

Excel Yourself 2018

By Neale Blackwood

www.a4accounting.com.au

a4@iinet.net.au



A4 ACCOUNTING
EXCEL CONSULTING, TRAINING AND WEBINARS

Contents

Introduction	3
February 2018 – Validations	4
March 2018 – Identify Current Row	8
April 2018 – Centralised Logic.....	11
November 2018 – Slope Charts	14

Introduction

This session will cover some of my Excel Yourself articles that were first published in the INTHEBLACK magazine during 2018. We will examine four of the 11 articles in the hour allotted. This is my seventh webinar covering my INTHEBLACK articles. The other six webinar recordings and materials are available for free on my website.

The companion Excel sheets are arranged in date order starting with the February article on the left and working across to the right to finish with the November article. The sheet tabs are grouped and coloured. There is an INDEX sheet to assist in navigation.

The content of this manual has been re-written and adapted from the original articles and includes some extra content.

INTHEBLACK Website

Note: I record a companion video for each month's article. You can see them all at this page

<https://www.intheblack.com/topics/excel>

Some of the articles also include a companion file.

New Excel Version

Excel 2019 has been released and will be rolled out progressively over the coming months to the subscription version users.

February 2018 – Validations

When building models it is common to include validations to ensure that values that should balance do balance. There are a number of different ways to carry out validations and we will go through some of them.

Typically validations are done close to the ranges that they are checking. This can make it difficult and time consuming to review all of the validations in a file.

I recommend including a centralised validation sheet where you bring together all of the validations into a single sheet to allow you to identify if there is a problem in the file.

Calculating differences

In its simplest form validation may simply be getting the difference between two numbers. This difference should be zero. One of the issues when just using a difference is that a validation may be out in one month (positive value) and corrected in another month (negative value) and so if you add up the whole range of differences they may equal zero. You need to be able to identify if there is a validation issue in any month.

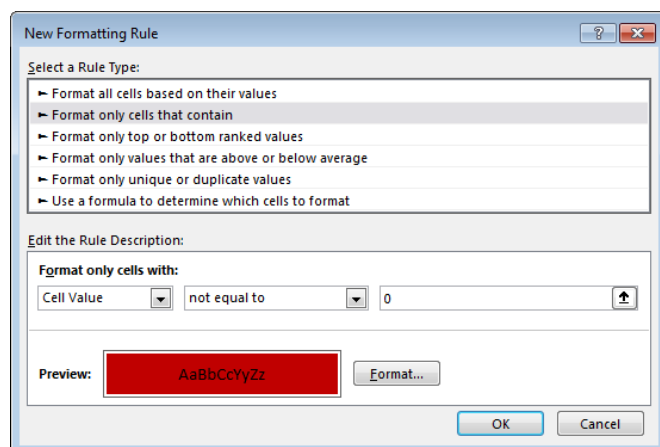
Using Conditional Formats

It is common to use conditional formats with validations so that you can identify cells that are invalid. Typically you only need to use a format to identify invalid cells rather than valid cells.

A standard conditional format to check a value equals zero is shown on the right.

This highlights the cell red if the value isn't zero.

Note: that if you are comparing to 0 but you have a format that doesn't show decimal places then it may display as zero but it might be out by cents and be shown as invalid.



Logical tests

Logical tests are typically used in conjunction with the IF function and a return TRUE or FALSE. These calculations are fast calculations and I use them in most of my validations. Many people are unaware that a cell can contain a logical test formula. A logical test formula returns TRUE or FALSE. I tend to display TRUE if a validation is correct and FALSE if it isn't. I use these in conjunction with cells that contain differences so that you also know the value you are looking for.

In its simplest form a logical test can be something like

=A1=B1

Where we are comparing cell A1 to cell B1. This formula will return TRUE if both cells are the same and FALSE if there is a difference.

Rounding issues

One of the common issues that you face when doing validations is created by rounding. Many files contain some form of rounding and this can affect the validations. When you are validating values with rounded numbers you need to have a tolerance factor, typically between -1 and 1.

