Data Manipulation and Analysis (Combined PC/Mac): Improved Queries with Database Functions

Welcome to our next lesson in this module on data manipulation and analysis. We're going to go through Database Functions here, building on what I described in the previous lesson where, to go up to Pivot Tables and Power Pivot, we need to first demonstrate some alternate methods of querying and aggregating data in Excel. So, here in the Summary tab, I've added this area to demonstrate how Database Functions work to summarize sales and orders. Database functions in Excel fix some, but not all, of the problems that you saw with SUMIFS and SUMPRODUCT.

With those, for example, you saw how in the previous lesson, sometimes the formulas can become very long and difficult to read. You get weird problems with dates, where sometimes the year and month functions, for example, will work with date columns, but then other times, as in the examples with SUMIFS, you actually have to construct your own dates using the DATE and EOMONTH functions.

[00:59]

If you have really complex criteria, such as dates being in between a certain range and then a certain Sales Rep ID being true but not a certain industry, yes, you could enter it with these types of functions. But it gets very clunky, and the formulas become even more difficult to read.

Database functions do require some more time and effort to set up because you need to create a separate area of the sheet to support them, as I've already done right here. And when you enter them, you need to get the syntax exactly right, you need to get the order of the fields correct, and so there is a bit of a learning curve.

If you have some type of small syntax error, it's going to break everything and Excel is not going to tell you what you are doing wrong. The data must be in tabular format, ideally in a Data Table, so these functions are less flexible than ones like INDEX and MATCH, which you can use for data retrieval anywhere even if the data is out of order, there are blank rows, blank columns, it's not exactly set up correctly. Also, with Database Functions, despite the misleading name, you still can't join together Tables based on relationships.

[02:05]

So, here, for example, in the Orders table, I still have the Industry and the Region listed even though these are redundant fields with redundant data because even if we've set up all the

Data Tables, and we have relationships established, these Database Functions, which have existed in Excel for a very long time, will not actually recognize those relationships.

So, if we were to enter something else here, let's say that we wanted to sum up everything for sales reps that were hired on or after a certain date, we could not really do that. To set that up, we'd have to modify the Orders table and then look up the Sales Rep Hire Date, and then add a separate column for that. So, we run into some of the same limitations for cases like that.

[02:53]

In general, Database Functions are most useful for setting up queries with complex criteria, such as multiple AND and OR conditions, usually ones that involve dates, amounts, regions, and industries. But if you have something simple, like a simple date check or just two criteria like we saw with some of the SUMPRODUCT and SUMIFS examples, Database Functions are usually overkill.

Let's go to Part 1 out and look at some simple Database Functions and queries; in Part 2, I will show you how to use multiple rows with Database Functions for criteria input, and then in Part 3, you will get an exercise where you have to enter your own AND/OR conditions into the Database Functions and sum up Orders and Commissions that meet a certain set of criteria.

So, the most common Database Functions are as follows: DSUM, DCOUNT, DCOUNTA, and DGET. And there are many others: DAVERAGE, DPRODUCT, DMAX, DMIN, and so on. There are even statistical functions for standard deviation and variance. But most of them are not going to be that useful for customer order analysis.

[03:57]

It's easiest to illustrate how all these work with a simple example for DSUM because everything else follows pretty much the same format. With the DSUM function, when we type in DSUM, we have to enter a database, a field, and then criteria. Let's go over and just demonstrate how this works, and I think that's the easiest way to learn this function.

So, you can see up here that we have some headers here, which, of course, match up to the exact headers that we have in the Orders table. Now, to do this, you could type in these names manually. So, for example, you could just type in Industry here. But the much smarter option is, especially if you have a Data Table, go over and link to this directly. In that way, if you do it like this, if the table ever changes, the order changes, the name changes, something like that, as long as you've linked to it directly, it will update correctly.

[04:50]

If you don't have a Data Table, that's okay, then you can just link to it and then anchor the cells so you have an absolute reference so that even if something changes, wherever you have this data, it'll be reflected right here as well. We don't recommend entering these names in manually because it's very error-prone. We also don't recommend entering the column numbers instead. So, here, for example, we could enter column 3 for industry instead of the actual name. We don't recommend it because it's much harder to read and understand like that.

