



Note :

All course guides are in PDF format and you must have an appropriate PDF reader installed on your computer to open and print these course guides. If you do not already have one, you can find and download a free copy of Acrobat Reader at: <https://get.adobe.com/reader/>



Course Reference Handout

MICROSOFT EXCEL - POWER USER

Contact us at Learn IT!
(415) 693-0250
www.learnit.com

Using the Index and Match Functions:

Many have heard of, if not used, a VLOOKUP function to retrieve data from a list or table based on a single key data value. However, the VLOOKUP function has limitations, which can be overcome by retrieving data using a different set of functions called INDEX & MATCH.

Limitations to VLOOKUP That Can Be Overcome with INDEX & MATCH:

- A VLOOKUP can only look for a single data value (such as a Part Number or a Customer ID) in a table or list in which it can retrieve other data values from the same record.
 - INDEX & MATCH, when used as an array function, can look up multiple data values.
- The lookup value that VLOOKUP searches for in a table array must be located to the left of any columns of data you want to retrieve.
 - With INDEX & MATCH, the lookup value can be in any column of the table array—to the left or right of the actual data being retrieved.

Looking at the INDEX Function Independently:

The INDEX function arguments:

=INDEX(Array, Row Number)

Purpose:

The function returns a data value from an array of values from a specific row within the array.

Array:

The range of data from which you want to return a value.

Row Number:

The row number of the array that contains the actual data value you want returned.

Example 1:

The INDEX function will retrieve the data value in row 3 from the array C2:C7 (Product).

Example 1		
Lookup	Function	Result
1003	=INDEX(C2:C7,3)	Shirts

Example 2		
Lookup	Function	Result
AZ	=INDEX(A2:A7,2)	1002

Example 2:

The INDEX function will retrieve the data value in row 2 from the array A2:A7 (ID). This is an example of data retrieval that the VLOOKUP is not capable of doing.

The Problem:

By itself, INDEX is not very efficient or effective. This is because the row number containing the actual value to be returned has to be known for this function. For example, in Example 1, we would have to know that ID 1003 is in row 3 of the array by looking it up ourselves and entering in "3" for the row number argument of the function. Looking up the value manually defeats the whole purpose of the function! We need another function that looks up the row number for our criteria automatically... and that's where MATCH comes in!

Looking at the MATCH Function Independently:

The MATCH function arguments:

=MATCH(Lookup Value, Lookup Array, Match Type)

Purpose:

The function looks for a value in an array, and then returns the row number of the array in which the value was found.

Lookup Value:

The value you want MATCH to look for.

Lookup Array:

The range of cells you want MATCH to search as it looks for the Lookup Value.

Match Type:

This determines if MATCH looks for an exact match or a closest possible match (enter "0" for exact matches).

Example 1:

The MATCH function will look for the value in cell A11 (1003) in the array A2:A7 and then return the row number of the array in which the value was found.

Example 1		
Lookup	Function	Result
1003	=MATCH(A11,A2:A7,0)	3

	A	B	C
1	ID	Region	Product
2	1001	CA	Hats
3	1002	AZ	Shoes
4	1003	OR	Shirts
5	1004	WA	Pants
6	1005	NV	Shorts
7	1006	TX	Belts

INDEX and MATCH
Sample Data

Using INDEX and MATCH together:

Goal:

One can use the INDEX function to retrieve a data value from an array, where MATCH is used to look up the key value in an array and determine the row number the value is in. MATCH will then pass this information on to the row number argument of the INDEX function.

The Nested Functions:

=INDEX(Array, MATCH(Lookup Value, Lookup Array, Match Type))

When nesting, or merging, the two functions, MATCH is used as the row number argument of the INDEX function.

	A	B	C
1	ID	Region	Product
2	1001	CA	Hats
3	1002	AZ	Shoes
4	1003	OR	Shirts
5	1004	WA	Pants
6	1005	NV	Shorts
7	1006	TX	Belts
8			
9			
10			
11	1003		

Array Formulas:

If you've done even a little programming, you've probably run across the term array. For our purposes, an array is just a collection of items. In Excel, those items can reside in a single row (called a one-dimensional, horizontal array), a column (a one-dimensional, vertical array), or multiple rows and columns (a two-dimensional array). You can't create three-dimensional arrays or array formulas in Excel.

