

TOP 10

EXCEL FORMULAS

Work Smarter Not Harder



X XELPLUS

*Leila
Ghazvini*

10 Excel Formulas to Work Smarter

© 2020 Leila Gharani – XelPlus.com

Leila Gharani, MA



XelPlus.com

You have permission to share this with anyone you think will benefit. The catch is: you have to share it for free. You cannot sell this digital copy or a printed version of it. You cannot change its content nor edit it in any form. I hope this document will serve you well.

Preface

The purpose of this eBook is to help you build a strong foundation in Excel and to solve complex problems fast. If you're already advanced in Excel, go through this document to "fill in the gaps". I still watch videos on topics I'm fairly certain I know, but I watch them because of all the little tips and tricks I could still discover.

The main aim of this eBook is to introduce you to the top 10 formulas you can use in Excel to work smarter.

I demonstrate the formulas in my [online courses](https://xelplus.com/courses) (XelPlus.com/Courses), on my [blog](https://xelplus.com/blog) (XelPlus.com/Tutorials) and my [YouTube Channel](#) (Leila Gharani).

Let's get started!

Table of Contents

10 Excel Formulas to Work Smarter	2
Introduction	5
Top Excel Formulas Every Analyst Should Know	7
Function #1: SUMIFS & COUNTIFS	8
Function #2: IF - Nested Functions	11
Function #3: IFERROR for Credible Reports	13
Function #4: VLOOKUP & HLOOKUP for Dynamic Lookups	15
Function #5: INDEX & MATCH for Flexible Lookups	18
Function #6: The Most Useful Text Functions	22
Function #7: Excel's Top Date Functions	25
Function #8: OFFSET for Dynamic Calculations	28
Function #9: FREQUENCY, SMALL & LARGE	30
Function #10: SUMPRODUCT for the Gurus	34
Concluding Words	38
Bonus Tip #11: Speed up Excel - Understand Volatile versus Non-Volatile formulas	39
Bonus Tip #12: Know the possibilities and the limitations	42
My Courses	45
Free Excel Tutorials	46
More Resources	46

Introduction

Advanced Excel skills are highly relevant in business. No matter which role you're currently in or you're planning to take on. Whether you'll be in finance, consulting, IT, or project management, your advanced Excel skills will set you apart from the competition.

The aim of the top 10 list is to increase your awareness of the most useful Excel formulas to work faster and smarter.

What's the difference between an Advanced & Average Excel user?

Advanced Excel users find the optimal solution for every task, and they find it fast!

The way I picked this specific top 10 is purely based on my experience of what I've generally seen as "missing" knowledge by many Excel users.

When you think of it, using one formula over another doesn't mean it's more "advanced", it just means you have more alternatives and more options of doing things faster and this puts you one step ahead of the average Excel user.

Use this document to include missing techniques in your Excel vocabulary

We are creatures of habit and it's difficult to get our head around learning new techniques. Becoming advanced in Excel takes **practice**, **readiness to learn** new methods and **willingness** to apply these to practical cases. When you go through each of these formulas, try to think how you can apply it to your own files.

My Educational / Work Experience

I have a master's in economics from University of Toronto in Canada. My first job title was Economist, then Consultant, Oracle HFM Accounting Systems Expert & Project Manager. My passion is teaching and solving difficult Excel problems. I am also addicted to learning and enjoy taking courses on a variety of topics.

Check out my Courses

If you'd like to learn Excel in a structured manner, check out my bestselling Excel courses. This way you can learn anytime that fits your schedule best:

<https://www.XelPlus.com/Courses>



10 Formulas every analyst should know....

*and not necessarily by heart – but know of its existence and
capabilities...*

Top Excel Formulas Every Analyst Should Know

Excel offers an incredible number of functions that permit dynamic analysis of data. In this document, I will introduce you to the top 10 Excel functions I believe every user who deals with Excel on a frequent basis should know. It's only when you are aware of what Excel can do in terms of dynamic formulas, will you be able to create solid spreadsheets and be fast and confident in your analysis.

A friend of mine, who works in the IT department and uses Excel daily, came to me with the following question. He wanted to find which day of the week most users utilize their system. He had imported a data file into Excel that contained the username of the person logging in and other relevant information. It had a lot of dates, a lot of users, going back many months.

He needed to find the day of the week from the date field and create some dynamic Pivot Charts to visualize the utilization. For example, to check how Monday looks in comparison with Tuesday and whether there is a specific week day that has the least users logging in.

The way he had gone about this was to crosscheck the dates with his calendar and assign the week day manually in the file in a separate column. What surprised me was he didn't ask Google if Excel had a function he could use. He didn't think Excel was capable.

This is the reason for my top 10 formulas. You don't necessarily have to know these by heart but it's important that you understand their usage and are aware of their existence, so you can use them whenever the situation arises.

Function #1: SUMIFS & COUNTIFS

Function #2: IF - Nested Functions

Function #3: IFERROR for Credible Reports

Function #4: VLOOKUP & HLOOKUP for Dynamic Lookups

Function #5: INDEX & MATCH for Flexible Lookups

Function #6: Most Useful Text Functions

Function #7: Excel's Important DATE Functions

Function #8: OFFSET for Dynamic Calculations

Function #9: FREQUENCY, SMALL & LARGE

Function #10: SUMPRODUCT for the Gurus

Now let's have a look at them in more detail.

Function #1: SUMIFS & COUNTIFS

Do you know the difference between the SUMIF and SUMIFS function?

SUMIF allows only one condition to be evaluated, whereas SUMIFS (introduced in Excel 2007) allows up to 29 criteria to be evaluated. It's the plural version of the SUMIF function.

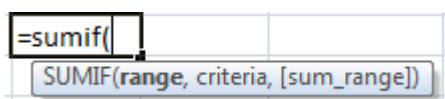
First, let us cover the SUMIF function. It sums the data in a column or row based on a condition.

In the example below, SUMIF adds the values in Column D, if Column A equals Feb.

	A	B	C	D	E	F	G
1	Month	Product	Country	Sales Revenue			
2	Jan	Shirt white	USA	500			
3	Jan	Shirt blue	UK	520			
4	Feb	Shirt white	USA	490			
5	Feb	Shirt blue	USA	550			
6	Feb	Shirt white	USA	590			
7	Feb	Shirt white	UK	530			
8	Feb	Shirt blue	USA	480			
9							
10	Find total sales for the below criteria:						
11	Feb			2,640	=SUMIF(A2:A8,A11,D2:D8)		

The syntax of the SUMIF function is comprised of the 3 following arguments:

- 1) the criteria range – in this case A2:A8
- 2) the criteria itself – in this case Feb
- 3) the sum range – in this case column D2:D8



What if you have multiple conditions?

How do you get the total sales for Shirt white, sold in the USA?

This is where the SUMIFS function comes in.

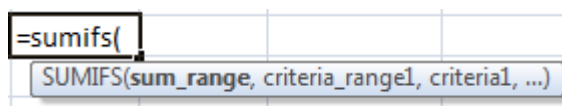
The syntax of SUMIFS is different to the SUMIF function. Notice for the SUMIF function, the sum range is at the end.

It's not possible to put the sum range at the end for the SUMIFS because the number of criteria to be evaluated can vary. This means the position of the sum range in the formula can vary.

For this reason, the sum range was switched to the first argument in the function. The original SUMIF formula couldn't be revised using this new format because it would cease to be compatible with older Excel versions.

The syntax of the SUMIFS function is

- First is the sum range
- Next the criteria range of your first criteria
- Followed by the criteria itself
- Add more criteria ranges followed by the criteria as necessary



The below example shows the formula that finds the total sales in Feb for shirt white in the USA.

	A	B	C	D	E	F	G	H	I
1	Month	Product	Country	Sales Revenue					
2	Jan	Shirt white	USA	500					
3	Jan	Shirt blue	UK	520					
4	Feb	Shirt white	USA	490					
5	Feb	Shirt blue	USA	550					
6	Feb	Shirt white	USA	590					
7	Feb	Shirt white	UK	530					
8	Feb	Shirt blue	USA	480					
9									
10	Find total sales for the below criteria:								
11	Feb			2,640	=SUMIF(A2:A8,A11,D2:D8)				
12	Feb	Shirt white	USA	1,080	=SUMIFS(D2:D8,A2:A8,A12,B2:B8,B12,C2:C8,C12)				

You can include in the criteria argument

- wildcards in the form of *
- mathematical operators such as >, <, =, or a combination of these

You need to surround the mathematical operators in quotation marks.

If you wanted total sales in Feb for shirt white, where sales are greater than 500, you would need to add additional criteria.

The criteria range is the sales value which is also your sum value. The ">500" can be written in different forms. It can also be written as ">"&500 or ">"&C13. C13 is the cell reference for the value 500.

Top Excel Formulas Every Analyst Should Know

	A	B	C	D	E	F	G	H	I
1	Month	Product	Country	Sales Revenue					
2	Jan	Shirt white	USA	500					
3	Jan	Shirt blue	UK	520					
4	Feb	Shirt white	USA	490					
5	Feb	Shirt blue	USA	550					
6	Feb	Shirt white	USA	590					
7	Feb	Shirt white	UK	530					
8	Feb	Shirt blue	USA	480					
9									
10	Find total sales for the below criteria:								
11	Feb			2,640	=SUMIF(A2:A8,A11,D2:D8)				
12	Feb	Shirt white	USA	1,080	=SUMIFS(D2:D8,A2:A8,A12,B2:B8,B12,C2:C8,C12)				
13	Feb	Shirt white	500	1,120	=SUMIFS(D2:D8,A2:A8,A13,B2:B8,B13,D2:D8,">"&C13)				

The same logic applies to the AVERAGEIFS function, which as opposed to the sum function calculates the average value of the range based on given criteria.

COUNTIFS function is shorter because it doesn't need the additional sum range. It only needs the criteria range. It counts the number of times the criteria are met.

=countifs(
COUNTIFS(criteria_range1, criteria1, ...)

	A	B	C	D	E
1	Company	Product	Sales Revenue		
2	Company A	Product A	428		
3	Company B	Product A	356		
4	Company C	Product B	549		
5	Company A	Product B	509		
6	Company B	Product B	499		
7	Company A	Product A	425		
8	Company C	Product A	516		
9					
10	COUNT Sales If Company = Company A & Product = Product A		2	Combination is found two times	

As you can see, the IFS version allows you to easily add exceptions to your sum, average and count calculations. Don't make your life more difficult by trying to use Excel filters to get your sum, use the IFS formula instead.

