1. Cell References and Naming

1.1 Using Cell References in Formulas

A cell reference identifies a cell or group of cells in a workbook. When you include cell references in a formula, the formula is linked to the referenced cells. The resulting value of the formula is dependent on the values in the referenced cells and changes automatically when the values in the referenced cells change. When you enter or edit a formula, press **[F4]** to change reference types quickly. *For example, from B2 to $B$2 to B$2 to $B2.*

1.1.1 Relative References

Relative References refer to cells by their position in relation to the cell that contains the formula. *For example, a relative reference to cell A1 looks like this: =A1.*

1.1.2 Absolute References

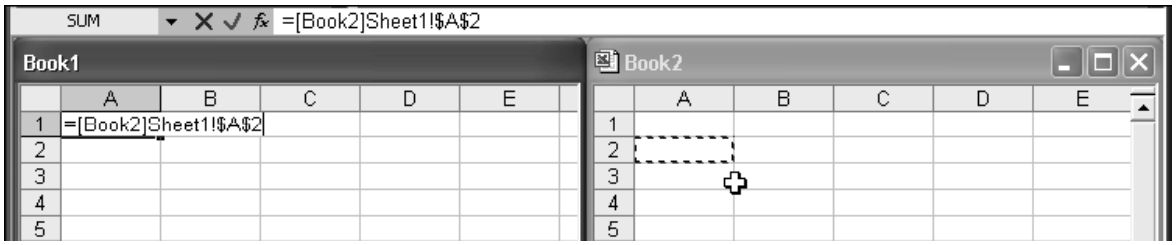
Absolute References refer to cells by their fixed position in the worksheet. *For example, an absolute reference to cell A1 looks like this: =$A$1.*

1.1.3 Mixed Reference

Mixed Reference contains a relative reference and an absolute reference. *For example, you can combine relative and absolute references to cell A1 to create these mixed references: =$A1 or =A$1.*

1.1.4 External References

You can refer to cells in worksheets in separate workbooks in the same way that you refer to cells in other worksheets within the same workbook.

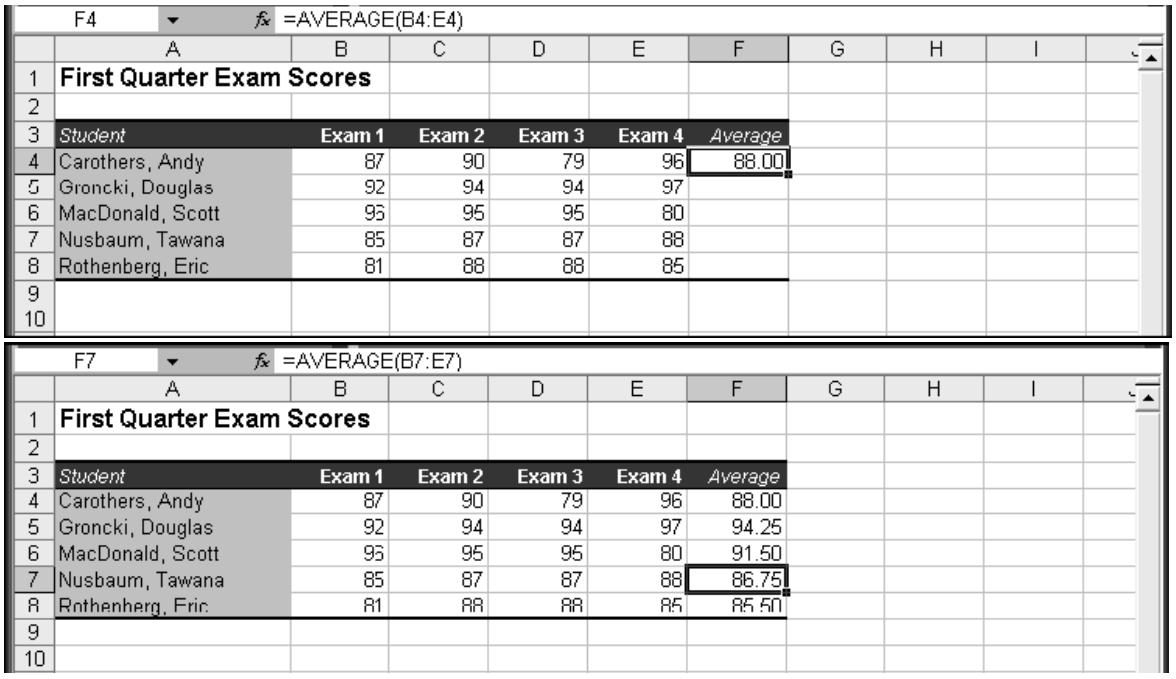


1.2 How Copying Affects Cell References

One of the handiest things about using references is the capability to copy and paste formulas. But you need to understand what happens to your references after you paste so that you can create formulas with references that operate the way you want them to.

1.2.1 Copying Relative References

When you copy a cell containing a formula with relative cell references, the references change automatically, relative to the position of the cell where you paste the formula. *For example, suppose you type the formula =AVERAGE(B4:E4) in cell F4. You want to repeat this calculation for the remaining rows as well, you select cell F4 and choose* ***Edit ➔ Copy****. Then you select cells F5:F8, choose* ***Edit ➔ Paste Special***. Because the formula in cell F4 contains a relative reference, Excel adjusts the references in each copy of the formula. *For example, cell F7 contains the formula =AVERAGE(B7:E7).*



F4 =AVERAGE(B4:E4)

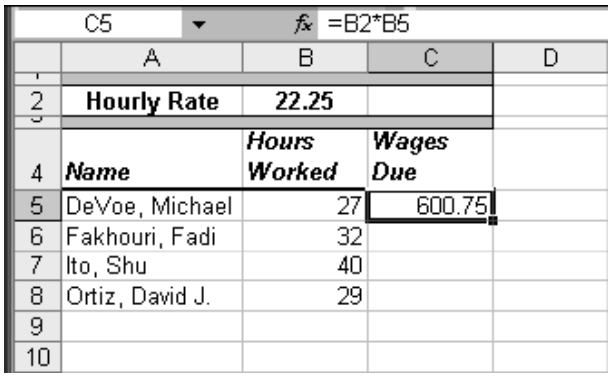
	A	B	C	D	E	F	G	H	I
1	**First Quarter Exam Scores**								
2									
3	*Student*	**Exam 1**	**Exam 2**	**Exam 3**	**Exam 4**	*Average*			
4	Carothers, Andy	87	90	79	96	88.00			
5	Groncki, Douglas	92	94	94	97				
6	MacDonald, Scott	95	95	95	80				
7	Nusbaum, Tawana	85	87	87	88				
8	Rothenberg, Eric	81	88	88	85				
9									
10									

F7 =AVERAGE(B7:E7)

	A	B	C	D	E	F	G	H	I
1	**First Quarter Exam Scores**								
2									
3	*Student*	**Exam 1**	**Exam 2**	**Exam 3**	**Exam 4**	*Average*			
4	Carothers, Andy	87	90	79	96	88.00			
5	Groncki, Douglas	92	94	94	97	94.25			
6	MacDonald, Scott	95	95	95	80	91.50			
7	Nusbaum, Tawana	85	87	87	88	86.75			
8	Rothenberg, Eric	81	88	88	85	85.50			
9									
10									

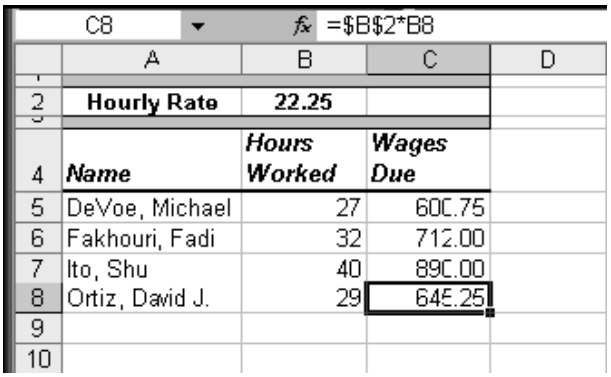
1.2.2 Copying Absolute References

If you want cell references to remain the same when you copy them, you must change the reference to an absolute reference by pressing [F4]. *For example, cell C5 contains the relative reference formula =B2*B5. After you change cell B2 to an absolute reference, the result in cell C5 is =$B$2*B5. When you copy this modified formula to cells C6:C8, cell C8 now contains the formula: =$B$2*B8*



C5 =B2*B5

	A	B	C	D
2	**Hourly Rate**	**22.25**		
4	***Name***	***Hours Worked***	***Wages Due***	
5	DeVoe, Michael	27	600.75	
6	Fakhouri, Fadi	32		
7	Ito, Shu	40		
8	Ortiz, David J.	29		
9				
10				

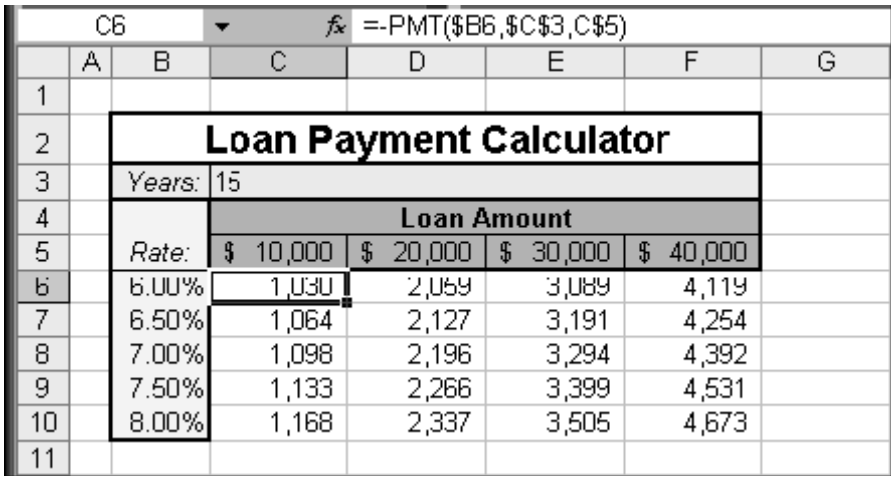


C8 =$B$2*B8

	A	B	C	D
2	**Hourly Rate**	**22.25**		
4	***Name***	***Hours Worked***	***Wages Due***	
5	DeVoe, Michael	27	600.75	
6	Fakhouri, Fadi	32	712.00	
7	Ito, Shu	40	890.00	
8	Ortiz, David J.	29	645.25	
9				
10				

1.2.3 Copying Mixed References

When you copy a mixed reference, Excel anchors the absolute portion and adjusts the relative portion to reflect the location of the cell to which you copied the formula. *For example, cell C6 contains the formula = –PMT ($B6,$C$3,C$5), the first cell reference, $B6, indicates that we always want to refer to the values in column B but the row reference can change. Similarly, the mixed reference, C$5, indicates that we always want to refer to the values in row 5 but the column reference can change. Therefore, cell E8 contains the formula =–PMT ($B8,$C$3,E$5).*



	A	B	C	D	E	F	G
1							
2		Loan Payment Calculator					
3		Years:	15				
4			Loan Amount				
5		Rate:	$ 10,000	$ 20,000	$ 30,000	$ 40,000	
6		6.00%	1,030	2,059	3,089	4,119	
7		6.50%	1,064	2,127	3,191	4,254	
8		7.00%	1,098	2,196	3,294	4,392	
9		7.50%	1,133	2,266	3,399	4,531	
10		8.00%	1,168	2,337	3,505	4,673	
11							

1.3 Naming Cells and Cell Ranges

1.3.1 Using Names in Formulas

When you use the name of a cell or a range in a formula by press **[Ctrl]** + **[F3]** to display the **Define Name** dialog box instantly, the result is the same as if you entered the cell or range address. *For example, suppose you entered the formula =A1+A2 in cell A3. If you defined the name "Test" as cell A1 and the name "Exam" as cell A2, the formula =Test+Exam has the same result*.