ABS Function

The ABS function simplifies using a tolerance factor. ABS stands for absolute and it is a mathematical concept that calculates a number's distance from zero. In practice this means that it removes negatives and leaves positive values unchanged.

Hence, we can create a simple formula to ensure the difference between two values is between -1 and 1.

```
=ABS(A1-B1)<=1
```

This formula will return TRUE if the values in A1 and B1 are within one of each other and FALSE if they're not.

Since it is more common to use the IF function when working with logical test if you want to convert the above formula into a text result you could use a formula like this

```
=IF(ABS(A1-B1)<=1,"OK","Error")
```

This formula will display the text OK if the two values are within one of each other. The word Error will be displayed if they aren't.

If you have already used a row to identify differences you can include another row to include the logical tests which could use the ABS function.

Overall validation

If you have used a centralised validation sheet to bring together all of the TRUE and FALSE results it is a simple process to review all of the cells in one calculation.

Assuming that the TRUE and FALSE validations are in the range B2 to Z100 then the following formula will identify how many validation errors are in the range.

```
=COUNTIF(B2:Z100,FALSE)
```

This formula counts how many validation errors they are. **Note** that the word FALSE is not in quotation marks. Both TRUE and FALSE are keywords within Excel and do not require quotation marks.

You can modify this formula to display TRUE and FALSE by comparing the result of the COUNTIF function with zero. Like this

```
=COUNTIF(B2:Z100,FALSE)=0
```

If you want to modify the above formula to work with the IF function you could use something like.

```
=IF(COUNTIF(B2:Z100,FALSE)=0,"OK","Error")
```

Which displays OK if no errors are found and Error if there are any errors.

Solving rounding problems

It is very common when you are using rounding and creating a report that adds up to a total that you will experience rounding issues with the total.

You either have to round up or round down. If most of the values in the range are rounded up or down then when you add up the total it won't match the correct total.

The two issues that you will find with totals are

1. The sum of the rounded values doesn't match the rounded total – see below.

	A	B	C	D
1	Description	Raw Value	Rounded	Formula in Column C
2	Cash	2,345.67	2,346.00	=ROUND(B2,0)
3	Bank	269,809.56	269,810.00	=ROUND(B3,0)
4	Stock	679,658.54	679,659.00	=ROUND(B4,0)
5	Trade Debtors	130,456.78	130,457.00	=ROUND(B5,0)
6	Other Debtors	34,567.89	34,568.00	=ROUND(B6,0)
7	Prepayments	7,890.90	7,891.00	=ROUND(B7,0)
8	Total Current Assets	1,124,729.34	1,124,731.00	=SUM(C2:C7)

2. The total figure of the rounded values is not the sum of the rounded values - see below.

	A	B	C	D
1	Description	Raw Value	Rounded	Formula in Column C
2	Cash	2,345.67	2,346.00	=ROUND(B2,0)
3	Bank	269,809.56	269,810.00	=ROUND(B3,0)
4	Stock	679,658.54	679,659.00	=ROUND(B4,0)
5	Trade Debtors	130,456.78	130,457.00	=ROUND(B5,0)
6	Other Debtors	34,567.89	34,568.00	=ROUND(B6,0)
7	Prepayments	7,890.90	7,891.00	=ROUND(B7,0)
8	Total Current Assets	1,124,729.34	1,124,729.00	=ROUND(B8,0)
9				
10	Column C Total Detail		1,124,731.00	=SUM(C2:C7)

This technique may not be applicable to all situations and involves using a balancing item. We will still include a validation to make sure that the balancing item is within a tolerance value.

In the image in the right column B contains the raw values for current assets.

Column C contains rounded values for all of the entries except the Stock row.

The Stock row has been used as the balancing item because it is the largest item in the list. The formula in the balancing item in cell C4 takes the rounded total

and deducts all of the rounded values to arrive at a value that balances.

