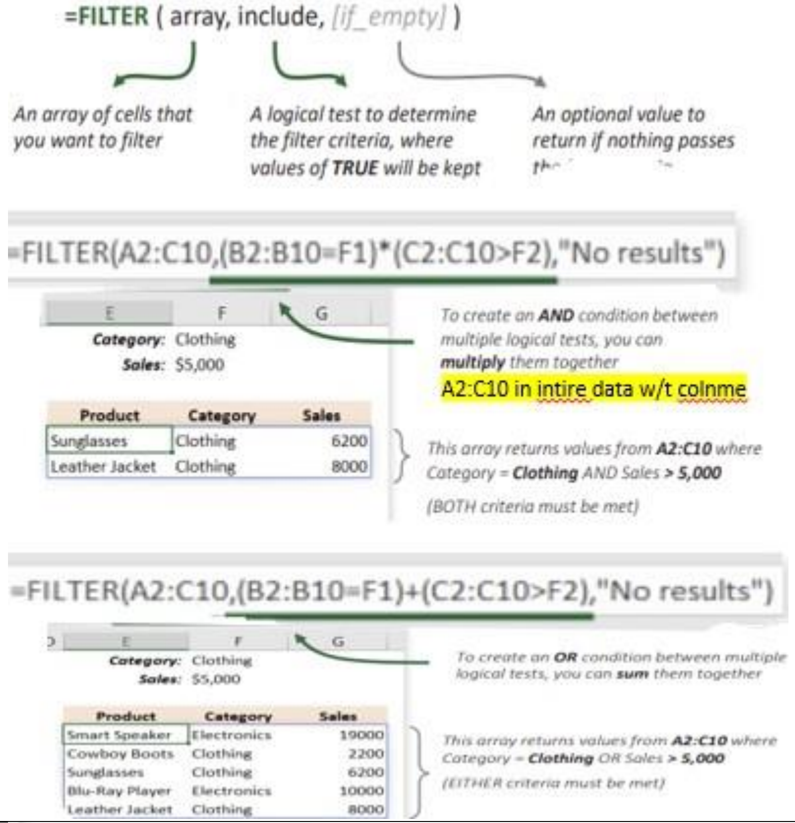


=FILTER (array, include, [if_empty])

An array of cells that you want to filter

*A logical test to determine the filter criteria, where values of **TRUE** will be kept*

An optional value to return if nothing passes th

=FILTER(A2:C10,(B2:B10=F1)*(C2:C10>F2),"No results")

E	F	G
Category:	Clothing	
Sales:	$5,000	

*To create an **AND** condition between multiple logical tests, you can **multiply** them together*

A2:C10 in intire data w/t colnme

Product	Category	Sales
Sunglasses	Clothing	6200
Leather Jacket	Clothing	8000

*This array returns values from **A2:C10** where Category = **Clothing** AND Sales **> 5,000***

(BOTH criteria must be met)

=FILTER(A2:C10,(B2:B10=F1)+(C2:C10>F2),"No results")