Keep the following basics in mind when using names in formulas:

- The Name box normally displays the address of the selected cell. If the selected cell or range is named, the name takes precedence over the address and is displayed in the Name box.
- When you define a name for a range of cells, the range name does not appear in the Name box unless the entire range is selected.
- When you click the Name box and select a name, the cell selection switches to the named cells.
- If you type a name that has already been defined into the Name box, Excel switches the selection instead of redefining the name.
- When you define a name, the address includes the worksheet name and the cell reference is absolute. For example, when you define the name TestName for cell C5 in Sheet1, the actual name definition is recorded as Sheet1!$C$5.

1.3.2 Rules for Naming

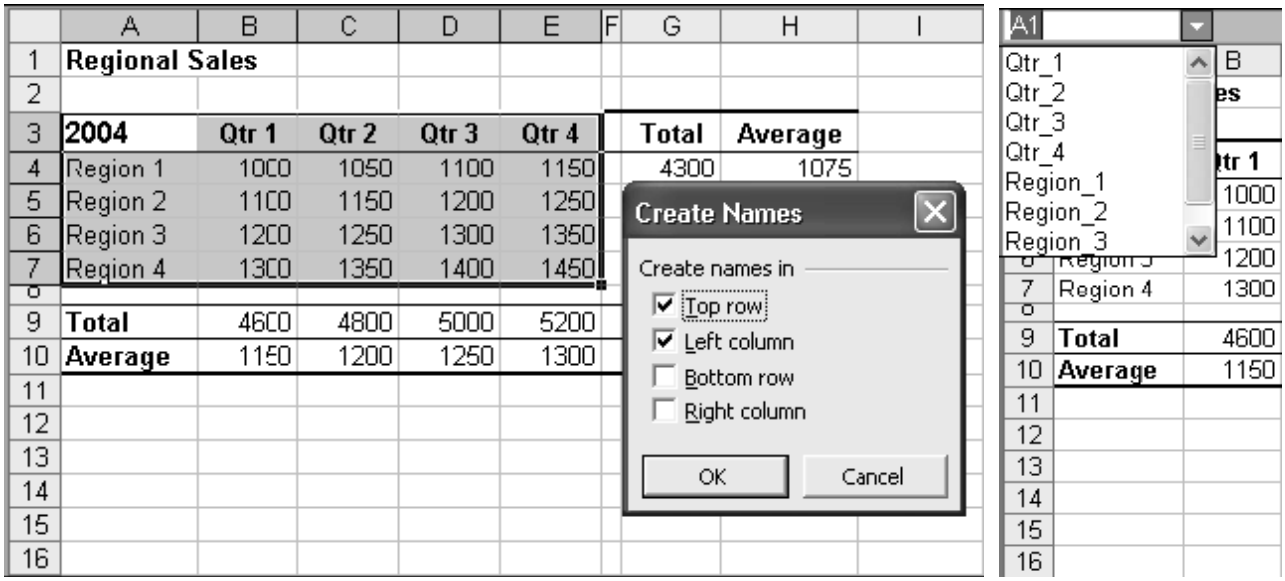
The following rules apply when you name cells and ranges in Excel:

- All names must begin with a letter, a backslash (\), or an underscore (_).
- Numbers can be used.
- Spaces cannot be used.
- Excel translates blank spaces in labels to underscores in defined names.
- Symbols other than backslash and underscore can't be used.
- Names that resemble cell references (*for example, AB$5 or R1C7*) can't be used.
- Single letters, with the exception of the letters R and C, can be used as names.

A name can contain 255 characters, but if it contains more than 253 characters, you can't select it from the Name box. Excel does not distinguish between uppercase and lowercase characters in names. *For example, if you create the name Test and then create the name TEST in the same workbook, the first name is overwritten by the second.*

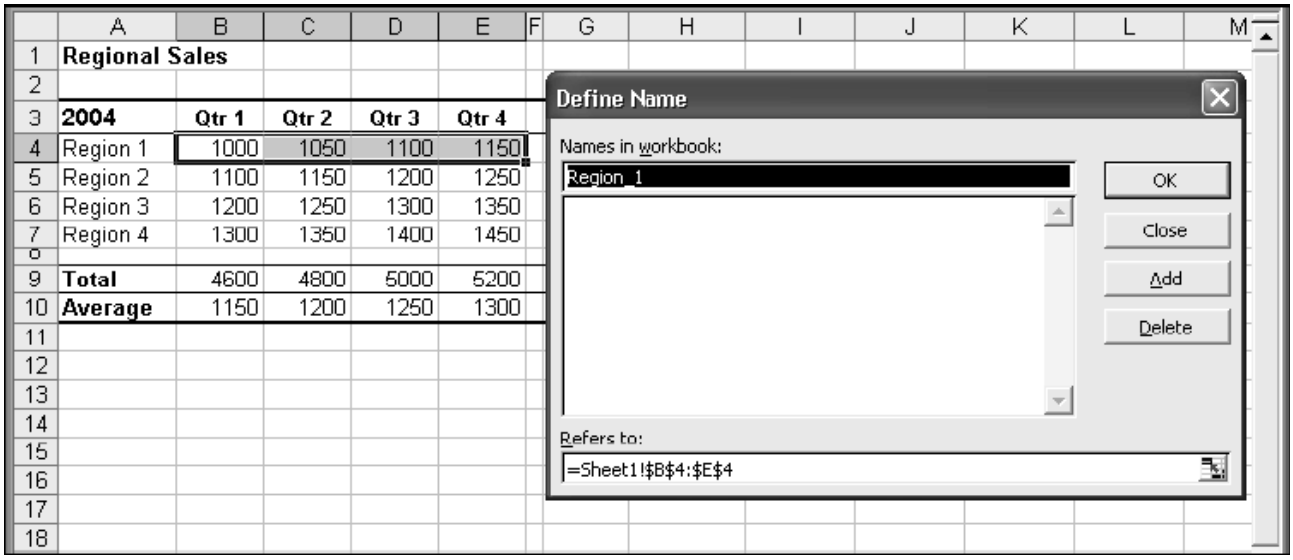
1.3.3 Creating Names

You can choose **Insert ➔ Name ➔ Create** to name several adjacent cells or ranges at once, using row labels, column labels, or both. Select the appropriate **Create Names In** options for the selected cells in the **Create Names** dialog box, and Excel uses the labels included in the selection to name each range. Please note that when using the Create Names command you actually need to select the labels as well as the data.



1.3.4 Defining and Managing Names

Instead of coming up with new names for cells and ranges, you can simply use existing text labels to create names. Choosing **Insert ➔ Name ➔ Define**, you can use text in adjacent cells to define cell and range names. You can choose this command also to redefine existing names.

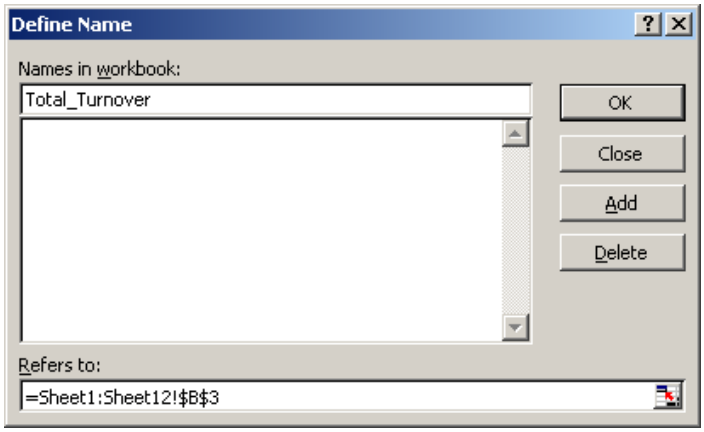


1.3.5 Creating Three-Dimensional Names

You can create three-dimensional names, which use 3-D references as their definitions. For example, suppose you have a 13-sheet workbook containing one identical sheet for each month plus one summary sheet. You can define a 3-D name that can be used to summarize totals from each monthly sheet by following these steps:

1. Select the target cell (*for example, cell B5*) in worksheet.
2. Choose **Insert ➔ Name ➔ Define**.
3. Type any name you choose in the **Names In Workbook** box and type
4. Type the formula (*for example, =Sheet1:Sheet12!B5*) in the **Refers To** box.
5. Press **[Enter]** to confirm

Now you can use this three-dimensional name in formulas that contain any of the following functions: SUM, MIN, VAR, COUNTA, PRODUCT, VARP, AVERAGE, STDEV, COUNT, MAX, and STDEVP.

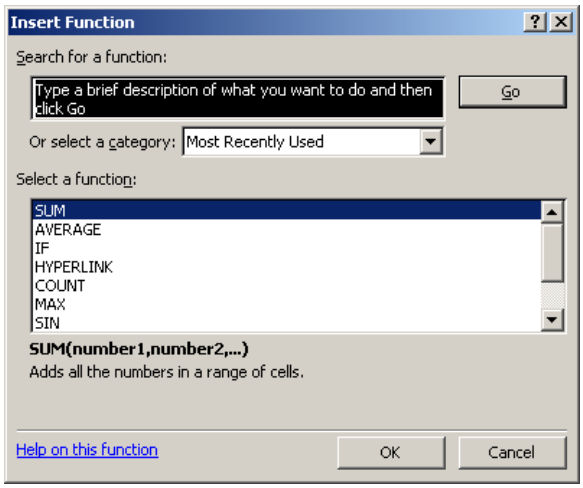


2. Using Function

In simplest terms, a function is a predefined formula. Many Excel functions are shorthand versions of frequently used formulas. For example, the SUM function adds a series of cell values by selecting a range by consolidate the formula =A1+A2+A3+A4+A5 to the formula =SUM(A1:A5).

2.1 Inserting Functions

When you want to use a built-in function, select **Insert ➔ Function** to display the **Insert Function** dialog box. This dialog box gives you access to all built-in functions in Excel.



