



Transcript for Session 008

Listen to the podcast session, see resources & links:

<http://chandoo.org/session8/>

Transcript:

Hello everyone. Welcome back to chandoo.org podcast session 8. This show is all about making you awesome in data analysis, charting, dashboards and automation using Microsoft Excel.

Thank you so much for joining me in session 8 of the podcast. Today we will talk about an interesting and familiar topic. As an Analyst or a Manager, you are likely to experience both sides of this topic at least once in your life. The topic is "Help me; somebody dumped this Excel workbook on me!" I'm going to call the title - "Hey, somebody dumped this Excel file on me, what do I do!" This is something very common for us as Analysts or Managers. Quite often, we inherit workbooks that have been created by somebody else and we have no idea what the files do and if we do, we don't know how they work. And, we are supposed to maintain these workbooks. As an Analyst, I've faced this both at the giving and the receiving end. Very early in my career, I would get these emails from my bosses who'd say something like, "Hey Purna (that's my first name), can you look at this pricing model that someone else has prepared and adapt it for ABC project?" This was the theme of the emails that I used to get as an Analyst. After a couple of years when I had built some scorecards, reports and dashboard templates, I would be the one to send emails like this! I would send files to colleagues and new Business Analysts who joined the team and I would ask them to use them to adapt them to their situation.

Sometimes, I wouldn't even send them the files, but they would somehow inherit them because their existing boss or someone else in their team had given them a file created by me. After a couple of weeks of wrestling with that file, they would come back to me or someone else in the team and ask the question, "Hey, what is this file really doing?" So, this is a situation that we all are familiar with. Today, I would like to highlight some of the tips and practical tactics that we all can use when we face the problem of inheriting a workbook and we have no idea of how it works.

The best and easiest tip that I can give you is this - whenever you inherit a file, always try to spend some time with the creator of the file. This is true for Excel or anything that has a little bit of programming built into it like an MS Access tool or a PowerPivot solution that they have put together etc. Whenever you are inheriting a piece of technical software or application that you will be responsible for maintaining, it is always a good idea to do some Knowledge Transfer (what we call 'KT' in the software industry). This sounds like a jargon term invented by Managers just to make it sound fancy and fashionable, but all it really takes is to take the creator out for a cup of coffee or inviting them for a working lunch to spend some time to understand how they have created the file, what their thinking process was, and what is the solution that they have adopted in the file.

Obviously, both of you are working in the same industry, so chances are high that you both know the business terms and that you both have some minimum knowledge of Excel. So, you don't exactly need



spoon-feeding, but getting a basic idea would be good so that you don't have to re-invent the wheel or figure out where the wheel is for this new workbook that you are inheriting. This is the best tip that I can give you!

But, let's be practical. Sometimes, we don't even know who created the file or we can't access the person. This is especially true of large organizations. In one of the companies that I was working at earlier, we had more than 100,000 employees. The biggest challenge for me was - I was always getting these emails from an email address, and although we had an exhaustive corporate intranet, messengers and email clients to network with the employees - still sometimes for practical reasons, it wasn't possible to reach the person because he was in a different geographical location, or on sabbatical, or on medical leave or he/she was no longer working with the company. Also, the person could have moved on to a higher domain and become the Vice President of the company and so had no time to help you out!

This is the reality. I have six helpful tips for a situation when we are supposed to maintain and change a workbook and we cannot reach out to the person who created it:

- For the sake of simplicity, let's say you have inherited a pricing model for a project in the form of workbook. You work for a company that does construction or infrastructure projects or something like that. Let's imagine that you construct bridges for a living. Someone really clever made an exhaustive Excel workbook where one could enter all the parameters for constructing a bridge and it would display the final cost that we should bid for the project. So now we are looking at a workbook that has 315 input variables and it gives you an output sheet that looks nice and clean and it gives you the detailed project plans, costs and timelines, but you have no idea how it works! It's almost like a black-box! You've inherited a pricing model and you are trying to de-mystify and understand it so that you can probably make some enhancements - for example, the bridges could be made with a new metal that they have just discovered or constructed, instead of with steel. You will need to change the model in such a case, but how do you do that?