So, that's some of the basic setup that you need here. **Notice how also these columns are in the same order as what we have here. So, Industry is first, and then we have Region because Region is after Industry in the table, and then we have Amount, and then we have Order Date, and then we have Sales Rep ID. So, you want to maintain that order to avoid problems. It's perfectly fine to have the same column listed twice, as we're doing here for the Order Date. It's just that you have to be conscious of the order and make sure that Industry is before State, which is before Region, which is before Amount, and so on and so forth.**

[05:53]

In any case, let's enter the DSUM function now so you can see how this works. For the database, we need to select the entire table, including the headers at the top. Yes, you need to include the header row or the Database Functions will not work correctly. So, let's select this whole thing. You can see the structured reference notation Excel puts all in brackets and puts the pound sign before it. Then for the field, you need to go up and select the field that you have right here. And again, if you don't have this in Data Table format, that's okay. But just make sure you anchor the cells and use an absolute reference so it doesn't shift around, and so it updates. You don't want to enter this type of thing manually, or type in the text here or something like that.

So, I've entered this. Notice how Excel automatically puts a comma between the headers and the amount to denote that this is a specific column within this header's row. It's a little bit confusing, but it's not actually a separate argument or a separate input here. That's just how Excel sets it up when you have these structured references.

[06:52]

And then for the criteria, we need to select the row that has all the field names, and then the row or rows right beneath it that we want to check conditions on, or the rows that contain the specific conditions we want to check. And I'll anchor everything there.

So, right now, as you can see, since we haven't entered any conditions, this simply sums up everything here. So, if we sum up the total amount, it comes out to around $408 million. And that's exactly what our DSUM function is doing.

Now, if you want this to actually work correctly, then we could start entering a criteria here. So, for the Region, remember, we have our region's list over here, let's start by entering Northeast. And now we can see the Order Total is a whole lot lower. The normal operators, greater than, less than, greater than or equal to, less than or equal to, equal to, not equal to, all work here as well. So, if we want the Order Dates to all be, say, greater than 2020, we can say >=1/1/2020.

[07:58]

With these dates, I will warn you in advance that depending on how your Windows and Excel are set up, if your default date format is different, so if you enter the year first, and the month, then the day, or you use dashes in between instead of slashes or something like that, or you enter day, month, year instead, you're going to have to enter it in a format that is accepted based on your system. This is a system-level setting, and it's not just something you can modify in Excel. So, just be aware of that.

Now, if we want to add another condition here, we can say that Sales Rep ID should be equal to 3. And you can see here how the Order Total keeps going down as we keep doing this. With DSUM, when you have just one row, everything is joined by an AND. So, we are only adding up entries where the Region is Northeast, and the Order Date is on or after January 1st, 2020, and the Sales Rep ID is equal to 3.

[08:50]

We could keep doing this as well. So, for example, let's add an industry here. And let's say, for example, that we want to use Financials for the industry. And now, you can see how the Order Total keeps going down. If we want to extend this, we can also count everything. So, let's just take this formula and copy it down. And I'm going to change this from DSUM to DCOUNT. And then if we want to add up the commissions here, let's take the same formula, copy it down, and here instead of amount, I will just enter Commissions. And so we have that.

So, these are the basics of how Database Functions work. Now there are a few finer points here that I want to go over because we tend to get a lot of confused questions whenever we try to cover this topic. **The first thing to note here is that you need to enter the entire table, including the headers. Even if you don't have this setup as a Data Table, even if it's just a normal range of cells in Excel, you need to enter the headers at the top. If you do not enter this exactly like I have it here, then it is not going to work.**

[09:57]

Let's go through this and show you a couple of examples of how you could easily mess this up and get a non-functional function. For example, here, let's say that I go in and let's say I have the bright idea of excluding the headers. So, I go in and I tried to select just the data in this table. It doesn't work. We get a value error. So, that's something you want to avoid. Let's say here that for the field, let's say that I want to select the entire column here. So, I select everything. You can see All, and then Order Date. Let me actually change that to All and then Amount instead.