	A	B	C	D
1	Description	Raw Value	Rounded	Formula in Column C
2	Cash	2,345.67	2,346.00	=ROUND(B2,0)
3	Bank	269,809.56	269,810.00	=ROUND(B3,0)
4	Stock	679,658.54	679,657.00	=C8-SUM(C2:C3,C5:C7)
5	Trade Debtors	130,456.78	130,457.00	=ROUND(B5,0)
6	Other Debtors	34,567.89	34,568.00	=ROUND(B6,0)
7	Prepayments	7,890.90	7,891.00	=ROUND(B7,0)
8	Total Current Assets	1,124,729.34	1,124,729.00	=ROUND(B8,0)
9				
10	Stock raw value	cell B4	679,658.54	=B4
11	Stock rounded	cell C4	679,657.00	=C4
12	Difference		1.54	=C10-C11
13	Valid		TRUE	=ABS(C12)<Tolerance

Note - row 13 Tolerance is a range name.

Tolerances

As mentioned it is common to use a tolerance value when working with rounded values. Also it is common to use a range name for that tolerance value. Using a range name makes it a lot easier to refer to the tolerance value throughout the file.

Tips for creating a Centralised Validation sheet

Excel has a feature that allows you to copy entries from one sheet and then paste link them to another sheet. This creates links to the cells that you have copied. The easy way to apply this is to copy the cells you want linked then return to the sheet where you want to paste them and press in sequence Alt H V N. This is the shortcut to paste link.

Once you have the links in place inside the validation sheet to follow a link you can use another keyboard shortcut. When on the links cell press Ctrl + [to follow the link. To return to where you were press F5 and then press Enter.

March 2018 – Identify Current Row

A retired accountant who was still using Excel on a daily basis in his community work wanted the ability to highlight a row in a data table to make it easier for him to see across the table.

There are a couple of techniques that will achieve this but they typically have issues with them.

Even the solution that I had in the magazine had an issue which I have since found a solution for.

We can use a conditional format to apply the colour. We can use two similar functions to identify the current row within the table. Then we can use a one line macro to avoid a calculation problem.

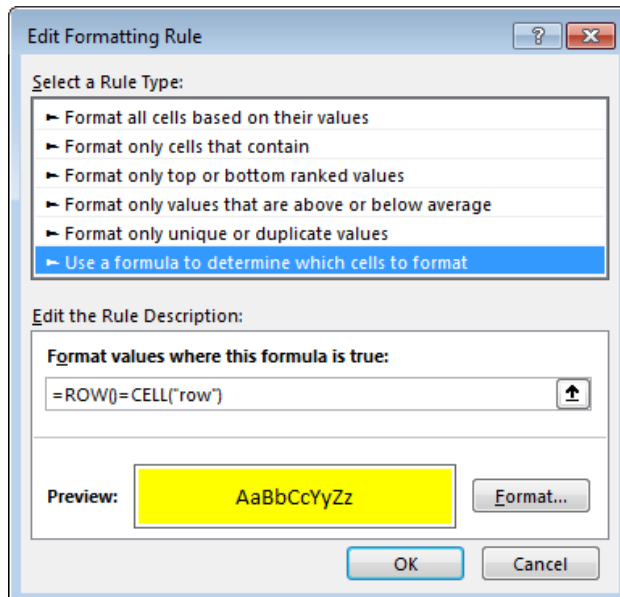
I have included a sheet called CELL_Options that demonstrates how the functions we need work.

Conditional Format

The conditional format option on the Home ribbon allows you to change the format of a cell or range based on a condition.

1. We will apply the conditional format on the whole table in the Highlight sheet. To select the whole table click a cell within the table and press Ctrl + a. Click the Conditional Format icon on the Home ribbon and choose New Rule. Use the last option "Use a formula ..." and enter the following formula. (This formula will be explained later in the session)

`=ROW()=CELL("row")`



2. Click the Format button and Choose a Fill color (I have used Yellow) and click OK.

The problem we face with the solution so far is that moving a cell around the table won't trigger a calculation. We need Excel to calculate so that the ROW and CELL functions will update. If you press F9 after you move around then you can see that the formula will update and the highlighting will work.