So, the very first tip that I have for you is called - 'modeling real life on paper'. Even before you look into the workbook and try to de-mystify it, imagine yourself in the shoes of the person who created the model. How would she have done it? The very first and basic step would not be to open up the Excel workbook and go to cell A1 or B1 or C2 and start typing the details. That would not have been the first step that she would have done. The first thing that she would have done is to create a mock-up model of the scenario on a piece of paper with some boxes which would be more like a mind map or a small diagram that depicts the individual components of the pricing model. You need to do the same thing - so take a pen and paper and try to draw the model using some boxes - draw the boxes, name them and try to connect them with one another.

For example, in the case of this bridge pricing model, the easiest thing that comes to my mind (and I haven't ever created a bridge outside of Leggo!) is that I would probably have to have a place where the raw material costs would be maintained, for example unit costs for steel, cement, bricks and a few other things that would go into the making of a bridge. So the raw material unit costs need to be maintained and we know from our experience that the unit cost of raw material is not a wildly fluctuating number. It's probably something that stays within a range of values. The second thing is that the construction of the bridge depends on obviously critical parameters like the length and



width, and whether it's a two-lane bridge for 100 meters or a four-lane for 1.5 miles. Depending on these parameters, the cost would go up or down. There are other inputs that talk about the nature of the bridge. These are the kinds of things that the boxes would depict on paper.

Once you've created a model of the workbook on paper, you can then try to identify all these boxes on paper in your Excel workbook. Obviously there would be a place where the raw material unit costs or some sort of thing like that is maintained in the workbook. Likewise, parameters like the dimensions, type of construction (hanging vs. column bridge etc.) will also be transferred to the Excel workbook. That will help you understand the overall layout of your workbook in a very easy way. It is a lot better than trying to jump into the file and figuring it out by yourself by changing values or tweaking things - that's a very crude approach. A better approach would be to make a realistic model on paper and then see how your workbook aligns with the model.

- Locate the engine - The challenge comes into the picture when we are looking into complex workbooks like the pricing model of a construction project or an Excel application that takes a lot of inputs and puts them in the database. In such cases, we don't know how things are working until we examine them. In all complex models, there would be an engine. An engine would usually be a bunch of formulas and VBA code that is hidden somewhere and isn't apparently visible. For the sake of simplicity, let us imagine that the engine here is a bunch of formulas. We will talk about VBA a little later in the podcast. Let's first address the concern of formulas. Any model where you're looking at a lot of calculations would be done by means of a bunch of formulas. How do we locate where these formulas are? If the model or workbook is used following some sort of conventions of the best practices of models, then it is easy for us to right away jump into the pages where the calculations are done. Sometimes inheritable workbooks are problematic because they are constructed in a very haphazard or complicated way. We don't know where the inputs are and which ones are assumptions and calculations. In such cases, there are a bunch of features in Excel that can help us. You can show all the formulas by using the 'show formulas' button in the Excel ribbon or you can press Ctrl and back quote.

I will print this shortcut on the show notes page which is located at <http://www.chandoo.org/session8/>. Here you can access the show notes, transcript, links and resources mentioned in this podcast. The shortcut for revealing all the formulas in your current workbook is Ctrl and back quote or the 'show formulas' button in the formulas ribbon.

Apart from this, there are two things that are also very useful - trace precedents and trace dependents. These two buttons in your formula ribbon are very useful as they help you understand how your formula is calculated. They're not going to tell you the calculation steps, but they can tell you which cells are feeding the calculations or which formulas are dependent on the cell value. So trace precedents and trace dependents, as the name suggests, can trace all the cells that are linked to this cell. These are very useful especially if you are looking at a complex model where everything depends on everything else. Using trace dependents and trace precedents is quite useful in such cases.

