

EXCEL <<<<<<<

By Bill Jelen

Mastering Conditional Formulas in Excel

You've probably used the IF function in Excel to perform simple conditional calculations. You can dramatically expand the power of the IF function by using functions such as AND, OR, and NOT. This month's article will take a look at these functions.

Say that your VP of sales proposes a sales promotion for the month of February. The sales rep will be paid a 2% bonus for any sales over \$100,000. The IF function is perfect for calculating this bonus. The syntax of the function is:

```
=IF(Logical Test, Value if True, Value if False)
```

Using the worksheet in Figure 1, the bonus calculation would be:

```
=IF(D2>=100000,0.02*D2,0)
```

The formula (cell G2) says, "If the revenue in cell D2 is greater than or equal to 100,000, then calculate the bonus as 2% of the revenue in D2; otherwise, the bonus is 0."

After reviewing the bonus results, the CFO suggests that the bonus shouldn't apply to any sales sold at a deep discount. The new rule for the bonus program is that revenue must be \$100,000 or more *and* the GP% has to be over 50%.

While the Logical Test portion of the IF function can't handle multiple conditions, you can overcome this limitation by using the AND function. The AND function can handle 1 to 30 logical tests separated by commas. The AND function will return a TRUE value only if *all* of the logical tests evaluate to TRUE.

The AND function to test for both conditions in the bonus plan would be:

```
=AND(D2>100000,F2>=0.5)
```

Remove the leading equals sign and use the AND function as the Logical Test argument in the IF function:

```
=IF(AND(D2>=100000,F2>0.05),0.02*D2,0)
```

While the AND function tests to see if all of the logical tests are TRUE, the OR function will test to see if *any* of the logical tests are TRUE. Perhaps the bonus plan seeks to reward any sales in the new Retail market. The original plan of paying the bonus for sales of \$100,000 is modified to pay a bonus on any records where the Market field is "Retail."

In this case, the OR function to test for the conditions would be:

```
=OR(D2>=100000,C2="Retail")
```

Again, remove the leading equals sign

from the OR function and insert it into the first argument of the IF function:

```
=IF(OR(D2>=100000,C2="Retail"),0.02*D2,0)
```

The NOT function can be used to change TRUE to FALSE and change FALSE to TRUE. Sometimes it's easier to test for a few exceptions and then reverse that answer using the NOT function. If you didn't want to pay a bonus to reps R99 or R01, you could use:

```
=IF(AND(D2>=100000,NOT(OR(B2="R99",B2="R01"))),0.02*D2,0)
```

Bonus plans in real life tend to have multiple conditions and exceptions. You can stack multiple AND, OR, and NOT functions, using parentheses to control the calculation order. You can imagine how a bonus plan can change over time to read this way: Pay the bonus for sales over \$100,000 with GP% over 50%, plus pay a bonus for any sales in Retail, but never pay the bonus for sales rep R99. While the formula gets long and hard to read, it will model all of that logic:

```
=IF(AND(B2<>"R99",OR(AND(D2>=100000,F2>0.5),C2="Retail")),0.02*D2,0)
```

Figure 1

=IF(OR(D2>=100000,C2="Retail"),0.02*D2,0)							
	A	B	C	D	E	F	G
1	Invoice	Rep	Mkt	Revenue	Cost	GP%	Bonus
2	43666	R14	FS	187,094	77,831	58.4%	3,742
3	43667	R32	Retail	62,646	39,718	36.6%	1,253
4	43668	R99	Mfg	140,063	119,434	14.7%	2,801
5	43669	R34	FS	78,803	32,940	58.2%	0

Handling Tiered Bonus Plans with Nested IF Statements

Another common situation is differing bonus rates for differing sales levels. While it might be easier to use the range version of VLOOKUP, you can handle a plan with a few levels using nested IF statements. Say that the bonus plan offers a 3% bonus for sales above \$150,000, a 2% bonus for sales above \$100,000, and a 1% bonus for sales above \$75,000. Be sure to start testing for the largest value first, and add a new IF function in the Value if False argument.

```
=IF(D2>150000,0.03*D2,IF
(D2>100000,0.02*D2,IF
(D2>75000,0.01*D2,0)))
```

Be careful when you start nesting IF statements. If you're using Excel 2003 or earlier versions, you can nest only seven IF statements in one formula. The new limit in Excel 2007 is 32 nested IF statements.

Performing Conditional Calculation on Many Records Using SUMIF

While the IF function is good for analyzing a few cells, sometimes you need to perform a conditional calculation on all of the records in a data set. The SUMIF and COUNTIF functions make this easy. Say that you want to look at all the data in your table and add up the revenue in column D for all records where the market in column C is "Retail."

The syntax for SUMIF is:

```
=SUMIF(Range to Check, Criteria,
Range to Sum)
```

In this case, the range to check is the markets in C2:C32. The criteria is the word "Retail" (in quotes), and the range to sum is the revenue in D2:D32. Note that the range to sum should be the same size and shape as the range to check. Use this formula:

```
=SUMIF(C2:C32,"Retail",D2:D32)
```

Unfortunately, you can't use the AND or OR functions in the Range to Check argument of SUMIF. Thus, SUMIF is limit-

ed to testing for one condition. You could build a new column in your dataset that used functions such as AND or OR to test for multiple conditions and then use that new column as the Range to Check in the SUMIF function.

Note that if you've upgraded to Excel 2007, a new plural version of SUMIF is available. The SUMIFS function can handle up to 32 different conditions. Be careful when using SUMIFS; you must ensure that everyone who opens the workbook has the new version of Excel. Otherwise, the SUMIFS cells will change to #NAME? errors.

By using various combinations of NOT, AND, and OR functions, you can dramatically increase the power of Excel's IF function. **SF**

Bill Jelen will bring his Power Excel seminar to IMA's 90th Annual Conference and Exhibition. Send questions for future articles to IMA@MrExcel.com.

Excel is a topic at IMA's Annual Conference, June 6-10, 2009, in Denver, Colo. For information, visit www.imaconference.org.