Event macro

We can use what is called an Event macro to force Excel to calculate each time we change the active cell. This will update the highlighted row in the table.

One of the problems that was identified with the original solution was that you couldn't copy and paste into the sheet because the Event macro cleared the clipboard. To get around that I have found code that will check the clipboard before calculating. To create the Event macro follow these steps.

- Press Alt + F11 to open the VBA window.
- On the left of screen double click the Sheet1(Highlight). This opens the code window for that particular sheet.
- In the code window on the right, click the left drop down above the code window and choose Worksheet. The SelectionChange Event macro will be created. Between the first and last line enter the following code.

If Application.CutCopyMode = 0 Then Calculate

This code checks whether there have been anything cut or copied and if it hasn't it will calculate. If something has been cut or copied it won't calculate which means the highlighting won't be updated.

After pasting you need to press Esc to clear the clipboard and the highlighting will start to work again. Whilst there is something on the clipboard the highlighting won't work.

How the Formula works

See the CELL_Options sheet for examples.

The ROW function returns a row number. If you type = ROW() in a cell it will return the row number of that cell. If you type =ROW(A17) in a cell it will always display 17.

The active cell in the image on the right is B19, so cells A14 to A17 all relate to that cell.

Those cells demonstrate other options you can use with the CELL function.

B19			
	A		B
1	ROW Function		Formula in Column A
2	2		=ROW()
3	3		=ROW()
4	4		=ROW()
5	17		=ROW(A17)
6			
7	CELL function has a cell reference		Formula in Column A
8	\$A\$7		=CELL("address",\$A\$7)
9	4\Training\Webinars\EY\EY 2018\[Excel Yourself 2018.xlsm]CELL_Options		=CELL("filename",\$A\$7)
10	7		=CELL("row",\$A\$7)
11	1		=CELL("col",\$A\$7)
12			
13	CELL function doesn't have a cell reference		Formula in Column A
14	\$B\$19		=CELL("address")
15	4\Training\Webinars\EY\EY 2018\[Excel Yourself 2018.xlsm]CELL_Options		=CELL("filename")
16	19		=CELL("row")
17	2		=CELL("col")
18			
19			
20			

When the CELL function doesn't have a second argument it returns information on the active cell.

When you use =ROW() in a conditional format formula it returns the row number of the cell it is formatting.

The CELL function can also return the row number of a particular cell - see cell A10 in the image on the previous page.

It can also return the row number of the active cell on the sheet see cell A16 in the image on the previous page. The formula below returns TRUE when the active cell matches a row in the table.

=ROW()=CELL("row")

Saving

When saving the file if you started with a normal .xlsx file you will be prompted with a message warning about losing the macros. Make sure you respond No and use Save As and save as a .xlsm or .xlsb file to preserve the macro.

April 2018 – Centralised Logic

When it comes to formulas in Excel nested IF functions are one of the most difficult functions to read. A single IF function is readable but as soon as you have three or four nested IF functions it becomes difficult to determine the logic used.

I would like to demonstrate an alternative to including logic within the IF function. Instead of having the Logic built-in to each IF function in a range consider building that logic into single rows. You can then refer to those rows within the IF function.

This technique shortens the IF function is making them easier to read and it also centralises the logic so that it is easier to use, understand and maintain.

