Taylor Moody

ISYS 520

December 9, 2010

Final Project

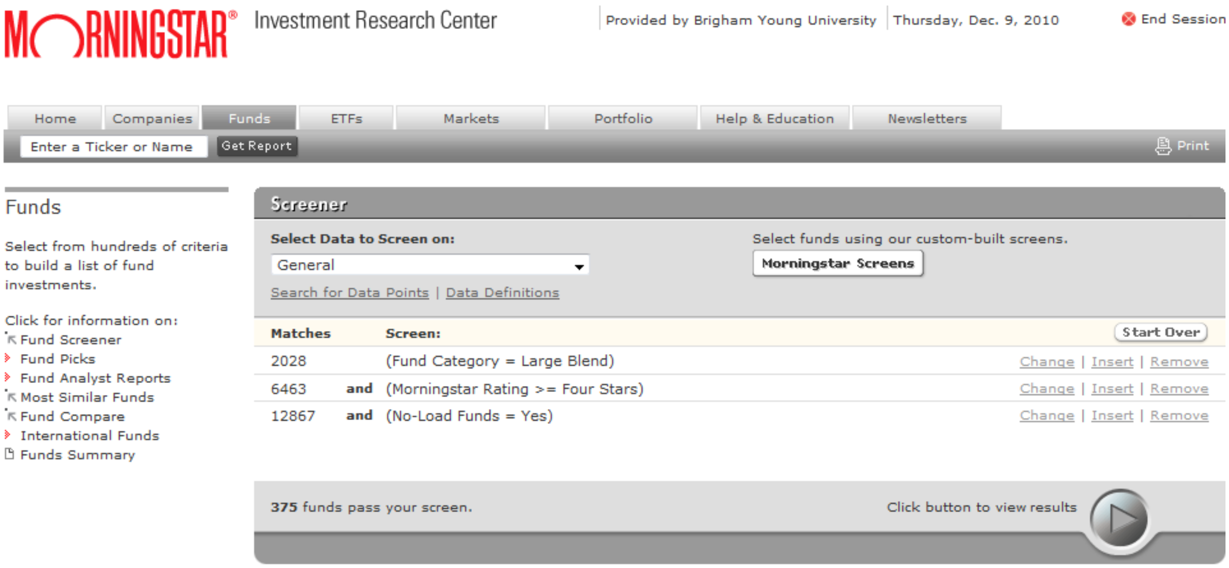
Mutual Fund Assistance Model

Executive Summary

This Project was inspired by BYU's BusM 418 Financial Planning course. It prepares students to develop a detailed Personal Financial Plan that they could implement the rest of their lives. There is quite a bit of financial analysis needed to select investments to build their portfolio. This Project mainly deals with selecting Mutual Funds, an asset class that give instant diversification to the average investor. Seeing as there are more than 8,000 mutual funds in the world today, it is quite difficult to target a mutual fund that is right for you. Throughout the course, students are encouraged to use Morningstar to identify Mutual Funds through a screening process. But, sometimes there are quite a few mutual funds after the screening making the task of identifying one mutual fund pretty daunting. That is why I created this program, to help aggregate mutual fund data onto this spreadsheet, then to organize data so it could be filtered and sorted on the students' terms. This Project will be given to the students to help guide them to financial success by assisting the process of identifying mutual funds. My goal of this project is that students may make a more well-informed decisions by using this project and in the end saving them time, and money.

Mutual Fund Assistance Model

This project was made to assist students in the targeting of mutual funds to be used in a personal financial plan. I spent hours in trying to target which mutual funds I wanted to use in my personal financial plan. Most of the time spent was switching back and forth between pages within Morningstar (a financial services data provider). For example, a screenshot of Morningstar's screening process is given below to help illustrate the problem with finding mutual funds.



Provided by Brigham Young University | Thursday, Dec. 9, 2010

Funds

Select from hundreds of criteria to build a list of fund investments.

Click for information on:

Select Data to Screen on: General

Select funds using our custom-built screens.

