

Analytics Hawk

AH Tools

Empowering your Analytical Skills

User Guide



www.analyticshawk.com

info@analyticshawk.com

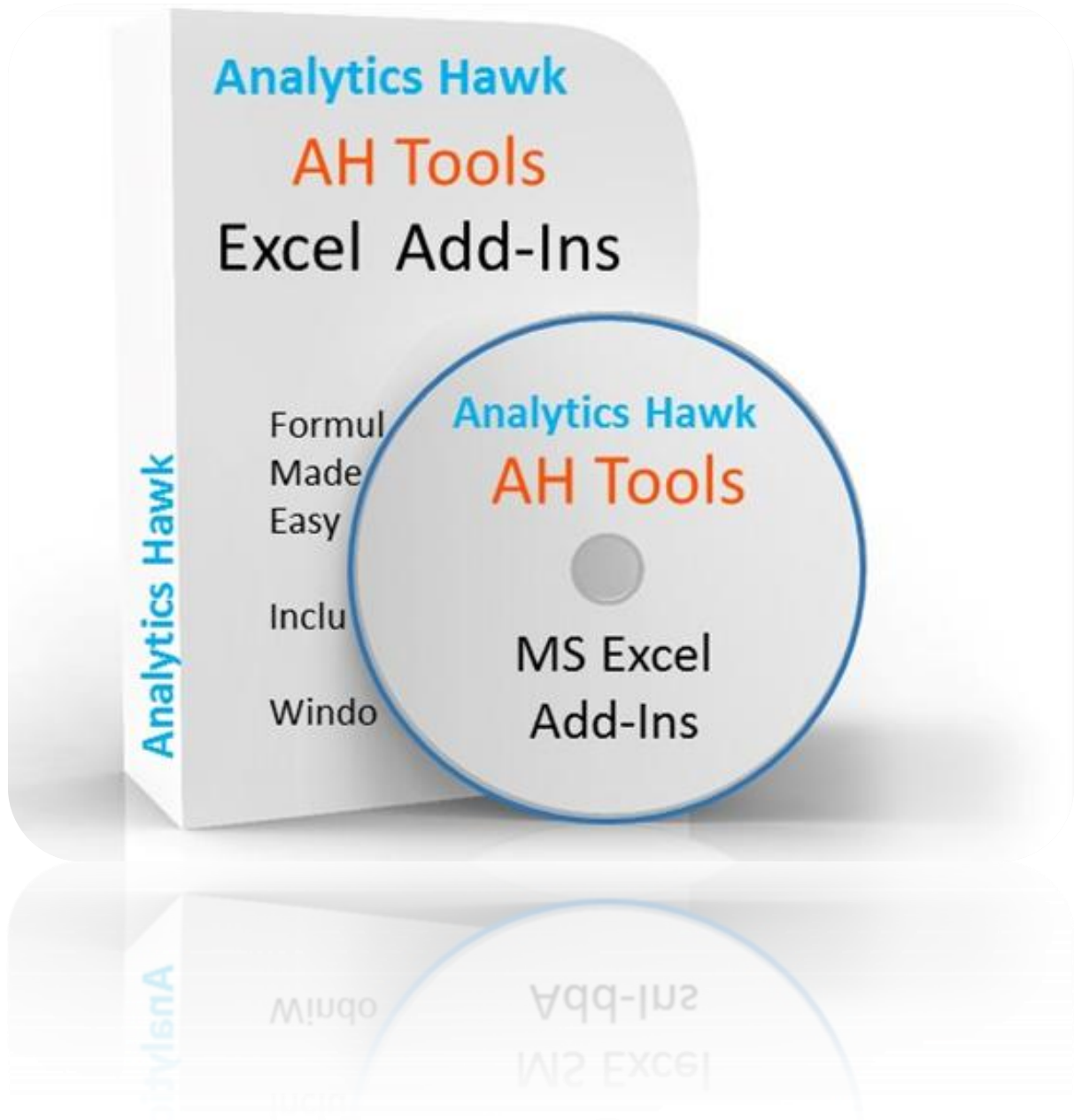
Version 1.0.5 rev 49

© USTECH. 2014 – Analytics Hawk

This document is copyright and protected. No information or part of this publication may be copied, printed, published, advertised or made available for download or retrieval electronically without prior written consent by USTECH.

While this document was prepared with thorough citation the publisher, developer, author and/or owners take no responsibility in any errors or omissions contained herein. You the user assume all responsibility and take ownership of any errors, faults, damages caused by this software.

MS, Microsoft, Microsoft Office, and MS Excel are registered trademarks of Microsoft Corporation.



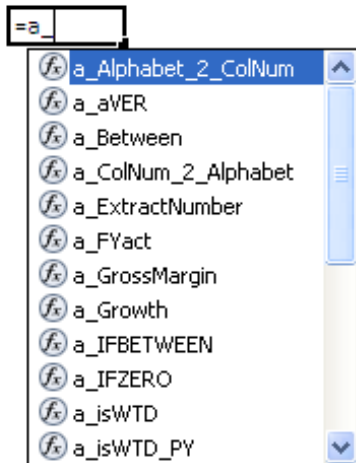
Contents

AH TOOLS FORMULAS	5
1. A_WEEKNO	6
2. A_FYACT	8
3. A_WTD	10
4. A_WTDPY	11
5. A_GROSSMARGIN	12
6. A_GROWTH	13
7. A_BETWEEN	14
8. A_EMAIL	15
9. A_COL2ALPHA	16
10. A_ALPHA2COL	17
11. A_EXTRACTNUMBER	18
12. A_SPLITME	19
13. A_WORKBOOK	20
14. A_MACID	21
15. A_ISWTD	22
16. A_ISWTDPY	23
17. A_ISYTD	24
18. A_ISYTDPY	25
19. A_ISZERO	26
20. A_IFBETWEEN	27
21. A_IFZERO	28
AH TOOLS RIBBON	29
22. COMMA STYLE	30
23. SHEET HIDE UNHIDE	31
24. TOC – TABLE OF CONTENTS	33
25. UNMERGE AND REPEAT	35
26. FIX BLANK CELLS	37
27. RESET STYLE	38

AH Tools Formulas

AH Tools boasts myriad of new formulas currently not available in MS Excel or requires complex nested formulas to achieve certain output.

All AH Tools formulas begin with **a_** so to use it, you'd need to type = **a_** e.g. below.



As these formulas are UDF (User Defined Functions) MS Excel treats it slightly differently where on a MS Excel built in formula e.g. =SUM, a popup appears showing you all the parameters required. In a UDF you will need to know what parameters are expected and this user guide should assist you to achieve this.

We have also attached a palm card of all formulas and functions at the end of this user guide. Feel free to print and stick on your desk or wall board.

So why custom formulas?