2.2 Error Values

An error value is the result of a formula that Excel can't resolve. The seven error values are

Error	Reason
#DIV/0!	You attempted to divide a number by zero. This error usually occurs when you create a formula with a divisor that refers to a blank cell.
#NAME?	You entered a name in a formula that isn't in the Define Name dialog box list. You might have mistyped the name or typed a deleted name. Excel also displays this error value if you do not enclose a text string in quotation marks.
#VALUE	You entered a mathematical formula that refers to a text entry.
#REF!	You deleted a range of cells whose references are included in a formula.
#N/A	No information is available for the calculation you want to perform.
#NUM!	You provided an invalid argument to a worksheet function. It also indicates that the result of a formula is too large or too small to be represented.
#NULL!	You included a space between two ranges in a formula to indicate an intersection, but the ranges have no common cells.

2.3 Mathematical Functions

2.3.1 The SUM Function

The SUM function totals a series of numbers. It takes the form =SUM(number1, number2, …). The number arguments are a series of as many as 30 entries that can be numbers, formulas, ranges, or cell references that result in numbers. SUM ignores arguments that refer to text values, logical values, or blank cells.

2.3.2 The MOD Function

The MOD function returns the remainder of a division operation (modulus). It takes the arguments (number, divisor). The result of the MOD function is the remainder produced when number is divided by divisor. *For example, the function =MOD(9, 4) returns 1 (the remainder).*.

2.3.3 The COMBIN Function

The COMBIN function determines the number of possible combinations that can be taken from a pool of items. It takes the arguments (number, number_chosen), where number is the total number of items in the pool and number_chosen is the number of items you want to group in each combination. *For example, to determine how many different combination for Mark Six, type the formula =COMBIN(49, 6). The result indicates that 13983816 combinations could be created.*

2.3.4 The RAND Functions

The RAND function generates a random number between 0 and 1. It is one of the few Excel functions that do not take an argument. Note that you must still type the parentheses after the function name. The result of a RAND function changes each time you recalculate your worksheet. If you use automatic recalculation, the value of the RAND function changes each time you make a worksheet entry.

2.3.5 The ROUND, ROUNDDOWN, and ROUNDUP Functions

The ROUND function rounds a number to a specified number of decimal places, rounding digits less than 5 down and digits greater than or equal to 5 up. It takes the arguments (number, num_digits). If num_digits is a positive number, then number is rounded to the specified number of decimal points; if num_digits is negative, the function rounds to the left of the decimal point; if num_digits is 0, the function rounds to the nearest integer. *For example, the formula =ROUND(123.4567, –2) returns 100, and the formula =ROUND(123.4567, 3) returns 123.457. The ROUNDDOWN and ROUNDUP functions take the same form as ROUND. As their names imply, they always round down or up, respectively*.

2.3.6 The EVEN and ODD Functions

The EVEN function rounds a number up to the nearest even integer. The ODD function rounds a number up to the nearest odd integer. Negative numbers are correspondingly rounded down. *For example, the formula =EVEN(23.4) returns 24, and the formula =ODD(–4) returns –5.*

2.3.7 The FLOOR and CEILING Functions

The FLOOR function rounds a number down to the nearest given multiple, and the CEILING function rounds a number up to the nearest given multiple. These functions take the arguments (number, multiple). *For example, the formula =FLOOR(23.4, 0.5) returns 23, and the formula =CEILING(5, 1.5) returns 6.*

2.3.8 The INT Function

The INT function rounds numbers down to the nearest integer. When a number is negative, INT also rounds that number down to the next integer. *For example, the formulas =INT(100.01) and =INT(100.99999999) both return the value 100, even though the number 100.99999999 is essentially equal to 101. If each of the numbers in the examples were negative, the resulting value would be –101.*

2.3.9 The TRUNC Function

The TRUNC function truncates everything to the right of the decimal point in a number, regardless of its sign. It takes the arguments (number, num_digits). If num_digits isn't specified, it's set to 0. Otherwise, TRUNC truncates everything after the specified number of digits to the right of the decimal point. *For example, the formula =TRUNC(13.978) returns the value 13; the formula =TRUNC(13.978, 1) returns the value 13.9.*

2.4 Text Functions

Text functions in Excel are some of the most useful word-processing and data-management tools you'll find anywhere because they do things that word-processing programs can't do.

2.4.1 The TEXT Function

The TEXT function converts a number into a text string with a specified format. Its arguments are (value, format_text), where value represents any number, formula, or cell reference; and format_text is the format for displaying the resulting string. *For example, the formula =TEXT(98/4,"0.00") returns the text string 24.50. You can use any Excel formatting symbol ($, #, 0, and so on) except the asterisk (*) to specify the format you want, but you can't use the General format.*

2.4.2 The DOLLAR Function

Like the TEXT function, the DOLLAR function converts a number into a string. DOLLAR, however, formats the resulting string as currency with the number of decimal places you specify. The arguments (number, decimals) specify a number or reference and the number of decimal places you want. *For example, the formula =DOLLAR(45.899, 2) returns the text string $45.90. Notice that Excel rounds the number when necessary.*

If you omit decimals, Excel uses two decimal places. If you add a comma after the first argument but omit the second argument, Excel uses zero decimal places. If you use a negative number for decimals, Excel rounds to the left of the decimal point.

2.4.3 The LEN Function

The LEN function returns the number of characters in an entry. The single argument can be a number, a string enclosed in double quotation marks, or a reference to a cell. Trailing zeros are ignored. *For example, the formula =LEN("Test") returns 4.*

2.4.4 The ASCII Functions: CHAR and CODE

Every computer uses numeric codes to represent characters. The most prevalent system of numeric codes is ASCII (American Standard Code for Information Interchange). ASCII uses a number from 0 to 127 (or in some systems, to 255) to represent each number, letter, and symbol.

The CHAR and CODE functions deal with these ASCII codes. The CHAR function returns the character that corresponds to an ASCII code number; the CODE function returns the ASCII code number for the first character of its argument. For example, the formula =CHAR(83) returns the text “S”. The formula =CODE("S") returns the ASCII code 83. If you type a literal character as the text argument, be sure to enclose the character in quotation marks; otherwise, Excel returns the #NAME? error value.

2.4.5 The Clean-Up Functions: TRIM and CLEAN

Leading and trailing blank characters often prevent you from correctly sorting entries in a worksheet or a database. If you use string functions to manipulate text in your worksheet, extra spaces can prevent your formulas from working correctly.

The TRIM function eliminates leading, trailing, and extra blank characters from a string, leaving only single spaces between words.

The CLEAN function is similar to TRIM, except it operates on only nonprintable characters, such as tabs and program-specific codes. CLEAN is especially useful if you import data from another program or operating system, because the translation process often introduces nonprintable characters that appear as symbols or boxes. You can use CLEAN to remove these characters from the data.

2.4.6 The EXACT Function

The EXACT function is a conditional function that determines whether two strings match exactly. The function ignores formatting, but it is case-sensitive, so uppercase letters are considered different than lowercase letters. If both strings are identical, the function returns TRUE. Both arguments must be literal strings enclosed in quotation marks, references to cells that contain text, numeric values, or formulas that evaluate to numeric values. *For example, if cell A5 and cell A6 of your worksheet both contain the text "Totals", the formula =EXACT(A5, A6) returns TRUE.*

2.4.7 The Case Functions: UPPER, LOWER, and PROPER

Three functions manipulate the case of characters in text strings. The UPPER and LOWER functions convert text strings to all uppercase or all lowercase letters. The PROPER function capitalizes the first letter in each word, capitalizes any other letters in the text string that do not follow another letter, and converts all other letters to lowercase. For example, if cell A1 contains the text "hello World", you can type the formula =UPPER(A1) to return "HELLO WORLD". Similarly, the formula =LOWER(A1) returns "hello world", and =PROPER(A1) returns "Hello World".

2.5 Substring Text Functions

The following functions locate and return portions of a text string or assemble larger strings from smaller ones: FIND, SEARCH, RIGHT, LEFT, MID, SUBSTITUTE, REPLACE, and CONCATENATE.

2.5.1 The FIND and SEARCH Functions

You use the FIND and SEARCH functions to locate the position of a substring within a string. Both functions return the position in the string of the character you specify (Excel counts blank spaces and punctuation marks as characters). These two functions work the same way, except FIND is case sensitive and SEARCH allows wildcards. Both functions take the same arguments: (find_text, within_text, start_num). The optional start_num argument is helpful when within_text contains more than one occurrence of find_text. If you omit start_num, Excel reports the first match it locates. *For example, to locate the "x" in the string "Advanced Microsoft Excel", you would type the formula =FIND("x", "Advanced Microsoft Excel"). The formula returns 21, because "x" is the 21st character in the string.*

If you're not sure of the character sequence you're searching for, you can use the SEARCH function and include wildcards in your find_text string. *Suppose you've used the names "Smith" and "Smyth" in your worksheet. To determine whether either name is in cell A1, type the formula =SEARCH("Sm?th", A1). If cell A1 contains the text "John Smith" or "John Smyth", the SEARCH function returns the value 6 where is the starting point of the string Sm?th.*

2.5.2 The RIGHT and LEFT Functions

The RIGHT function returns the rightmost series of characters from a specified string; the LEFT function returns the leftmost series of characters. These functions take the same arguments: (text, num_chars). The num_chars argument indicates the number of characters to extract from the text argument.

These functions count blank spaces in the text argument as characters; if text contains leading or trailing blank characters, you might want to use a TRIM function within the RIGHT or LEFT function to ensure the expected result. *For example, suppose you type "Advanced Microsoft Excel" in cell A1 of your worksheet. The formula =RIGHT(A1,5) returns the word "Excel*".

2.5.3 The MID Function

You can use the MID function to extract a series of characters from a text string. This function takes the arguments (text, start_num, num_chars). *For example, if cell A1 contains the text "Advanced Microsoft Excel", you can type the formula =MID(A1, 10, 9) to extract the text "Microsoft" from the entry in cell A1.*

2.5.4 The REPLACE and SUBSTITUTE Functions

The REPLACE and SUBSTITUTE functions substitute new text for old text. The REPLACE function replaces one string of characters with another string of characters and takes the arguments (old_text, start_num, num_chars, new_text). *Suppose cell A1 contains the text "Advanced Microsoft Excel". To replace the first four characters with the string "Beginning", type the formula =REPLACE(A1, 1, 8, "Beginning"). The result is "Beginning Microsoft Excel"*.

With the SUBSTITUTE function, you specify the text to replace. The function takes the arguments (text, old_text, new_text, instance_num). The instance_num argument optionally replaces only the specified occurrence of old_text. If you don't include instance_num, Excel changes all occurrences of old_text to new_text. *Suppose cell A1 contains the text "Advanced Microsoft Excel" and you want to place it in cell A2 but change it to "Advanced Microsoft Word". Type this formula in cell A2 =SUBSTITUTE(A1,"Excel","Word").*

2.5.5 The CONCATENATE Function

To assemble strings from up to 30 smaller strings or references, the CONCATENATE function is the function equivalent of the & character. *For example, if cell A1 contains the text "Hello" with a trailing space character, the formula =CONCATENATE(A1 "World") returns "Hello World"*.