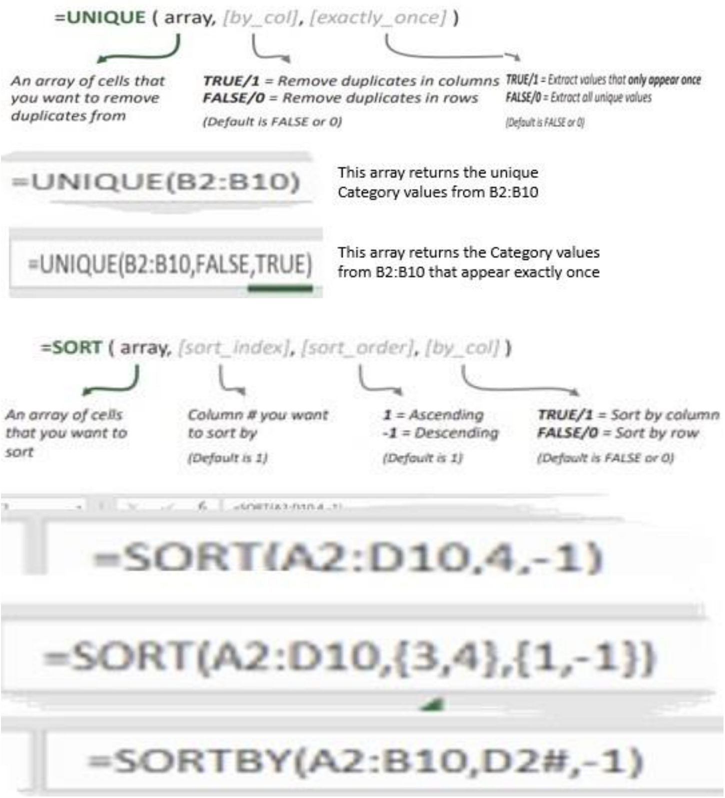
E	F	G
Category:	Clothing	
Sales:	$5,000	

*To create an **OR** condition between multiple logical tests, you can **sum** them together*

Product	Category	Sales
Smart Speaker	Electronics	19000
Cowboy Boots	Clothing	2200
Sunglasses	Clothing	6200
Blu-Ray Player	Electronics	10000
Leather Jacket	Clothing	8000

*This array returns values from **A2:C10** where Category = **Clothing** OR Sales **> 5,000***

(EITHER criteria must be met)



=UNIQUE (array, [by_col], [exactly_once])

An array of cells that you want to remove duplicates from

***TRUE/1** = Remove duplicates in columns*
***FALSE/0** = Remove duplicates in rows*
(Default is FALSE or 0)

TRUE/1 = Extract values that **only appear once**
FALSE/0 = Extract all unique values
(Default is FALSE or 0)

=UNIQUE(B2:B10)

This array returns the unique Category values from B2:B10

=UNIQUE(B2:B10,FALSE,TRUE)

This array returns the Category values from B2:B10 that appear exactly once

=SORT (array, [sort_index], [sort_order], [by_col])

An array of cells that you want to sort

Column # you want to sort by
(Default is 1)

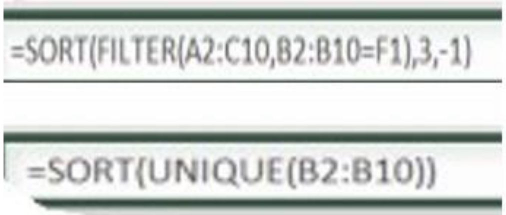
***1** = Ascending*
***-1** = Descending*
(Default is 1)

***TRUE/1** = Sort by column*
***FALSE/0** = Sort by row*
(Default is FALSE or 0)

=SORT(A2:D10,4,-1)

=SORT(A2:D10,{3,4},{1,-1})

=SORTBY(A2:B10,D2#,-1)

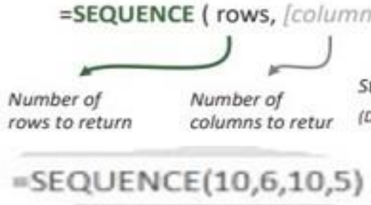


=SORT(FILTER(A2:C10,B2:B10=F1),3,-1)

Here we're combining SORT and FILTER to return an array of products in the Clothing category, sorted by Sales

=SORT(UNIQUE(B2:B10))

Here we're combining SORT and UNIQUE to return
an array of unique categories in ascending order



=SEQUENCE (rows, [columns], [start], [step])

Number of rows to return

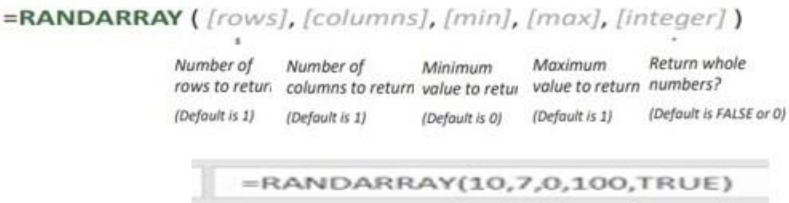
Number of columns to retur

Starting number
(Default is 1)

Increment between each number
(Default is 1)

=SEQUENCE(10,6,10,5)

Remember select cell u want to put sequent to the left is happen then put = se… and put value accordingly