There is also another tool in the Excel formula ribbon called 'evaluate formulas' that is quite useful to evaluate the formula and step through each and every step of the formula and see how it is calculating the value that is printed on the screen. This is especially useful if you are trying to



understand how a certain formula is arriving at the result. We will talk a little bit more about formulas, VBA and how to understand them in the latter half of this podcast.

Finally, there is one more very powerful tip that not many Analysts know about which is called 'Go to Special'. This is an almost secret functionality in Excel because it's not apparently visible on the main screen. If you press the F5 function from within any cell in a workbook, it will show the 'Go To' box. This is your ticket to go to special type of Excel workbook cells. In your model, you would obviously have a few cells or range of cell addresses that are dedicated for inputs and some others will have formulas. When everything is on the same page and you don't know what is what, you can use 'Go to Special' to highlight just the formulas only. This is one of my favorite ways to quickly identify which cells contain formulas. When you select all the formulas with 'Go to Special', it will select all the cells that only contain formulas. Once it has selected all the cells with formulas, you could for example fill up these cells with a yellow fill in color or make the cell font color red, so that they stand out because these are formula cells.

As somebody who inherited the workbook, and since your job is to analyze and understand it better, you are going to spend a lot of time looking at the engine part of the workbook which is the formulas. So it's important to isolate those formulas before you can do anything meaningful with them and that's where the 'Go to Special >> Formulas' comes into the picture. This is the second tip - "locate the engine", or in plain English "locate the formulas, see where they are and how they are linked to each other".

- The third tip is that under the hood there is more than the engine. I'm trying to be cryptic here, but what I really mean is that apart from formulas there is a lot more in your model that is not visible but still helping. These are things like hidden sheets which are hidden but are still there and adding a lot of functionality to the workbook. So you want to make sure that there are no hidden worksheets. Sometimes these are not even visible when you try to unhide them, as they are set up as 'very hidden'. Using the VBA window of Excel, you can select any worksheet and set it as 'very hidden'. If this is the first time that you are hearing about the very hidden feature of Excel, I would encourage you to bookmark the keyword 'very hidden' in your mind and search for it on Google later when you have some time. When you right click and choose 'unhide', a 'very hidden' worksheet will not get unhidden. You will have to go to the VBA console or Visual Basic Editor to unhide those sheets. There could be hidden or very hidden sheets, hidden columns, hidden rows and my favorite 'white-washed' cells.

Have you ever heard of 'white-washed' cells? Well, I'd never heard about them till two years after I started working as an Analyst. A 'white-washed' cell is a worksheet cell that contains a value (whether an input, value or output) but it appears as a blank cell to our eyes. This is because somebody decided to select the cell and change the font color to white. Since the cell background is also white, it looks blank when you look at it but it contains a value and it is doing something in the background. This is called a 'white-washed' cell. Of course, there is a better way to white-wash a cell which is to change the cell's custom formatting code so that it shows up as nothing on the screen even though it contains a value. Either way, whether you are painting white color and changing it or whether you are custom formatting the code to change it, 'white-washed' cells can be a lot of pain especially for people who are trying to understand a workbook and figure out what it is doing. This is where it is important to locate those things.



Of course, there is no quick and easy way, especially if somebody has used custom cell formatting as finding those cells could be a little tricky but it can be done using 'Go to Special' etc. Look out for hidden sheets, hidden columns, hidden rows and 'white-washed' cells etc.

There are two things that people probably know about - one is VBA code which is usually not visible on the workbook itself but you have to right click on the worksheet name area at the bottom and then go to 'View Code' and all the modules and VBA code will be visible there. So, in other terms, VBA code is behind the Excel workbook and you have to go and look for it, otherwise you won't realize that it is there.