2.6 Logical Functions

You use logical functions to test for specific conditions. These functions are often called logical operators in discussions of Boolean logic. You use logical operators to arrive at one of two conclusions: TRUE or FALSE. We'll discuss the most useful logical functions in this section.

2.6.1 The SUMIF and COUNTIF Function

With SUMIF, you can add specific values in a range, based on a criterion you supply. This performs all the calculations you need in one cell, and eliminates having to create a column of IF formula. *For example, you can type the formula =SUMIF(C12:C27, "Pass", A12:A27) to find the total of all numbers in A12:A27 in which the cell in the same row in column C contains the word "Pass". s.*

Similarly, COUNTIF counts the cells that match specified criteria and takes the arguments (range, criteria). *For example, you can find the number of months in which sales fell below $600 using a conditional test, as in the formula =COUNTIF(Sales, "<600").*

2.6.2 The IF Function

The IF function returns values based on supplied conditional tests. It takes the arguments (logical_test, value_if_true, value_if_false). *For example, the formula =IF(A1<22, 5, 10) returns 5 if the value in cell A1 is less than 22; otherwise, it returns 10.*

You can nest other functions and use text arguments to return nothing instead of zero if the result is false. *For example, the formula =IF(SUM(A1:A10)>0, SUM(A1:A10), " ") returns a null string (" ") if the conditional test is false.*

2.6.3 The AND, OR, and NOT Functions

The functions work with the logical operators =, >, <, >=, <=, and <>. The AND and OR functions can each have as many as 30 logical arguments. The NOT function takes only one argument which can be conditional tests, arrays, or references to cells that contain logical values).

The OR function returns the logical value TRUE if any one of the conditional tests is true; the AND function returns the logical value TRUE only if all the conditional tests are true. NOT instructs Excel to return the logical value TRUE if the argument is false or the logical value FALSE if the argument is true. The truth table for AND, OR and NOT functions are listed below:

x	y	x AND y
T	T	T
T	F	F
F	T	F
F	F	F

x	y	x OR y
T	T	T
T	F	F
F	T	F
F	F	F

x	NOT x
T	F
F	T

Suppose you want to return the text "Pass" only if a student has an average score above 75 and fewer than five unexcused absences. If we typed the formula =IF(AND(G4<5, F4>75), "Pass", "Fail"). This fails the student in row 5 because of the five absences. If you use OR instead of AND in the formula, all students would pass.

2.7 Information Functions

Information functions allow you to gather information about the contents of cells, their formatting, and the computing environment as well as perform conditional tests for the presence of specific types of values.

2.7.1 The TYPE and ERROR.TYPE Functions

The TYPE function determines whether a cell contains text, a number, a logical value, an array, or an error value. The result is a code for the type of entry in the referenced cell: 1 for a number (or a blank cell), 2 for text, 4 for a logical value (TRUE or FALSE), 16 for an error value, and 64 for an array. *For example, if cell A1 contains the number 100, the formula =TYPE(A1) returns 1. If A1 contains the text "Hello", the formula returns 2.*

Like the TYPE function, the ERROR.TYPE function detects the contents of a cell, except it detects different types of error values. The result is a code for the type of error value in the referenced cell: 1 for #NULL!, 2 for #DIV/0!, 3 for #VALUE!, 4 for #REF!, 5 for #NAME!, 6 for #NUM!, and 7 for #N/A. Any other value in the referenced cell returns the error value #N/A. *For example, if cell A1 contains a formula that displays the error value #NAME!, the formula =ERROR.TYPE(A1) returns 5. If A1 contains the text Microsoft Excel, the formula returns #N/A.*

2.7.2 The COUNTBLANK Function

The COUNTBLANK function counts the number of empty cells in the specified range, which is its only argument. However, this function is tricky because formulas that evaluate to null text strings, such as =" ", or to zero might seem empty, but they aren't and therefore won't be counted.

2.7.3 Using the IS Information Functions

You can use the ISBLANK, ISERR, ISERROR, ISLOGICAL, ISNA, ISNONTEXT, ISNUMBER, ISREF, and ISTEXT functions to determine whether a referenced cell or range contains the corresponding type of value. All IS Information functions take a single argument. *For example, the ISBLANK function takes the form =ISBLANK(value). If value refers to a blank cell, the function returns the logical value TRUE; otherwise, it returns FALSE.*

2.7.4 An ISERR Example

You can use ISERR to avoid getting error values as formula results. *For example, the FIND function returns the position at which a substring is found within a larger string. If the substring isn't there, FIND returns #VALUE!. Adding an ISERR function, such as =IF(ISERR(FIND("12A", A1)), " ", "Yes"). Because you're not interested in the error, which is simply a by-product of the calculation, this traps the error, leaving only the results that you are interested in.*

2.8 Lookup Functions

Lookup functions help you use your own worksheet tables as sources of information to be used elsewhere in formulas. You can use three primary functions to look up information stored in a list or a table or to manipulate references: LOOKUP, VLOOKUP, and HLOOKUP.

VLOOKUP and HLOOKUP are nearly identical functions that look up information stored in tables you have constructed. VLOOKUP and HLOOKUP operate in either vertical or horizontal orientation (respectively), but LOOKUP works either way.

When you look up information in a table, you normally use a row index and a column index to locate a particular cell. Excel derives the first index by finding the largest value in the first column or row that is less than or equal to a lookup value you supply and then uses a row number or column number argument as the other index. Make sure the table is sorted by the row or column containing the lookup values. These functions take the following forms:

=VLOOKUP(lookup_value, table_array, col_index_num, range_lookup)

=HLOOKUP(lookup_value, table_array, row_index_num, range_lookup)

The LOOKUP function takes two forms, the first is called the vector form, and the second is called the array form:

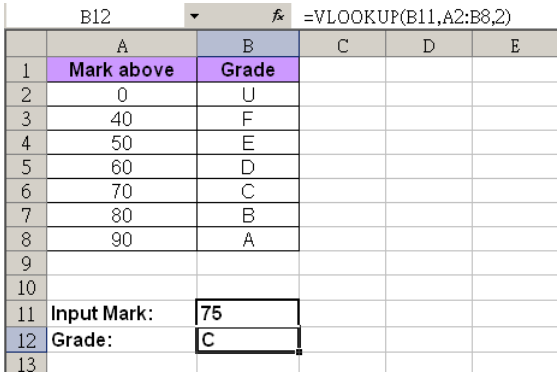
=LOOKUP(lookup_value, lookup_vector, result_vector)

=LOOKUP(lookup_value, array)

LOOKUP Function Argument	Description
lookup_value	The value, cell reference, or text (enclosed in quotation marks) that you want to find in a table or a range.
table_array	A cell range or name that defines the table to look in.
row_index_num, col_index_num	The row or column number of the table from which to select the result, counted relative to the table (not according to the actual row and column numbers).
range_lookup	A logical value that determines whether the function matches the lookup_value exactly or approximately. Type FALSE to match the lookup_value exactly. The default is TRUE, which finds the closest match.
lookup_vector	A one-row or one-column range that contains numbers, text, or logical values.
result_vector	A one-row or one-column range that must be the same size as lookup_vector.
array	A range containing numbers, text, or logical values to compare with lookup_value.

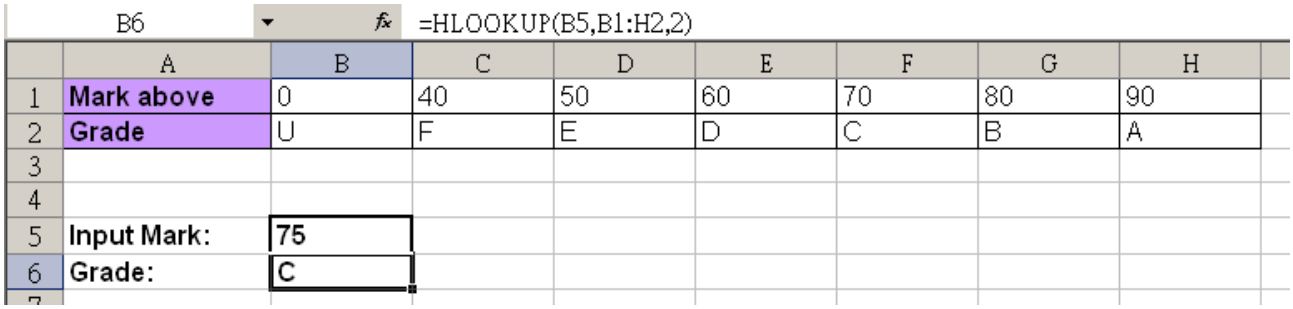
2.8.1 The VLOOKUP and HLOOKUP Functions

For the VLOOKUP and HLOOKUP functions, whether a lookup table should be considered vertical or horizontal depends on where the comparison values (the first index) are located. If the values are in the leftmost column of the table, the table is vertical; if they are in the first row of the table, the table is horizontal. The comparison values can be numbers or text, but they must be arranged in ascending order. No comparison value should be used more than once in a table.



B12 =VLOOKUP(B11,A2:B8,2)

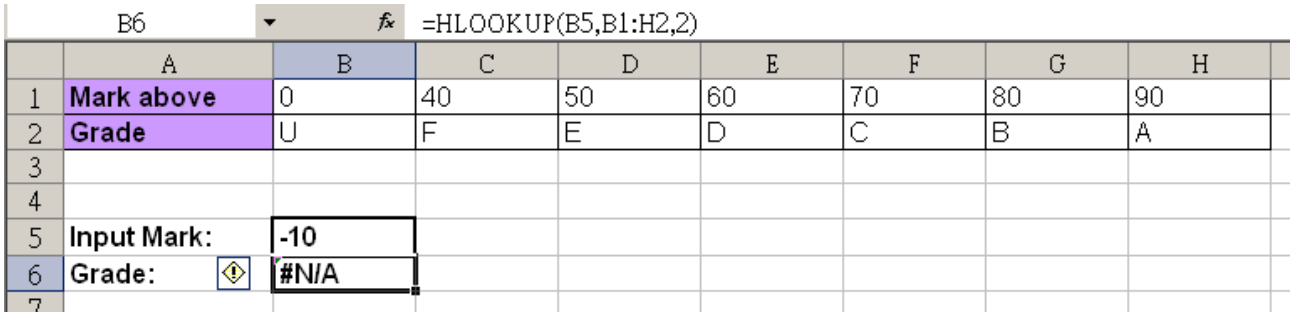
	A	B	C	D	E
1	Mark above	Grade			
2	0	U			
3	40	F			
4	50	E			
5	60	D			
6	70	C			
7	80	B			
8	90	A			
9					
10					
11	Input Mark:	75			
12	Grade:	C			
13					



B6 =HLOOKUP(B5,B1:H2,2)

	A	B	C	D	E	F	G	H
1	Mark above	0	40	50	60	70	80	90
2	Grade	U	F	E	D	C	B	A
3								
4								
5	Input Mark:	75						
6	Grade:	C						

Remember that these lookup functions normally search for the greatest comparison value that is less than or equal to the lookup value, not for an exact match between the comparison values and the lookup value. If all the comparison values in the first row or column of the table range are greater than the lookup value, the function returns the #N/A error value. If all the comparison values are less than the lookup value, however, the function returns the value that corresponds to the last (largest) comparison value in the table, which might not be what you want. If you require an exact match, type FALSE as the range_lookup argument.



