change.

Maker	Spare	Cost
Vauxhall	Ignition	£50
VW	GearBox	£600
Ford	Engine	£1,200
VW	Steering	£275
Ford	Ignition	£70
Ford	CYHead	£290
Vauxhall	GearBox	£500
Ford	Engine	£1,200

Lookup Table

	Vauxhall	Ford	VW
GearBox	500	450	600
Engine	1000	1200	800
Steering	250	350	275
Ignition	50	70	45
CYHead	300	290	310

=VLOOKUP(C81,F75:I79,MATCH(B81,G74:I74,0)+1,FALSE)

Example 3

In the following example a builders merchant is offering discount on large orders.
The Unit Cost Table holds the cost of 1 unit of Brick, Wood and Glass.
The Discount Table holds the various discounts for different quantities of each product.
The Orders Table is used to enter the orders and calculate the Total.

All the calculations take place in the Orders Table.
The name of the Item is typed in column C of the Orders Table.

The Unit Cost of the item is then looked up in the Unit Cost Table.
The FALSE option has been used at the end of the function to indicate that the product names down the side of the Unit Cost Table are not sorted.
Using the FALSE option forces the function to search for an exact match. If a match is not found, the function will produce an error.
=VLOOKUP(C126,C114:D116,2,FALSE)

The discount is then looked up in the Discount Table
If the Quantity Ordered matches a value at the side of the Discount Table the =VLOOKUP will look across to find the correct discount.
The TRUE option has been used at the end of the function to indicate that the values down the side of the Discount Table are sorted.
Using TRUE will allow the function to make an approximate match. If the Quantity Ordered does not match a value at the side of the Discount Table, the next lowest value is used.
Trying to match an order of 125 will drop down to 100, and the discount from the 100 row is used.
=VLOOKUP(D126,F114:I116,MATCH(C126,G113:I113,0)+1,TRUE)

Unit Cost Table

Brick	£2
Wood	£1
Glass	£3

Discount Table

	Brick	Wood	Glass
1	0%	0%	0%
100	6%	3%	12%
300	8%	5%	15%

Orders Table

Item	Units	Unit Cost	Discount	Total
Brick	100	£2	6%	£188
Wood	200	£1	3%	£194
Glass	150	£3	12%	£396
Brick	225	£2	6%	£423
Wood	50	£1	0%	£50
Glass	500	£3	15%	£1,275

Formula for :

Unit Cost	=VLOOKUP(C126,C114:D116,2,FALSE)
Discount	=VLOOKUP(D126,F114:I116,MATCH(C126,G113:I113,0)+1,TRUE)
Total	=(D126*E126)-(D126*E126*F126)

WEEKDAY

Date	Weekday	
Thu 01-Jan-98	5	=WEEKDAY(C4)
Thu 01-Jan-98	5	=WEEKDAY(C5)
Thu 01-Jan-98	5	=WEEKDAY(C6,1)
Thu 01-Jan-98	4	=WEEKDAY(C7,2)
Thu 01-Jan-98	3	=WEEKDAY(C8,3)

What Does It Do?

This function shows the day of the week from a date.

Syntax

=WEEKDAY(Date,Type)

Type : This is used to indicate the week day numbering system.
1 : will set Sunday as 1 through to Saturday as 7
2 : will set Monday as 1 through to Sunday as 7.
3 : will set Monday as 0 through to Sunday as 6.
If no number is specified, Excel will use 1.

Formatting

The result will be shown as a normal number.
To show the result as the name of the day, use **Format**, **Cells**, **Custom** and set the **Type** to **ddd** or **dddd**.

Example

The following table was used by a hotel which rented a function room.
The hotel charged different rates depending upon which day of the week the booking was for.
The Booking Date is entered.
The Actual Day is calculated.
The Booking Cost is picked from a list of rates using the =LOOKUP() function.

Booking Date	Actual Day	Booking Cost
7-Jan-98	Wednesday	£30.00

=LOOKUP(WEEKDAY(C34),C39:D45)

Booking Rates	
Day Of Week	Cost
1	£50
2	£25
3	£25
4	£30
5	£40
6	£50
7	£100

WORKDAY

StartDate	Days	Result	
1-Jan-98	28	35836	=WORKDAY(D4,E4)
1-Jan-98	28	10-Feb-98	=WORKDAY(D5,E5)

What Does It Do?