The other place where some of the logic can be hidden is the 'name manager'. People who have created the model could have created some name formulas and named ranges where data is kept or manipulated. Go to the 'name manager' in the formula ribbon and make sure that if there are any names you understand what they are doing, what the intention is and how they calculate and where they are being used etc. This is the third tip - i.e., understand the hidden components and go and examine them.

- The fourth tip, when you have inherited a clumsy or complicated workbook and you are trying to figure out what it is doing is to annotate as you go. Do not wait until you understand the entire workbook. As and when you discover something that is strange or interesting, make sure you apply a cell comment to that cell and put down your thoughts there itself. If that is a difficult thing for you to do, another practical suggestion that I often follow is to print out the workbook if it is in a reasonably good printable format and add notes on it so that it's easy for you to see where things are and how they link up with each other. So, annotate as you go. Don't finish until you finish understanding the whole thing because you'll forget what cell C3 is doing by that time!
- The fifth tip is to locate the controls. Any workbook, especially workbooks like our pricing model workbook will have a lot of things that control the output. Based on the input and values that you type, the output will change. You need to locate these input values, mark them, understand their purpose and how they interact with the formulas. Likewise there would obviously be some sort of settings or assumptions. In our case, for example, if you are a company based in New Zealand and you construct bridges all over the world and you are preparing a pricing mode where the output is in US dollars and not New Zealand dollars. So, in your workbook model there would be a cell somewhere which would tell you the conversion rate for one New Zealand dollar to one US dollar. How many US dollars per New Zealand dollar? There would be some variable somewhere. This is more like an assumption than an input factor because we just assume that for every New Zealand dollar you would get about 75 cents in US. So you just assume that and you put that value in a cell, and if you don't locate that particular cell in your analysis or understanding of this workbook, you would always wonder why the wages and material costs that are being entered in New Zealand dollars are displaying as US dollars in the output. So, it's important to locate those settings or assumption cells.

Likewise, you also need to locate any scenarios. Somebody might choose to set up the conversion rate as a scenario where the exchange rate fluctuates between 60 cents to 90 cents. They would set up a scenario and use one of those values. How would the model know which value is being used? - Probably by picking a scenario that uses manual input or a form control or something like that. So,



again, you need to figure out what the form controls that could control or determine the scenarios are, and what the assumption and input cells are. These are important. Only when you know these things can you understand the engine or the core component better.

- The last tip that I have for you is that sometimes the workbook that you've inherited is so complicated, clumsy and unintelligible, that you just have to re-construct it. This has happened to me several times in my life. It hasn't happened so many times that I hate it, but there are many times when I would get a workbook like say a customer-relationship management tool. But the workbook contained so many formulas, input cells and an entire rainbow of colors, and I wouldn't understand anything. In such cases, I would always feel the urge to re-construct the whole thing instead of bothering with breaking it down, locating the formulas, VBA code etc. Instead, I would just figure out what it's supposed to do and go and construct it again. This is the kind of thing I used to do when I was a young Analyst and as a very new person in the company, I always had an urge to create things so that I could learn faster. I felt that instead of trying to understand a workbook that somebody else had built, I could probably re-construct it if I had 3 or 4 hours' time. And, along the way, I could learn a lot more a lot faster. I encourage this.

Even today, if I am looking at a problem and I go and search for a solution on the internet for an appropriate VBA code snippet or a formula technique, most of the times I don't copy and paste it into my workbooks. I try to see the kind of logic that they applied and come back and try to re-implement it and maybe add my own twists to it, so that I can learn a little more than I can instead of just copying and pasting. This is something that I encourage you to do, especially if you are fed up of the whole understanding thing or you don't know what the colors used mean etc. You can always re-construct. Just keep in mind that sometimes re-construction is a very costly activity especially if you have a deadline. If you have a lot of time, or if you have the luxury of convincing your boss that if they wait for another week you can give them a better model, then go and make it by all means as you will learn a lot more through this method.

These are the tips that I have for you for understanding an inherited workbook better. Just to give you a recap, tip zero is to talk to the person who made it. Take them out to tea, coffee or lunch and discuss it.