So, I have the entire column here. And I think this is going to work fine. But of course, it doesn't. Excel doesn't even let me enter this properly. Even if I change this, and let's say that I only want to get the data here, Excel simply gives us a #VALUE! as the error message. So, you need to get this exactly right. And you need to enter only the name of the field that you actually want to sum up here, so the Amount in this case. If you try to enter the whole column or something else like that, it is simply not going to work and Excel is going to give you no clue as to why it is not working.

[11:09]

All the normal operators here work. I put the Spiderman principle, referring to that quote, "With great power comes great responsibility," because you have to be really careful about what you're doing if you get into complicated references here.

For example, one thing that is not going to work is if you want to do something more complex with the Order Dates. So, let's just delete the Industry for now, and let's say that we want to have multiple conditions here. And let's say that we want to make this between 2020 and 2023. This is not going to work. It doesn't work like this because Excel will only allow you to enter one specific condition in one column here. So, if you really want to do something like this, then you need to enter it right here instead, so <=12/31/2023, and then that'll work.

[12:04]

So, these are just examples of a few things that could potentially go wrong when you try to do this. I'm going to delete the Sales Rep ID for now because we don't really want that. That's pretty much all there is to know about the basic functionality of Database Functions. I have some more notes here in the side as always, if you want to look at these.

Let's go to Part 2 now and look at Multiple Rows for the Criteria Input. So, just like you can include multiple columns in the Criteria, the example with the Order Dates being the best one that lets you select a range of dates, you can also include multiple rows if you change the criteria range to include them

Now, in a single row, all the conditions are joined by an AND. So, everything in that row has to be TRUE in order for Excel to add it up or to count. But with multiple rows, each row is joined to the other row with an OR, so everything in Row 1 must be true OR everything in Row 2 must be true. So, you could have criteria like this, Industrials AND Midwest AND After January 1, 2020 OR Sales Rep 1 AND After January 1, 2020 AND Before December 31, 2023.

[13:11]

Let's just copy and paste these, so don't forget these criteria. And I'll just put them in a text file. And then let's enter them, and let's see how this works. Let's enter Industrials for the industry. By the way, we could also use Data Validation for this, and that would probably be smarter in a lot of cases. I'm skipping it to save some time here. But, for example, we could easily apply Data Validation, and Data Validation should still work within Database Functions and the setup for them like this. For the Region, we'll say Midwest, and then we'll say After January 1st, 2020, so 1/1/2020; it's not greater than or equal to because in the instructions it just said after January 1st, 2020, OR Sales Rep 1 AND After 1/1/2020, but before 12/31/2023. So, now we have an OR condition right here that is joining everything together. And so nothing happens here.

[14:15]

Of course, the problem isn't that nothing should actually happen, the problem is that we haven't expanded the criteria range. So, let's expand this by a few rows first, and then go down to row 9. And then let's do the same thing here, and then the same thing here. **And now we can see another problem, which is that if we make it too broad and we don't have anything in this third row, Excel just interprets this row as being everything in the table and having no conditions, so it sums up everything.**

So, we have to be really careful, and we have to only include the rows that actually have entries here as I'm doing right now. I'm just changing the P9 to P8 in each case and now this will get us what we want. Now, if we delete the second row, look at this, the data clearly changes and once again, we are now summing up everything because of this blank row.

[15:02]

So, it goes back to what I was saying before, "With great power comes great responsibility," and you have to be really careful with the syntax here. If you really want to make this more open ended, we could just delete these two criteria and then we get a lot more, but we're still not summing up everything exactly.

So, these are just a few things to keep in mind. The bottom line is that when you include multiple rows like this, you have to be very certain of what you're looking for. You cannot enter blank rows or Excel will interpret it as you wanting to sum up or count everything in the range, which you normally don't want to do. And you have to be really careful about what these types of AND and OR conditions mean across very large sets of data.

So, that's a little bit about how to do this and some of the things to watch out for here. Overall, I don't think it's terribly useful to be able to input complex conditions like this because you hardly ever need to find something so specific. Even if you do want to look at something this specific, normally, you want to look at it in the context of some type of larger trend or pattern, like sales by region, by industry, by year or something like that. So, we don't think that Database Functions are terribly useful in cases like this, although they may still come up from time to time.