Use this function to calculate a past or future date based on a starting date and a specified number of days. The function excludes weekends and holidays and can therefore be used to calculate delivery dates or invoice dates.

Syntax

=WORKDAY(StartDate,Days,Holidays)

Formatting

The result will normally be shown as a number which can be formatted to a normal date by using Format,Cells,Number,Date.

Example

The following example shows how the function can be used to calculate delivery dates based upon an initial Order Date and estimated Delivery Days.

Order Date	Delivery Days	Delivery Date
Mon 02-Feb-98	2	Wed 04-Feb-98
Tue 15-Dec-98	28	Tue 26-Jan-99

=WORKDAY(D25,E25,D28:D32)

Holidays	
Fri 01-May-98	Bank Holiday
Fri 25-Dec-98	Xmas
Wed 01-Jan-97	New Year
Thu 01-Jan-98	New Year
Fri 01-Jan-99	New Year

YEAR

Date	Year	
25-Dec-98	1998	=YEAR(C4)

What Does It Do?

This function extracts the year number from a date.

Syntax

=YEAR(Date)

Formatting

The result is shown as a number.

YEARFRAC

Start Date	End Date	Fraction	
1-Jan-98	1-Apr-98	0.25	=YEARFRAC(C4,D4)
1-Jan-98	31-Dec-98	1	=YEARFRAC(C5,D5)
1-Jan-98	1-Apr-98	25%	=YEARFRAC(C6,D6)

What Does It Do?

This function calculates the difference between two dates and expresses the result as a decimal fraction.

Syntax

=YEARFRAC(StartDate,EndData,Basis)

Basis : Defines the calendar system to be used in the function.

0 : or omitted USA style 30 days per month divided by 360.
1 : 29 or 30 or 31 days per month divided by 365.
2 : 29 or 30 or 31 days per month divided by 360.
3 : 29 or 30 0r 31 days per month divided by 365.
4 : European 29 or 30 or 31 days divided by 360.

Formatting

The result will be shown as a decimal fraction, but can be formatted as a percent.

Example

The following table was used by a company which hired people on short term contracts for a part of the year.
The Pro Rata Salary which represents the annual salary is entered.
The Start and End dates of the contract are entered.
The =YEARFRAC() function is used to calculate Actual Salary for the portion of the year.

Start	End	Pro Rata Salary	Actual Salary	
1-Jan-98	31-Dec-98	£12,000	£12,000	=YEARFRAC(B32,C32+1,4)*D32
1-Jan-98	31-Mar-98	£12,000	£3,000	=YEARFRAC(B33,C33+1,4)*D33
1-Jan-98	30-Jun-98	£12,000	£6,000	=YEARFRAC(B34,C34+1,4)*D34

Note

The extra 1 has been added to the End date to compensate for the fact that the =YEARFRAC() function calculates from the Start date up to, but not including, the End date.

	A	B	C	D	E	F	G	H	I
1	**Project Dates**			**House Building**					
2									
3		Target Delivery	Tue 27-Jan-98				Target Budget	£12,000	
4									
5		Job Stage	Start Date	Days Required	End Date		Daily Cost	Total	
6		Survey	Mon 05-Jan-98	5	Fri 09-Jan-98		£200	£1,000	
7		Foundation	Mon 12-Jan-98	4	Thu 15-Jan-98		£1,000	£4,000	
8		Walls	Fri 16-Jan-98	3	Tue 20-Jan-98		£800	£2,400	
9		Roof	Wed 21-Jan-98	6	Wed 28-Jan-98		£400	£2,400	
10		Electrics	Thu 29-Jan-98	4	Tue 03-Feb-98		£300	£1,200	
11									
12		Actual Delivery	Tue 03-Feb-98				Total Cost	£11,000	
13									
14		Against Target	5 days behind				Budget %	92%	
15									
16		Total Days	22						