Function #2: IF - Nested Functions

Nesting is defined as “using formulas inside other formulas”. There are many reasons to nest formulas, but the biggest reason is that data analysis (or manipulation) isn’t a simple and straight forward task. In many situations, you may need to make exceptions or change one or more outcomes based upon a situation. In the example below, nested formulas are “housed” inside the IF statement.

Start with IF (what should happen if true, else what should happen if false). In the false statement, you can open with another IF.

One gigantic IF formula can contain a myriad of IF statements.

Imagine you have the below data set:

	A	B	C
1	Sales Agent	Quantity	Revenue
2	Agent A	5	280
3	Agent B	6	300
4	Agent C	1	50
5	Agent D	5	350
6	Agent E	2	210
7	Agent F	3	180
8	Agent G	4	350

You’d like to do the following analysis:

You’d like to flag the agents that sold less than 3 pieces. You can do this:

	A	B	C	D	E
1	Sales Agent	Quantity	Revenue	Flag if quantity <=3	
2	Agent A	5	280		=IF(B2<=3,"Flag", "")
3	Agent B	6	300		=IF(B3<=3,"Flag", "")
4	Agent C	1	50	Flag	=IF(B4<=3,"Flag", "")
5	Agent D	5	350		=IF(B5<=3,"Flag", "")
6	Agent E	2	210	Flag	=IF(B6<=3,"Flag", "")
7	Agent F	3	180	Flag	=IF(B7<=3,"Flag", "")
8	Agent G	4	350		=IF(B8<=3,"Flag", "")

The formula above translates to:

If the value in B is less than 3 then write “Flag” in E, otherwise leave E empty.

Maybe you decide to incorporate a second exception to the rule. Perhaps you need to flag agents who sold less than or equal to 3 pieces AND generated revenue that was less than 200.

Here, the logical operator AND comes in handy.

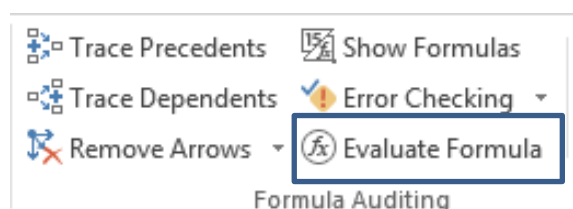
	A	B	C	D	E
1	Sales Agent	Quantity	Revenue	Flag if quantity <= 3 and revenue < 200	
2	Agent A	5	280		=IF(AND(B2<=3,C2<200),"Flag","")
3	Agent B	6	300		=IF(AND(B3<=3,C3<200),"Flag","")
4	Agent C	1	50	Flag	=IF(AND(B4<=3,C4<200),"Flag","")
5	Agent D	5	350		=IF(AND(B5<=3,C5<200),"Flag","")
6	Agent E	2	210		=IF(AND(B6<=3,C6<200),"Flag","")
7	Agent F	3	180	Flag	=IF(AND(B7<=3,C7<200),"Flag","")
8	Agent G	4	350		=IF(AND(B8<=3,C8<200),"Flag","")

Now let's say you need to make further exceptions like the below. This is where you can nest one IF function inside another IF formula.

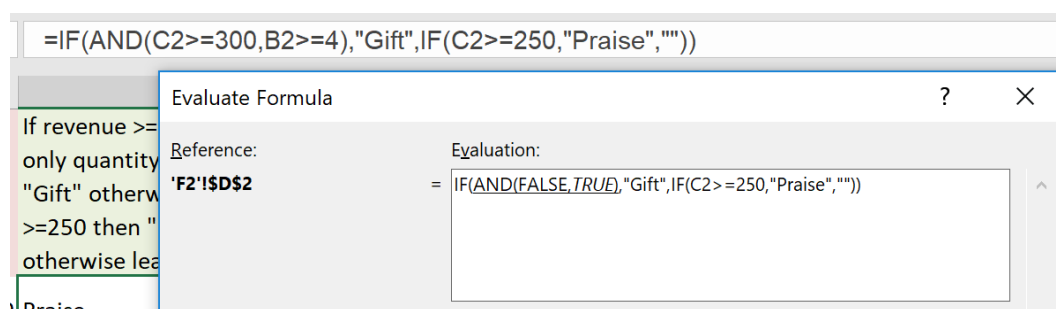
	A	B	C	D	E
1	Sales Agent	Quantity	Revenue	If revenue >= 300 and if only quantity >=4 then "Gift" otherwise if revenue >=250 then "Praise" otherwise leave empty	
2	Agent A	5	280	Praise	=IF(AND(C2>=300,B2>=4),"Gift",IF(C2>=250,"Praise",""))
3	Agent B	6	300	Gift	=IF(AND(C3>=300,B3>=4),"Gift",IF(C3>=250,"Praise",""))
4	Agent C	1	50		=IF(AND(C4>=300,B4>=4),"Gift",IF(C4>=250,"Praise",""))
5	Agent D	5	350	Gift	=IF(AND(C5>=300,B5>=4),"Gift",IF(C5>=250,"Praise",""))
6	Agent E	2	210		=IF(AND(C6>=300,B6>=4),"Gift",IF(C6>=250,"Praise",""))
7	Agent F	3	180		=IF(AND(C7>=300,B7>=4),"Gift",IF(C7>=250,"Praise",""))
8	Agent G	4	350	Gift	=IF(AND(C8>=300,B8>=4),"Gift",IF(C8>=250,"Praise",""))

Nested formulas, as you've probably observed, can be difficult to read.

In all these "hard to read" situations use the Evaluate Formula tool in Formulas, Formula Auditing to follow the intermediate results of the nested formula.



The Evaluate Formula feature displays the result for both True or False in the IF statement. Each answer is displayed separately as shown in the graphic below. Just click on Evaluate to see the intermediary results. It's a great tool for formula debugging - especially for deeply nested formulas.



Function #3: IFERROR for Credible Reports

If you create, run, or alter Excel reports often enough you'll eventually encounter formula errors. Understanding the types of errors that "pop up" and why they occur is essential, so you know how to properly correct them.

The below table provides an overview of the error types and their description.

Error	Description
#DIV/0!	This is probably the most common error and easiest to pinpoint. When you divide a number by zero you will get this error.
#VALUE!	This error is returned when the wrong type of argument is used. For example, if you try to multiply a value by a cell that has text in it.
#REF!	This error results when you move your formulas and haven't correctly specified which cells are fixed and which should move with your formula. Also, if you delete some cells that were a part of your formula.
#NAME?	The most common reason for this error is because the formula has been mistyped and Excel doesn't recognize what function you are referring to.
#N/A!	This is a frequent error if you are using Lookup formulas (especially in VLOOKUP) and the matching cannot be done.
#NULL!	This is an uncommon error and occurs if you haven't specified your ranges correctly, e.g. typing SUM(B4 B6) instead of SUM(B4:B6) or SUM(B4,B6).
#NUM!	This is another uncommon error and occurs when you supply an invalid number to your formula e.g. if you are using negatives where a positive number is required by your function.

The ISERROR() function, returns TRUE if any of the above errors occur in the formula. If you use the =IF(ISERROR(), ,) combination then you have the capability to instruct Excel to return that exact message (value) that you want shown if one of these errors are encountered. For example:

	A	B	C	D
1	#DIV/0!	=2/0		
2	Error	=IF(ISERROR(A1),"Error","")		

IFERROR was introduced in Excel 2007 and it's a better alternative to ISERROR. It's shorter, easier to write, and provides an elegant and simple way to hide errors.

	A	B	C	E
1	#DIV/0!	=2/0		
2	Error	=IF(ISERROR(A1),"Error","")		
3	Error	=IFERROR(A1,"Error")		

The main difference between them is that ISERROR evaluates the formula twice. IFERROR on the other hand, evaluates it only once making it easier to read, write and process.

Below, in column D, you see an example of an error free **% change** calculation. The IFERROR function hides the error values.

	A	B	C	D	E
					% Change with IFERROR
1	Customer	Quantity 2016	Quantity 2015	% Change	
2	Customer A	2,739	4,465	-39%	-39%
3	Customer B	2,667	Follow up	#VALUE!	
4	Customer C	2,447	1,468	67%	67%
5	Customer D	2,324		#DIV/0!	
6	Customer E	2,274	3,664	-38%	-38%
7	Customer F	2,273		#DIV/0!	
8	Customer G	1,915	3,043	-37%	-37%
9	Customer H	1,883	2,874	-34%	-34%
10	Customer I	1,814	1,600	13%	13%
11	Customer J	2,130		#DIV/0!	
12	Customer K	2,360	1,950	21%	21%

Note: For large array formulas, it's best to use an IF instead of IFERROR – it's not always possible – but if your logic allows you to make an exception using IF, go with the IF version. IFERROR can slow down your formulas because it needs to calculate the entire formula to see if it runs into an error. The IF will only run the calculation, if the exception is TRUE. This video shows an example of a case where IFERROR is replaced with an IF formula: <https://youtu.be/fDB1Ktyhp3Y>

Function #4: VLOOKUP & HLOOKUP for Dynamic Lookups

Most of the Excel user population has worked with a file that includes VLOOKUP or HLOOKUP functions. These two functions are easy to write and understand and they are a major time-saver for simpler lookups.

The VLOOKUP function can search for a specific category or value in a table and return the “opposing” value from adjacent columns.

HLOOKUP works exactly in the same way except that it looks horizontally rather than vertically. “V” in VLOOKUP refers to vertical and “H” in HLOOKUP refers to horizontal.

If the data in your table has column headers, use VLOOKUP. If the data table has row headers and you need to search horizontally across the rows for a “match” then use HLOOKUP.

The main limitation is that the value looked with VLOOKUP needs to be in the left-most table column and in HLOOKUP it must be in the top row of the data set.