=RANDARRAY ([rows], [columns], [min], [max], [integer])

Number of rows to retur
(Default is 1)

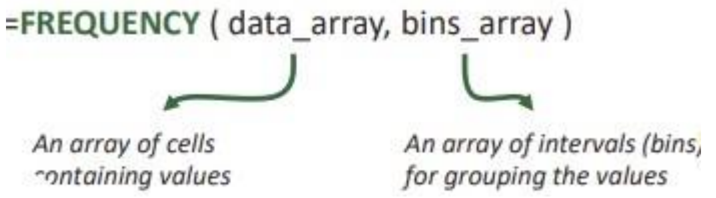
Number of columns to return
(Default is 1)

Minimum value to retui
(Default is 0)

Maximum value to return
(Default is 1)

Return whole numbers?
(Default is FALSE or 0)

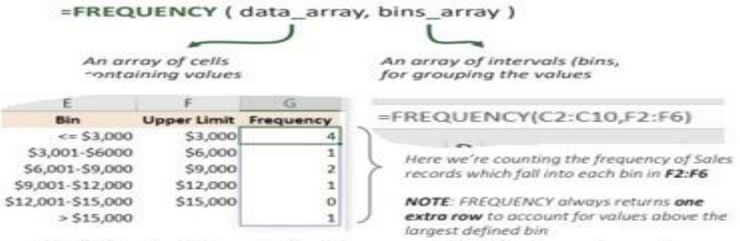
=RANDARRAY(10,7,0,100,TRUE)



=FREQUENCY (data_array, bins_array)

An array of cells ontaining values

An array of intervals (bins, for grouping the values



=FREQUENCY (data_array, bins_array)

An array of cells ontaining values

An array of intervals (bins, for grouping the values

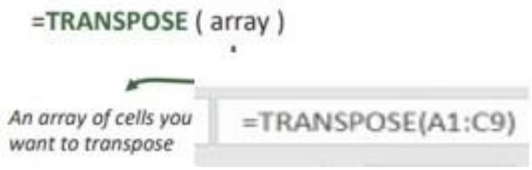
E	F	G
Bin	**Upper Limit**	**Frequency**
<= $3,000	$3,000	4
$3,001-$6000	$6,000	1
$6,001-$9,000	$9,000	2
$9,001-$12,000	$12,000	1
$12,001-$15,000	$15,000	0
> $15,000		1

=FREQUENCY(C2:C10,F2:F6)

*Here we're counting the frequency of Sales records which fall into each bin in **F2:F6***

***NOTE**: FREQUENCY always returns **one extra row** to account for values above the largest defined bin*

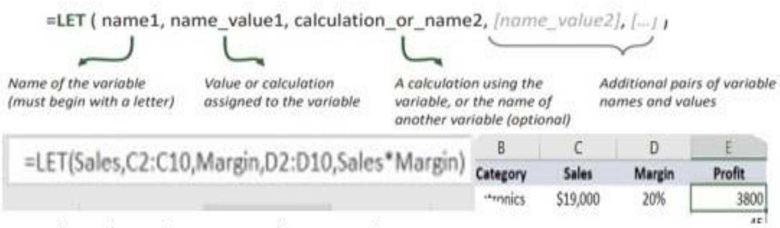
C2:C10 is the Colum to find frequency F2:F6 is the col we put on frequency that we want



=TRANSPOSE (array)

An array of cells you want to transpose

=TRANSPOSE(A1:C9)



=LET (name1, name_value1, calculation_or_name2, [name_value2], [...])

Name of the variable (must begin with a letter)

Value or calculation assigned to the variable

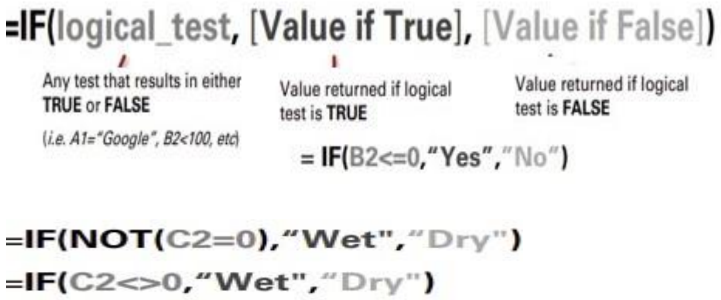
A calculation using the variable, or the name of another variable (optional)