If that's not possible, then the six tips that I have for you are:

- Model the entire thing on paper and try to locate the parts in your workbook. Make a sketch on paper and then see how it correlates with the workbook that you have at hand.
- Locate the engine components which are essentially the formulas in the workbook. To locate them, you could use the 'show formulas' button or trace precedents and trace dependents buttons, or the 'Go to Special' formulas features of Excel.
- Likewise, there could also be some hidden components in the workbook especially hidden worksheets, 'very hidden' worksheets, hidden rows, hidden columns, names and 'white-washed' cells etc. You need to identify these hidden components.
- Annotate things as and when you find something interesting, by using the cell comments feature.



- Locate the controls, i.e. the values, cells and inputs that control your model. Locate them and annotate them. There are mainly three types of controls - input values, assumptions and scenarios. Identify and demarcate them.
- If all else fails and you are just frustrated trying to understand the workbook, you can just reconstruct it from scratch.

Now that we have talked at length about various aspects and techniques that you can use for inheriting a workbook and maintaining it better, let us deep dive into two key areas of workbooks. Obviously when you inherit a workbook like this, everything else could be easy to understand like the business concepts since you are in the same industry and so concepts like pricing are easy to understand from a modeling point of view. But the things that we struggle with are the formulas and the VBA code. These are the core engine components of the workbooks. To understand them, let's talk about formulas and VBA a little bit.

When it comes to the topic of understanding formulas, the very basic and crucial thing for us is that we need to know the language. Well formulas, in themselves, are not a language. But, what I am referring to as language here is the terminology of formulas. As an Analyst, it is very important for you to know the basic and important formulas like the LOOKUP formulas, SUM(), SUMIF(), COUNTIF() and work-specific functions. So, if you are in the financial modeling industry you would probably be talking a lot about modeling of depreciation, interest etc. If you are in an industry related to statistics, then you need to know the statistical formulas. If you don't know what a T-test is, it would be very difficult for you to understand what the T-formula is doing in Excel. When it comes to formulas, the most important thing would be to know the lingo. To know the lingo, you can do a lot of things, but the easiest thing that comes to my mind since you are already listening to the podcast is to just go back to the archives of our episodes and look for episodes 2 and 7. Episode 2 of our podcast talks about the LOOKUP functions in Excel, with special emphasis on VLOOKUP() functions. Episode 7 talks about SUM() formulas in Excel. It talks about SUMIFS(), COUNTIFS() etc. These are two very good resources for you to pick up all the vital skills required when it comes to learning important formulas.

The second important thing, when it comes to understanding formulas, is to use the F9 key. This is probably your best friend when it comes to understanding a long, complicated formula. The way the F9 key works is that it can evaluate a portion of the formula for you. Let's say that you have a very long formula involving IF, VLOOKUP, SUM etc.; when you press enter in the cell you just get one value for the entire formula. That makes it very difficult to understand how the formula is working and what the nested components of the formula are and what each little component is calculating.

That's where F9 comes handy. You can select a little portion of the formula like a sub-formula or a smaller expression of the formula and press the F9 key or the Ctrl and = key. Excel will tell you the answer for only that portion. I know it is kind of tricky for you to imagine this in a podcast, so I encourage you to visit the <http://www.chandoo.org/session8/> where I will provide a resource and a demo of how the F9 key works. So go and check it out. Don't forget that the link is <http://www.chandoo.org/session8/>. That's about the F9 key.