Below is an example of the less commonly used HLOOKUP function.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Sales Agent	Agent A	Agent B	Agent C	Agent D	Agent E	Agent F	Agent G	Agent H	Agent I	Agent Y	Agent Z
2	Quantity	5	6	1	9	2	3	4	9	10	10	3
3	Revenue	200	300	50	350	90	200	350	333	410	350	102
4	Price	40	50	50	39	45	67	88	37	41	35	34
5	Customer	Customer A	Customer B	Customer C	Customer D	Customer E	Customer F	Customer G	Customer H	Customer I	Customer Y	Customer Z
6												
7				Select Agent:	Agent D							
8				Find Price	39	=HLOOKUP(E7,B1:L5,4,FALSE)						
9				Find Customer	Customer D	=HLOOKUP(E7,B1:L5,5,FALSE)						

HLOOKUP is used to search along the top row and then look down to retrieve the result from a specified row – the second row in this instance.

The example below uses the VLOOKUP function.

	A	B	C	D	E	F	G	H	I	J
1	Sales Agent	Quantity	Revenue	Price	Customer					
2	Agent A	5	200	40	Customer A		Agent Report			
3	Agent B	6	300	50	Customer B	Select Agent:	Agent D			
4	Agent C	1	50	50	Customer C	Find Price	39	=VLOOKUP(H3,A2:E8,4,FALSE)		
5	Agent D	9	350	39	Customer D	Find Customer	Customer D	=VLOOKUP(H3,A2:E8,5,FALSE)		
6	Agent E	2	90	45	Customer E					
7	Agent F	3	200	67	Customer F					
8	Agent G	4	350	88	Customer G					

The VLOOKUP formula syntax is the following

```
=vlookup(  
VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])
```

- Lookup_value: H3 or Agent D in this example.
- Table_array: the range includes the location of the column lookup value and the columns where the return value is located (i.e. A2:E8).

- **Col_index_num**: the number of columns it should move to the right. In this case, the range contains 5 columns with the match found in the 4th column for price and 5th column for customer.
- **Range_lookup**: this argument is important to use correctly if the lookup value is text and non-numeric. If the parameter is left empty, it defaults to “TRUE” meaning that an exact match is unnecessary. But, to work properly, the lookup column must be sorted in ascending order. Otherwise, an error or #NA will appear in the cell if the column isn’t sorted in ascending order and this argument is left empty.

In the example above, if FALSE is excluded from the last argument then the formula will work without a problem. This is because column A is sorted. If column A (the lookup column) wasn’t sorted, the formula will return a wrong value.

Column A is sorted – we can omit FALSE for the last argument:

	A	B	C	D	E	F	G	H	I
1	Sales Agent	Quantity	Revenue	Price	Customer				
2	Agent A	5	200	40	Customer A		Agent Report		
3	Agent B	6	300	50	Customer B		Select Agent:	Agent D	
4	Agent C	1	50	50	Customer C		Find Price	39	=VLOOKUP(H3,A2:E8,4)
5	Agent D	9	350	39	Customer D				
6	Agent E	2	90	45	Customer E				
7	Agent F	3	200	67	Customer F				
8	Agent G	4	350	88	Customer G				

Column A is not sorted – If we omit FALSE for the last argument, we get the wrong value.

	A	B	C	D	E	F	G	H	I
1	Sales Agent	Quantity	Revenue	Price	Customer				
2	Agent C	1	50	50	Customer C		Agent Report		
3	Agent E	2	90	45	Customer E		Select Agent:	Agent D	
4	Agent F	3	200	67	Customer F		Find Price	50	=VLOOKUP(H3,A2:E8,4)
5	Agent G	4	350	88	Customer G				
6	Agent A	5	200	40	Customer A				
7	Agent B	6	300	50	Customer B				
8	Agent D	9	350	39	Customer D				

If you’re not 100% sure that your lookup column is sorted, make sure you add FALSE for the last argument. You can also type in a 0 instead of writing FALSE.

	A	B	C	D	E	F	G	H	I	J
1	Sales Agent	Quantity	Revenue	Price	Customer					
2	Agent C	1	50	50	Customer C		Agent Report			
3	Agent E	2	90	45	Customer E		Select Agent:	Agent D		
4	Agent F	3	200	67	Customer F		Find Price	39	=VLOOKUP(H3,A2:E8,4,FALSE)	
5	Agent G	4	350	88	Customer G					
6	Agent A	5	200	40	Customer A					
7	Agent B	6	300	50	Customer B					
8	Agent D	9	350	39	Customer D					

Building up on the VLOOKUP Function

Wild Cards: Various wild cards can be used with VLOOKUP. You have the option to use the asterisk (*) to match any sequence of characters or the question mark (?) to match any single character. Note that if you want to find an actual question mark or asterisk type a ~ before the character.

```
=VLOOKUP("???????",A1:C9,2,FALSE)
```

Returns the value for the first 7 characters.

```
=VLOOKUP("*ast*",A1:C9,2,FALSE)
```

Returns the match value for the first "ast" in the name.

```
=VLOOKUP("*ast?03*",A1:C9,2,FALSE)
```

Uses both "?" and "*" to search for the first match of a text.

Limitations of VLOOKUP

One major limitation of VLOOKUP is that it cannot look to the left. The values to lookup must always be on the left-most column of the range and the values to return must be on the right-hand side.

You cannot use the standard VLOOKUP to look at the columns and the rows to find an exact match. The INDEX and MATCH functions are used in this instance and are much more flexible than the classical VLOOKUP.

Tip: Although the CHOOSE function nested in VLOOKUP is a "workaround" tool, the Index and Match combination is more effective to look data up in any direction.

Function #5: INDEX & MATCH for Flexible Lookups

Here are some of the reasons why Excel experts generally substitute VLOOKUP with INDEX and MATCH.

- Unlike VLOOKUP, which searches only to the right, INDEX and MATCH can look in both directions – left and right.
- INDEX & MATCH can perform two-way lookups by both looking along the rows and along the columns to find the intersection within a matrix.
- INDEX & MATCH is less prone to errors. Assume you have a VLOOKUP where the final value you want returned is in column N. Your lookup value is in column A. You need to highlight the entire A to N range and then provide your index number to be 14. If you happen to delete any of the in-between columns, you would have to update that index number. You don't need to worry about this when you use INDEX & MATCH.

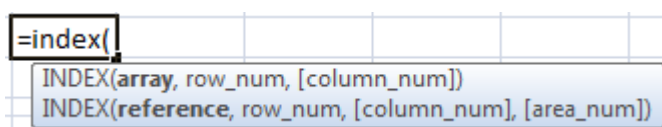
All in all, INDEX and MATCH is more flexible than VLOOKUP.

What it is

INDEX and MATCH are “nested” functions - two functions used together.

The first argument of INDEX is to give it an array. This array (range) should include your answer. You then need to specify how many rows to go down and how many columns to move to find the correct value. You cannot move outside the INDEX range.

The syntax of INDEX is:



- The range where the return value resides.
- Number of rows to move down (the row index)
- Number of columns to move to the right (the column Index). This argument is optional. If you only have one column, you can leave the column argument empty, otherwise, you need to specify the number of columns to move over, in the range.

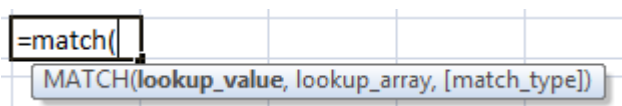
Here is an example of INDEX without MATCH:

	A	B	C	D	E
1	Sales Agent	Quantity	Revenue	Price	Customer
2	Agent A	5	200	40	Customer A
3	Agent B	6	300	50	Customer B
4	Agent C	1	50	50	Customer C
5	Agent D	9	350	39	Customer D
6	Agent E	2	90	45	Customer E
7	Agent F	3	200	67	Customer F
8	Agent G	4	350	88	Customer G
9					
10				39	=INDEX(D2:D8,4)

As shown above, INDEX alone is not useful for identifying the number of rows or columns to move. This is where MATCH comes into play. It helps the INDEX function to locate the answer.

The MATCH function is designed to return the position or address of a lookup value within a range. This is what differentiates MATCH with other functions. It doesn't return the VALUE in the cell but returns the POSITION of the cell within a specified range.

The MATCH function's syntax is as follows:



- Like VLOOKUP, Match needs a lookup (target) value. The user can reference a cell or directly type the value into the formula.
- Lookup_array: The "list" (range) where the return values are located.
- Match_type: 0 for an exact match.

In the below example, the MATCH function returns the value of 4. The lookup value is Agent D. 4 means, Agent D is the 4th position in the range that was selected.

The important point to note on the MATCH function is that it's a one-way street – you can only have one row, or one column selected.

	A	B	C	D	E	F	G
1	Sales Agent	Quantity	Revenue	Price	Customer		
2	Agent A	5	200	40	Customer A		
3	Agent B	6	300	50	Customer B		
4	Agent C	1	50	50	Customer C		
5	Agent D	9	350	39	Customer D		
6	Agent E	2	90	45	Customer E		
7	Agent F	3	200	67	Customer F		
8	Agent G	4	350	88	Customer G		
9							
10				39	=INDEX(D2:D8,4)		
11				4	=MATCH("Agent D",A2:A8,0)		

Putting it together

When INDEX and MATCH are used together, the MATCH function finds the look up value's row / column index and then hands this value off to the INDEX function to get the lookup value.

Think of it like a GPS: You have a map – this map includes your final destination – which includes the cell that contains your answer. You need to define the latitude and longitude to get to your destination. This is where the row number and column number arguments come in. This way Excel knows how many rows to move down and how many columns to move over to get to the destination. It starts to count from the top left-hand corner.

The example below shows how these work in tandem.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Sales Agent	Quantity	Revenue	Price	Customer			Agent Report						
2	Agent A	5	200	40	Customer A			Select Agent:	Agent D					
3	Agent B	6	300	50	Customer B			Find Price	39	=INDEX(D2:D8,MATCH(I2,A2:A8,0),1)				
4	Agent C	1	50	50	Customer C			Find Customer	Customer D	=INDEX(E2:E8,MATCH(I2,A2:A8,0))				
5	Agent D	9	350	39	Customer D									
6	Agent E	2	90	45	Customer E									
7	Agent F	3	200	67	Customer F									
8	Agent G	4	350	88	Customer G									

The VLOOKUP function must include all the columns in your range even if in-between columns aren't needed. INDEX and MATCH only requires the return value column which can significantly reduce processing time in larger worksheets.