MS Excel is built with 100s of formulas and while it does what most users require, we believe there can be some more additional formulas or in some cases a better easy way of doing it hence the need of AH Formulas. AH Formulas are easy to use and we haven't recreated the wheel where it already exists.

Our expert team of analysts have wide range of industry experience and develop only what is required and easy to use or to make you efficient in your job.

1. a_WeekNo

a_WeekNo returns the week number (where week starts on first week of July – not necessarily 1st July but the week where financial year starts). aWeekNo is useful for Australian financial weeks for Australian companies where a week is measured from Sunday to Saturday inclusive.

Note : This function is slightly different from MS Excel's WeekNumber function as it's designed for Australian calculations where MS Excel's WeekNumber function calculates from 1st Jan.

How to use?

Syntax: =a_WeekNo (<DATE>)

DATE = Any date for which the Week Number is required.

Example

Table 1 below contains Sales Date in Column A and Sales Amount in Column B and here we'd like to derive the Total Sales by week. By using the formula a_WeekNo we can achieve this easily in 2 steps.

1. Apply the formula a_WeekNo as per Column C and drag down. The usage of this formula is shown in Column D.
2. Pivot Table the data and summarise by Week No (Column C)

Table 1

	A	B	C	D
1	Sales Date	Sales Amount	Week No	Formula
2	22/03/2015	\$7,569	39	=a_WeekNo(A2)
3	23/03/2015	\$7,473	39	=a_WeekNo(A3)
4	24/03/2015	\$8,289	39	=a_WeekNo(A4)
5	25/03/2015	\$6,344	39	=a_WeekNo(A5)
6	26/03/2015	\$7,257	39	=a_WeekNo(A6)
7	27/03/2015	\$7,502	39	=a_WeekNo(A7)
8	28/03/2015	\$8,925	39	=a_WeekNo(A8)
9	29/03/2015	\$8,444	40	=a_WeekNo(A9)
10	30/03/2015	\$6,992	40	=a_WeekNo(A10)
11	31/03/2015	\$6,042	40	=a_WeekNo(A11)
12	1/04/2015	\$6,353	40	=a_WeekNo(A12)
13	2/04/2015	\$4,767	40	=a_WeekNo(A13)
14	3/04/2015	\$8,840	40	=a_WeekNo(A14)
15	4/04/2015	\$8,533	40	=a_WeekNo(A15)
16	5/04/2015	\$8,499	41	=a_WeekNo(A16)
17	6/04/2015	\$6,593	41	=a_WeekNo(A17)
18	7/04/2015	\$7,683	41	=a_WeekNo(A18)
19	8/04/2015	\$8,777	41	=a_WeekNo(A19)
20	9/04/2015	\$7,476	41	=a_WeekNo(A20)
21	10/04/2015	\$5,771	41	=a_WeekNo(A21)
22	11/04/2015	\$9,088	41	=a_WeekNo(A22)
23				
24	Total	\$157,217		

Design the Pivot Table

PivotTable Field List

Choose fields to add to report:

- Sales Date
- Sales Amount
- Week No
- Formula

Drag fields between areas below:

Report Filter:

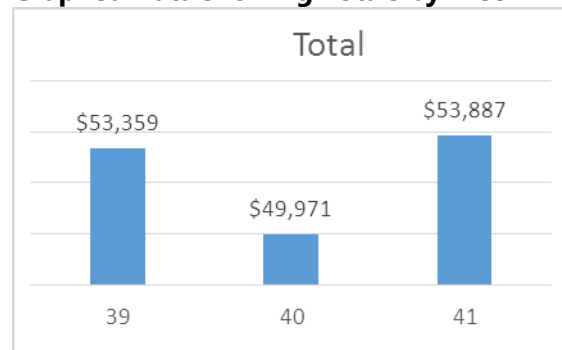
Column Labels:

Row Labels: Week No

Values: Sum of Sales ...

Row Labels	Sum of Sales Amount
39	\$53,359
40	\$49,971
41	\$53,887
Grand Total	\$157,217

Graphed Data showing Totals by Week



Other common usage for a_WeekNo;

- Show trends by week
- Compare weekly totals Year on Year (Variance)
- Portray seasonality in the business (Peaks / Troughs)

2. a_FyAct

a_FyAct returns the Financial Year for any given date (Actual Financial Year). This function is designed for Australian Financial Year calculations where Financial Year starts on 1 July and Ends on 30 June.

This function has multiple uses such as determining the FY of one date vs FY of another date or in a real example calculate FY for a range of dates. The return / output by this function can be customised based on the control arguments e.g. below

Assume we're looking date 23/03/2015 with Control Arguments as below;

Control Argument	Return / Output
1	FY15
2	FY2015
3	15
4	MYTEXT 15

How to use?

Syntax: =a_fyAct (<DATE>, <CONTROL ARGUMENT>, <PREFIX>)



DATE = Date for which FY is required

CONTROL ARGUMENT = 1 to 4 as above

PREFIX = Optional text which can be set as prefix to the return Value

Usage of PREFIX

PREFIX adds optional text before the actual FY Year. It has very important unique usage and not always required by most users however, some uses are;

-  Output FY as custom text e.g. 'Fin Year 2015' 'F2015' 'F.Y. 2015' ...
-  Gives user the control on what to display before the actual Financial Year number.

Example

Table 1 below shows Sales Date in Column A and Sales Amount in Column B. For each date, we'd like to see the all 4 FY outputs.

Here's how;

Column C shows Control Argument of 1.

Column D shows the formula used in Column C to achieve this. You can see ① which is the Control Argument to output as **FY14**.

Column E shows Control Argument of 2.

Column F shows Control Argument of 3.

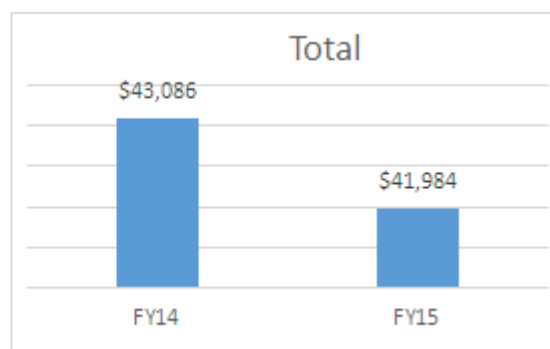
Column G shows Control Argument of 4.

