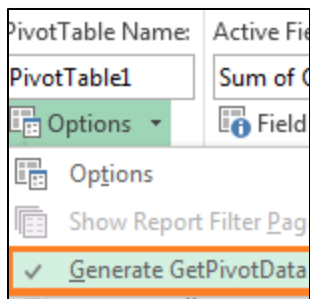# Excel GETPIVOTDATA Function

In its simplest form Excel's GETPIVOTDATA function enables you to extract values from a PivotTable report, but if you're like me when you first came across the GETPIVOTDATA function you were less than pleased with the results. Understandably so, because in its default form it's quite inflexible.

However, the benefit in using GETPIVOTDATA, as opposed to a regular cell reference, is huge in terms of reducing your ongoing workload in maintaining your reports.

Why? Well, PivotTable reports are typically being updated and, or changing shape as you add new data to the source and alter filters etc. If you link to a cell in a PivotTable with a regular cell reference and then the location of that data changes due to a refresh, or a filter or Slicer being applied, then all of a sudden your formula is returning the wrong information.

Not so with GETPIVOTDATA. If the location of your data moves then GETPIVOTDATA will still return it, assuming it's still visible somewhere in your PivotTable report.

So, my objective with this tutorial is to convert you from a GETPIVOTDATA hater (harsh, but it rhymes ;-)) to a GETPIVOTDATA lover! I know; I've got my work cut out for me.

The trick with leveraging GETPIVOTDATA power is to replace the hard keyed arguments with nested formulas so the GETPIVOTDATA formula becomes dynamic. Sounds complicated but it's not.

As an example, the 'hard keyed' arguments are those in red below:

=GETPIVOTDATA("*Order Amount*",A4,"*Order Date*",*1*,"*Years*",*2009*)
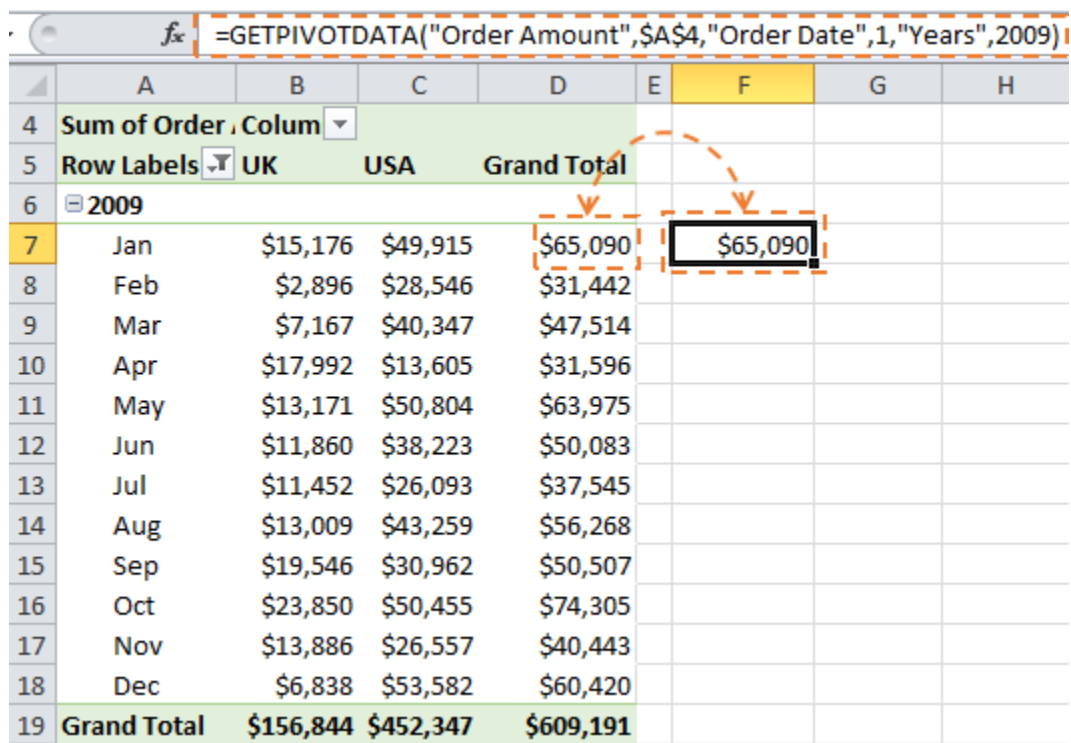
Turn GETPIVOTDATA On

First, in order to have Excel generate the GETPIVOTDATA formulas you must have the preference turned on. If you've turned it off in frustration then you can turn it back on in the PivotTable Options tab of the Ribbon:
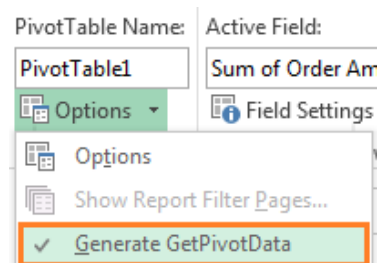
Let's look at a simple scenario. You can see my PivotTable below. When I enter an equals sign in cell F7 and then click on cell D7 in the PivotTable, Excel automatically enters this formula:

=GETPIVOTDATA("Order Amount",A4,"Order Date",1,"Years",2009)

In English the GETPIVOTDATA formula reads, return the Order Amount for the Order Date (month) 1 (which is January) for the Year 2009.

Seems like a big formula for what would usually just be =D7, but remember there are big advantages to using GETPIVOTDATA.