The easy way to remember the syntax of the INDEX and MATCH is:

=INDEX (The range location of my answer, MATCH(The value I'm looking for, The look-up range to find the Value, 0 for exact match))

You can also do a lookup the other way around: Something the standard VLOOKUP can't do:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Sales Agent	Quantity	Revenue	Price	Customer			Agent Report						
2	Agent A	5	200	40	Customer A			Select Customer:	Customer D					
3	Agent B	6	300	50	Customer B			Get Agent	Agent D	=INDEX(A2:A8,MATCH(I2,E2:E8,0))				
4	Agent C	1	50	50	Customer C									
5	Agent D	9	350	39	Customer D									
6	Agent E	2	90	45	Customer E									
7	Agent F	3	200	67	Customer F									
8	Agent G	4	350	88	Customer G									

Matrix Lookups

Use INDEX MATCH & a second MATCH for cases where you need to look up a value in a matrix.

MATCH is used twice: once for the rows and once for the columns. In this example, the return range in the Index function involves multiple columns and the MATCH function instructs the formula to move down a certain number of rows and move over a number of columns across the sheet to retrieve the desired value.

Instead of just selecting a single row or column you need to index the entire table with multiple rows and columns as your array since your value is somewhere inside this matrix.

The syntax of the INDEX MATCH MATCH function is:

=INDEX(The full range where your answer can be found in, MATCH(The value I'm looking for in the rows, The range I need to find the Value, 0), MATCH(The value I'm looking for in the columns, The range I need to find the Value, 0))

	A	B	C	D	E	F	G	H	I
1	Sales Agent Quantity Revenue Price Customer					Agent Report			
2	Agent A	5	200	40	Customer A			Select Agent:	Agent E
3	Agent B	6	300	50	Customer B			Select Variable:	Price
4	Agent C	1	50	50	Customer C			Answer:	45
5	Agent D	9	350	39	Customer D				
6	Agent E	2	90	45	Customer E				
7	Agent F	3	200	67	Customer F				
8	Agent G	4	350	88	Customer G				

Building on the INDEX and MATCH Function

You can do a lot more with INDEX and MATCH by slightly tweaking the function or including it in other formulas.

One common question with this function is how do I handle multiple row or column lookups? So, what if my header isn't unique and I need to match for a combination of price and year? Find out how in chapter 2 of this post:

<https://www.xelplus.com/excel-index-and-match/>

By nesting INDEX and MATCH in other formulas you can create more complex, dynamic calculations.

For example, you can nest INDEX and MATCH in the SUMIFS function. This way you can show the SUM of different columns depending on what the user wants to see. You can see an example of this in this post:

<https://www.xelplus.com/excel-sumifs-sum-alternate-columns-based-on-header/>

INDEX & MATCH is one of my favorite functions because it can be used in many complex situations. A complete understanding of this function is what separates the gurus from the norms.

Function #6: The Most Useful Text Functions

Text formulas are a great time saver for cleaning data and transforming them into formats you need. Sometimes you need to manipulate the product descriptions or remove a particular wording in a certain portion of a text string. The below list includes the most oft-used Text formulas.

UPPER(), **LOWER()**, **PROPER()** to format text accordingly
LEN() to get the length of text in terms of number of characters
TRIM() to remove unnecessary spaces at start mid or end of text
LEFT(), **RIGHT()**, **MID()** to extract portions of text
SEARCH() or **FIND()** to find the position of a character in text

UPPER, LOWER and PROPER

These functions “toggle” between capital letters and small letters.

UPPER switches the selected characters to upper-case and LOWER to lower-case characters. The PROPER function is the combination of upper and lower cases: the first character is changed to upper case while the other characters are switched (or kept) in lower-case.

	A	B	C	D	E	F	G
1	CUSTOMER NAME	UPPER		LOWER		PROPER	
2	john anderson	JOHN ANDERSON	=UPPER(A2)	john anderson	=LOWER(A2)	John Anderson	=PROPER(A2)
3	Valery Mong	VALERY MONG		valery mong		Valery Mong	
4	toby cartER	TOBY CARTER		toby carter		Toby Carter	

LEN

The **LEN** formula counts the number of characters in a cell. This includes any spaces before, between or after words.

	A	B	C
1	CUSTOMER NAME	Get Length	
2	john anderson	13	=LEN(A2)
3	Valery Mong	11	
4	toby cartER	11	

TRIM

The TRIM function removes any trailing spaces in a cell that occur before and/or after any words. It doesn't remove spaces between words unless they have more than one space.

TRIM is generally used to remove trailing spaces from data that is pulled from another database or data copied and pasted into Excel from non-Excel files.

Leading or trailing spaces cause VLOOKUP or INDEX and MATCH formulas not to be able to find your data due to the extra space(s). In these cases, use TRIM to “trim” those unwanted spaces before performing the lookup.

C8				
fx =TRIM(A8)				
	A	B	C	D
		Get Length		Get Length
7	CUSTOMER NAME	BEFORE trim	TRIM	AFTER trim
8	john anderson__	17	john anderson	13
9	Valery Mong	12	Valery Mong	11
10	toby_carter	12	toby carter	11

LEFT, RIGHT, MID

These functions are quite useful for parsing out parts of text. The LEFT function, for example, will strip out the left part of the text. This is accomplished by providing the function with the number of characters that should be “stripped out”.

For example, =LEFT(“TRY ME”, 3) results in TRY.

The RIGHT function behaves the same way except it strips out the right part of the text or text string.

The MID function will strip out the mid sections of the text. However, The MID function requires the starting character count in the text (or text string) to work.

	A	B	C	D
1	CUSTOMER NAME	First Name	Middle Name	Last Name
2	Toby Jane Carter	Toby	Jane	Carter
3		=LEFT(A2,4)	=MID(A2,6,4)	=RIGHT(A2,6)

These functions are not very flexible because they need to know exactly how many characters to strip out and can’t “find” spaces between characters.

To find spaces where they occur between words and to also cause the LEFT, RIGHT and MID functions to become more dynamic use the SEARCH and FIND functions.

SEARCH and FIND

These two functions are alike except the FIND function is case sensitive and the SEARCH function is not. SEARCH also allows for wild cards.

The SEARCH and FIND functions help find spaces, any specific words and/or letters in your characters.

	A	B	C
1	CUSTOMER NAME	Find t	Search for t
2	Toby Jane Carter	14	1
3		=FIND("t",A2)	=SEARCH("t",A2)

Putting it all Together

The above functions used as standalone functions can be useful if working with static text length strings. If, however, the text length constantly changes, and you need to remove certain parts due to the varying text lengths then it's necessary to combine the formulas.

	A	B	C	D	E
1	CUSTOMER NAME	First Name		Last Name	
2	john anderson	John	=TRIM(LEFT(PROPER(A2),SEARCH(" ",A2)))	Anderson	=RIGHT(PROPER(A2),LEN(A2)-SEARCH(" ",A2))
3	Valery Mong	Valery		Mong	
4	toby cartER	Toby		Carter	

In the example above, LEFT, PROPER and SEARCH are “nested” to strip out the first name. The RIGHT, PROPER, LEN and SEARCH are then “nested” to strip out the family name.

These Text formulas can save a great deal of time cleaning up your data and getting them in the format you need.

Keep in mind this is not an exhaustive list. It's what I find to be the most useful text functions in Excel. Whenever you need to manipulate text, codes or descriptions think of these functions and how you can combine them to get the result you want.

Function #7: Excel's Top Date Functions

Excel's date functions store dates as serial numbers representing the days since January 1, 1900. The following date functions are very useful:

DATE(), **YEAR()**, **MONTH()**, **DAY()** to turn numbers to a date or get the corresponding year, month or day from a given date

WEEKDAY() to get the day of the week for a given date

WORKDAY() to get the **end date** from a start date and number of working days in between

NETWORKDAYS() to get the **number of working days** between two dates excluding the weekends

DATE

The DATE () function combines values from different cells that denote the year, month and day date elements into a single cell to create a single date.

The YEAR(), MONTH() and DAY() functions do the opposite: they strip out each relevant section from the date and display it separately in a cell.

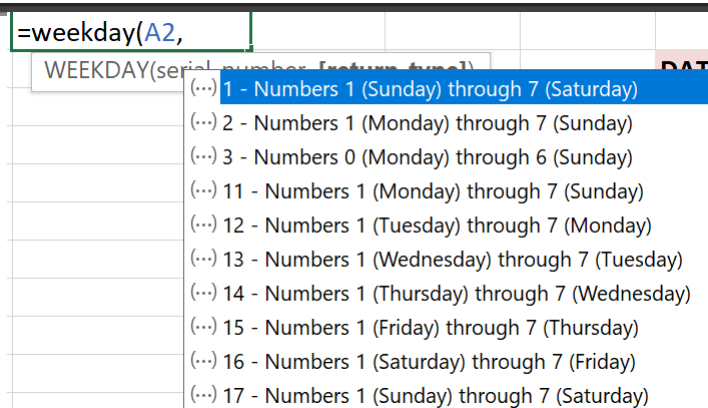
	A	B	C	D	E	F
1	YEAR	MONTH	DAY	DATE		
2	2019	3	21	Thu, 21/03/2019	=DATE(A2,B2,C2)	
3						
4	DATE	YEAR	MONTH	DAY		
5	3/21/2019	2019	3	21		
6		=YEAR(A5)	=MONTH(A5)	=DAY(A5)		

WEEKDAY

WEEKDAY() function returns the weekday number from a given date. This function is useful for showing trends by weekdays. For example, to compare Monday sales to Tuesday sales or weekend sales, write the Weekday formula as follows:

=WEEKDAY(Date , variable that determines when the week starts)

By selecting the desired value from Excel's argument list, you can define if the weekday should start on a Sunday, Monday etc.



1 means the weekday should start on a Sunday. This is Excel's default value for the start of the weekday.

	A	B	C
1	DATE	DAY	
2	3/21/2016		2 =WEEKDAY(A2,1)

WORKDAY

WORKDAY is another great function for creating project timelines or reporting deadlines.

WORKDAY returns the end date based on a starting date plus the number of days needed to complete the task. This function EXCLUDES weekends by default so keep this in mind when using it. Holidays can also be excluded if you provide the dates for the holidays to the formula.