B6 =HLOOKUP(B5,B1:H2,2)

	A	B	C	D	E	F	G	H
1	Mark above	0	40	50	60	70	80	90
2	Grade	U	F	E	D	C	B	A
3								
4								
5	Input Mark:	-10						
6	Grade:	#N/A						

2.8.2 The LOOKUP Function

The array form of LOOKUP is similar to VLOOKUP and HLOOKUP but works with either a horizontal or a vertical table, using the dimensions of the table to figure out the location of the comparison values. If the table is taller than it is wide or the table is square, the function treats it as a vertical table and assumes that the comparison values are in the leftmost column. If the table is wider than it is tall, the function views the table as horizontal and assumes that the comparison values are in the first row of the table. The result is always in the last row or column of the specified table; you can't specify column or row numbers. Because HLOOKUP and VLOOKUP are more predictable and controllable, you'll generally find using them preferable to using LOOKUP.

The lookup_vector and result_vector arguments are often adjacent ranges, but they don't have to be when you use LOOKUP. They can be located in separate areas of the worksheet, and one range can be horizontal and the other vertical. The only requirement is that they must have the same number of elements.

B14 =LOOKUP(B13, B2:B8,B10:H10)

	A	B	C	D	E	F	G	H
1	**Comment**	**Mark Above**						
2	Unclassified	0						
3	Fail	40						
4	Poor	50						
5	Fair	60						
6	Average	70						
7	Good	80						
8	Excellent	90						
9								
10	**Grade**	U	F	E	D	C	B	A
11	**No. of Student**	10	20	30	40	30	20	10
12								
13	**Input Mark:**	75						
14	**Grade:**	C						
15								

2.9 Reference Functions

2.9.1 The CHOOSE Function

You use the CHOOSE function to retrieve an item from a list of values. The function takes the arguments (index_num, value 1, value 2, ...) and can include up to 29 values. The index_num argument is the position in the list you want to return; it must be positive and can't exceed the number of elements in the list. The function returns the value of the element in the list that occupies the position indicated by index_num. *For example, the function =CHOOSE(2, "A", "B", "C", "D", "E") returns the value "B".*

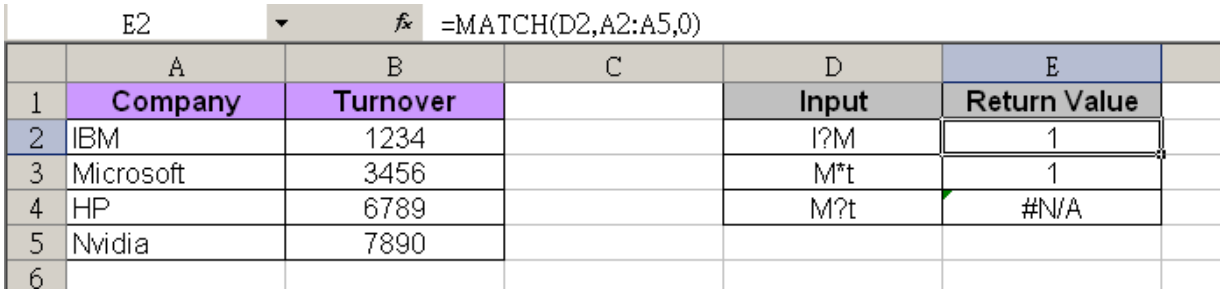
You can use individual cell references for the list, but you can't specify ranges. You might be tempted to create a function, *such as =CHOOSE(A10, C1:C5), to take the place of the longer function in the preceding example. If you do, however, the result is a #VALUE! error value.*

2.9.2 The MATCH Function

The MATCH function returns the position of the item in the list that most closely matches a lookup value. This function takes the arguments (lookup_value, lookup_array, match_type), where lookup_value and the items in the lookup_array can be numeric values or text strings, and match_type defines the rules for the search.

match_type	Description
1 (default)	Finds the largest value in the specified range (which must be sorted in ascending order) that is less than or equal to lookup_value. If no items in the range meet these criteria, the function returns #N/A.
0	Finds the first value in the specified range (no sorting necessary) that is equal to lookup_value. If no items in the range match, the function returns #N/A.
-1	Finds the smallest value in the specified range (which must be sorted in descending order) that is greater than or equal to lookup_value. If no items in the range meet these criteria, the function returns #N/A.

When you use MATCH to locate text strings, you should specify a match_type argument of 0 (an exact match). You can then use the wildcards * and ? in the lookup_value argument.



	A	B	C	D	E
1	Company	Turnover		Input	Return Value
2	IBM	1234		I?M	1
3	Microsoft	3456		M*t	1
4	HP	6789		M?t	#N/A
5	Nvidia	7890			
6					

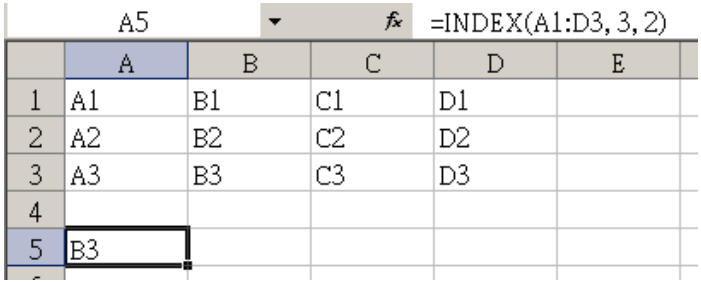
2.9.3 The INDEX Function

The INDEX function has two forms: an array form, which returns a value, and a reference form, which returns a cell reference. The forms of these functions are

=INDEX(array, row_num, column_num)

=INDEX(reference, row_num, column_num, area_num)

The array form works only with an array argument; it returns the value of the result, not the cell reference. The result is the value at the position in array indicated by row_num and column_num. The reference form returns a cell address instead of a value and is useful when you want to perform operations on a cell, rather than on its value.



	A	B	C	D	E
1	A1	B1	C1	D1	
2	A2	B2	C2	D2	
3	A3	B3	C3	D3	
4					
5	B3				

2.10 Date and Time Functions

2.10.1 The TODAY and NOW Functions

You can enter =TODAY( ) into a cell or a formula to insert the serial value of the current date. Similarly, you can enter =NOW( ) into a cell or formula to insert the current date and time. The result of the function is a serial date and time value that includes an integer (the date) and a decimal value (the time).

2.10.2 The YEAR, MONTH, and DAY Functions

The YEAR, MONTH, and DAY functions return the value of the year, month, and day portions of a serial date value. All three take a single argument, which can be a serial date value; a reference to a cell that contains either a date function or a serial date value; or a text date enclosed in quotation marks. *For example, if cell A1 contains the date 31/12/1999, the formula =YEAR(A1) returns the value 1999, the formula =MONTH(A1) returns the value 12, and the formula =DAY(A1) returns the value 31*

.

2.10.3 The HOUR, MINUTE, and SECOND Functions

Just as the YEAR, MONTH, and DAY functions extract the value of the year, month, and day portions of a serial date value, the HOUR, MINUTE, and SECOND functions extract the value of the hour, minute, and second portions of a serial time value. *For example, if cell A1 contains the time 12:15:35 PM, the formula =HOUR(A1) returns the value 12, the formula =MINUTE(A1) returns the value 15, and the formula =SECOND(A1) returns the value 25.*

2.10.4 The WEEKDAY Function

The WEEKDAY function returns the day of the week for a specific date and takes the arguments (serial_number, return_type). The serial_number argument can be a serial date value; a reference to a cell that contains either a date function or a serial date value; or text, such as 31/12/1999. The function returns a number that represents the day of the week that the specified date falls on. The optional return_type argument determines the way the result is displayed.

If return_type is	WEEKDAY returns
1 (default)	A number from 1 through 7 where 1 is Sunday and 7 is Saturday
2	A number from 1 through 7 where 1 is Monday and 7 is Sunday
3	A number from 0 through 6 where 0 is Monday and 6 is Sunday

3. Auditing Worksheet

Excel has a number of powerful and flexible features that help you audit and debug your worksheets and document your work. Most auditing features can be accessed via the Formula Auditing toolbar. This toolbar can be displayed by selecting **View ➔ Toolbars ➔ Formula Auditing**.

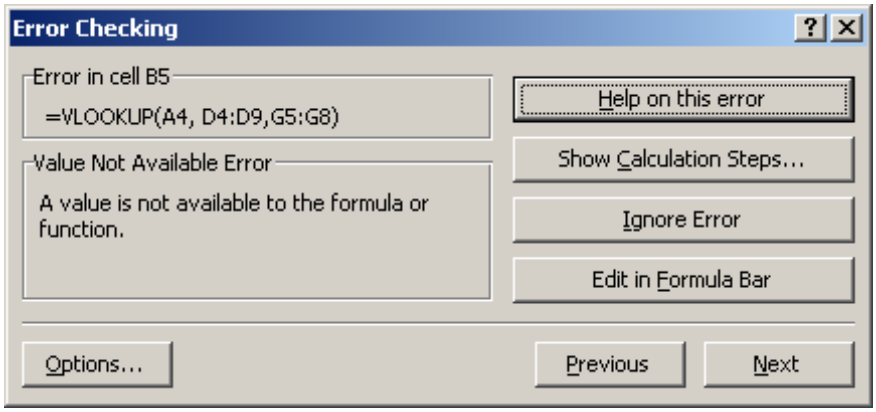
3.1 Redisplay the Active Cell

If you scroll through your worksheet and the active cell is no longer visible, you can redisplay it by pressing **[Ctrl] + [Backspace]** button.