In the image below is in the sheet Report. This is an extract and just shows the sales. We are creating a revised forecast for the full year based on actuals and either forecast for budget. The actual columns are determined by the month entry cell B1. Cell B2 determines whether a future month is going to use the budget or forecast value.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Current Month	Feb-18												
2	Budget/Forecast	Forecast												
3														
4		Jul-17	Aug-17	Sep-17	Oct-17	Nov-17	Dec-17	Jan-18	Feb-18	Mar-18	Apr-18	May-18	Jun-18	Total
5	Revised Year													
6	Sales	246,524	281,588	225,105	291,596	214,239	241,225	230,690	249,207	259,905	269,449	202,072	218,313	2,929,913
7														
8	Actuals													
9	Sales	246,524	281,588	225,105	291,596	214,239	241,225	230,690	249,207					1,980,174
10														
11	Budget													
12	Sales	225,551	251,462	216,047	243,834	244,931	252,697	247,430	225,045	227,352	250,285	240,935	224,895	2,850,464
13														
14	Forecast													
15	Sales	234,556	224,883	280,458	200,174	283,100	263,282	224,991	220,630	259,905	269,449	202,072	218,313	2,881,813
16														

The typical formula in cell B6 which has been copied across is

`=IF(B$4<=$B$1,B9,IF($B$2="Budget",B12,IF($B$2="Forecast",B15,0)))`

As you can see this is a reasonably complex formula that contains three IF functions. Typically this complex nested IF function would be repeated in other lines when we are bringing in other metrics.

There are three logical decisions being made in this formula. If we can capture the three logical results in separate rows we can simplify the final formula.

The image below shows a different structure in the Report_1 sheet where we have inserted rows to handle the logic calculations.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Current Month	Feb-18												
2	Budget/Forecast	Forecast												
3		Actuals	Actuals	Actuals	Actuals	Actuals	Actuals	Actuals	Actuals	Forecast	Forecast	Forecast	Forecast	
4		Jul-17	Aug-17	Sep-17	Oct-17	Nov-17	Dec-17	Jan-18	Feb-18	Mar-18	Apr-18	May-18	Jun-18	Total
5	Use Actuals	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	
6	Use Budget	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	
7	Use Forecast	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	
8	Revised Year													
9	Sales	246,524	281,588	225,105	291,596	214,239	241,225	230,690	249,207	259,905	269,449	202,072	218,313	2,929,913
10														
11	Actuals													
12	Sales	246,524	281,588	225,105	291,596	214,239	241,225	230,690	249,207					1,980,174
13														
14	Budget													
15	Sales	225,551	251,462	216,047	243,834	244,931	252,697	247,430	225,045	227,352	250,285	240,935	224,895	2,850,464
16														
17	Forecast													
18	Sales	234,556	224,883	280,458	200,174	283,100	263,282	224,991	220,630	259,905	269,449	202,072	218,313	2,881,813
19														

Most people are unaware that the logical test at the start of a IF function can be placed in a cell by itself and it will return TRUE or FALSE.

In determining what value should go in row nine in the revised report we must first determine if the month is an actual month. The table below shows the formulas for column B in rows 3, 5, 6 and 7.

Cell	Formula	Alternate Formula
B3	=IF(B5,"Actuals",\$B\$2)	
B5	=B\$4<=\$B\$1	
B6	=AND(B5=FALSE,\$B\$2="Budget")	=AND(NOT(B5),\$B\$2="Budget")
B7	=AND(B5=FALSE,\$B\$2="Forecast")	=AND(NOT(B5),\$B\$2="Forecast")

Row 3 displays the metric being used based on row 5 and cell B2.

Row 5 determines if the column is an actual column by comparing the date in cell B4 to the date in B1.

Row 6 determines if the column is going to be a budget. The AND function requires all the logical tests to be TRUE for it to return TRUE. For row 6 to display TRUE there must be a FALSE in row 5 and Budget must be displayed in cell B2.

The alternate formulas for row 6 use the NOT function. The NOT function is the same as comparing the cell to FALSE. The NOT function reverses the logic and changes TRUE to FALSE and FALSE to TRUE.

Row 7 is the same as row 6 except it is looking for Forecast in cell B2. The alternate formula is also the same except for looking for Forecast in cell B2.