	A	B	C	D	E	F
1	START DATE	Days	Holidays	Result		
2	Mon, 31/12/2018	3	Include holidays	Thu, 03/01/2019	=WORKDAY(A2,B2)	
3	Mon, 31/12/2018	3	Exclude holidays	Fri, 04/01/2019	=WORKDAY(A3,B3,B5:B6)	
4						
5	Holidays	1/1/2019				
6		1/6/2019				
7						
8	Days overview					
9	MON	31				
10	TUE	1	holiday			
11	WED	2				
12	THU	3				
13	FRI	4				
14	SAT	5	weekend			
15	SUN	6	weekend/holiday			
16	MON	7				

Prior to Excel 2010, weekends were always assumed to be Saturday and Sunday. Starting from Excel 2010, it's possible to select when weekends start depending on the country.

In the above example, assume you are creating a deadline report memo. Enter the starting date followed by the number of working days (column B). Include any holidays that might occur during these days (as in B5 & B6). Note that you need to create a holiday date table somewhere in your spreadsheet. Once these have been listed, you can write your formula as in E3.

=WORKDAY("Starting Date", "number of working days", "List of holidays")

That's it! You now have a dynamic formula that can calculate your end date based on any start date and the number of working days you specify. This saves time and potential errors when designing your timeline.

NETWORKDAY

NETWORKDAY function determines the number of working days between two dates. It also excludes weekends by default. Just like the WORKDAYS function, you can exclude holidays as long as a list of these are included in your spreadsheet.

It is also important to note that NETWORKDAY also includes start and end dates.

The function is written as follows:

NETWORKDAY(Starting date, End Date, List of Holidays)

	A	B	C	D	E	F
1	START DATE	END DATE	Holidays	Result		
2	Mon, 31/12/2018	Fri, 04/01/2019	Include holidays	5	=NETWORKDAYS(A2,B2)	
3	Mon, 31/12/2018	Fri, 04/01/2019	Exclude holidays	4	=NETWORKDAYS(A3,B3,B5:B6)	
4						
5	Holidays	1/1/2019				
6		1/6/2019				
7						
8	Days overview					
9	MON 31					
10	TUE 1	holiday				
11	WED 2					
12	THU 3					
13	FRI 4					
14	SAT 5	weekend				
15	SUN 6	weekend/holiday				
16	MON 7					

In the above example, you can see that if holidays are included in the calculation then the number of working days between 31.dec and 04 Jan is 4. If holidays are excluded from calculation, the number of working days is 5.

To Recap

Date functions are great when you need to create a timeline for your project, reporting timetable or list of tasks with deadlines. What people often do is to look up the days in the calendar.

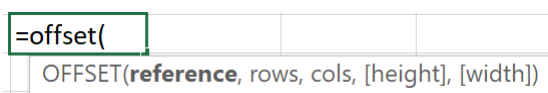
Use Excel's DATE functions instead because they don't just save time but save you from making mistakes.

Function #8: OFFSET for Dynamic Calculations

OFFSET tells Excel to “fetch” a cell location (address) from within a data range.

The offset formula looks like:

=OFFSET(starting point, rows to move, columns to move, height, width)



Explained visually:



Here are practical uses of OFFSET:

- **Dynamic Calculations:** OFFSET can be included in other Excel functions to get dynamic ranges. For example, calculating the average sales of the last 3 months based on a selection
- **Flexible Chart Ranges:** In conjunction with charts it can be used in the NAME manager to add flexibility to chart data. The OFFSET function behaves like an Excel table where the data range automatically expands and contracts when chart data is updated.
- **Dynamic dashboards:** It can be used together with Excel’s Form Controls to allow for dynamic analysis and reports based on user-selection.

You can tell the OFFSET function to return one cell if you specify height and width as 1. It can return an array of cells when the height and width are greater than 1.

Height and/or width CANNOT be zero, because you either need 1 cell or a range of cells. In the illustration above, the height and width are 2.

The example below illustrates how it’s used in practice. You’d like a report to provide the average sales totals for the last 6 months.

Top Excel Formulas Every Analyst Should Know

	A	B	C	D	E	F
1	Month	Sales		Average of last 6 months		
2	01/2014	100		113		
3	02/2014	120		=AVERAGE(OFFSET(B1,COUNTA(B2:B12),0,-6,1))		
4	03/2014	130				
5	04/2014	90				
6	05/2014	105				
7	06/2014	110				
8	07/2014	125				

Assume, July is the last month - row 8. A static formula would be written as:

```
=AVERAGE(B3:B8)
```

When August is added, the 6-month range would have to be updated manually from B3:B8 to B4:B9. To avoid having to change this manually each month use the OFFSET and COUNTA functions together to create a dynamic range that updates automatically each month.

COUNTA counts the number of non-empty cells in the range – notice that a larger range than used has been specified so that once new data are added, the COUNTA formula is automatically updated. COUNTA returns 7 and once data is added to row 9, COUNTA will return 8.

Here's what the formula does:

It starts at B1 which is the header. It then moves down 7 cells (get to B8) and moves 0 columns (stays on B8). For the height it includes 6 cells going back (B8 included) and 1 column. This means range B3 to B8 are selected.

Once we add more data to the bottom, we end up changing the COUNTA part of the formula, which in turn means we always move to the bottom of the data range.

OFFSET is an interesting formula and it's one that can do so much more than meets the eye. It can be used inside many formulas to provide them with flexible ranges instead of fixed ranges. This makes it a perfect candidate for dynamic reports and dashboards.

Function #9: FREQUENCY, SMALL & LARGE

The FREQUENCY, SMALL and LARGE functions are very effective in organizing and presenting data. FREQUENCY function is often overlooked because it's an array function that requires Control Shift Enter (CSE). This means it has to be entered in a special way. It is, however, an extremely simple formula that saves a tremendous amount of time. Simply put - FREQUENCY organizes data between categories (bins).

The MAX and MIN functions are popular choices for returning the high and low values in a data set. Their useful counterparts, LARGE and SMALL, work in a similar manner except you can define whether you want the largest, second largest, third largest etc.

The class of functions that help you organize your data {FREQUENCY()} is a special array formula that organizes data between categories

MIN() & MAX() find the smallest and the largest data

SMALL() & LARGE() find the nth smallest or the nth largest data (with n being any integer value)

FREQUENCY

	A	B
1	Age	Age group
2	17	20
3	18	30
4	20	40
5	21	50
6	26	60
7	27	100
8	27	
9	28	
10	30	
11	32	
12	33	
13	40	
14	46	
15	50	
16	57	
17	66	
18	70	

FREQUENCY function counts the number (occurrence) of times an element appears in specified groupings (bins) within the defined data set (range).

For example, assume you need to report the number of people in your department who fall between specific age categories.

Column A shows people's ages and column B defines the age groupings (bins). These "age" bins fall into the following group:

of people less than 20;

of people between 20 and 30;

of people between 30 and 40;

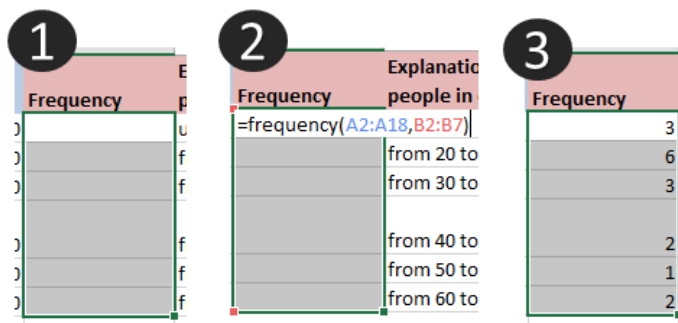
and the # of people between the ages of 60 to 100.

FREQUENCY quickly counts and allocates the number of people that fall within these categories.

Once catch is that FREQUENCY is an array formula. To work properly, it needs to be written between two curly brackets --"{FREQUENCY()}"}. You cannot manually input the curly brackets. You need to follow these steps:

1. Highlight the cell range where the formula is to be used – i.e. the cells where you want the answers to appear. In this example, it would be range (C2:C7)

- Type =FREQUENCY(Highlight raw data in column A , highlight your groups in Column B)
- Press “Control + Shift + Enter” (CSE) instead of Enter. CSE tells Excel to put the curly brackets (“{}”) around the formula which converts it to an array formula.



PRESS CONTROL + SHIFT + ENTER

	A	B	C	D
1	Age	Age group	Frequency	Explanation: Number of people in each category
2	17	20	3	up to and including 20
3	18	30	6	from 20 to 30 (30 included)
4	20	40	3	from 30 to 40 (40 included)
5	21	50	2	from 40 to 50 (50 included)
6	26	60	1	from 50 to 60 (60 included)
7	27	100	2	from 60 to 100
8	27			
9	28			
10	30			
11	32			
12	33			
13	40			
14	46			
15	50			
16	57			
17	66			
18	70			

With one simple formula, you get the number of people that are assigned to each age group.

The same concept applies to reports that groups companies based on revenue or other factors. Using this simple formula avoids having to do this manually or having to create complicated nested formulas.

MIN, MAX, SMALL and LARGE

The MIN and MAX functions find the smallest and largest value in your data set. The SMALL and LARGE functions find the nth smallest or the nth largest data (with n being any integer value). If you wanted to show the highest sales value, for example, in a data range you would use the MAX function. However, to show the second and the third highest sales values you would need the LARGE ("data range",2) and LARGE("data range",3) formulas.

These are very useful functions for creating dynamic reports. Coupled with INDEX and MATCH you can find the actual company that had the highest, second highest and third highest sales. The example below demonstrates how this works.

	F	G	H	I	J	K
1	Names	Age	youngest person	oldest person	3rd youngest person	3rd oldest person
2	Mini	17	17	70	20	57
3	Mia	18	=MIN(G2:G18)	=MAX(G2:G18)	=SMALL(G2:G18,3)	=LARGE(G2:G18,3)
4	Mio	20	Mini	Loli	Mio	Papi
5	Alex	21	=INDEX(F2:F19,MATCH(MIN(G2:G18),G2:G18,0))	=INDEX(F2:F19,MATCH(MAX(G2:G18),G2:G18,0))	=INDEX(F2:F19,MATCH(SMALL(G2:G18,3),G2:G18,0))	=INDEX(F2:F19,MATCH(LARGE(G2:G18,3),G2:G18,0))
6	Anton	26				
7	Petra	27				
8	Stefan	27				
9	Martin	28				
10	Maja	30				
11	Sara	32				
12	Michael	33				
13	Milan	40				
14	Jasper	46				
15	Loren	50				
16	Papi	57				
17	Lali	66				
18	Loli	70				

MIN(number1, [number2], ...)

