

Excel performance analysis the professional way

Analysing performance data in Excel using array formula and dynamic range name methods.

The recording and analysis of performance data is the quintessential spreadsheet application. In every company, probably in every department, performance data is collected in Excel. Somewhere there are coneheads and bean counters who then work their magic to analyse these numbers. This article will provide an introduction for us mortals on exactly how they do it. The good news is that we can do it without any VBA or macros at all.

By 'performance data' I mean whatever numbers you use to show your company or department doing business. Usually, this will take the form of transaction data like this:

	A	B	C	D	E	F	G
1	Date	Customer	Item	No sold	Item price	Total cost	Tax @ 10%
2	01-02-03	Smith	Box	1	£ 20.00	£ 20.00	£ 2.00
3	01-03-03	Jones	Box	2	£ 20.00	£ 40.00	£ 4.00
4	18-09-03	Smith	Bag	4	£ 10.00	£ 40.00	£ 4.00
5	18-09-03	White	Box	8	£ 20.00	£ 160.00	£ 16.00
6	04-10-03	Jones	Bag	16	£ 11.00	£ 176.00	£ 17.60
7	04-10-03	Smith	Box	32	£ 22.00	£ 704.00	£ 70.40
8							
9							
10							

Ultimately, there will probably be several entries a day to this table, but we'll use this simplified model for now. Type in the table on a new Excel sheet, exactly as shown

We're going to start by naming some ranges. This will make it a lot easier to work out what's happening in the various formulae later on, when things get complex. It will also mean we don't have to keep tinkering with the formulae every time we add new data to the table. To start with, we'll define some simple static ranges. Just hit CTRL+F3, and define the following ranges:

Dates = A2:A1000
Customer = B2:B1000
Item = C2:C1000
NoSold = D2:D1000
ItemPrice = E2:E1000
TotalCost = F2:F1000
Tax = G2:G1000

Let's assume all the totals are based on column D, the number sold.

I would use either a D-function or an array function to work out weekly and running totals. Here, I'll use arrays for simplicity. Put all these formulae on a different sheet, or over to the right, past column G at any rate.

The total amount sold to date to a customer called 'Smith' would read:

```
=SUMIF(Customer,"Smith",NoSold)
```

No arrays yet, but that's a simple one. SUMIF and COUNTIF will work really well on a single condition, but cannot handle multiple conditions.

Let's narrow it down to the number of items sold to Smith in the last seven days:

```
{=SUM(IF(Customer="Smith",IF(Date>=(TODAY()-7),NoSold,0),0))}
```

Note that the curly brackets can't be typed in. They denote that you're making this an array function. You get them by typing the rest of the formula, then rather than just pressing ENTER, you hold SHIFT and CTRL down together while pressing ENTER.

Once you get the hang of this, you can add multiple conditions to have lots of fun! For instance, want to know how boxes you sold in the first 10 days of any month so far? Easy:

```
{=SUM(IF(DAY(Date)<=10,IF(Item="Box",NoSold,0),0))}
```

Or what if you wanted to know how much Smith had spent with you just on boxes this year?

```
{=SUM(IF(Customer="Smith",IF(Item="Box",IF(YEAR(Date)=YEAR(TODAY()),TotalCost,0),0),0))}
```

Now then. Remember earlier when we defined the ranges? We set them to look at the first 999 lines only. You might well go beyond this, particularly if you just keep adding data all year.

What we'll do is go back and change the range definitions so we have dynamic ranges, ones that will automatically stretch to cater for however many lines of data we add as we go.

CTRL+F3 will get you back to the range names dialog. You need to change the definition for the range called Date to the following:

```
=OFFSET(Sheet1!$A$2,0,0,COUNTA(Sheet1!$A$2:$A$60000),1)
```

So long as you don't go over 60,000 lines of data, this will keep the range properly defined for you. Now do the same thing for the other named ranges; all you have to change are the column letters each time.

The analysis of performance data is very much what Excel was designed to do. A little persistence with these two tools of array formulae and dynamic range names will enable any Excel user to conduct sophisticated and powerful analysis with ease.