An array formula is a formula that can perform multiple calculations on one or more of the items in an array. Array formulas can return either multiple results or a single result. For example, you can place an array formula in a range of cells and calculate a column or row of subtotals. You can also place a formula in a single cell and calculate a single amount. An array formula that resides in multiple cells is called (logically enough) a multi-cell formula, and an array formula that resides in a single cell is called a single-cell formula.

Entering an Array Formula:

- When creating an array formula or function, in order to obtain the correct result, you must enter the calculation(s) using a combination of CTRL + SHIFT + ENTER.
- When referencing multiple arrays of data in an array function, use the Concatenation symbol (&) to combine the arrays in the function.
- Note that you cannot enter a multi-cell array formula in a table. Tables are meant to automatically copy formula results for you, and therefore will not support multi-cell array formulas.
- Array formulas can be identified by the braces that surround them, such as in the following example: {=A2:A5*B2:B5}.

Benefits of an Array Formula:

Consistency:

If you click any of the cells within a range that you have inserted array formulas, you see the same formula. That consistency can help ensure greater accuracy.

Safety:

You cannot overwrite part of a multi-cell array formula. For example, click a cell within an array containing an array formula and press DELETE. Excel will not allow you to delete the formula. You have to either select the entire range of cells and change the formula for the entire array, or leave it as is.

Smaller File Sizes:

You can often use a single array formula instead of several intermediate formulas, which takes up less memory in the long run.

Array Formula Example Data

	A	B	C	D	E
1	HRS	HOURLY RATE	GROSS PAY		
2	35.5	\$12.50	\$443.75		
3	35.5	\$13.30	\$472.15		
4	42	\$16.75	\$703.50		
5	40	\$8.75	\$350.00		
6	40	\$12.60	\$504.00		
7	35	\$24.00	\$840.00		
8	35	\$12.10	\$423.50		
9	40	\$21.50	\$860.00		

Entering a Multi-Cell Array Formula:

- Using the example data seen in the image to the right, the Gross Pay amounts were all created at once using a multi-cell array formula. Here's how:
 - Select all cells you want results to appear within. In this example, cell range C2:C9 were selected.
 - Start writing the formula by typing an equals (=) sign.
 - Select A2:A9. Then type the asterisk (*) symbol (for multiplication in this example), and then select the range B2:B9.
 - To enter the array formula, press CTRL + SHIFT + ENTER. All of the cells will now contain the same, multi-cell array formula.

Entering a Single-Cell Array Function:

- Using the same example data, you could use a single cell array function to calculate the SUM total of all HRS multiplied by their respective HOURLY RATE.
 - Click the cell you want the SUM total to appear in.
 - Type "=SUM", then select the range A2:A9, type the asterisk * symbol, and then select the range B2:B9.
 - Press CTRL + SHIFT + ENTER to enter the sum total of all the multiplied values.

Single Cell Array Function fx {=SUM(A2:A9*B2:B9)}

Building Dynamic Ranges Using the Offset Function:

One of the great things about using formatted tables in Excel is that the table automatically expands as new data records are added. This allows you to dynamically reference a table's data by name in other Excel tools, such as formulas, Pivot Tables, and charts. When new data is added to the table, the other Excel tools can read the new data, meaning you will not have to modify a predefined cell range that had been used as a data source.

However, there may be instances when you need to reference data in a dynamic way, and using a table is simply not an option. In these cases, you can create a Dynamic Named Range, which relies on the OFFSET function.

Inserting Names:

- By using Named Ranges, you can reference a single cell or a range of cells by name. You can also use formulas and functions to provide the source data that the Named Range refers to.
- To create a new Named Range
 - Click the Formulas tab of the Ribbon, and in the Define Name group, click the Define Name button.
 - Type the new name in the Name: box, define the Scope of the range, and use a cell range or a formula to determine the values the name refers to in the Refers to: Area.