SMALL(array, k)

The functions on their own

Combines the functions with INDEX & MATCH to find the name of the person

MIN and MAX functions are used in cells H3 and I3 to find the age of the youngest and the oldest person respectively. SMALL is used to find the age of the 3rd youngest person and LARGE is used to find the age of the 3rd oldest person.

To find the names associated with each, INDEX and MATCH is used in row 4 (formula is shown in row 5).

To Recap

FREQUENCY function is a very simple formula that provides you with the number of occurrences within your data for specific intervals. You can use this for salaries, revenue, quantities or anything numeric. Don't forget Ctrl + Shift + Enter. The MAX, MIN, LARGE and SMALL functions are great when you need to organize your data and provide a specific ranking or order to your data.

#EXCELHASH

Check out this Excel Challenge to see different ways you can use the FREQUENCY and MAX functions. You'll be surprised at what they can do.

What is #ExcelHash?

It's the Iron Chef Tournament between 6 Excel MVP YouTubers (you probably know most of them):

- Mike Girvin from Excel is Fun
- Bill Jelen from MrExcel
- Mynda Treacy from MyOnlineTrainingHub
- Jon Acampora from Excel Campus
- Oz du Soleil from Excel on Fire
- Leila Gharani – That's me

The final 4 ingredients that were randomly picked were:

Form Controls, MAX, FREQUENCY, 3D Model

Not the easiest set of ingredients to work with!

This is my solution: https://XelPlus.com/excelhash_1



All videos are great!

You'll be surprised at the random tips and tricks you discover watching them.

The competition is intense!

Find out more and watch all solutions here: https://XelPlus.com/excelhash_1

Function #10: SUMPRODUCT for the Gurus

The SUMPRODUCT function is **THE** function to work smart in Excel. This function wears many hats and has many uses other than the traditional sum of products formula, which the name implies.

SUMPRODUCT is an array function but is different than most of Excel's array functions in that it doesn't require Ctrl + Shift + Enter (CSE). SUMPRODUCT can be used to help solve many problems in non-obvious ways. SUMPRODUCT can be used like SUMIFS, it can be used to COUNT, and it can also be used like INDEX and MATCH.

First off let's demonstrate the classical use of SUMPRODUCT:

	A	B	C	D	E
1	Company	Product	Price	Quantity	Revenue
2	Customer A	Product A	222	200	44,400
3	Customer B	Product A	400	33	13,200
4	Customer C	Product A	200	100	20,000
5	Customer D	Product B	2	90	180
6	Customer E	Product B	200	100	20,000
7	Customer F	Product A	100	110	11,000
8				Total	108,780
9				Sumproduct	108,780

This function can be written in two ways which returns the same result. Using either Excel separator “,” (comma) or using the multiply sign “*” (multiplication) performs the matrix algebra calculation. However, when SUMPRODUCT is used in non-classical ways, it is generally necessary to use “*” operator instead of the “,”.

Let me explain:

Let's say we want to get the total quantity for Product A only. We add the exception inside brackets (B2:B7=H2) and then multiply this with the quantity range (D2:D7). Using brackets to group the criteria is crucial for the SUMPRODUCT function to interpret the different criteria groupings properly.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Company	Product	Price	Quantity								
2	Customer A	Product A	222	200			Only for:	Product A				
3	Customer B	Product A	400	33			Quantity	443	=SUMPRODUCT((B2:B7=H2)*D2:D7)			
4	Customer C	Product A	200	100			Revenue	88,600	=SUMPRODUCT((B2:B7=H2)*C2:C7*D2:D7)			
5	Customer D	Product B	2	90								
6	Customer E	Product B	200	100								
7	Customer F	Product A	100	110								

The first check B2:B7=H2 results in TRUE & FALSE values.

We get: {TRUE;TRUE;TRUE;FALSE;FALSE;TRUE}

The multiplication “*” forces the TRUE and FALSE values to become 1 and 0 values – i.e. we end up multiplying the values in the quantity column with 1s and 0s. This way we get rid of products that aren’t Product A.

There’s another way you can write this formula (last formula below)

Only for:	Product A	
Quantity	443	=SUMPRODUCT((B2:B7=H2)*D2:D7)
Revenue	88,600	=SUMPRODUCT((B2:B7=H2)*C2:C7*D2:D7)
	88,600	=SUMPRODUCT(--(B2:B7=H2),C2:C7,D2:D7)

Notice I’m using “,” instead of “*”. I use double negation to force the TRUE and FALSE Boolean values of the first check to become 1s and 0s.

There’s more:

One of the advantages that SUMPRODUCT has over SUMIFS, is that it can understand OR conditions besides checking for AND conditions. OR conditions are written in SUMPRODUCT using the “+” operator. AND conditions are denoted using the “*” operator. Using brackets to group the And / OR criteria is crucial for the SUMPRODUCT function to interpret the different criteria groupings properly.

The below illustrates some of the different ways of writing the SUMPRODUCT function.

The formula translation for “sum of sales in cell I6” is as follows:

SELECT SUM(quantity) FROM Quantity (column E) WHERE Entity=Company A or Entity=Company C, Year=2019 and Month=Jan

	A	B	C	D	E	F	G	H	I	J	K
1	Company	Year	Month	Price	Quantity	Revenue		Conditions			
2	Company A	2019	Feb	300	200	60,000		Company	Company A or Company C		
3	Company B	2019	Jan	400	33	13,200		Year	2019		
4	Company A	2019	Jan	100	100	10,000		Month	Jan		
5	Company A	2019	Jan	200	90	18,000		Count quantity (SUMPRODUCT)	4	=SUMPRODUCT(((A2:A10=I2)+(A2:A10=K2))*(B2:B10=I3)*(C2:C10=I4))	
6	Company C	2019	Jan	320	100	32,000		Sum quantity (SUMPRODCUT)	390	=SUMPRODUCT(((A2:A10=I2)+(A2:A10=K2))*(B2:B10=I3)*(C2:C10=I4)*E2:E10)	
7	Company D	2019	Feb	350	110	38,500		Sum Revenue (SUMPRODCUT)	82,200	=SUMPRODUCT(((A2:A10=I2)+(A2:A10=K2))*(B2:B10=I3)*(C2:C10=I4)*E2:E10*D2:D10)	
8	Company A	2019	Jan	222	100	22200					
9	Company A	2018	Feb	400	90	36000					
10	Company A	2018	Jan	200	80	16000					

Below shows the SUMPRODUCT function for the sum of revenue in slow motion:

The criteria:

Conditions	
Company	Company A or Company C
Year	2019
Month	Jan

Behind the scenes:

Company	Year	Month	Price	Quantity	Revenue		Company	Year	Month	Price	Quantity	Revenue
TRUE	TRUE	FALSE	300	200	60,000		1	1	0	300	200	0
FALSE	TRUE	TRUE	400	33	13,200		0	1	1	400	33	0
TRUE	TRUE	TRUE	100	100	10,000		1	1	1	100	100	10,000
TRUE	TRUE	TRUE	200	90	18,000		1	1	1	200	90	18,000
TRUE	TRUE	TRUE	320	100	32,000		1	1	1	320	100	32,000
FALSE	TRUE	FALSE	350	110	38,500		0	1	0	350	110	0
TRUE	TRUE	TRUE	222	100	22,200		1	1	1	222	100	22,200
TRUE	FALSE	FALSE	400	90	36,000		1	0	0	400	90	0
TRUE	FALSE	TRUE	200	80	16,000		1	0	1	200	80	0
												82,200

Note that in the first step, the function translates each calculation to either TRUE or FALSE. Excel cannot add TRUE or FALSE values with one another since these are considered to be text strings (Boolean Logic). A mathematical operation of some type (e.g. addition, multiplication, etc.) must be performed on these Boolean values to convert them to 1's and 0's.

The use of the "*" operator here performs the math operation (multiplication) and converts the Boolean values (i.e. TRUE and FALSE) to the integers 1 and 0. After converting to 1's and 0's, multiplication is performed on the remaining formula members. Therefore, you need to use "*" operator instead of "," if you use SUMPRODUCT in this form.

The 1's and 0's resulting from the Boolean math allows the SUMPRODUCT to behave as a "Counting" function because each value being compared ends up occurring either 1 time (the TRUE result) or 0 times (the false result).

Can you see why SUMPRODUCT is such an unusual and useful function? It's not just the sum of products, but you can use it as a COUNTIF function, a SUMIF function, a LOOKUP function and more!

Quiz Time

You come across this data:

	A	B	C	D	E	F
1	240	32	45		45	34
2	240	32	32		34	240

Someone has entered this formula:

=SUMPRODUCT(--(A1:F1=A2:F2),--(A1:F1<>""))

What does the formula return?

Let's step into the formula:

A1:F1=A2:F2 gives us {TRUE,TRUE,FALSE,TRUE,FALSE,FALSE}

--(A1:F1=A2:F2) becomes {1,1,0,1,0,0}

A1:F1<>"" gives us{TRUE,TRUE,TRUE,FALSE,TRUE,TRUE}

--(A1:F1<>"") becomes {1,1,1,0,1,1}

{1,1,0,1,0,0} multiplied by {1,1,1,0,1,1} (the product part of sumproduct) gives us

{1,1,0,0,0,0} and the Sum part of sumproduct gives us 2.

Answer = 2

This formula could also be written as:

=SUMPRODUCT((A1:F1=A2:F2)*(A1:F1<>""))

The * forces the TRUE and FALSE values to become 1 and 0 values.

What the Formula Does:

It counts identical numbers in row 1 and row 2 only if they are below one another. The count excludes empty cells.

Concluding Words

Let's take a look at what we covered:

1. SUMIFS & COUNTIFS
2. IF - Nested Functions
3. IFERROR for Credible Reports
4. VLOOKUP & HLOOKUP for Dynamic Lookups
5. INDEX & MATCH for Flexible Lookups
6. The Most Useful Text Functions
7. Excel's Top Date Functions
8. OFFSET for Dynamic Calculations
9. FREQUENCY, SMALL & LARGE
10. SUMPRODUCT for the Gurus

With these functions you are well equipped with the knowledge you need to handle pretty much any situation that requires you to analyse data or grab a piece or pieces of information from a large table.