Table 1

	A	B	C	D	E	F	G
1			Control Argument				
2			①		2	3	4
3	Sales Date	Sales Amount	FY	Formula	FY	FY	FY
4	1/07/2013	\$7,569	FY14	=a_FYact(A2,①)	FY2014	14	2014
5	2/07/2013	\$7,473	FY14	=a_FYact(A3,1)	FY2014	14	2014
6	3/07/2013	\$8,289	FY14	=a_FYact(A4,1)	FY2014	14	2014
7	...						
8	...						
9	28/06/2014	\$6,721	FY14	=a_FYact(A10,1)	FY2014	14	2014
10	29/06/2014	\$6,992	FY14	=a_FYact(A10,1)	FY2014	14	2014
11	30/06/2014	\$6,042	FY14	=a_FYact(A11,1)	FY2014	14	2014
12							
13	1/07/2014	\$6,353	FY15	=a_FYact(A12,1)	FY2015	15	2015
14	2/07/2014	\$4,767	FY15	=a_FYact(A13,1)	FY2015	15	2015
15	3/07/2014	\$8,840	FY15	=a_FYact(A14,1)	FY2015	15	2015
16	...						
17	...						
18	28/06/2015	\$8,777	FY15	=a_FYact(A19,1)	FY2015	15	2015
19	29/06/2015	\$7,476	FY15	=a_FYact(A20,1)	FY2015	15	2015
20	30/06/2015	\$5,771	FY15	=a_FYact(A21,1)	FY2015	15	2015
21							
22	Total	\$85,070					

We can go further by applying a Pivot Table to this data to show Year on Year totals as below.

Row Labels	Sum of Sales Amount
FY14	\$43,086
FY15	\$41,984



3. a_WTD

a_WTD returns the latest Week To Date. Since Week To Date is defined as Sunday to Saturday, a_WTD will return the Saturday date (i.e. the last day). It can also be called 'Week Ending on date'.

Week To Date is based on a Sunday to Saturday of the week that just completed. The week that is in running i.e. current week or this week is NOT considered as Week To Date.

e.g. if today is Wednesday 16th December 2015, it will be considered as current week since this week hasn't completed yet. (Note Sunday to Saturday as being the week. Sunday first day and Saturday last day).

Week To Date is calculated to be up to Saturday 12th December 2015.

From Sunday 13th December 2015 it's considered to be current week or also known as this week.

How to use?

Syntax: =a_WTD()

4. a_WTDPY

a_WTDPY returns the latest Week To Date for Prior Year. Since Week To Date is defined as Sunday to Saturday, a_WTDPY will return the Saturday date (i.e. the last day). It can also be called 'Week Ending on date' for Prior Year.

In simple terms, it's the same week but for last year.

Week To Date is based on a Sunday to Saturday of the week that just completed. The week that is in running i.e. current week or this week is NOT considered as Week To Date.

This function is very useful in determining the date range and whether to include as part of WTD calculation or not.

e.g. if today is Wednesday 16th December 2015, it will be considered as current week since this week hasn't completed yet. (Note Sunday to Saturday as being the week. Sunday first day and Saturday last day).

Week To Date is calculated to be up to Saturday 12th December 2015.

From Sunday 13th December 2015 it's considered to be current week or also known as this week.

How to use?

Syntax: =a_WTDPY()

5. a_GROSSMARGIN

a_GROSSMARGIN calculates the margin between the SELL_VALUE and COST_VALUE. The margin is returned in decimal format and can be converted to percentage to show the actual value.

How to use?

Syntax: =a_GROSSMARGIN (<SELL_VALUE>, <COST_VALUE>)

SELL_VALUE = Value in numeric format.

COST_VALUE = Value in numeric format.

The function will return numeric value as a result. Change the format to % is required.

The underlying formula calculates Gross Margin as;

$$=(\text{SELL VALUE} - \text{COST VALUE}) / \text{SELL VALUE}$$

Surely this is simple enough and why would you use a custom formula?

Using a_GROSSMARGIN clearly shows the user that a margin is getting calculated. In a complex formula, e.g. nested formulas, Gross Margin can be easily identified and makes troubleshooting of formulas much easy.

Example

The example below calculates the margin in Column E.

Column F shows the formula used in Column E to achieve this.

	A	B	C	D	E	F
1						
2	Item	Cost	Sell	Profit	Margin	Formula
3	Blue Pen	\$1.50	\$2.10	\$0.60	29%	=a_GrossMargin(C3,B3)
4	Black Pen	\$1.70	\$2.00	\$0.30	15%	=a_GrossMargin(C4,B4)
5	Stapler	\$2.10	\$2.50	\$0.40	16%	=a_GrossMargin(C5,B5)
6	Paper	\$3.50	\$4.00	\$0.50	13%	=a_GrossMargin(C6,B6)
7	Ruler	\$0.50	\$0.70	\$0.20	29%	=a_GrossMargin(C7,B7)

6. a_GROWTH

a_GROWTH calculates the growth or change between the NEW_VALUE and OLD_VALUE. The growth is returned in decimal format and can be converted to percentage to show the actual value.

This function is specifically useful when analysing % growth or % change between 2 numbers. E.g. Sales of Current Year vs Sales of Prior Year.

How to use?

Syntax: =A_GROWTH (<NEW_VALUE>, <OLD_VALUE>)

NEW_VALUE = Value in numeric format.

OLD_VALUE = Value in numeric format.

The function will return numeric value as a result. Change the format to % is required.

The underlying formula calculates Growth as;

$$=(\text{NEW VALUE} - \text{OLD VALUE}) / \text{OLD VALUE}$$

Surely this is simple enough and why would you use a custom formula?

Using a_GROWTH clearly shows the user that a Growth is getting calculated. In a complex formula, e.g. nested formulas, Growth of % change can be easily identified and makes troubleshooting of formulas much easy.

Example

The example below calculates the growth / change in Column E. Column F shows the formula used in Column E to achieve this.

	A	B	C	D	E	F
1						
2	Item	2014	2013	Variance	Growth	Formula
3	Blue Pen	\$2,500.00	\$1,200.00	\$1,300.00	108%	=a_Growth(B3,C3)
4	Black Pen	\$3,250.00	\$4,500.00	-\$1,250.00	-28%	=a_Growth(B4,C4)
5	Stapler	\$1,900.00	\$1,400.00	\$500.00	36%	=a_Growth(B5,C5)
6	Paper	\$15,980.00	\$18,500.00	-\$2,520.00	-14%	=a_Growth(B6,C6)
7	Ruler	\$1,030.00	\$1,300.00	-\$270.00	-21%	=a_Growth(B7,C7)

7. a_BETWEEN

a_BETWEEN is very similar to VLOOKUP however it looks in the first 2 columns as a condition. This function validates the LOOKUP_VALUE in LOOKUP_ARRAY and will return the value from COL_INDEX_NUMBER.

This function is specifically useful when searching between values such as trends, incentives, payments or scaled numerical terms.

How to use?

Syntax: =a_Between(<LOOKUP_VALUE>, <LOOKUP_ARRAY>, <COL_INDEX_NUM>)

LOOKUP_VALUE = Value which you're looking for or searching.

LOOKUP_ARRAY = List / Array where you're looking in. The structure of this Array needs to be as per below example for it to function properly.

COL_INDEX_NUM = Form the Array above, this is the Column Index Number which you want returned. Note: Index starts from 1 e.g. First Column is 1, second Column is 2 ...