Likewise, it is also a good idea to examine the names in the 'name manager' as I've already told you in the earlier part of this podcast. Name manager can contain a lot of mysteries especially if you have



inherited a workbook from a good Excel craftsman like someone who's listened to a few podcast episodes before you did or someone who has been following chandoo.org for a while. Check out the 'name manager' and see if there are any clues or mysteries there. Most of the models that are constructed follow a kind of pattern which goes something like this - the person would first develop one formula and then the same formula would be copied and pasted in various places and only a variable or two would be changed, usually through relative referencing. If your model contains 300 formulas, it doesn't mean that you have to understand 300 formulas. You may have to just understand 3 formulas which are pasted 100 times each. The formulas may look different because in one formula the cell reference looks like A1 and in the second formula it looks like A2. But the entire formula in itself is the same. So, it's important for you to identify these patterns in the formulas that are repeated, isolate them and understand just one of them. Once you get it, you get the whole bunch. Look for these kinds of big wins.

The other tip that I have for you is not about understanding formulas but about working with them. The tip is 'tackle the errors' using IFERROR() formulas or something like that. Make sure that any errors in the model may not really be errors. The creator may have just forgotten to tackle them. If you see an error in the workbook that you have inherited, see if the error handling was done properly or not. And, if not, apply some IFERROR() formulas on top of it. Again, I will provide a link to the IFERROR() formula in the podcast show-notes at <http://www.chandoo.org/session8/>.

The second component, once you understand the formulas, is to understand the VBA code behind it. Again, trying to understand the VBA code is a huge subject in itself, and I doubt we can talk about it in a short while. We won't be done with it, even if I talk about it till Christmas! So, I want to highlight some of the important features of VBA that can help us understand the formulas better.

Any VBA code in the workbook has to be in one of the modules or it has to be attached to a worksheet or workbook level. Usually, good practice is to place the code in modules and event code that is attached to the workbooks is placed in the workbooks along with some public variables and a function or two. This is good practice. Your first clue would be to go and look for modules that are part of the workbook and also examine and make sure that there is no code lying in the worksheet pages or workbook. If it is, then see if it is really attached to something or it was just copy-pasted and not used properly. Another thing that you might recognize is that many times when people are constructing models, they might use the VBA recorder in Excel to record the macro and events to just see how they work. This creates a lot of unnecessary code and they might forget to clean it up. So your Excel model or workbook might contain a lot of code that is never used or called by anything else. Don't waste a lot of time trying to understand everything. Look at the entire code first and make sure that you are only focusing your energy on the things that matter. The person who has created the model might have just recorded the macro to see what code it produces and only copied one line from it and pasted it in another place, and left the entire recorded macro there. So, that's that.

Now we'll talk about the practical tips of VBA that can help us understand code better. My most favorite tip has to be 'stepping through the code'. This is a built-in functionality of Visual Basic editor where you can go to a module, sub or macro and run it one line at a time. After each and every line, VBA will stop so that you can examine everything, and then go to the next line. Instead of running a 20 line macro in one go, Excel will run it in 20 little steps pausing after each step, so that you can understand the code and its effects better. Stepping through the code is the best and most obvious way to understand the



code.

Another common technique that I use is to insert a number of breakpoints in my code. If I don't want to step through the code because I understand what most of it is doing, I'll put a breakpoint at line number 21 for example if I don't understand what is happening at line number 21. Once I put a breakpoint at that line and wait for Excel to stop there, I'll then examine all the variables and make sure whether or not the calculations are as per my expectations.

A tip here is that in either break-mode or stepping mode, if you place your mouse pointer on any variable and wait for a fraction of a second, Excel will tell you what the value is in the form of a small yellow-colored pop-up. You can also add a watch to a variable so that you can examine it in real time as the macro is progressing.

But, I'm more like a crude person because I learned my programming in C language and command prompt with Basic and those kinds of things way back when I was a kid. So, I personally prefer using print statements when I am testing my macros. I suggest that you do the same. You can use 'debug.print' in VBA to print the value after every few lines and understand how the calculation is done. This will print the values in the immediate window. Immediate window is a window in your VB editor that shows these kinds of values. If you don't see the immediate window, press Ctrl+G to turn it on or off. You can also toggle it from the 'view' menu.

You can also use a message box if you want to see the values pop-up in real time.