3.2 Error Checking

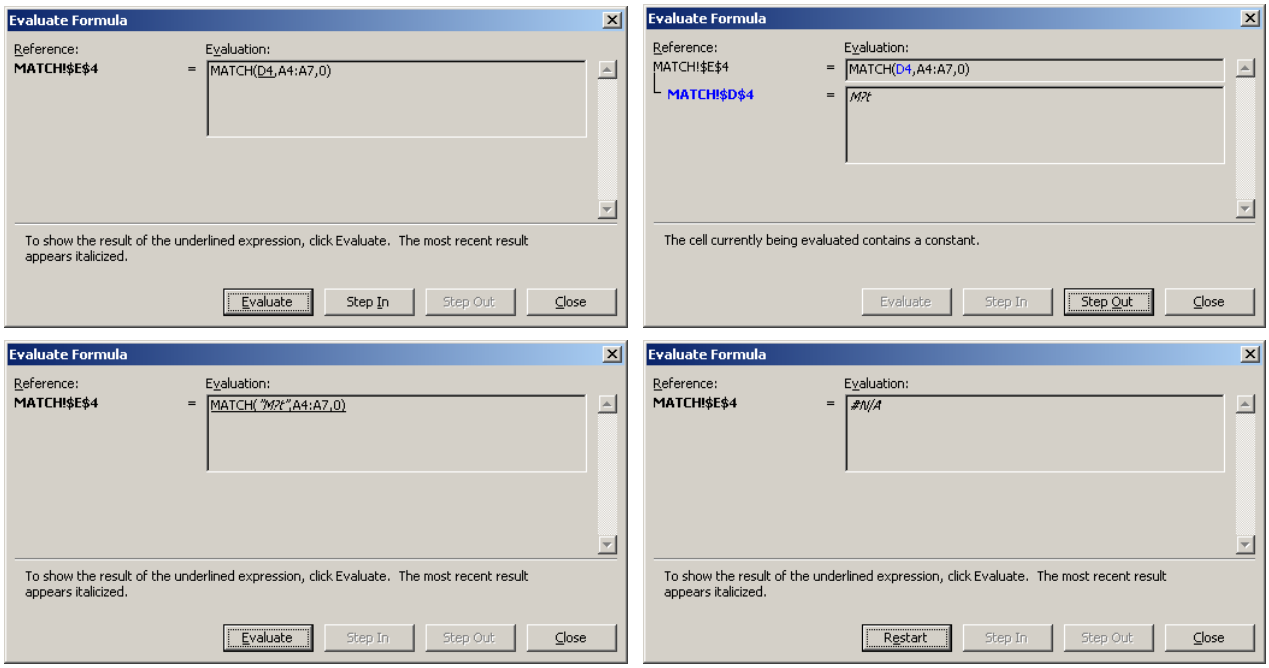
Choose **Tools ➔ Error Checking** to quickly find any error values in the current worksheet and display the **Error Checking** dialog box. The first erroneous cell in the worksheet is selected and its contents are displayed in the dialog box, along with a suggestion about the nature of the problem. When your problem appears in the dialog box, the following selections are available:

- **Help on this Error** displays a Help topic relating to the problem cell.
- **Show Calculation Steps** displays the Evaluate Formula dialog box.
- **Ignore Error** skips over the selected cell. To reset the ignored errors, click **[Options]** button and then click **[Reset Ignored Errors]** button.
- **Edit in Formula Bar** opens the selected cell in the formula bar for editing. When you're finished, click **[Resume]** button.



3.3 Evaluating and Auditing Formulas

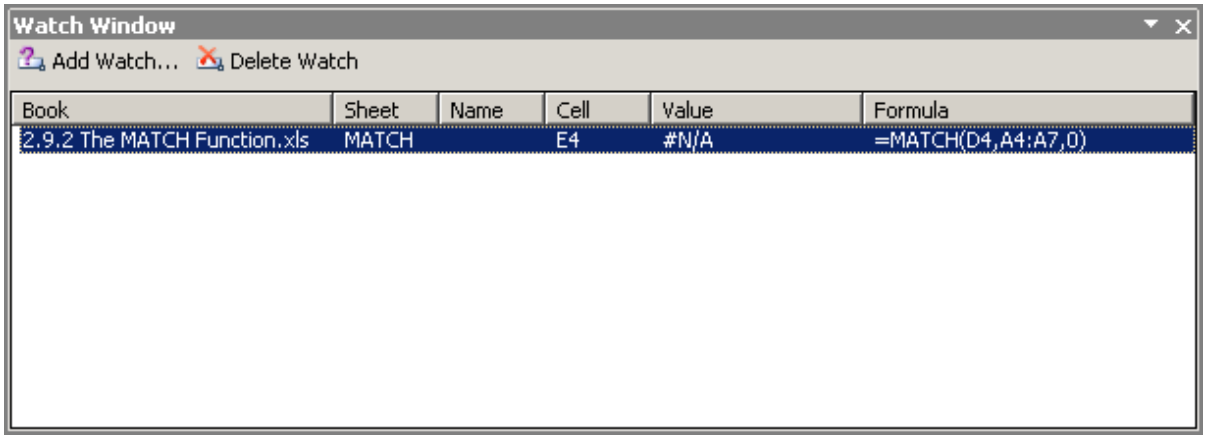
Sometimes it's difficult to tell what's going on in a complex nested formula. When you choose **Tools ➔ Formula Auditing ➔ Evaluate Formula**, you can check complex formulas easily. Click **[Evaluate]** button to replace calculable arguments with their resulting values. You can click Evaluate as many times as necessary if your formula contains many nested levels.



3.4 Watching Formulas

Sometimes you might want to keep an eye on a formula as you make changes to other parts of a worksheet, or even when working on other workbooks that supply information to a worksheet. Instead of constantly having to return to the formula's location to see the results of your ministrations, you can use the Watch Window, which provides remote viewing for any cell on any open worksheet.

Select a cell you want to keep an eye on and choose **Tools ➔ Formula Auditing ➔ Show Watch Window**. Then select the cell and click **[Add Watch]** button in the **Add Watch** window.



3.5 Tracing Cell References

You can also use cell tracers to help find the source of those pesky errors that occasionally appear in your worksheets.

3.5.1 Understanding Precedents and Dependents

The terms precedent and dependent crop up quite often in this section. They refer to the relationships that cells containing formulas create with other cells. A lot of what a spreadsheet is all about is wrapped up in these concepts, so here's a brief description of each term:

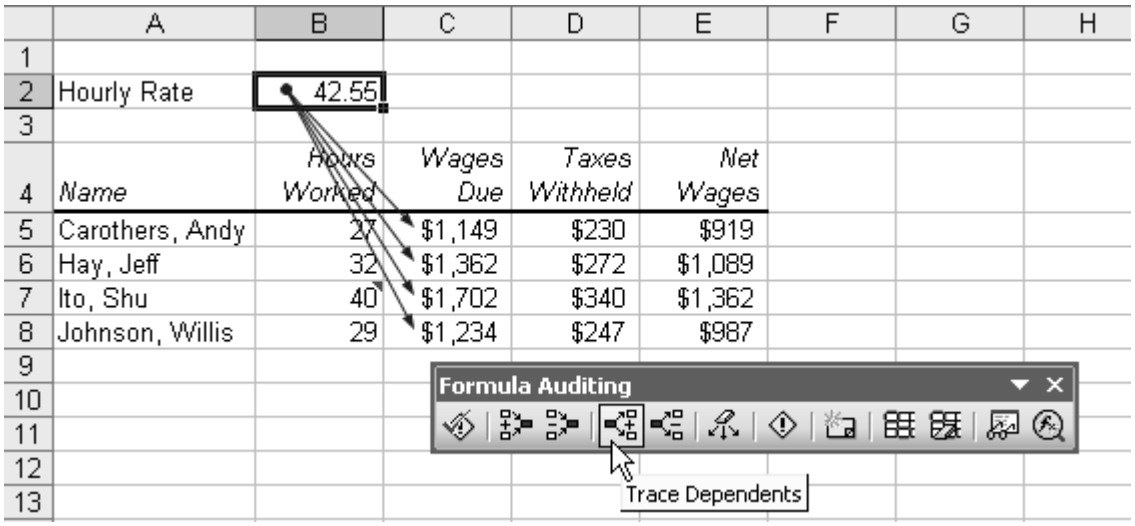
- Precedents are cells whose values are used by the formula in the selected cell. A cell that has precedents always contains a formula.
- Dependents are cells that use the value in the selected cell. A cell that has dependents can contain either a formula or a constant value.

For example, if the formula =SUM(A1:A5) is in cell A6, cell A6 has precedents (A1:A5) but no apparent dependents. Cell A1 has a dependent (A6), but no apparent precedents. A cell can be both a precedent and a dependent if the cell contains a formula and is also referenced by another formula.

3.5.2 Tracing Dependent Cells

To find out which cells contain formulas that use this value, you can click the **[Trace Dependents]** button on the **Formula Auditing** toolbar. Although this worksheet is elementary, to make it easier to illustrate the cell tracers, consider the ramifications of using the cell tracers in a large and complex worksheet.

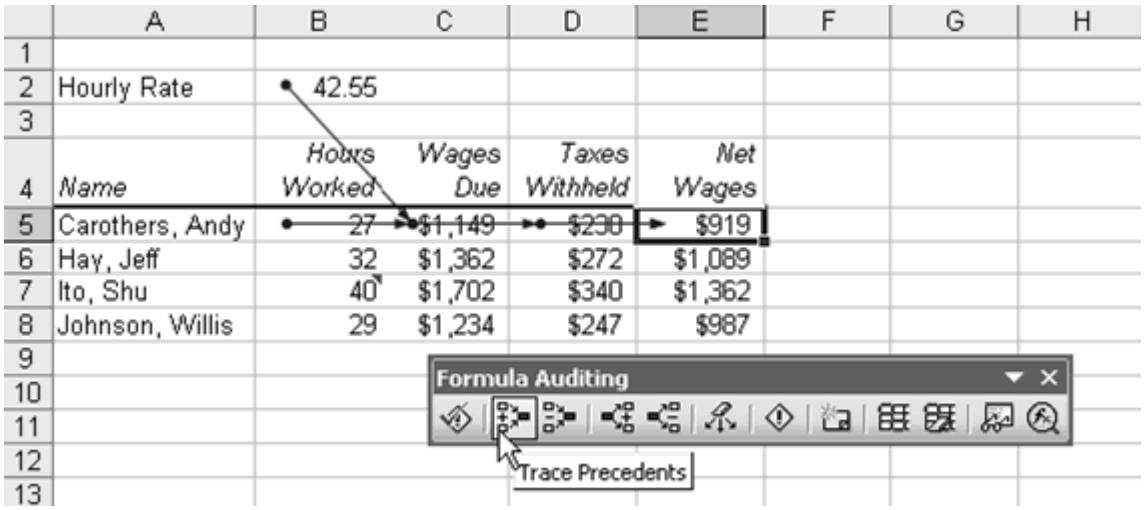
The tracer arrows indicate that cell is directly referred to by the formulas in another cells. The dot appears in the cell indicating that it is has dependents. If you click Trace Dependents again, another set of arrows appears, indicating the next level of dependencies or indirect dependents.



3.5.3 Tracing Precedent Cells

You can also trace in the opposite direction by starting from a cell that contains a formula and tracing the cells that are referred to in the formula. To find out which cells this formula refers to, we clicked **[Trace Precedents]** button.

The dots identify these cells as precedents in the data flow. Notice that the arrow still points in the same direction (toward the formula and in the direction of the data flow) even though we started from the opposite end of the path. To continue the trace, click the **[Trace Precedents]** button again.