Example

The example below shows the returned value in Column C (Commission).

Column D shows the formula used in Column C to achieve this.

	A	B	C	D
1				
2	Sales Commission Structure			
3	FROM	TO	% COMMISSION	
4	\$0.00	\$ 6,999.99	1.0%	
5	\$7,000.00	\$ 7,999.99	10.0%	
6	\$8,000.00	\$ 8,999.99	15.0%	
7	\$9,000.00	\$ 100,000.00	20.0%	
8				
9				
10	Sales Transactions			
11	SALES DATE	SALES AMOUNT	COMMISSION	FORMULA
12	22/03/2015	\$7,569	10%	=a_Between(B12,\$A\$4:\$C\$7,3)
13	23/03/2015	\$7,473	10%	=a_Between(B13,\$A\$4:\$C\$7,3)
14	24/03/2015	\$8,289	15%	=a_Between(B14,\$A\$4:\$C\$7,3)
15	25/03/2015	\$6,344	1%	=a_Between(B15,\$A\$4:\$C\$7,3)
16	26/03/2015	\$7,257	10%	=a_Between(B16,\$A\$4:\$C\$7,3)
17	27/03/2015	\$7,502	10%	=a_Between(B17,\$A\$4:\$C\$7,3)
18	28/03/2015	\$8,925	15%	=a_Between(B18,\$A\$4:\$C\$7,3)
19	29/03/2015	\$8,444	15%	=a_Between(B19,\$A\$4:\$C\$7,3)
20	30/03/2015	\$6,992	1%	=a_Between(B20,\$A\$4:\$C\$7,3)
21	31/03/2015	\$6,042	1%	=a_Between(B21,\$A\$4:\$C\$7,3)
22	1/04/2015	\$6,353	1%	=a_Between(B22,\$A\$4:\$C\$7,3)
23	2/04/2015	\$4,767	1%	=a_Between(B23,\$A\$4:\$C\$7,3)
24	3/04/2015	\$8,840	15%	=a_Between(B24,\$A\$4:\$C\$7,3)
25	4/04/2015	\$8,533	15%	=a_Between(B25,\$A\$4:\$C\$7,3)
26	5/04/2015	\$8,499	15%	=a_Between(B26,\$A\$4:\$C\$7,3)
27	6/04/2015	\$6,593	1%	=a_Between(B27,\$A\$4:\$C\$7,3)
28	7/04/2015	\$7,683	10%	=a_Between(B28,\$A\$4:\$C\$7,3)
29	8/04/2015	\$8,777	15%	=a_Between(B29,\$A\$4:\$C\$7,3)
30	9/04/2015	\$7,476	10%	=a_Between(B30,\$A\$4:\$C\$7,3)
31	10/04/2015	\$5,771	1%	=a_Between(B31,\$A\$4:\$C\$7,3)
32	11/04/2015	\$9,088	20%	=a_Between(B32,\$A\$4:\$C\$7,3)

8. a_EMAIL

Due to MS Outlook's security settings, this function has been removed in this version.

9. a_COL2ALPHA

a_COL2ALPHA returns the corresponding column Character in Alphabet Format when any number is passed.

This function has no limits and is not specifically designed to provide 256 columns but it can go beyond that. It has multiple uses and most common usages are;

- Dashboard / Report automation where you're counting x number for columns
- Retrieving the column position of any given number

How to use?

Syntax: =a_Col2Alpha(<INTEGER>)

INTEGER = Any Integer (whole number) value for which the column is required.

Example

Cell A1 contains the formula.

Cell B1 shows the returned result.

This shows that 29th Column is AA.

	A	B	C	D	E
1	=ah_col2Alpha(27)	AA			
2					
3					
4					
5					
6					

10. a_ALPHA2COL

a_ALPHA2COL returns the corresponding column number (such as 1, 2, 3, 26, 27, etc) when a Column Alphabet is passed. This function is useful to determine a specific number for any given Column. E.g. A=1, B=2, C=3 ...

This function has no limits and is not specifically designed to provide 256 columns but it can go beyond that. It has multiple uses and most common usages are;

- Dashboard / Report automation where you're counting x number for columns
- Retrieving the column position of any given Column Name

How to use?

Syntax: =a_ALPHA2COL(<COLUMN NAME>)

COLUMN NAME = Any Column or Alphabet Character for which the column number is required.

Example

Cell A1 contains the formula.

Cell B1 shows the returned result.

This shows that AA falls on the 27th Column.

	A	B	C	D	E
1	=ah_Alpha2Col("AA")	27			
2					

11. a_EXTRACTNUMBER

a_EXTRACTNUMBER extracts all numbers from a mixed text. The purpose of this function is to extract the numbers from a text where it's mixed with printable characters. E.g. APP76LE will return 76.

How to use?

Syntax: =a_EXTRACTNUMBER(<TEXT>)

TEXT = Text which contains alpha numeric characters.

Example

Below we have texts with mixed characters. Numbers are extracted in Column B.

	A	B
1	APP76LE	76
2	A7PP6LE	76
3	BANA987NA654 TASTES 321 GREAT	987654321
4		

12. a_SPLITME

a_SPLITME splits a string or cell value based on a delimited character. Surely you can use Excel's built in "Text to Columns" but this will spread across all columns. a_SplitMe works directly like a formula and returns the nth value in a text.

This function is really useful if you're trying to split a URL, unformatted text and series of fields in one dump.

How to use?

Syntax: =a_SPLITME (<TEXT TO SPLIT>, <DELIMITER CHARACTER>, <RETURN Nth POSITION>)

TEXT TO SPLIT = Text which contains the characters or Cell Value

DELIMITER CHARACTER = Delimiter to split the Text by. E.g. Comma, Semi Colon, Pipe, = etc.

RETURN Nth POSITION = Once text has been split by the Delimiter into number of parts, which position or part do you want to return.

Example

Cell A1 = "http://www.analyticshawk.com/pages.html?file=abc;size=100;colour=blue;user=sa"

Formula

=a_SPLITME (A1, ";", 3) will return blue

=a_SPLITME (A1, ";", 4) will return sa

=a_SPLITME (A1, ";", 2) will return 100

=a_SPLITME (A1, ";", 1) will return http://www.analyticshawk.com/pages.html?file=abc
But if we only wanted to see file=abc we could nest another a_SPLITME function as such;

a_SPLITME (a_SplitMe (A1, ";", 1), "?", 2)

This will then Split the text by ? and return the 2nd Text.

13. a_Workbook

a_Workbook Contains various Excel information regarding the file, location, size, date created, last saved, author etc.

Some of these information can be easily found using CELL () function but for most of these you'll need to write VBA code or look elsewhere.

The results returned by a_Workbook () can be useful in dashboard environment or even for labelling purpose.

How to use?