Additional pairs of variable names and values

=LET(Sales,C2:C10,Margin,D2:D10,Sales*Margin)

B	C	D	E
Category	**Sales**	**Margin**	**Profit**
tronics	$19,000	20%	3800

Not that sale and margin are the same column name



=IF(logical_test, [Value if True], [Value if False])

Any test that results in either **TRUE** or **FALSE**
(i.e. A1="Google", B2<100, etc)

Value returned if logical test is **TRUE**

Value returned if logical test is **FALSE**

= IF(B2<=0,"Yes","No")

=IF(NOT(C2=0),"Wet","Dry")

=IF(C2<>0,"Wet","Dry")

= IF(B2<40,"COLD",IF(B2>80,"HOT","MILD"))

If temp<40, climate = "Cold", if temp>80, climate = "Hot", otherwise climate = "Mild"

=IF(OR(F2="Rain",F2="Snow"),"Wet","Dry")

Here we're categorizing conditions as "Wet" if the precipitation type equals "rain" OR "snow". otherwise Conditions = "Dry"

=IF(AND(D2="Yes",C2>0),"Snow",IF(AND(D2="No",C2>0),"Rain","None"))

If the temp is below freezing AND the amount of precipitation > 0, then Precip Type = "Snow", if the temp is above freezing AND the amount of precipitation >0, then Precip Type = "Rain", otherwise Precip Type = "None"

=IF(AND(D2="Yes",C2>0),"Snow",IF(AND(D2="No",C2>0),"Rain","None"))

If the temp is below freezing AND the amount of precipitation > 0, then Precip Type = "Snow", if the temp is above freezing AND the amount of precipitation >0, then Precip Type = "Rain", otherwise Precip Type = "None"

=IFERROR(value. value_if_error)

Formula or value that may or may not result in an error

Value returned in the case of an error

ISBLANK = *Checks whether the reference cell or value is blank*

ISNUMBER = *Checks whether the reference cell or value is numerical*

ISTEXT = *Checks whether the reference cell or value is a text string*

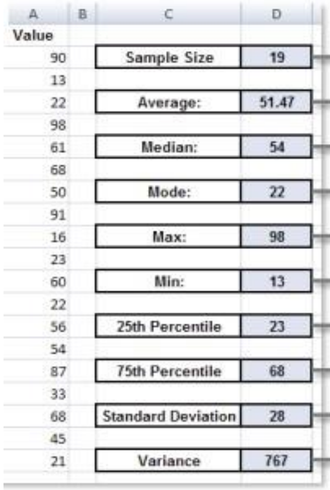
ISERROR = *Checks whether the reference cell or value returns an error*

ISEVEN = *Checks whether the reference cell or value is even*

ISODD = *Checks whether the reference cell or value is odd*

ISLOGICAL = *Checks whether the reference cell or value is a logical operator*

ISFORMULA = *Checks whether the reference cell or value is a formula*



A	B	C	D
Value			
90		Sample Size	19
13			
22		Average:	51.47
98			
61		Median:	54
68			
50		Mode:	22
91			
16		Max:	98
23			
60		Min:	13
22			
56		25th Percentile	23
54			
87		75th Percentile	68
33			
68		Standard Deviation	28
45			
21		Variance	767

=COUNT(A2:A20)

=AVERAGE(A2:A20)

=MEDIAN(A2:A20)

=MODE(A2:A20)

=MAX(A2:A20)

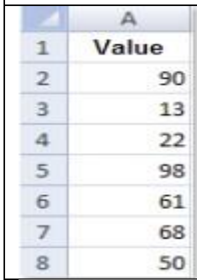
=MIN(A2:A20)

=PERCENTILE(A2:A20,.25)

=PERCENTILE(A2:A20,.75)

=STDEV(A2:A20)

=VAR(A2:A20)