Bonus Tip #11: Speed up Excel - Understand Volatile versus Non-Volatile formulas

The design of your Excel spreadsheet is key in defining how efficient and fast your workbook calculations run. Often spreadsheets run slow, not because of the number of formulas used or the workbook size, but due to the number of cell formatting and the efficiency of the functions used.

It's also important to understand how Excel runs calculations and which cells are calculated. Knowing the difference between volatile and non-volatile functions can also help you define which formulas to use if your files slow down.

Make sure your work is optimized:

- Delete all invalid links, names or formulas
- Keep as much information as possible on the same workbook
- Delete everything that is "out of date". Press Ctrl + END to determine the actual used range on each one of your sheets. Remove rows or columns that aren't needed by deleting them. Save the workbook. The used range will reset once you re-open the workbook.

Make sure your formulas are optimized to:

- Use the most efficient function possible to get your results. It's extremely important to thoroughly understand Excel's more advanced formulas and from this understanding you can judge which formulas work best for a given situation. Knowledge like this reduces workarounds and complex calculations. Expand your Excel Vocabulary. For example where possible, use IFERROR() instead of IF(ISERROR(),,) and where possible use IF() instead of IFERROR().
- Minimize the number of cells in a function. For example, if you're using VLOOKUP or SUMIFS do not reference the entire column (D:D), instead use (D1:D200).

Be careful with Volatile functions

Volatile functions are formulas that automatically recalculate any time data is entered. They get calculated independent of whether your formula is using these cells or not; Volatile cells are also recalculated when a new row or column is inserted; a worksheet is renamed or the worksheets tab order changes. Any task performed in the open file that contains volatile formulas causes them to recalculate. Examples of volatile functions are:

- =OFFSET()
- =TODAY()
- =NOW()
- =INDIRECT()
- =RAND() & RANDBETWEEN()
- =CELL ()
- = INFO()

Chances are, if you're an intermediate Excel user, you've used some of these.

Conditional formatting formulas are also volatile. Conditional formats are evaluated every time data change. Since conditional formatting is comprised of both formatting and calculations, both can decrease calculation time twofold. Note that conditional formats are only updated based on what is visible on the screen, everything else is not. Extensive use of highly complex conditional formatting formulas can cause scrolling up and down to be slower as well.

Using volatile functions may not necessarily be the reason for a slow workbook. The file "might" just slow down because you have a large and rather complex workbook. If your workbook is quite slow and volatile functions are being used, then it would be wise to look for alternative calculation methods. Sometimes you can replace volatile functions with non-volatile ones - substituting OFFSET with the INDEX and MATCH function for example.

VBA (Visual Basic for Applications) is also an option for complicated workbooks and calculations.

How do normal formulas work?

Non-volatile formulas recalculate any time a cell is changed that impacts the formula. Conversely, if the updated cells impact these formulas then nothing gets recalculated. Let's assume you have a large Excel file that has thousands of formulas and many of these formulas reference other cells which might in themselves also be formulas referencing other cells. This dependency chain is memorized and stored by Excel in a dependency tree. Excel uses this dependency tree to decide which cells to recalculate and which cells to ignore. For normal formula calculation, Excel always refers to this dependency chain. If you make one change to a cell, it updates its dependency tree.

Follow the steps below to speed up any of your slow-calculating workbooks.



1 Go to **Formulas / Name Manager**

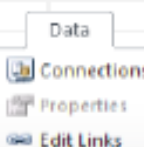
Do you have any #Ref or other invalid references? If yes sort on these and delete them completely.

Name Manager

Name	Value	Refers To
x	#REF!	=#REF!
y	#REF!	=#REF!
z	#REF!	=#REF!

2 Go to **Data / Connections**

- Is Edit Links clickable or greyed out? If greyed out, move to the next step.
- If not, click **Edit Links** and check which files are connected. Are these still valid? If not break the link now.



3 **Remove Invalid Formulas**

Do you have old formulas that return errors in cells which you no longer need? Remove these now.

4 **Clear unused cells from memory**

Go to each worksheet - Press **Ctrl + End**. If there are many columns and rows you don't need, highlight the last row/column in your sheet and press **Ctrl + Shift + End**. From **Home** tab, select **Clear / Clear All**. Save the Workbook. Close and when you re-open, the last cell will have changed to the last one in your sheet.

5 **Double check your formulas**

Are you using too many workarounds? Is there a better function that can be used? Are you referencing too many cells? Update these now, or schedule a time when you will revise this. Don't skip this step!



6 **Check your usage of Volatile functions and conditional formatting**

Do you have too many volatile functions in your sheet? Remove these and see if there is an impact. If yes, replace these with alternate non-volatile formulas. Is too much conditional formatting making your file slow? Revise these as necessary

List of Volatile Functions

- TODAY()
- NOW()
- OFFSET()
- RAND()
- RANDBETWEEN()
- INDIRECT()
- CELL()
- INFO()

DONE

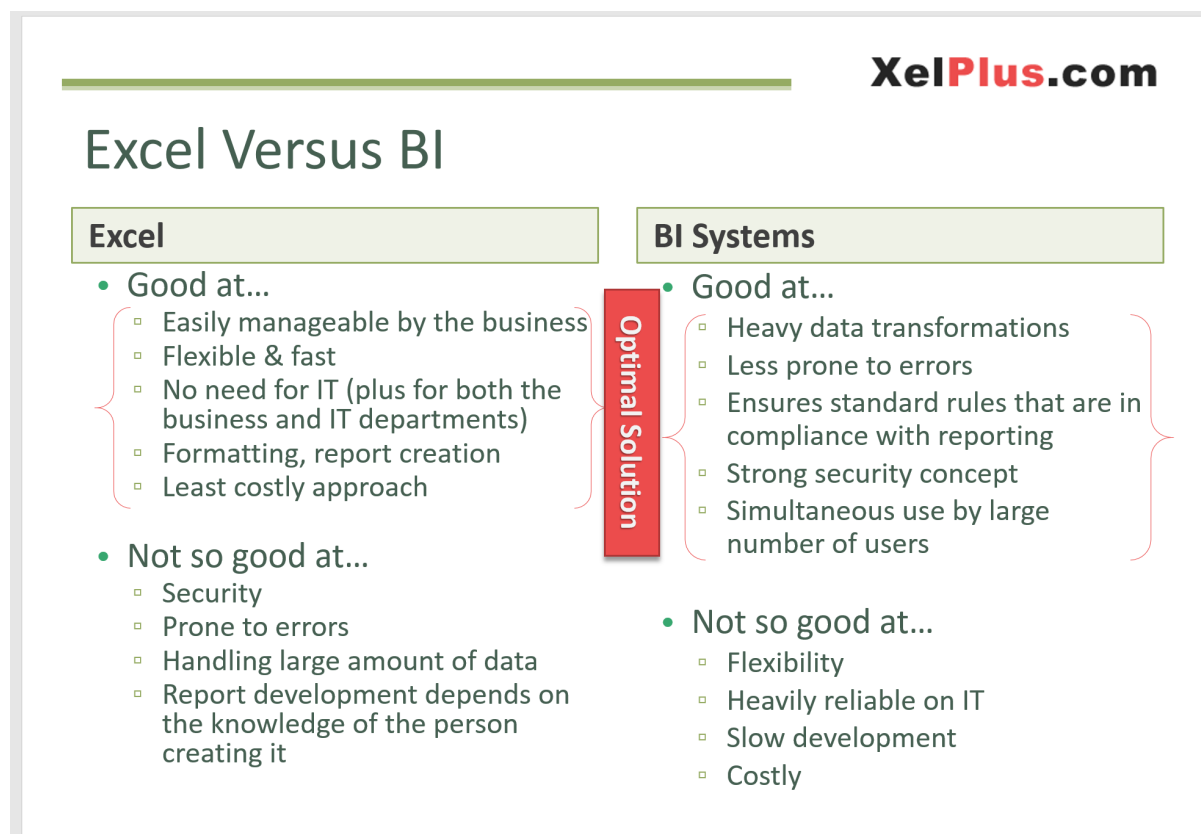
Infographic made in Excel :-)

XelPlus.com

Bonus Tip #12: Know the possibilities and the limitations

Just like with any program, the user should know both Excel's potential and limitations. Companies may need and use more complicated systems to handle databases and consolidations, but they still rely heavily on Excel as a data reporting and analysis tool.

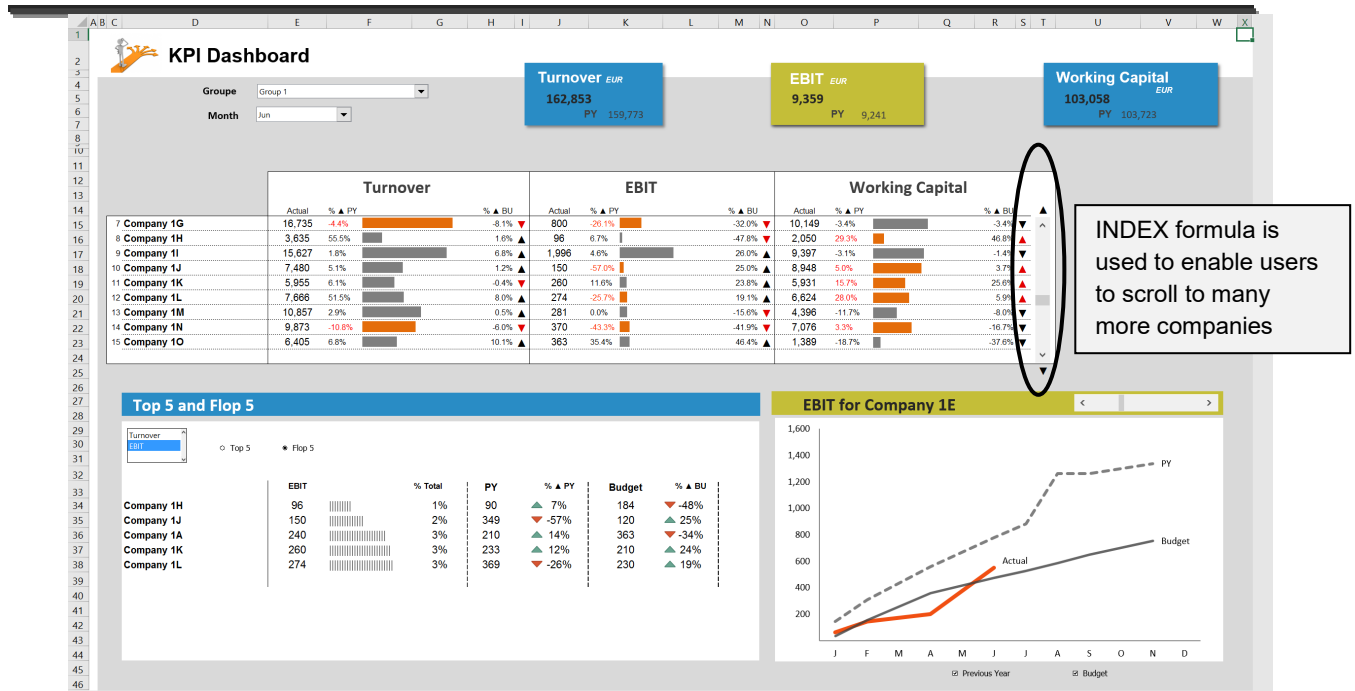
The below illustration shows my view on BI systems and Excel. The optimal solution is to use the best of both worlds. Use Excel where it's good at – flexible reports, graphics & analysis – and use BI where it's good at – handling large amount of data & users with a strong security concept.