Syntax: =a_Workbook (<PARAMETER>)

PARAMETER = 1 to xx see below for xx representations

Example

PARAMETER = 1 to xx	Formula	Result
1. Active Workbook Location (Directory)	=a_Workbook (1)	C:\my Directory\Project1\
2. Active Workbook Name	=a_Workbook (2)	Monthly Sales Report
3. Active Workbook File Size	=a_Workbook (3)	768kb
4. Active Workbook Author	=a_Workbook (4)	Richard Gear
5. Active Workbook Last Saved Time	=a_Workbook (5)	22/01/2014 10:30
6. Active Workbook Date of Creation	=a_Workbook (6)	20/01/2014 17:30
7. Active Workbook Visible Number of Sheets	=a_Workbook (7)	3
8. Active Workbook Hidden Number of Sheets	=a_Workbook (8)	2
9. Active Workbook Active Sheet Name	=a_Workbook (9)	Sheet 1
10. Active Workbook File Version (2003, 2007, 2010, 2013 etc)	=a_Workbook (10)	2010
11. Active Workbook File Extension	=a_Workbook (11)	.xlsx
12. Active Workbook File Attribute	=a_Workbook (11)	Read, Write

14. a_MACID

a_MACID returns the MAC ID of the PC. MAC ID can be used to uniquely identify a computers Network Adapter. If the PC has several Network Adapters, MAC ID lists all.

This formula can be used to determine if specific tasks need to be performed on a specific PC (although computer name can be used but you'll need to write a VBA code). MAC ID can be used to authenticate communication with other workbooks (allow only valid MAC ID's to update the workbook etc.)

How to use?

See compatibility guide below for your MS Excel version.

Syntax: =a_MACID ()

15. a_isWTD

a_isWTD returns TRUE or FALSE based on date parameter passed.

Week To Date is based on a Sunday to Saturday of the week that just completed.

The week that is in running i.e. current week or this week is NOT considered as Week To Date.

This function is very useful in determining the date range and whether to include as part of WTD calculation or not.

e.g. if today is Wednesday 16th December 2015, it will be considered as current week since this week hasn't completed yet. (Note Sunday to Saturday as being the week. Sunday first day and Saturday last day).

Week To Date is calculated to be up to Saturday 12th December 2015.

From Sunday 13th December 2015 it's considered to be current week or also known as this week.

How to use?

Syntax: =a_isWTD (<DATE>)

DATE = Date which you're checking

Example

Assume today is Wednesday 11th March 2015.

Table 1 below shows Date To Check in Column A and isWTD formula in Column B. As you can see, for the dates between 1/03/2015 to 7/03/2015 WTD is TRUE. Column C shows the formula used in Column B.

Table 1

	A	B	C
1	DATE TO CHECK	isWTD	Formula
2	20/02/2015	FALSE	=a_isWTD(A2)
3	21/02/2015	FALSE	=a_isWTD(A3)
4	22/02/2015	FALSE	=a_isWTD(A4)
5	23/02/2015	FALSE	=a_isWTD(A5)
6	24/02/2015	FALSE	=a_isWTD(A6)
7	25/02/2015	FALSE	=a_isWTD(A7)
8	26/02/2015	FALSE	=a_isWTD(A8)
9	27/02/2015	FALSE	=a_isWTD(A9)
10	28/02/2015	FALSE	=a_isWTD(A10)
11	1/03/2015	TRUE	=a_isWTD(A11)
12	2/03/2015	TRUE	=a_isWTD(A12)
13	3/03/2015	TRUE	=a_isWTD(A13)
14	4/03/2015	TRUE	=a_isWTD(A14)
15	5/03/2015	TRUE	=a_isWTD(A15)
16	6/03/2015	TRUE	=a_isWTD(A16)
17	7/03/2015	TRUE	=a_isWTD(A17)
18	8/03/2015	FALSE	=a_isWTD(A18)
19	9/03/2015	FALSE	=a_isWTD(A19)
20	10/03/2015	FALSE	=a_isWTD(A20)
21	11/03/2015	FALSE	=a_isWTD(A21)

How do you use this?

You can now Pivot this data and apply as filter. The corresponding sales data will be filtered to show Week To Date sales only.

16. a_isWTDPY

a_isWTDPY returns TRUE or FALSE based on date parameter passed. It evaluates the date and returns TRUE if the date is in Week To Date Prior Year or FALSE otherwise.

Week To Date Prior Year is determined by Sunday to Saturday of the week that just past in Prior Year i.e. Until last Saturday of Prior Year.

This function is very useful in determining range of dates and whether to include as part of calculation or not.

How to use?

Syntax: =a_isWTDPY (<DATE>)

DATE = Date which you're evaluating

17. a_isYTD

a_isYTD returns TRUE or FALSE based on date parameter passed.

Year To Date is defined as all dates between 1st July and latest Week To Date. 1st July is the beginning of the Financial Year hence why it's the starting point. Latest Week To Date simply means the last Sunday date.

How to use?

Syntax: =a_isYTD (<DATE>)

DATE = Date which you're checking

18. a_isYTDPY

a_isYTDPY returns TRUE or FALSE based on date parameter passed.

Year To Date is defined as all dates between 1st July and latest Week To Date. 1st July is the beginning of the Financial Year hence why it's the starting point. Latest Week To Date simply means the last Sunday date.

However in this case it checks for YTD but for the Year Prior. It's useful when comparing Year On Year dates.

How to use?

Syntax: =a_isYTDPY (<DATE>)

DATE = Date which you're checking

19. a_isZERO

a_isZERO checks if the specified value is "0" and returns TRUE or FALSE. There are multiple uses for a_isZERO for example it can be used to validate the input before performing a calculation. One can argue that a_isZERO can be performed via IF statement. This is true but IF statements tends to get complex and creates the formulae length longer than necessary.

How to use?

Syntax: =a_isZERO (<PARAMETER>)

Example

Cell Value	Result
-200	FALSE
ABC	FALSE
0	TRUE
	TRUE
999	FALSE
100	FALSE
ABC123	FALSE
ABC000	FALSE
'0	TRUE

20. a_IFBETWEEN

a_IFBETWEEN checks a value to see if it's within the supplied MIN and MAX value, then returns the OUTPUT based on the parameter.

How to use?

Syntax:

= a_IFBETWEEN (<VALUE TO CHECK>, <MIN RANGE>, <MAX RANGE>, <OUTPUT IF TRUE>)

a_IFBETWEEN () works with text values as well. You can evaluate if a text is within MIN Alphabet and MAX Alphabet. However, keep in mind that UPPERCASE comes before LOWERCASE as per example below on row 14 and 15.