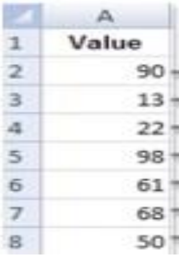
	A
1	Value
2	90
3	13
4	22
5	98
6	61
7	68
8	50

LARGE(A2:A8,2) = 90

(the 2nd largest number in the array is 90)

SMALL(A2:A8,3) = 50

(the 3rd smallest number in the array is 50)



	A
1	Value
2	90
3	13
4	22
5	98
6	61
7	68
8	50

This just give rank of value

RANK(A2,A2:A8) = 2

RANK(A3,A2:A8) = 7 *(lowest)*

RANK(A4,A2:A8) = 6

RANK(A5,A2:A8) = 1 *(highest)*

RANK(A6,A2:A8) = 4

RANK(A7,A2:A8) = 3

RANK(A8,A2:A8) = 5

Quantity of goods sold at Shaws:

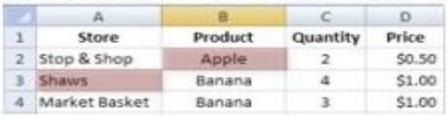
SUMPRODUCT((A2:A17="Shaws")*C2:C17) = 16

Total revenue from Shaws:

SUMPRODUCT((A2:A17="Shaws")*C2:C17*D2:D17) = $21.80

Revenue from apples sold at Shaws:

SUMPRODUCT((A2:A17="Shaws")*(B2:B17="Apple")*C2:C17*D2:D17) = $0.50



	A	B	C	D
1	Store	Product	Quantity	Price
2	Stop & Shop	Apple	2	$0.50
3	Shaws	Banana	4	$1.00
4	Market Basket	Banana	3	$1.00

=COUNTIF(range, criteria)

=SUMIF(range, criteria, sum_range)

=AVERAGEIF(range, criteria, average_range)

COUNTIF(B2:B20,22) = 2

SUMIF(A2:A20,"Ryan",B2:B20) = 190

SUMIF(A2:A20,"<>Tim",B2:B20) = 702

AVERAGEIF(A2:A20,"Maria",B2:B20) = 45.75

Note ryan,tim,maria are in that column of a2:a20 then give sum om b

=COUNTIFS(criteria_range1, criteria1, criteria_range2 , criteria2...)

=SUMIFS(sum_range, criteria_range1, criteria1, criteria_range2 , criteria2...)

=AVERAGEIFS(average_range, criteria_range1, criteria1, criteria_range2 , criteria2...)

COUNTIFS(B2:B13,"Search", D2:D13,">200") = 3

SUMIFS(D2:D13, A2:A13,"Feb",B2:B13,"Display") = 734

AVERAGEIFS(D2:D13, A2:A13,"Jan",C2:C13,"MSN") = 263

Another way of use

D2=HLOOKUP(A2, $H$1:$L$2, 2, 0)

A2 is the cell in the vertical table we want to fill and that have looking value

This is the table with horizontal value we want

We want the stuff from the second col in the horizontal table

COLUMN(C10) = 3

Colunms give nbr of col reference nbr c is nbr 3

COLUMNS(A10:D15) = 4

Colunms give nbr of col btwen a10 and d15

ROW(C10) = 10

Colunms give nbr of row reference look c10 is on row nbr 10

ROWS(A10:D15) = 6

Colunms give nbr of row btwen a10 and d15

The INDEX function returns the value of a specific cell within an array

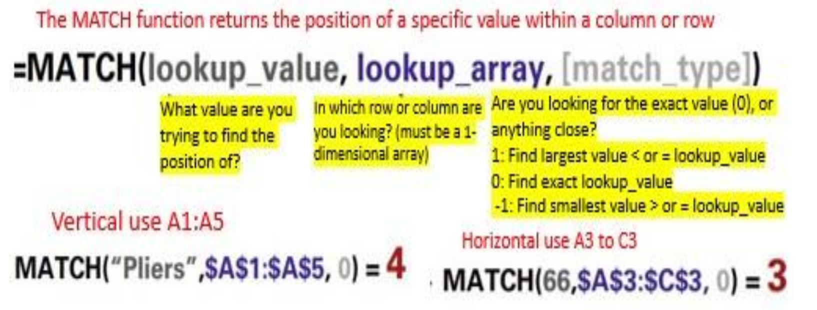
=INDEX(array, row_num, column_num)

What range of cells are you looking at?