Another common technique that I apply when one is looking at a large piece of code, or code with several functions and you don't know what one of them is doing, you could isolate that and run it by copy-pasting that code and putting it in a new workbook and supplying it with some dummy data. This is probably a costlier technique in terms of time and investment on your part, but it is a good one.

The best practice that you should follow when you are examining VBA code is that, as and when you identify and learn something, add comments. One of the reasons why you are wasting a lot of time examining the VBA code is because the earlier creator has not commented it out. So you shouldn't repeat the same mistake; so as you understand a bunch of lines and why a variable is calculating the way it is, just add the comments in the code itself so that anybody else who inherits the workbook from you will not go through these kinds of problems.

I am sure there is a lot more to VBA than what I have discussed, so I encourage you to spend some time learning various aspects of VBA. You can go to our website <http://www.chandoo.org/wp> and click on the Excel VBA link at the top to learn more about it. I will provide a link to this on the show notes page at <http://www.chandoo.org/session8/>.

To summarize, the basic approach whenever you inherit a workbook from a colleague or a past worker and you don't understand most of it - the best and easiest method would be to take out the colleague for a cup of coffee and understand from them how it works. If that is not possible, then follow the tips that we have talked about in the first part of the podcast. In brief, these tips are:

- make a real world model and try to co-relate it with the workbook
- locate the formulas



- locate the hidden components
- annotate as you go
- locate all the input controls for the workbook and how they interact with various formulas
- and, in the worst case scenario, re-construct the workbook.

When it comes to examining formulas and VBA in-depth, it is always a good idea to use techniques like the F9 key to evaluate portions of the formula and examine the names.

For VBA code, you can always use techniques like 'debug.print', break-points, stepping through the code and comments so that you can understand it better.

In a nutshell, that's how I would go about understanding a workbook that I inherit. You would think, "hey, Chandoo, you're not working with anybody now since you are working for yourself, so why would you need to understand workbooks that are made by somebody else?" But, believe me, I do get a lot of emails from our readers and some of them just send me an Excel workbook and ask me to help them with that. I don't usually reply to them via email as I encourage them to do it via our forum. But sometimes, out of curiosity or out of the excitement of taking up a challenge, I open up the workbook and examine it.

But, the most important reason that I have to look at other's workbooks is when we run contests. Recently we ran a contest on our website where more than 40 people emailed me their dashboard workbooks and many of them contained VBA code. As somebody who has to judge this contest, I need to look at all these workbooks that contain a lot of complex formulas and VBA code as I'm trying to understand and figure out whether they have done it correctly or not. This is something that I always use. I get a workbook from somebody who I have never met and I have to understand it. So, these are the techniques that I follow and I encourage you to follow the same.

If you have any thoughts about how you would understand and interpret a workbook made by a colleague, please share it in the comments of our show notes page. Please visit <http://www.chandoo.org/session8/> for that.

Before I wrap up, I just want to thank you so much for tuning into our podcast and listening to it and making yourself a little more awesome. If you have some time, please go to our i-Tunes page and leave an honest feedback about this show. Your reviews and feedback help us a lot to find new listeners. And, in return, I can make them awesome - that's my life goal. So, thank you so much for taking the time to do that. If you need a link, just visit <http://www.chandoo.org/iTunes/> and that'll take you to our podcast page where you can access the link to leave a review on i-Tunes.

I also want to thank some wonderful people who have left us raving reviews on i-Tunes. We have 12 reviews on i-Tunes so far and all of them are 5-star! That makes me very happy. I want to thank Bob, sem, inrtp, shanilr, binarysolo, bigmartha and spokainwexcler. Thank you so much to all of you for reviewing our show and leaving your wonderful, thoughtful and kind feedback on i-Tunes. If you haven't done it already, I request you to just take a minute or two and provide feedback on i-Tunes so that I can help you better.

Thank you so much and you have a wonderful day ahead. Bye.