Example

	A	B	C	D	E	F
1	VALUE TO EVALUATE	MIN RANGE	MAX RANGE	IF BETWEEN MIN AND MAX, RETURN	a_IFBETWEEN	Formula in COLUMN E
2	50	0	100	0 to 100	0 to 100	=a_IFBETWEEN(A2,B2,C2,D2)
3	0	0	100	0 to 100	0 to 100	=a_IFBETWEEN(A3,B3,C3,D3)
4	100	0	100	0 to 100	0 to 100	=a_IFBETWEEN(A4,B4,C4,D4)
5	-1	0	100	0 to 100		=a_IFBETWEEN(A5,B5,C5,D5)
6	100.1	0	100	0 to 100		=a_IFBETWEEN(A6,B6,C6,D6)
7	Alice	A	F	a to f	a to f	=a_IFBETWEEN(A7,B7,C7,D7)
8	FRED	A	F	a to f		=a_IFBETWEEN(A8,B8,C8,D8)
9	RICHARD	A	F	a to f		=a_IFBETWEEN(A9,B9,C9,D9)
10	Brett	A	F	a to f	a to f	=a_IFBETWEEN(A10,B10,C10,D10)
11	BOB	A	F	a to f	a to f	=a_IFBETWEEN(A11,B11,C11,D11)
12	NATHAN	A	F	a to f		=a_IFBETWEEN(A12,B12,C12,D12)
13	f albert	A	F	a to f		=a_IFBETWEEN(A13,B13,C13,D13)
14	abc	A	F	a to f		=a_IFBETWEEN(A14,B14,C14,D14)
15	abc	A	f	a to f	a to f	=a_IFBETWEEN(A15,B15,C15,D15)
16						

21. a_ifZERO

Returns a value you specify if a formula evaluates to "0"; otherwise, returns the result of the formula. Use the a_ifZERO function to trap and handle "0" in a formula.

This formula is really useful when creating dashboards or evaluating zero value. Zero value is a common check field e.g. division by zero errors can be captured.

How to use?

Syntax:

=a_ifZERO (<PARAMETER 1>, <PARAMETER 2>)

=a_ifZERO (<VALUE TO EVALUATE>, <VALUE TO RETURN IF RESULT IF ZERO>)

Returns PARAMETER 1 if value is not "0". If Value is "0", returns PARAMATER 2.

Example

	A	B	C	D	E
1	Territory	2014 Sales	2015 Sales	% Change	Formula used in Column D
2	Area A	1500	2000	33%	=a_ifZERO((C2-B2)/B2,"No Change")
3	Area B	1300	1200	-8%	=a_ifZERO((C3-B3)/B3,"No Change")
4	Area A	1900	1900	No Change	=a_ifZERO((C4-B4)/B4,"No Change")
5	Area B	1800	1600	-11%	=a_ifZERO((C5-B5)/B5,"No Change")
6	Area A	1200	1200	No Change	=a_ifZERO((C6-B6)/B6,"No Change")
7	Area B	1000	1100	10%	=a_ifZERO((C7-B7)/B7,"No Change")
8	Area A	1750	1750	No Change	=a_ifZERO((C8-B8)/B8,"No Change")
9					

AH Tools Ribbon

AH Tools is equipped with some of the fastest and effective tools you will ever need in MS Excel.

These tools are powered using Macros and custom codes written in the backend.

A Note Of Warning:

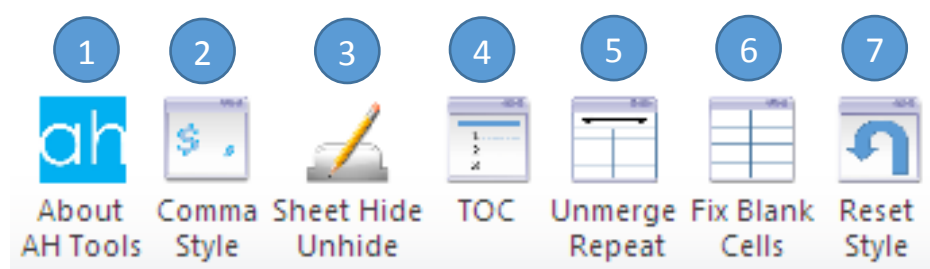
As all macros built in MS Excel, there are certain precautions you should take and be mindful of when using it. Macros are custom built for a purpose and in most cases once the macro starts, it can't be easily stopped. The output of the macro or the functionality can't be undone using the "UNDO" button of MS Excel.

This is not AH TOOLS issue but such is the way MS Excel performs and treats macros. You should also know that macros can be harmful and may not always work as intended.

Analytics Hawk is not responsible for any mishaps or functionality issues or data loss or changes arising out of this.

These statements were not meant to scare you or to put you off using AH Tools. It's the case with all macros and addins just in case if you weren't aware of it.

AH Tools Ribbon currently showcases the following functionality.



1. About AH Tools – shows version
2. Comma Style – changes the cells format to comma style
3. Sheet Hide Unhide – Allows user to see all sheets and hide / unhide in a batch
4. TOC – Creates Table of Contents for all sheets in the workbook
5. Unmerge Repeat – Unmerge the Cells and Repeat the Data
6. Fix Blank Cells – Fixes MS Excel error where Blank Cell sometimes doesn't format correctly
7. Reset Style – Reset all Style sin the workbook to default MS Excel default only

22. Comma Style

Comma Style formats the selected cells (of numeric value) to easily readable comma format.

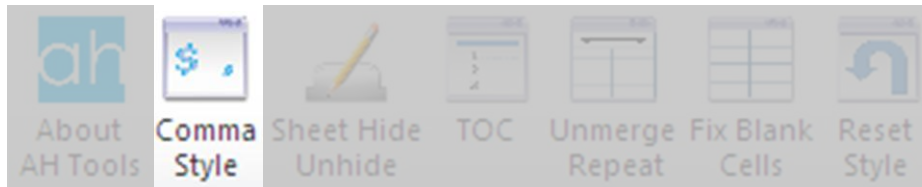
This is not a formula but a function applied to the selected cells.

How to use?

1. Data below shows Sales Revenue. However, the figures can be bit hard to read e.g. making out thousands, 10 thousands etc. Select the cells which you want the formatting to be applied.

Sales Revenue	
	35216.25
	21543.39
	21546.25
	34568.78
	35987.14
	29834.25
	34794.89
	31937.29
	31649.93
	27349.16
Total	304427.33

2. From **AH Tools** Ribbon, Click on **Comma Style**



3. Your data should now be formatted as below.

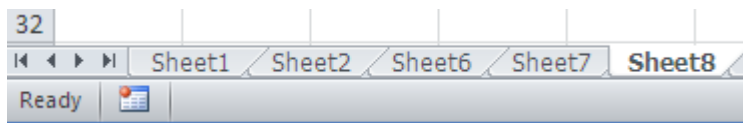
Sales Revenue	
	35,216
	21,543
	21,546
	34,569
	35,987
	29,834
	34,795
	31,937
	31,650
	27,349
Total	304,427

23. Sheet Hide Unhide

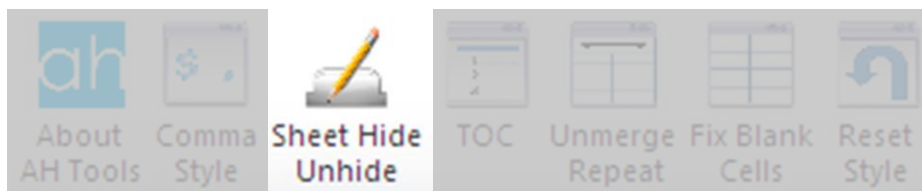
Sheet Hide Unhide allows the user to easily view all sheets and its visible status.