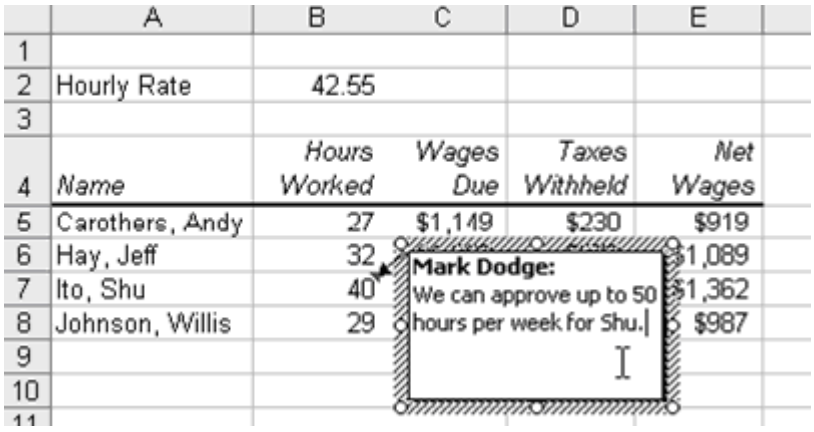
	A	B	C	D	E
2	Hourly Rate	42.55			
4	*Name*	*Hours Worked*	*Wages Due*	*Taxes Withheld*	*Net Wages*
5	Carothers, Andy	27	$1,149	$230	$919
6	Hay, Jeff	32	$1,362	$272	$1,089
7	Ito, Shu	40	$1,702	$340	$1,362
8	Johnson, Willis	29	$1,234	$247	$987

3.5.4 Clearing Tracer Arrows

Each time you trace another cell's precedents or dependents, additional tracer arrows appear. It's a good idea to start fresh each time you want to trace cells. To remove all the tracer arrows from the screen, click the **[Remove All Arrows]** button on the **Formula Auditing** toolbar.

3.6 Adding Comments to Cells

You can attach comments to cells to document your work, explain calculations and assumptions, or provide reminders. Select the cell you want to annotate and click the **[New Comment]** button on the **Formula Auditing** toolbar. Then type your message in the comment box that appears.



	A	B	C	D	E
2	Hourly Rate	42.55			
4	*Name*	*Hours Worked*	*Wages Due*	*Taxes Withheld*	*Net Wages*
5	Carothers, Andy	27	$1,149	$230	$919
6	Hay, Jeff	32			$1,089
7	Ito, Shu	40			$1,362
8	Johnson, Willis	29			$987

4. Function Reference

4.1 Database Functions

Function	Description
DAVERAGE	Returns the average of selected database entries
DCOUNT	Counts the cells that contain numbers in a database
DCOUNTA	Counts nonblank cells in a database
DGET	Extracts from a database a single record that matches the specified criteria
DMAX	Returns the maximum value from selected database entries
DMIN	Returns the minimum value from selected database entries
DPRODUCT	Multiplies the values in a particular field of records that match the criteria in a database
DSTDEV	Estimates the standard deviation based on a sample of selected database entries
DSTDEVP	Calculates the standard deviation based on the entire population of selected database entries
DSUM	Adds the numbers in the field column of records in the database that match the criteria
DVAR	Estimates variance based on a sample from selected database entries
DVARP	Calculates variance based on the entire population of selected database entries

4.2 Date and Time Functions

Function	Description
DATE	Returns the serial number of a particular date
DATEVALUE	Converts a date in the form of text to a serial number
DAY	Converts a serial number to a day of the month
DAYS360	Calculates the number of days between two dates based on a 360-day year
EDATE	Returns the serial number of the date that is the indicated number of months before or after the start date
EOMONTH	Returns the serial number of the last day of the month before or after a specified number of months
HOUR	Converts a serial number to an hour
MINUTE	Converts a serial number to a minute
MONTH	Converts a serial number to a month
NETWORKDAYS	Returns the number of whole workdays between two dates

NOW	Returns the serial number of the current date and time
SECOND	Converts a serial number to a second
TIME	Returns the serial number of a particular time
TIMEVALUE	Converts a time in the form of text to a serial number
TODAY	Returns the serial number of today's date
WEEKDAY	Converts a serial number to a day of the week
WEEKNUM	Converts a serial number to a number representing where the week falls numerically with a year
WORKDAY	Returns the serial number of the date before or after a specified number of workdays
YEAR	Converts a serial number to a year
YEARFRAC	Returns the year fraction representing the number of whole days between start_date and end_date

4.3 Engineering Functions

Function	Description
BESSELI	Returns the modified Bessel function In(x)
BESSELJ	Returns the Bessel function Jn(x)
BESSELK	Returns the modified Bessel function Kn(x)
BESSELY	Returns the Bessel function Yn(x)
BIN2DEC	Converts a binary number to decimal
BIN2HEX	Converts a binary number to hexadecimal
BIN2OCT	Converts a binary number to octal
COMPLEX	Converts real and imaginary coefficients into a complex number
CONVERT	Converts a number from one measurement system to another
DEC2BIN	Converts a decimal number to binary
DEC2HEX	Converts a decimal number to hexadecimal
DEC2OCT	Converts a decimal number to octal
DELTA	Tests whether two values are equal
ERF	Returns the error function
ERFC	Returns the complementary error function
GESTEP	Tests whether a number is greater than a threshold value
HEX2BIN	Converts a hexadecimal number to binary
HEX2DEC	Converts a hexadecimal number to decimal
HEX2OCT	Converts a hexadecimal number to octal
IMABS	Returns the absolute value (modulus) of a complex number

IMAGINARY	Returns the imaginary coefficient of a complex number
IMARGUMENT	Returns the argument theta, an angle expressed in radians
IMCONJUGATE	Returns the complex conjugate of a complex number
IMCOS	Returns the cosine of a complex number
IMDIV	Returns the quotient of two complex numbers
IMEXP	Returns the exponential of a complex number
IMLN	Returns the natural logarithm of a complex number
IMLOG10	Returns the base-10 logarithm of a complex number
IMLOG2	Returns the base-2 logarithm of a complex number
IMPOWER	Returns a complex number raised to an integer power
IMPRODUCT	Returns the product of from 2 to 29 complex numbers
IMREAL	Returns the real coefficient of a complex number
IMSIN	Returns the sine of a complex number
IMSQRT	Returns the square root of a complex number
IMSUB	Returns the difference between two complex numbers
IMSUM	Returns the sum of complex numbers
OCT2BIN	Converts an octal number to binary
OCT2DEC	Converts an octal number to decimal
OCT2HEX	Converts an octal number to hexadecimal

4.4 Financial Functions

Function	Description
ACCRINT	Returns the accrued interest for a security that pays periodic interest
ACCRINTM	Returns the accrued interest for a security that pays interest at maturity
AMORDEGRC	Returns the depreciation for each accounting period by using a depreciation coefficient
AMORLINC	Returns the depreciation for each accounting period
COUPDAYBS	Returns the number of days from the beginning of the coupon period to the settlement date
COUPDAYS	Returns the number of days in the coupon period that contains the settlement date
COUPDAYSNC	Returns the number of days from the settlement date to the next coupon date
COUPNCD	Returns the next coupon date after the settlement date
COUPNUM	Returns the number of coupons payable between the settlement date and maturity date

COUPPCD	Returns the previous coupon date before the settlement date
CUMIPMT	Returns the cumulative interest paid between two periods
CUMPRINC	Returns the cumulative principal paid on a loan between two periods
DB	Returns the depreciation of an asset for a specified period by using the fixed-declining balance method
DDB	Returns the depreciation of an asset for a specified period by using the double-declining balance method or some other method that you specify
DISC	Returns the discount rate for a security
DOLLARDE	Converts a dollar price, expressed as a fraction, into a dollar price, expressed as a decimal number
DOLLARFR	Converts a dollar price, expressed as a decimal number, into a dollar price, expressed as a fraction
DURATION	Returns the annual duration of a security with periodic interest payments
EFFECT	Returns the effective annual interest rate
FV	Returns the future value of an investment
FVSCHEDULE	Returns the future value of an initial principal after applying a series of compound interest rates
INTRATE	Returns the interest rate for a fully invested security
IPMT	Returns the interest payment for an investment for a given period
IRR	Returns the internal rate of return for a series of cash flows
ISPMT	Calculates the interest paid during a specific period of an investment
MDURATION	Returns the Macauley modified duration for a security with an assumed par value of $100
MIRR	Returns the internal rate of return where positive and negative cash flows are financed at different rates
NOMINAL	Returns the annual nominal interest rate
NPER	Returns the number of periods for an investment
NPV	Returns the net present value of an investment based on a series of periodic cash flows and a discount rate
ODDFPRICE	Returns the price per $100 face value of a security with an odd first period
ODDFYIELD	Returns the yield of a security with an odd first period
ODDLPRICE	Returns the price per $100 face value of a security with an odd last period
ODDLYIELD	Returns the yield of a security with an odd last period
PMT	Returns the periodic payment for an annuity
PPMT	Returns the payment on the principal for an investment for a given period
PRICE	Returns the price per $100 face value of a security that pays periodic interest

PRICEDISC	Returns the price per $100 face value of a discounted security
PRICEMAT	Returns the price per $100 face value of a security that pays interest at maturity
PV	Returns the present value of an investment
RATE	Returns the interest rate per period of an annuity
RECEIVED	Returns the amount received at maturity for a fully invested security
SLN	Returns the straight-line depreciation of an asset for one period
SYD	Returns the sum-of-years' digits depreciation of an asset for a specified period
TBILLEQ	Returns the bond-equivalent yield for a Treasury bill
TBILLPRICE	Returns the price per $100 face value for a Treasury bill
TBILLYIELD	Returns the yield for a Treasury bill
VDB	Returns the depreciation of an asset for a specified or partial period by using a declining balance method
XIRR	Returns the internal rate of return for a schedule of cash flows that is not necessarily periodic
XNPV	Returns the net present value for a schedule of cash flows that is not necessarily periodic
YIELD	Returns the yield on a security that pays periodic interest
YIELDDISC	Returns the annual yield for a discounted security; for example, a Treasury bill
YIELDMAT	Returns the annual yield of a security that pays interest at maturity

4.5 Information Functions

Function	Description
CELL	Returns information about the formatting, location, or contents of a cell
ERROR.TYPE	Returns a number corresponding to an error type
INFO	Returns information about the current operating environment
ISBLANK	Returns TRUE if the value is blank
ISERR	Returns TRUE if the value is any error value except #N/A
ISERROR	Returns TRUE if the value is any error value
ISEVEN	Returns TRUE if the number is even
ISLOGICAL	Returns TRUE if the value is a logical value
ISNA	Returns TRUE if the value is the #N/A error value
ISNONTEXT	Returns TRUE if the value is not text
ISNUMBER	Returns TRUE if the value is a number
ISODD	Returns TRUE if the number is odd

ISREF	Returns TRUE if the value is a reference
ISTEXT	Returns TRUE if the value is text
N	Returns a value converted to a number
NA	Returns the error value #N/A
TYPE	Returns a number indicating the data type of a value