Using the OFFSET Function:

- Purpose: to reference a single cell or a range of cells, which may start a certain number of rows and/or columns away from the initial cell reference, and may consist of a single cell or a range of cells consisting of multiple rows and/or columns of data
- =OFFSET(Reference, Rows, Cols, Height, Width)
 - Reference = the initial cell reference, or "anchor" cell, from which to start
 - Rows = the number of cells up or down from the anchor cell reference you want to begin referencing. Use positive numbers to move down, and negative numbers to move up.
 - Cols = the number of cells to the right or to the left from the anchor cell reference you want to begin referencing. Use positive numbers to move right and negative numbers to move left.
 - Height = how many rows tall the reference should be
- Often times the COUNT or COUNTA functions may be used to count how many data values are present within a range and to set the Height argument to equal that same amount.
 - Width = how many columns wide the reference should be.

Example:

Using the image of the range to the right, =OFFSET(A1,3,1,3,1) would reference the range B4:B6.

Start with cell A1. Move 3 cells down and 1 cell to the right, and select a range 3 rows deep and 1 column wide.

	A	B
1	Name	Units
2	JAN	130
3	FEB	45
4	MAR	55
5	APR	90
6	MAY	140
7	JUN	79
8	JUL	90
9	AUG	220

Creating a Named Range with OFFSET:

- Goal: We want to dynamically reference a range of month names for use as labels in a chart. As months are added or removed, we want the named range to automatically update within the chart.
- Create a new Name by clicking the Define Name button (Formulas tab of the Ribbon) and type in "Months" for the name.
- In the Refers to: area, insert an OFFSET that will start with cell A2 and capture any cells below that may also contain references to months. Though you may have absolute cell references and references to sheet names, the resulting function should look similar to this: =OFFSET(A2,0,0,COUNTA(A:A),1).
- In the above example, the reference cell, or "anchor" cell, is A2.
- The reference starts with that actual cell. Because ROWS & COLS are both set to 0, our starting point is not adjusted a certain number of rows or columns away from the initial reference cell.
- The height of the range is determined by the result of the COUNTA function counting data values in all of column A.
- The width of the range is 1 column.

Conditional Formatting Formulas:

Though there are many fantastic built in Conditional Formatting options available in Excel, there may be times when it will be necessary to create your own custom Conditional Formatting formulas.

Creating a Conditional Formatting Formula:

- Click the Conditional Formatting button (Home tab of the Ribbon) and then select New Rule.
- In the New Formatting Rule window, select the Use a formula to determine which cells to format option.
- Type in the formula and select the formatting to apply when the formula is true.

	A	B	C	D	E	F	G
1	Rec #	Name	Type Code	Vendor Code	Price		Enter Vendor Code:
2	1	Biscotto al Nero	D	301	\$2.15		430
3	2	Ciocolacino	B	430	\$2.95		
4	3	Oranciata	B	406	\$1.95		
5	4	Cappuccino	B	430	\$2.50		
6	5	Caffe Americana - Decaf.	B	430	\$1.50		
7	6	Pizzetta	M	395	\$5.95		

Applying a Conditional Formatting Formula to a Range of Cells:

- You can apply the same Conditional Formatting rules using a formula with multiple rows and columns of data by first selecting the entire range of cells you want to format and then creating a single Conditional Formatting formula.
- The trick, however, is knowing that Excel will apply the conditional format to the first cell and then automatically copy the Conditional Formatting formula to all other cells. This means you have to use a combination of absolute and relative cell references, so that when the conditional format is copied, the cells referenced in the Conditional Formatting formula will adjust accordingly.
- Using the illustration to the right, cell range A2:E7 was selected and formatted so that each record with a Vendor Code that matched the code in cell G2 appeared with a fill color.
- The Conditional Formatting formula created and applied to all selected cells at once, starting with the Vendor Code in row 2, is as follows: =\$D2=\$G\$2. (Note how the reference to D2 will adjust down to other rows but not across to other columns).