How to use?

With the example below Sheet3, Sheet4 and Sheet5 are hidden. We'll use **Sheet Hide Unhide** button to make them all visible.



1. On Sheet8 click on **Sheet Hide Unhide**.



You should get a listing of sheets as below.

Update	
Sheet1	Visible
Sheet2	Visible
Sheet3	Hidden
Sheet4	Hidden
Sheet5	Hidden
Sheet6	Visible
Sheet7	Visible
Sheet8	Visible

2. Now change the visibility on sheet3, sheet4 and sheet5 to **Visible** as below

Update	
Sheet1	Visible
Sheet2	Visible
Sheet3	Visible
Sheet4	Visible
Sheet5	Visible
Sheet6	Visible
Sheet7	Visible
Sheet8	Visible

3. Now click on **Update** and all sheets should be visible. You can do the same to hide 1 or multiple sheets.

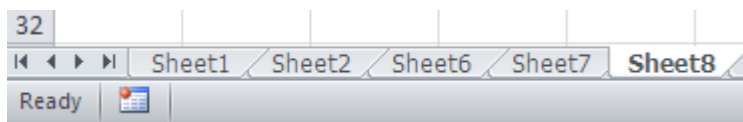
24. TOC – Table Of Contents

TOC or Table Of Contents is exactly what the name says. It creates TOC of all sheets in the document and hyperlinks to it. This is really useful when creating dashboard or a list of sheets in the document without having to manually type each of it.

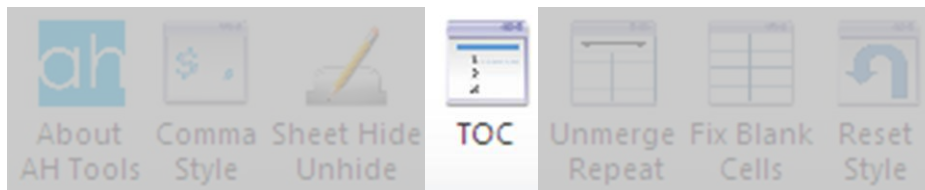
TOC works with visible sheets only.

How to use?

With the example below Sheet1, Sheet2, Sheet6, Sheet7 and Sheet8 are visible. We'll use **TOC** button to create Table Of Contents in Sheet1.

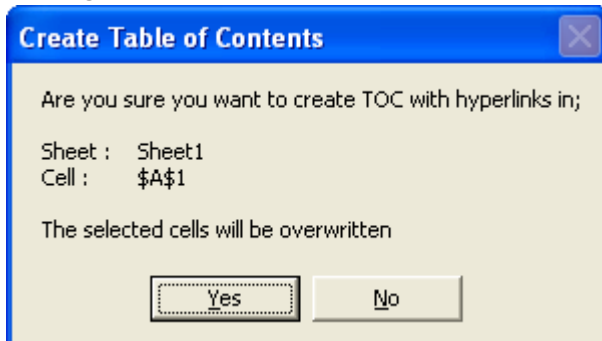


1. On Sheet1 click on **TOC**.

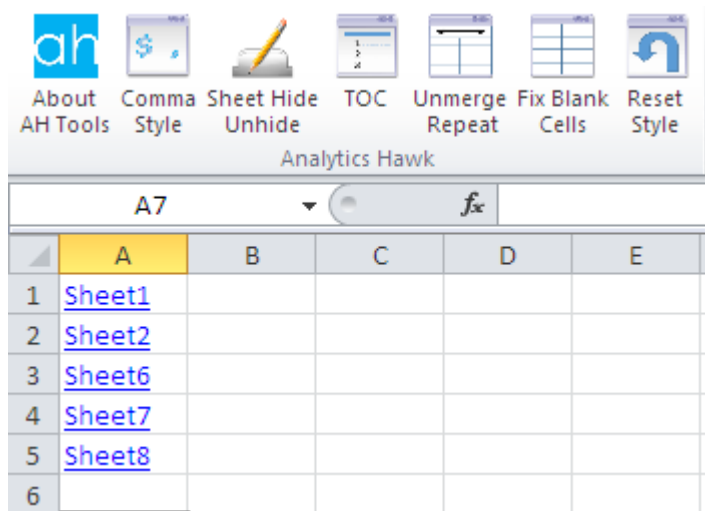


2. You will get a message to confirm TOC listing.

Note: Ensure you have enough empty cells without data (i.e blank). TOC will overwrite any existing data.



3. When your TOC is created, it will look as follows. You can customise the formatting but in the event if you ever recreate TOC, all formatting will be lost.



25. Unmerge and Repeat

Unmerge and Repeat is very useful and allows the user to repeat rows of data for blank cells. This tool is used particularly when 3rd party analytics software output data in merged cells. As analysts, we can't use the data to summarise, pivot, append to database etc. You will need to unmerge each section at a time and repeat the rows. MS Excel will allow you to unmerge all at once but **no repeating functionality**.

How to use?

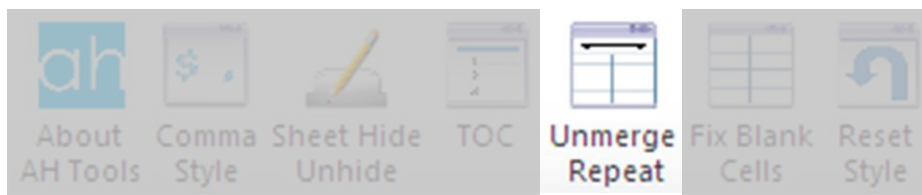
With the example below, Country is merged and shows Quarterly sales. Before we can Pivot this data, we need to repeat the Country for each blank cell.