Helper cells

I would describe rows 5, 6 and 7 as helper cells because they simplify the final formula by capturing the logic. Row 3 is an information cell as it displays what is being used for each month.

The formula in cell B9 which has been copied across is

=IF(B\$5,B12,IF(B\$6,B15,IF(B\$7,B18,0)))

When referring to cells that contain TRUE or FALSE you do not need to compare the cell to TRUE in the logical test part of the IF function. You can just refer to the cell as the logical test.

Centralised logic

The beauty of centralised logic is that if one of the logical test needs to change you only need to change it in one row. If you have built the logic into all of the other formulas then you need to amend all of them.

Grouping

Since rows 5,6 and 7 contain helper cells we do not need to display them. The simple way to hide these rows is to use grouping. Select rows 5, 6 and 7 and press Shift + Alt + Right Arrow. This is the shortcut to apply grouping. This adds icons to the left of screen which allows you to easily hide and unhide the rows. To remove grouping use Shift + Alt + Left Arrow.

Multiplying by TRUE and FALSE

Rows 5, 6 and 7 are mutually exclusive. That means in any one month there is only one TRUE in the range. Because of that we can take advantage of how Excel treats TRUE and FALSE in calculations to create an even shorter formula.

In Excel TRUE equals one and FALSE equals zero. This means if you multiply by TRUE you will leave the value unchanged. If you multiply by FALSE you will zero the value. Because the three entries are mutually exclusive we can multiply each of the sales values by their respective logic test and add up the results. The Report_2 sheet contains the alternate formulas.

The revised formula for cell B9 is

`=(B5*B12)+(B6*B15)+(B7*B18)`

This formula works because the results are mutually exclusive.

November 2018 – Slope Charts

Slope charts are an effective way to show movement between two periods. They are not a built-in chart in Excel but you can create them. I have also included a macro that can simplify the creation process. The macro has an Excel 2010 version and an Excel 2013 and later version.

You can create slope charts in Excel 2010 but the interface changed in Excel 2013. I will be using Excel 2016 during the webinar.

I will take you through the steps in creating a slope chart from scratch.

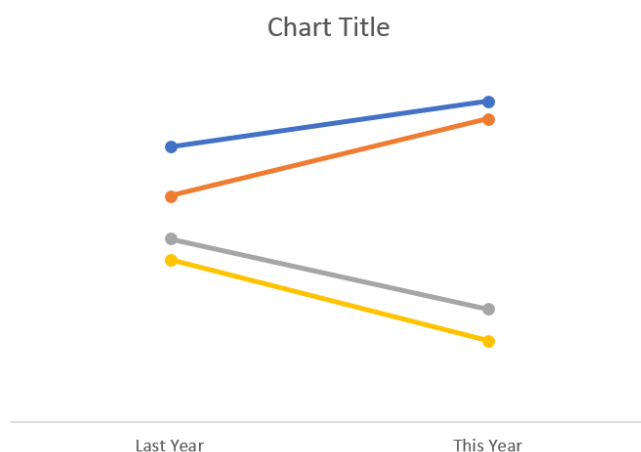
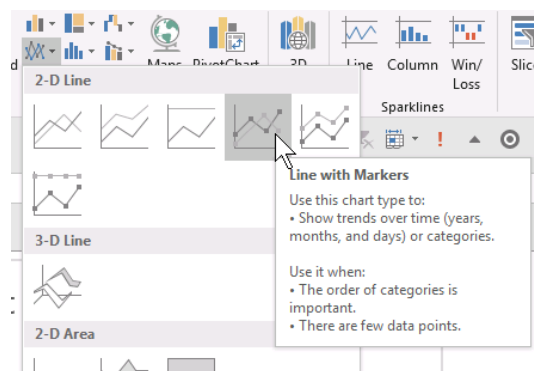
The steps are reasonably repetitive which is why a macro is so useful.

In our example we will plot the defect percentage between this year and last year for different states. The data we will use is shown on the right.