The following examples demonstrate some uses of Excel. Below is a KPI dashboard created by retrieving data from Oracle's Hyperion Financial system. The dashboard is dynamic and works flawlessly with formulas. The form controls, such as the drop-down boxes, selection and option boxes, allow the users to switch between different views. The scroll bar on the right enables all companies to be viewed in the same window. Graphs are animated and dynamic so that the user can tick mark "budget" and/or "Previous year" for direct comparisons.

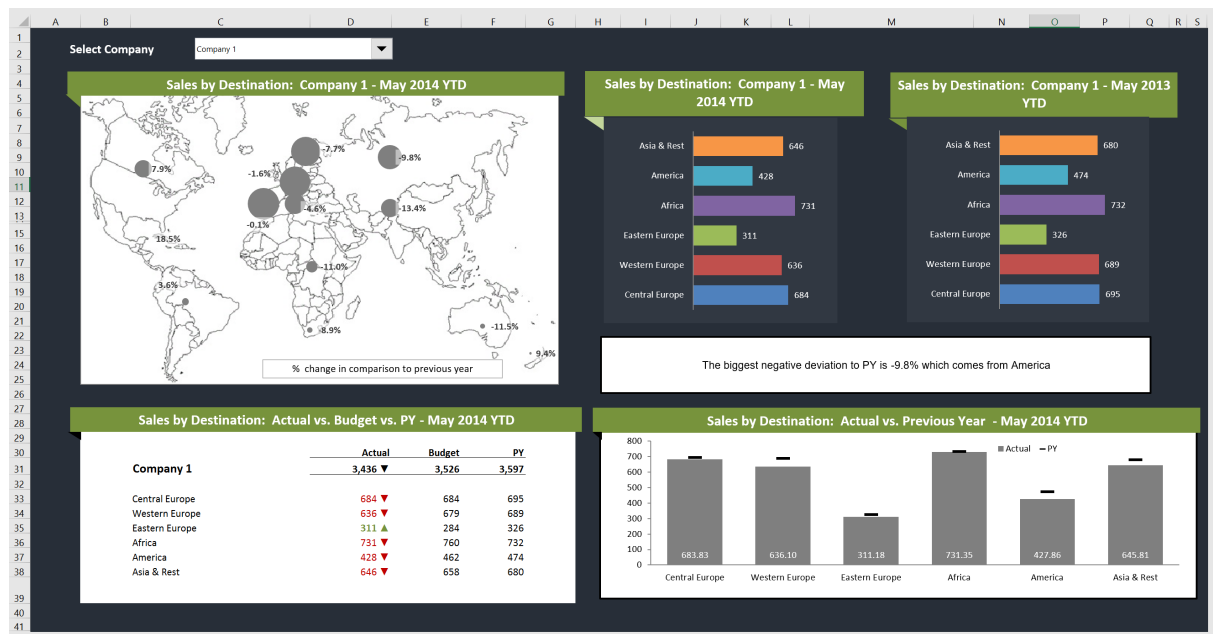
No macros or VBA programming are necessary.

Top Excel Formulas Every Analyst Should Know

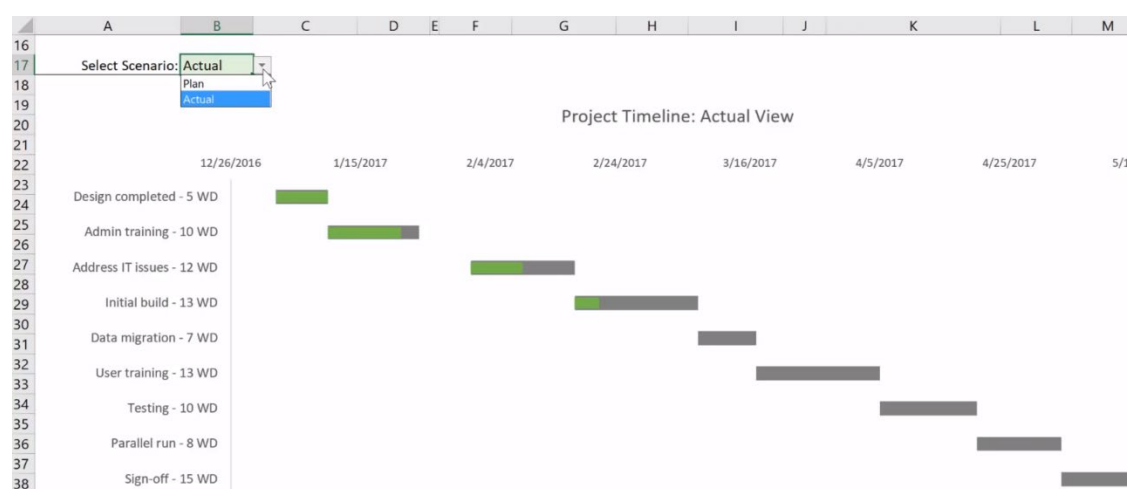


The second dashboard below shows the geographical sales revenue distribution of the selected company. A scatter chart is deftly used to dynamically visualize the company's annual sales. A combined graph is also used to compare the current year values to those from the previous year.

If you'd like to learn how to create impressive dashboards like these, check out my "Visually Effective Excel Dashboards" Course: <https://XeIPlus.com/Courses>



The next example is an Excel Template that can be used for project management purposes. It's fully dynamic; it uses formulas and some chart tricks to dynamically illustrate the project completion percentage and the current week.



This article shows you how to create Gantt charts from scratch:

<https://www.xelplus.com/quick-gantt-chart/>

You need to create a quick Gantt chart to visualize milestones? Check out Chapter 1 in the article.

You'd like to have a fully featured Gantt chart where you can switch between Plan and Actual and also keep an eye on task completion? Check out Chapter 2.

VBA can be used as well to create real standalone programs and tools that enable you do certain tasks. Learning some Visual Basic is a critical skill for any Excel power user.

With VBA, you can create sophisticated dashboards. You can create tools that map data from one database to another, you can also control other applications from Excel.

If you're interested to learn VBA, check out my bestselling VBA course "Unlock Excel VBA & Macros" course: <https://XelPlus.com/Courses>

PowerQuery, PowerPivot and PowerBI take Excel to a whole new level. Check out my resource page: <https://www.xelplus.com/resources/#power> if you'd like to learn these tools in a structured manner. I've listed a few resources which personally helped me with my learning.

My Courses

Check out my Courses

If you'd like to learn Excel in a structured manner, check out my bestselling Excel courses. This way you can learn anytime that fits your schedule best:

<https://www.XelPlus.com/Courses>



What people are saying about the courses:

Sara says: *I wanted to send you a message and let you know how much I enjoy and appreciate your class! I recently took a new job where one of my responsibilities is creating dashboards. I was incredibly discouraged about my depth of knowledge in this area and your class has completely turned that around. It's now my favorite part of the job because I feel like I'm actually learning, exploring new things, and contributing in a valuable way to my company. I've been able to tweak the things you've shown us in your class and apply them to our data and needs in a way I did not expect to be able to in such a short amount of time. I still have a long way to go but it's made it an exciting process instead of a discouraging one. You've made me feel like I can actually figure it out and accomplish amazing things in Excel!*

Jiyaad says: *Very detailed explanations, I took another highly rated (VBA) course for learning Macros and it covered maybe 10% of what this covers, so **literally 10x more knowledge** - which was all very important. As I progress through this course, there's so many good exercises I can follow along with and I find myself trying to solve ahead of time and just follow along where I get stuck. I'm really enjoying the course, and it's very comprehensive - thank you for the great course!*

Boby says: *Excellent presentation and easy to understand of an extremely useful information. I can only say thank you for the new knowledge that I have now gained in this field, which has been very useful in the improvement of my efficiency in this very highly competitive working environment. Giving five stars still does not do the justice for the quality of this course. Please keep up the good work and look forward to see more course programs from you.*

Melani says: *Thank you Leila for this very valuable informative training course. I now feel much more confident in my work.*

And many more...

Free Excel Tutorials

I hope you've found a lot of useful information in this free PDF document. If you'd like to learn more for free, check out my Excel Tutorials here: <https://www.xelplus.com/tutorials>

And if you're someone like me, who likes to have everything in one file, download the workbook at the bottom of this post: <https://www.xelplus.com/free-excel-tutorials/>



You'll find the list as well as the links to all the Excel tutorials I have so far. I'll be updating the workbook on a monthly basis.

More Resources

When I set out to learn something new, I don't just buy one book or one course, I learn from a variety of sources. Each person has their unique view and way of presenting. Each one inspires me in a different way. If you're serious about learning Excel, check out my list of favorite books and courses.

You'll find resources on data visualization, Power Query, Power BI and much more:

<https://xelplus.com/resources>