How many rows down is the value you want?

How many columns over is the value you want?

INDEX($A$1:$C$5, 5, 3) = 234



The MATCH function returns the position of a specific value within a column or row

=MATCH(lookup_value, lookup_array, [match_type])

What value are you trying to find the position of?

In which row or column are you looking? (must be a 1-dimensional array)

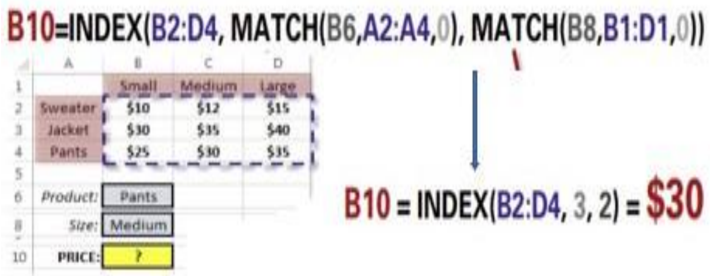
Are you looking for the exact value (0), or anything close?
1: Find largest value < or = lookup_value
0: Find exact lookup_value
-1: Find smallest value > or = lookup_value

Vertical use A1:A5

MATCH("Pliers",$A$1:$A$5, 0) = 4

Horizontal use A3 to C3

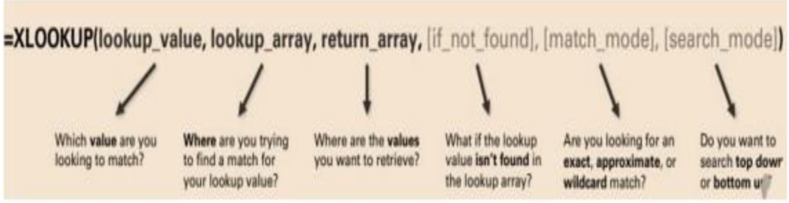
MATCH(66,$A$3:$C$3, 0) = 3



B10=INDEX(B2:D4, MATCH(B6,A2:A4,0), MATCH(B8,B1:D1,0))

	A	B	C	D
1		Small	Medium	Large
2	Sweater	$10	$12	$15
3	Jacket	$30	$35	$40
4	Pants	$25	$30	$35
5				
6	Product:	Pants		
8	Size:	Medium		
10	PRICE:	?		

B10 = INDEX(B2:D4, 3, 2) = $30



=XLOOKUP(lookup_value, lookup_array, return_array, [if_not_found], [match_mode], [search_mode])

Which **value** are you looking to match?

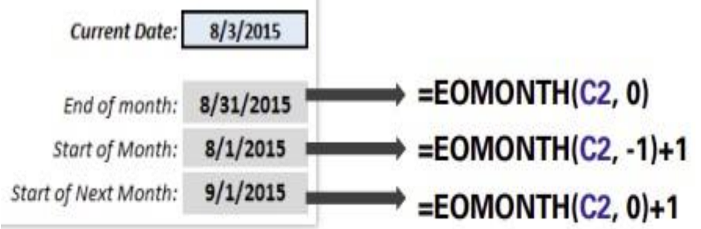
Where are you trying to find a match for your lookup value?

Where are the **values** you want to retrieve?

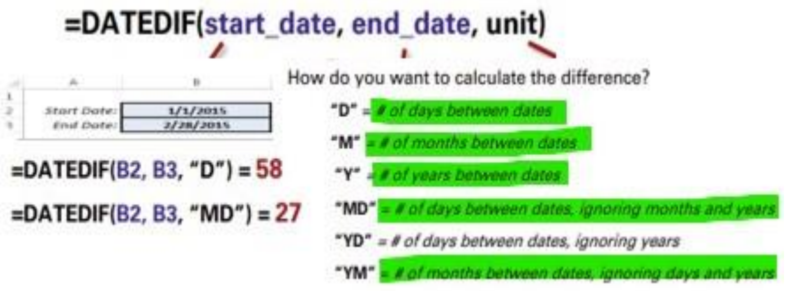
What if the lookup value **isn't found** in the lookup array?