Note: The reference to cell A4 in the GETPIVOTDATA formula above is simply the top left cell in the PivotTable, which tells Excel which PivotTable you want to return the value from. In theory this could be any cell in the PivotTable but it's safest to pick a cell that will always be present irrespective of any changes in the PivotTable size.

GETPIVOTDATA Relative References

The annoying thing with GETPIVOTDATA is that when you copy and paste the formula, for example down column F, the references aren't relative. That is they don't update like normal cell references to pick up the next cell in the range. So you end up with a column of identical values like these in column F:

	A	B	C	D	E	F
4	**Sum of Order Amount**	Column Labels ▾				
5	**Row Labels**	ⷦUK	USA	**Grand Total**		
6	⊟2009					
7	Jan	$15,176	$49,915	$65,090		$65,090
8	Feb	$2,896	$28,546	$31,442		$65,090
9	Mar	$7,167	$40,347	$47,514		$65,090
10	Apr	$17,992	$13,605	$31,596		$65,090
11	May	$13,171	$50,804	$63,975		$65,090
12	Jun	$11,860	$38,223	$50,083		$65,090
13	Jul	$11,452	$26,093	$37,545		$65,090
14	Aug	$13,009	$43,259	$56,268		$65,090
15	Sep	$19,546	$30,962	$50,507		$65,090
16	Oct	$23,850	$50,455	$74,305		$65,090
17	Nov	$13,886	$26,557	$40,443		$65,090
18	Dec	$6,838	$53,582	$60,420		$65,090
19	**Grand Total**	$156,844	$452,347	$609,191		
20						

Right about now you're annoyance levels are rising and if you're working on a deadline you're likely to revert to typing in the cell references because you haven't got time to figure out how to STOP IT!

Making GETPIVOTDATA Dynamic

Let's start by making the formula update so that it picks up each month when copied down the column. The trick here is knowing how PivotTables represent dates; while those months appear to be the month names Jan, Feb, Mar…, for the purpose of the GETPIVOTDATA function, those months are actually numbers 1,2,3… through 12.

So we need to replace the month number argument in the GETPIVOTDATA formula with something that will count up automatically as we copy the formula down the column. For this we can use the ROW function.

ROW Function

The ROW function simply returns the row number of the reference. So, the formula ROW(A1) will return a 1, ROW(A2) will return a 2 and so on.

And we can put ROW in our formula in cell F7 like this:

=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A1),"Years",2009)

Which evaluates to this:

=GETPIVOTDATA("Order Amount",A4,"Order Date",1,"Years",2009)

So now when you copy the formula in cell F7 down the column the relative reference to cell A1 in the ROW function dynamically counts up by 1. I've written out the formulas from column F in column G in the image below:

D	F	G
Grand Total		
$65,090	$65,090	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A1),"Years",2009)
$31,442	$31,442	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A2),"Years",2009)
$47,514	$47,514	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A3),"Years",2009)
$31,596	$31,596	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A4),"Years",2009)
$63,975	$63,975	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A5),"Years",2009)
$50,083	$50,083	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A6),"Years",2009)
$37,545	$37,545	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A7),"Years",2009)
$56,268	$56,268	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A8),"Years",2009)
$50,507	$50,507	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A9),"Years",2009)
$74,305	$74,305	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A10),"Years",2009)
$40,443	$40,443	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A11),"Years",2009)
$60,420	$60,420	=GETPIVOTDATA("Order Amount",A4,"Order Date",ROW(A12),"Years",2009)
$609,191		

Now you might be thinking that's a fair bit of work just to make GETPIVOTDATA dynamic, when you could have just as easily entered =D7 and copied that down the column but remember, **the benefits of GETPIVOTDATA mean a more robust formula able to withstand significant changes to your PivotTable without losing its place. The same can't be said for =D7. In this case you get what you pay for.**

Tips:

- If you wanted to make the Year argument dynamic you could use ROW(A2009)….

- Or if your report had the years and months going across the columns you could use ROW's cousin function, COLUMN instead:
  The COLUMN function returns the column number of the cell reference, so COLUMN(A1) will return a 1, COLUMN(B1) will return a 2 and so on.

- The ROW/COLUMN function can be used in other formulas to create a dynamic reference too.

GETPIVOTDATA with Data Validation

What if we wanted to get the grand total for each country and toggle between the two using a Data Validation List like this:

Our formula for the UK grand total would look like this:

=GETPIVOTDATA("Order Amount",A4,"Country","UK")

So all we need to do is link the country name argument to cell F3 which contains the Data Validation list like this:

=GETPIVOTDATA("Order Amount",A4,"Country",F3)

Now when you choose a different country from the data validation list the GETPIVOTDATA formula dynamically updates.

Tip: *Notice how there aren't any month or year arguments in the formula above? That's because we're picking up the grand total.*

GETPIVOTDATA Tips

- I usually let Excel write my GETPIVOTDATA formulas by typing an = in the cell I want the value returned to, and then click on the cell in the PivotTable. This gives me a pre-written formula and all I need to do is edit it to make it dynamic. It's a lot quicker than writing it out from scratch.

- You can make any of the arguments dynamic, not just the examples I've show you here.

- You can only use GETPIVOTDATA to pick up values that are visible in the PivotTable report. It can't query the source data itself.

- You can use it to return values from Calculated Columns.

- Just like any other function, you can nest it in formulas or apply math/logic calculations on the result. e.g. you can add 10% to the value returned by =GETPIVOTDATA("Order Amount",A4,"Country",F3)*1.1

- If the value described by the arguments isn't present in the PivotTable it will return the #REF! error.