[16:14]

So, with all that said, now let's go into your exercise, where you'll get some practice with these database functions. **What I want you to do here is sum up all the orders and commissions from the Midwest or Northeast from companies in the Industrials or Energy sectors, placed between 2021 and 2024.**

So, take these criteria and then go over here. You may have to modify this area. You may have to delete these formulas and start over again to get all the orders and commissions that match the criteria I just laid out. So, pause this video right now. Give it a shot yourself. When you're done, come back, un-pause it, and then we'll go through this together.

Okay. So, first off, I'm going to go over here and just copy and paste the criteria into a separate text file I have off-screen, so I don't forget what the actual criteria is. And now this is stored in a text file. Let's now go back here and start seeing what we have to do.

[17:08]

So, the first thing to note about this is that with a set of criteria like this, where we have Midwest or Northeast and then Industrials or Energy, basically we're going to have to create four rows here, because, remember, with database functions, within each row the criteria are joined together with an AND. But when you have two rows, those are joined together with an OR. Three rows, again it's an OR that connects them. Four rows, it's an OR that connects them. So, to get an OR with all four of these different criteria, we are going to need four rows.

I'm going to start by just deleting the existing functions here. Then I will cut this and move down one row. So, now we have one, two, three, four rows up here that we can enter. And then for the Industry, I will enter Industrials for these first two and then Energy for these next two. State will be blank. And then for Region, it's Midwest or Northeast. So, we want Midwest and then Northeast, and then Midwest and Northeast again for the Energy one. So, we have all four combinations: Industrials and Midwest, Industrials and Northeast, Energy and Midwest, and Energy and Northeast.

[18:17]

Now for the Order Dates between 2021 and 2024. So, I could say greater than or equal to 1/1/2021 and then less than or equal to 12/31/2024. That matches our criteria. Let's now copy these down.

And then for the Order Total, so let's enter a =DSUM function here. And then let's go over to our orders and let's select the whole table here. For the field, we want to get the Amount for this one. So, let's go and take Amount right there. And then for the criteria, we want to get these headers at the top and then these four rows. So, we have that.

And then for the next one, we can take this and just paste it down and then just change the =DSUM to =DCOUNT right there. Notice how we're still linking to J6 through P10. So, we still have those same four rows right there.

[19:10]

And then for the Commission Total, same idea, but we just need to change the column here in the =DSUM function. So, I'll just take this and then paste this down. And then let's change the field criteria right there. Go back and link it to Commissions instead, and so now we have that.

[19:27]

Now, we don't know for sure whether or not this works. If we want to, we could go in and check. But it looks reasonable to me based on the total amount of data and how specific we're making these criteria. If you want to, you could do a manual check of this yourself. But based on these formulas and the criteria we have, this seems to be correct.

So, that's it for this lesson. Let's now do a quick recap and summary.

The database functions in Excel fix some of the problems with the SUMIFS and SUMPRODUCT formulas where they can often get very difficult to read and edit and interpret. They do take some time and effort to set up because you need to create a separate area of the spreadsheet to support them. You have to get the syntax exactly right. The fields have to be in the same order. The data must be in tabular format. So, database functions are most useful for complex queries where you have AND and OR, and maybe it's too difficult or time-consuming to express with the SUMIFS or the SUMPRODUCT.

[20:17]

There are many different types of database functions. =DSUM, =DCOUNT, =DCOUNTA, and =DGET are some of the key ones that we look at and use here.

You have enter the database, ideally a data table, then the field, that header or column or field name that you're looking for, then the criteria, which is just this area with the headers, and then the actual criteria that you're using up here.

All of the normal operators, greater than, less than, greater than or equal to, less than or equal to, not equal to, equal to, all of work those.

In a single row, all the conditions are joined with an AND. But then when you have multiple rows, each row is joined with an OR instead. So, you saw how, for example, if you want to enter something like this, two OR conditions, you have to create two rows in the database criteria. And then you saw here how with these four OR conditions, we had to create these four separate rows.

[21:07]

That's about it for database functions. Coming up next, we'll get into pivot tables, and you will learn how to slice and dice data without even having to enter any type of formula in Excel, and how you can set up everything using a graphical interface.