Are you looking for an **exact, approximate**, or **wildcard** match?

Do you want to search **top dowr** or **bottom u**



Current Date:	8/3/2015	
End of month:	8/31/2015	=EOMONTH(C2, 0)
Start of Month:	8/1/2015	=EOMONTH(C2, -1)+1
Start of Next Month:	9/1/2015	=EOMONTH(C2, 0)+1



=DATEDIF(start_date, end_date, unit)

	A	B
1		
2	Start Date:	1/1/2015
3	End Date:	2/28/2015

=DATEDIF(B2, B3, "D") = 58

=DATEDIF(B2, B3, "MD") = 27

How do you want to calculate the difference?

"D" = # of days between dates

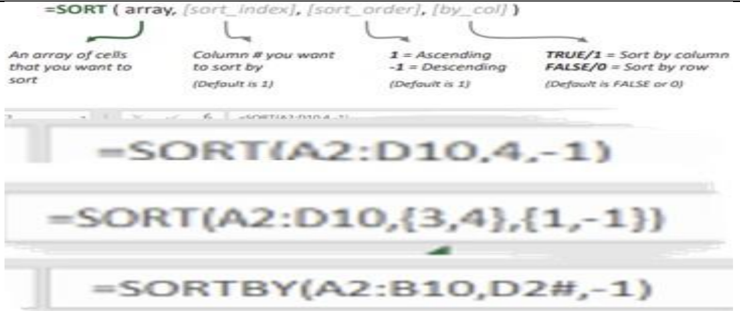
"M" = # of months between dates

"Y" = # of years between dates

"MD" = # of days between dates, ignoring months and years

"YD" = # of days between dates, ignoring years

"YM" = # of months between dates, ignoring days and years



=SORT (array, [sort_index], [sort_order], [by_col])

An array of cells that you want to sort

Column # you want to sort by
(Default is 1)

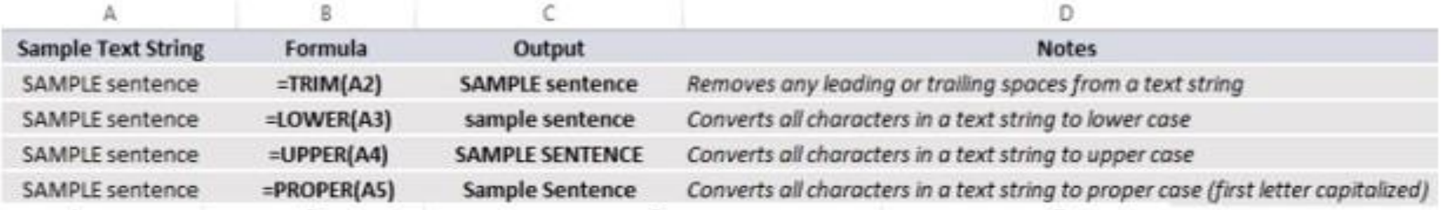
1 = Ascending
-1 = Descending
(Default is 1)

TRUE/1 = Sort by column
FALSE/0 = Sort by row
(Default is FALSE or 0)

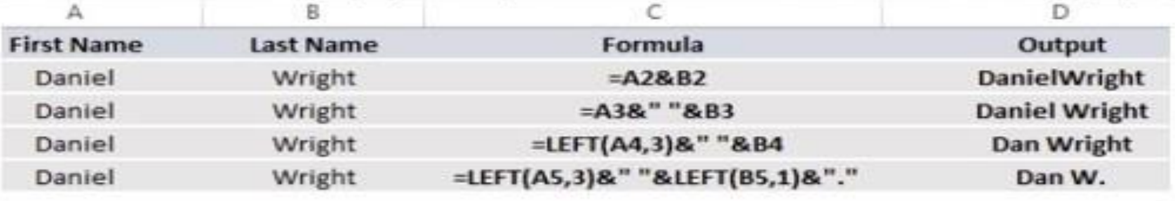
=SORT(A2:D10,4,-1)

=SORT(A2:D10,{3,4},{1,-1})

=SORTBY(A2:B10,D2#,-1)