	A	B	C
1	Country	Quarter	Sales
2	Australia	Q1	170,999
3		Q2	198,924
4		Q3	298,273
5		Q4	247,982
6	UK/EUROPE	Q1	204,975
7		Q2	137,233
8		Q3	272,649
9		Q4	145,877
10	SINGAPORE	Q1	125,606
11		Q2	296,081
12		Q3	195,883
13		Q4	210,583
14	USA	Q1	66,318
15		Q2	176,888
16		Q3	255,622
17		Q4	250,139

1. Select the Column where merged data exists. In this case we'll select Column A.

	A	B	C
1	Country	Quarter	Sales
2	Australia	Q1	170,999
3		Q2	198,924
4		Q3	298,273
5		Q4	247,982
6	UK/EUROPE	Q1	204,975
7		Q2	137,233
8		Q3	272,649
9		Q4	145,877
10	SINGAPORE	Q1	125,606
11		Q2	296,081
12		Q3	195,883
13		Q4	210,583
14	USA	Q1	66,318
15		Q2	176,888
16		Q3	255,622
17		Q4	250,139
18			
19			
20			

2. Click on **Unmerge Repeat**



3. Your data should now be **Unmerged** and **Repeated** as below

	A	B	C
1	Country	Quarter	Sales
2	Australia	Q1	230,665
3	Australia	Q2	133,745
4	Australia	Q3	153,900
5	Australia	Q4	73,746
6	UK/EUROPE	Q1	109,875
7	UK/EUROPE	Q2	92,919
8	UK/EUROPE	Q3	75,500
9	UK/EUROPE	Q4	65,663
10	SINGAPORE	Q1	124,141
11	SINGAPORE	Q2	256,840
12	SINGAPORE	Q3	228,846
13	SINGAPORE	Q4	286,604
14	USA	Q1	103,234
15	USA	Q2	204,785
16	USA	Q3	56,775
17	USA	Q4	74,110

26. Fix Blank Cells

Fix Blank Cells is advanced user tool. Generally most workbooks are fine and work as intended. However, we have seen several case where a workbook or worksheet seems to look blank but the cells are formatted in a specific style.

There is no fix for this and the user has to select all cells and change format to general or whatever is required. This can be tedious if you have several sheets and have worked on it for hours/days/months...

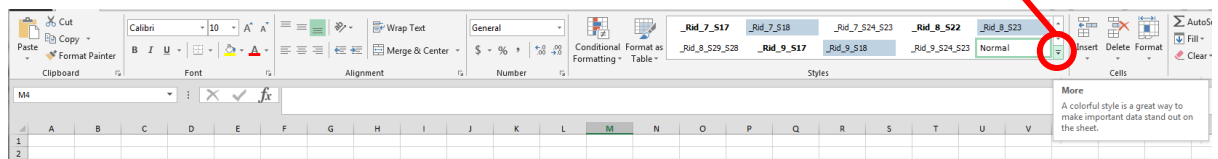
Fix Blank Cells selects all **blank cells** in the worksheet and converts the format to general.

27. Reset Style

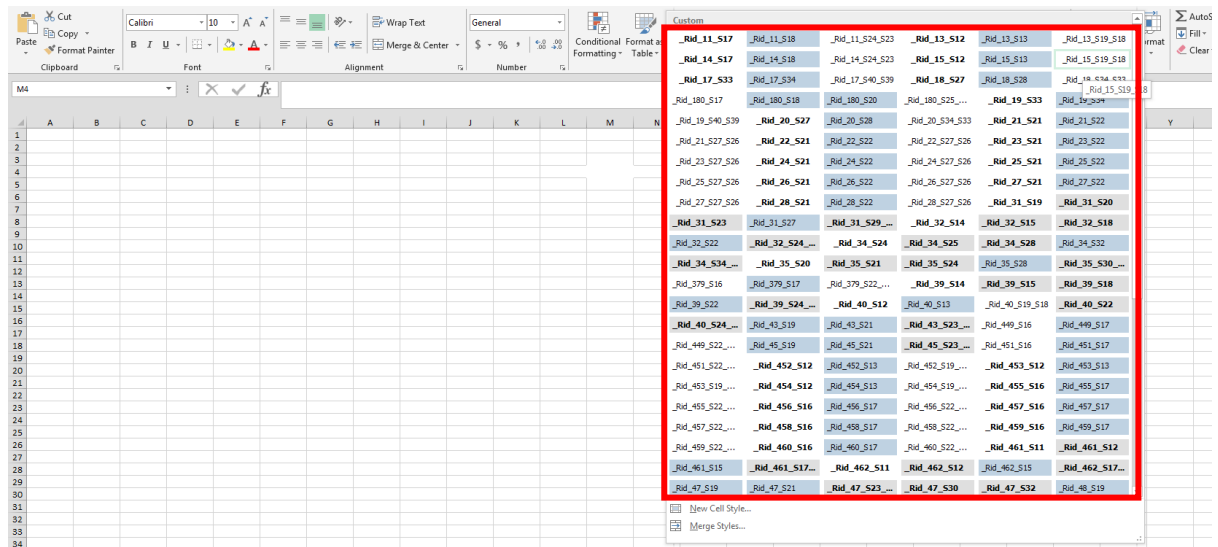
As your workbook grows overtime, so do the styles. Generally most users should be ok with custom style but in some cases where 3rd party addins are involved, workbook can become real slow with updates and will grow large in size.

Steps below shows how to view all styles in MS Excel then delete all custom styles.

1. Under **HOME** ribbon, click on the **more** arrow as follows

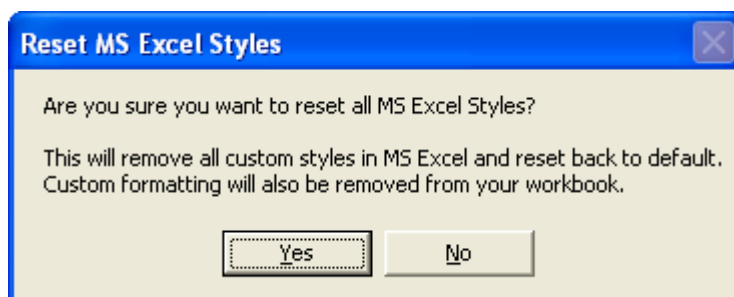
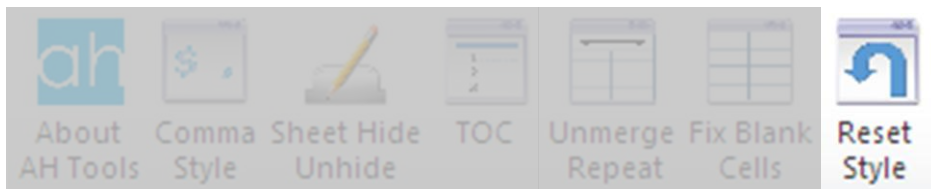


2. This will expand and show all style available in this workbook. Here you can see these style are non MS Excel built in. They have been created by 3rd party addins. The issue is, the 3rd party addin recreates these style each time the report is refreshed and it quickly increases the file size and lags the performance of the workbook.



3. Click on **Reset Style**. You will get a prompt to confirm.
Note: Reset Style will delete all custom styles in the workbook. Which means you will lose all custom colour formatting (if not MS Excel stands ones).

Tip: First create a copy of the workbook and test if this works for you.



Once the custom styles are deleted, the **Styles** drop down should only show MS Excel default colour styles.