

## Microsoft Excel: Nested Functions

Although there are very many functions built into Excel it is not unusual to find the no single function will complete your required calculation, so it becomes necessary to combine functions to achieve the desired result. When one function is placed inside another it is called a *nested* function.

### Nesting Calculations

Consider this example of calculating a person's age from their birth date (*Fig. 1*). The calculation subtracts the birth date from the current date (supplied by Excel's TODAY function) which gives the difference between the two dates in days, then divides the resulting value by the average number of days in a year (365.25 taking account of leap years). However the result provides a value with several decimal places.

	A	B	C	D	E
1	Birth Date:	27/09/1950			
2	Age:	59.04996578			
3					
4					

*Fig. 1* The basic calculation produces several decimal places.

We normally think about age as a whole number of years so this value needs to be rounded down to the nearest whole number. This can be achieved with either the ROUND function (rounds to a specific number of decimal places) or the INT function (rounds down to a whole number). This additional part of the calculation could be placed in another cell (*Fig. 2*) or the calculations could be nested into a single calculation in the same cell (*Fig. 3*).

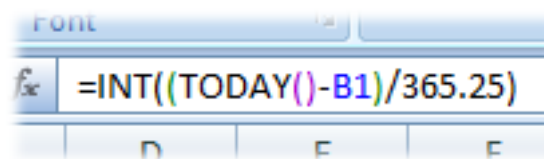
	A	B	C	D
1	Birth Date:	27/09/1950		
2	Age:	59.04996578		
3		59		
4				

*Fig. 2* The INT function gives a whole number.

	A	B	C	D	E	F
1	Birth Date:	27/09/1950				
2	Age:	59				
3						
4						

*Fig. 3* The functions are nested in a single calculation.

When placing one function inside another it is not necessary to repeat the equals sign (=). Nested calculations often include several pairs of brackets. Excel helps you make sure you have the correct number of opening and closing brackets by colour-coding the pairs of brackets as you type (*Fig. 4*).



*Fig. 4* Excel colour codes nested brackets.

### Nested IF Functions

The IF function allows you to specify two outcomes to a calculation. It is in three parts: a *Logical Test* (an expression that will evaluate to either TRUE or FALSE); a *Value if true* (an expression that determines what to return if the test evaluates to True; and a *Value if false* (an expression that determines what to return if the test evaluates to False).

Suppose your mail order company applies a shipping charge to orders under a certain amount. Above this amount shipping is free. This can be determined using a simple IF function. In this example (*Fig. 5*) a shipping charge of £5 is applied to orders under £30 using **=IF(B1<30,5,0)** where the total value of the goods is in cell B1. The logical test is **B1<30** so the function returns 5 if the value in B1 is less than 30 (i.e. the test is TRUE) or zero if the value in B1 is 30 or more (i.e. the test is FALSE).

	A	B	C	D	E
1	SubTotal:	£ 27.85			
2	Shipping:	£ 5.00			
3	Total:	£ 32.85			
4					

Fig. 5 A simple IF function with two conditions.

NOTE: If you want the cell to remain empty instead of returning a zero type a pair of quotes instead of the zero: **=IF(B1<30,5,"")**. Alternatively, if you omit this argument completely (e.g. **=IF(B1<30,5)**) the function will return the value FALSE.

Used simply like this the IF function can deal with just two conditions. You can apply more conditions by *nesting* if functions. Suppose you have a sliding scale of shipping charges. Orders below £30 are charged £5 shipping, orders from £30 to £49.99 are charged £2.50 shipping, and for orders of £50 and above shipping is free.

The first part of the IF function (if the order is less than £30) remains the same, but the second part needs to change. If the order is £30 or more there is another condition – is the order less than £50? So you need to add another IF function (Fig. 6):

	A	B	C	D	E	F
1	SubTotal:	£ 38.45				
2	Shipping:	£ 2.50				
3	Total:	£ 40.95				
4						

Fig. 6 A second IF provides an additional condition.

The second part of the original IF function has been replaced by a second IF function, thus adding an additional condition. You can continue adding conditions like this until Excel's limit for nested functions is reached (Excel 2007 permits up to 64 levels of nesting).

## VLOOKUP Is An Alternative

When you want to apply a large number of conditions, using nested IFs can become complex. An alternative is to use the VLOOKUP function which gets a value by referring to a reference table as in this example (Fig. 7) where there are four different charge bands for shipping:

	A	B	C	D	E	F
1	SubTotal:	£ 83.25			Goods	Shipping
2	Shipping:	£ 2.50			0	10
3	Total:	£ 85.75			25	5
4					50	2.5
5					100	0
6						

Fig. 7 A VLOOKUP can be used if there are many conditions.

The VLOOKUP function **=VLOOKUP(B1,E1:F5,2,TRUE)** looks at the value in cell **B1** then tries to match it to a value in the first column of the table located in the cells **E1:F5**. The word **TRUE** at the end of the function tells it that if it can't find an exact match an approximate match is OK (the values in the first column of the table must be sorted in *ascending* order for this to work!). It then returns the corresponding value from the specified column (column **2** in this case). Using VLOOKUP and a reference table means that the function is simpler to write and read, and also that the shipping rates can be easily changed without rewriting the function. Nested IFs would look like this:

**=IF(B1<25,10,IF(B1<50,5,IF(B1<100,2.5,0)))**