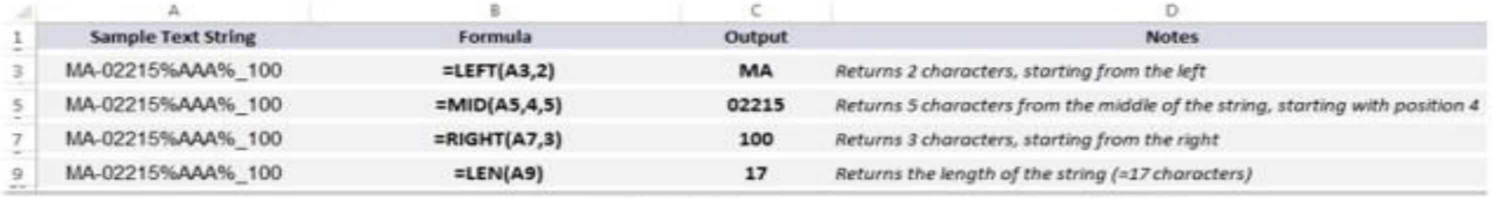


A	B	C	D
Sample Text String	Formula	Output	Notes
SAMPLE sentence	=TRIM(A2)	SAMPLE sentence	Removes any leading or trailing spaces from a text string
SAMPLE sentence	=LOWER(A3)	sample sentence	Converts all characters in a text string to lower case
SAMPLE sentence	=UPPER(A4)	SAMPLE SENTENCE	Converts all characters in a text string to upper case
SAMPLE sentence	=PROPER(A5)	Sample Sentence	Converts all characters in a text string to proper case (first letter capitalized)

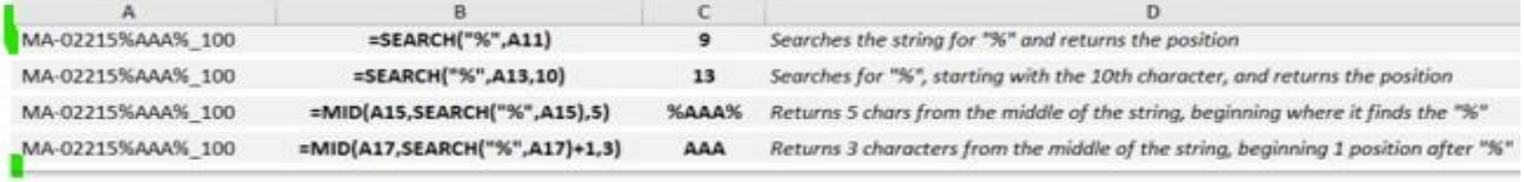


A	B	C	D
First Name	Last Name	Formula	Output
Daniel	Wright	=A2&B2	DanielWright
Daniel	Wright	=A3&" "&B3	Daniel Wright
Daniel	Wright	=LEFT(A4,3)&" "&B4	Dan Wright
Daniel	Wright	=LEFT(A5,3)&" "&LEFT(B5,1)&"."	Dan W.

concate
nate



	A	B	C	D
1	Sample Text String	Formula	Output	Notes
3	MA-02215%AAA%_100	=LEFT(A3,2)	MA	Returns 2 characters, starting from the left
5	MA-02215%AAA%_100	=MID(A5,4,5)	02215	Returns 5 characters from the middle of the string, starting with position 4
7	MA-02215%AAA%_100	=RIGHT(A7,3)	100	Returns 3 characters, starting from the right
9	MA-02215%AAA%_100	=LEN(A9)	17	Returns the length of the string (=17 characters)



A	B	C	D
MA-02215%AAA%_100	=SEARCH("%",A11)	9	Searches the string for "%" and returns the position
MA-02215%AAA%_100	=SEARCH("%",A13,10)	13	Searches for "%", starting with the 10th character, and returns the position
MA-02215%AAA%_100	=MID(A15,SEARCH("%",A15),5)	%AAA%	Returns 5 chars from the middle of the string, beginning where it finds the "%"
MA-02215%AAA%_100	=MID(A17,SEARCH("%",A17)+1,3)	AAA	Returns 3 characters from the middle of the string, beginning 1 position after "%"