4.6 Logical Functions

Function	Description
AND	Returns TRUE if all of its arguments are TRUE
FALSE	Returns the logical value FALSE
IF	Specifies a logical test to perform
NOT	Reverses the logic of its argument
OR	Returns TRUE if any argument is TRUE
TRUE	Returns the logical value TRUE

4.7 Lookup and Reference Functions

Function	Description
ADDRESS	Returns a reference as text to a single cell in a worksheet
AREAS	Returns the number of areas in a reference
CHOOSE	Chooses a value from a list of values
COLUMN	Returns the column number of a reference
COLUMNS	Returns the number of columns in a reference
GETPIVOTDATA	Returns data stored in a PivotTable
HLOOKUP	Looks in the top row of an array and returns the value of the indicated cell
HYPERLINK	Creates a shortcut or jump that opens a document stored on a network server, an intranet, or the Internet
INDEX	Uses an index to choose a value from a reference or array
INDIRECT	Returns a reference indicated by a text value
LOOKUP	Looks up values in a vector or array
MATCH	Looks up values in a reference or array
OFFSET	Returns a reference offset from a given reference
ROW	Returns the row number of a reference
ROWS	Returns the number of rows in a reference
RTD	Retrieves real-time data from a program that supports COM automation (Automation: A way to work with an application's objects from another application or development tool. Formerly called OLE

	Automation, Automation is an industry standard and a feature of the Component Object Model (COM).)
TRANSPOSE	Returns the transpose of an array
VLOOKUP	Looks in the first column of an array and moves across the row to return the value of a cell

4.8 Math and Trigonometry Functions

Function	Description
ABS	Returns the absolute value of a number
ACOS	Returns the arccosine of a number
ACOSH	Returns the inverse hyperbolic cosine of a number
ASIN	Returns the arcsine of a number
ASINH	Returns the inverse hyperbolic sine of a number
ATAN	Returns the arctangent of a number
ATAN2	Returns the arctangent from x- and y-coordinates
ATANH	Returns the inverse hyperbolic tangent of a number
CEILING	Rounds a number to the nearest integer or to the nearest multiple of significance
COMBIN	Returns the number of combinations for a given number of objects
COS	Returns the cosine of a number
COSH	Returns the hyperbolic cosine of a number
DEGREES	Converts radians to degrees
EVEN	Rounds a number up to the nearest even integer
EXP	Returns e raised to the power of a given number
FACT	Returns the factorial of a number
FACTDOUBLE	Returns the double factorial of a number
FLOOR	Rounds a number down, toward zero
GCD	Returns the greatest common divisor
INT	Rounds a number down to the nearest integer
LCM	Returns the least common multiple
LN	Returns the natural logarithm of a number
LOG	Returns the logarithm of a number to a specified base
LOG10	Returns the base-10 logarithm of a number
MDETERM	Returns the matrix determinant of an array
MINVERSE	Returns the matrix inverse of an array
MMULT	Returns the matrix product of two arrays

MOD	Returns the remainder from division
MROUND	Returns a number rounded to the desired multiple
MULTINOMIAL	Returns the multinomial of a set of numbers
ODD	Rounds a number up to the nearest odd integer
PI	Returns the value of pi
POWER	Returns the result of a number raised to a power
PRODUCT	Multiplies its arguments
QUOTIENT	Returns the integer portion of a division
RADIANS	Converts degrees to radians
RAND	Returns a random number between 0 and 1
RANDBETWEEN	Returns a random number between the numbers you specify
ROMAN	Converts an arabic numeral to roman, as text
ROUND	Rounds a number to a specified number of digits
ROUNDDOWN	Rounds a number down, toward zero
ROUNDUP	Rounds a number up, away from zero
SERIESSUM	Returns the sum of a power series based on the formula
SIGN	Returns the sign of a number
SIN	Returns the sine of the given angle
SINH	Returns the hyperbolic sine of a number
SQRT	Returns a positive square root
SQRTPI	Returns the square root of (number * pi)
SUBTOTAL	Returns a subtotal in a list or database
SUM	Adds its arguments
SUMIF	Adds the cells specified by a given criteria
SUMPRODUCT	Returns the sum of the products of corresponding array components
SUMSQ	Returns the sum of the squares of the arguments
SUMX2MY2	Returns the sum of the difference of squares of corresponding values in two arrays
SUMX2PY2	Returns the sum of the sum of squares of corresponding values in two arrays
SUMXMY2	Returns the sum of squares of differences of corresponding values in two arrays
TAN	Returns the tangent of a number
TANH	Returns the hyperbolic tangent of a number
TRUNC	Truncates a number to an integer

4.9 Statistical Functions

Function	Description
AVEDEV	Returns the average of the absolute deviations of data points from their mean
AVERAGE	Returns the average of its arguments
AVERAGEA	Returns the average of its arguments, including numbers, text, and logical values
BETADIST	Returns the beta cumulative distribution function
BETAINV	Returns the inverse of the cumulative distribution function for a specified beta distribution
BINOMDIST	Returns the individual term binomial distribution probability
CHIDIST	Returns the one-tailed probability of the chi-squared distribution
CHIINV	Returns the inverse of the one-tailed probability of the chi-squared distribution
CHITEST	Returns the test for independence
CONFIDENCE	Returns the confidence interval for a population mean
CORREL	Returns the correlation coefficient between two data sets
COUNT	Counts how many numbers are in the list of arguments
COUNTA	Counts how many values are in the list of arguments
COUNTBLANK	Counts the number of blank cells within a range
COUNTIF	Counts the number of nonblank cells within a range that meet the given criteria
COVAR	Returns covariance, the average of the products of paired deviations
CRITBINOM	Returns the smallest value for which the cumulative binomial distribution is less than or equal to a criterion value
DEVSQ	Returns the sum of squares of deviations
EXPONDIST	Returns the exponential distribution
FDIST	Returns the F probability distribution
FINV	Returns the inverse of the F probability distribution
FISHER	Returns the Fisher transformation
FISHERINV	Returns the inverse of the Fisher transformation
FORECAST	Returns a value along a linear trend
FREQUENCY	Returns a frequency distribution as a vertical array
FTEST	Returns the result of an F-test
GAMMADIST	Returns the gamma distribution
GAMMAINV	Returns the inverse of the gamma cumulative distribution
GAMMALN	Returns the natural logarithm of the gamma function, $\Gamma(x)$

GEOMEAN	Returns the geometric mean
GROWTH	Returns values along an exponential trend
HARMEAN	Returns the harmonic mean
HYPGEOMDIST	Returns the hyper-geometric distribution
INTERCEPT	Returns the intercept of the linear regression line
KURT	Returns the kurtosis of a data set
LARGE	Returns the k-th largest value in a data set
LINEST	Returns the parameters of a linear trend
LOGEST	Returns the parameters of an exponential trend
LOGINV	Returns the inverse of the lognormal distribution
LOGNORMDIST	Returns the cumulative lognormal distribution
MAX	Returns the maximum value in a list of arguments
MAXA	Returns the maximum value in a list of arguments, including numbers, text, and logical values
MEDIAN	Returns the median of the given numbers
MIN	Returns the minimum value in a list of arguments
MINA	Returns the smallest value in a list of arguments, including numbers, text, and logical values
MODE	Returns the most common value in a data set
NEGBINOMDIST	Returns the negative binomial distribution
NORMDIST	Returns the normal cumulative distribution
NORMINV	Returns the inverse of the normal cumulative distribution
NORMSDIST	Returns the standard normal cumulative distribution
NORMSINV	Returns the inverse of the standard normal cumulative distribution
PEARSON	Returns the Pearson product moment correlation coefficient
PERCENTILE	Returns the k-th percentile of values in a range
PERCENTRANK	Returns the percentage rank of a value in a data set
PERMUT	Returns the number of permutations for a given number of objects
POISSON	Returns the Poisson distribution
PROB	Returns the probability that values in a range are between two limits
QUARTILE	Returns the quartile of a data set
RANK	Returns the rank of a number in a list of numbers
RSQ	Returns the square of the Pearson product moment correlation coefficient
SKEW	Returns the skewness of a distribution
SLOPE	Returns the slope of the linear regression line
SMALL	Returns the k-th smallest value in a data set

STANDARDIZE	Returns a normalized value
STDEV	Estimates standard deviation based on a sample
STDEVA	Estimates standard deviation based on a sample, including numbers, text, and logical values
STDEVP	Calculates standard deviation based on the entire population
STDEVPA	Calculates standard deviation based on the entire population, including numbers, text, and logical values
STEYX	Returns the standard error of the predicted y-value for each x in the regression
TDIST	Returns the Student's t-distribution
TINV	Returns the inverse of the Student's t-distribution
TREND	Returns values along a linear trend
TRIMMEAN	Returns the mean of the interior of a data set
TTEST	Returns the probability associated with a Student's t-test
VAR	Estimates variance based on a sample
VARA	Estimates variance based on a sample, including numbers, text, and logical values
VARP	Calculates variance based on the entire population
VARPA	Calculates variance based on the entire population, including numbers, text, and logical values
WEIBULL	Returns the Weibull distribution
ZTEST	Returns the one-tailed probability-value of a z-test

4.10 Text Functions

Function	Description
ASC	Changes full-width (double-byte) English letters or katakana within a character string to half-width (single-byte) characters
BAHTTEXT	Converts a number to text, using the ß (baht) currency format
CHAR	Returns the character specified by the code number
CLEAN	Removes all nonprintable characters from text
CODE	Returns a numeric code for the first character in a text string
CONCATENATE	Joins several text items into one text item
DOLLAR	Converts a number to text, using the $ (dollar) currency format
EXACT	Checks to see if two text values are identical
FIND, FINDB	Finds one text value within another (case-sensitive)
FIXED	Formats a number as text with a fixed number of decimals
JIS	Changes half-width (single-byte) English letters or katakana within a

	character string to full-width (double-byte) characters
LEFT, LEFTB	Returns the leftmost characters from a text value
LEN, LENB	Returns the number of characters in a text string
LOWER	Converts text to lowercase
MID, MIDB	Returns a specific number of characters from a text string starting at the position you specify
PHONETIC	Extracts the phonetic (furigana) characters from a text string
PROPER	Capitalizes the first letter in each word of a text value
REPLACE, REPLACEB	Replaces characters within text
REPT	Repeats text a given number of times
RIGHT, RIGHTB	Returns the rightmost characters from a text value
SEARCH, SEARCHB	Finds one text value within another (not case-sensitive)
SUBSTITUTE	Substitutes new text for old text in a text string
T	Converts its arguments to text
TEXT	Formats a number and converts it to text
TRIM	Removes spaces from text
UPPER	Converts text to uppercase
VALUE	Converts a text argument to a number

4.11 External Functions

Function	Description
EUROCONVERT	Converts a number to Euros, converts a number from Euros to a euro member currency, or converts a number from one euro member currency to another by using the euro as an intermediary (triangulation)
SQL.REQUEST	Connects with an external data source and runs a query from a worksheet, then returns the result as an array without the need for macro programming