	A	B	C
1	Defect %	Last Year	This Year
2	QLD	7.8%	9.1%
3	NSW	6.4%	8.6%
4	WA	5.2%	3.2%
5	VIC	4.6%	2.3%

Steps

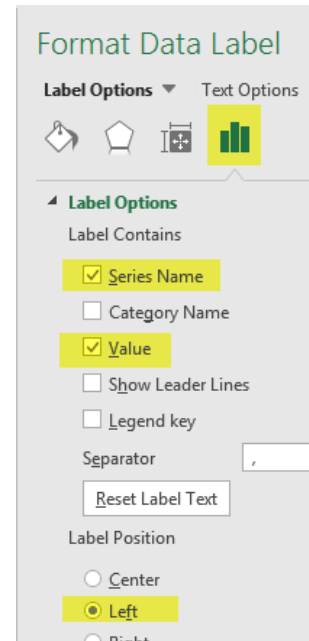
1. Select the range A1:C5.
2. Click the Insert ribbon tab and click the Line icon in the Charts section and choose a Line with Markers as shown on the right.
3. The chart that is created isn't quite what we need so we need to change a setting to make it look like a slope chart. With the chart selected click the Design ribbon tab and click the Switch Row/Column icon.
4. This looks more like a slope chart. We must remove chart elements to make it a bit cleaner. We will delete the vertical axis, the gridlines and the legend. The chart will then look like this.



5. This is where it gets repetitive.

For each line we need to do the following

- Add Data Labels – right click the line and choose Add Data Labels.
 - Click the left Data Label twice and right click it and Format it – choose the last icon in the top section – Options.
 - Add the Series Name and change the Label position to Left- see image on right.
6. Repeat the parts of Step 5 for the other three lines.
7. Select the Chart Title click in the Formula Bar and press = and click cell A1 and press Enter. See below.



A1							
	A	B	C	D	E	F	G
1	Defect %	Last Year	This Year				
2	QLD	7.8%	9.1%				
3	NSW	6.10%	8.50%				

8. That's it - chart done.

Slope Chart Issues

To ensure the slope chart and the data are aligned sort the table in descending order by the last period.

Slope charts don't work in all instances. If you have too many points to plot or the values are too close together then they don't work very well because the labels overwrite each other.

You can avoid a few labels overwriting on the righthand side by using rounding so that the values are the same. This works for the second period not for the first period where the series name labels will still overwrite.

You can also increase the height of the chart to separate values and labels.

Be careful with the label length because Excel will wrap them if they get too long. The only way to fix the wrapping is to make the chart wider.

Emphasising a line

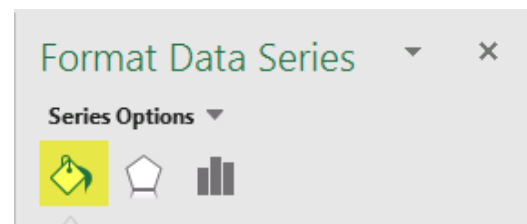
One common technique when using a slope chart is to emphasis one line with one colour, say blue, and using grey for the other lines. Blue is a good colour to use as it isn't affected by colour blindness. This may seem straight forward but the problem is that each line has three elements that require a colour change so to change three lines to grey you must change 9 elements.

The three colours options you need to change for each line are

- Line
- Marker Fill
- Marker Border

To change the colours you will need to use the first icon – the paint can - see image on right.

There are different sections for the elements mentioned above.



Macros

I have included two macros that create a slope chart. There is a macro for Excel 2010 and earlier versions and one for Excel 2013 and later versions.

To use a macro select the range to use and press Alt + F8. Choose This Workbook from Macros in drop down.

Then double click the correct macro for your version.

To see the macro code you can either

- press Alt + F8 and choose the macro and then click the Edit button (see image on right) or
- press Alt + F11 to open the VBA window and double click the modSlope module on the left of screen - see image on right.

Note: these are not recorded macros. I did record macros to discover the code required. Then I modified the recorded code to do what was required.