Matches		Screen:
2028		(Fund Category = Large Blend)
6463	and	(Morningstar Rating >= Four Stars)
12867	and	(No-Load Funds = Yes)

375 funds pass your screen.

Problem

I chose three criteria that are important for me in my financial plan. I want a mutual fund that is a Large Blend fund that has a Morningstar rating of Four stars and above with no load fees (including front and back end loads). After screening through those criteria, there were approximately 375 mutual funds that meet my above mentioned criteria. I guess I could screen those mutual funds even more, but it takes a little more time in doing so. The problem is clear. There are just too many mutual funds to feel safe that you are making a well-informed investment decision.

With regards to a Personal Financial Plan, you don't ever want to make a bad decision with regards to your investments. You want to minimize your risk as an investor by taking into account everything that is in your control (fees, management type, investment drift, etc). You can't ever control the market, but you can control many things. With so many mutual funds, it can be easy to make a poor decision if you just don't know what your options are and if you are not looking at the right data.

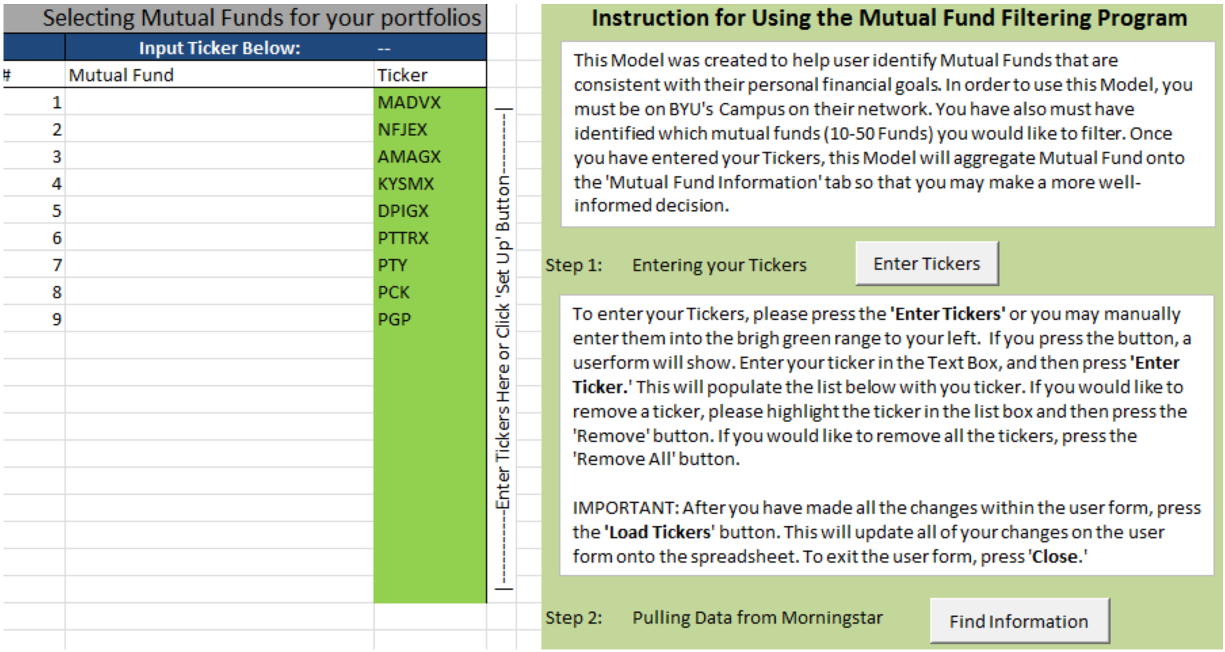
Solution

I have created this VBA model to help BYU students, even Professors, make better investment decisions by helping them pull data from Morningstar, aggregating it, and then allowing the end users to manipulate data either through filtering, sorting, or a little of both, all in hopes that the end user will be able to narrow their many mutual funds down to one or two.

This model is created for all types of users, both beginner and expert-level investors, and computer-inept to the computer-savvy. This model was created to walk users through the process of entering mutual funds tickers and pulling the data. The model is based on the fact that BYU students have already targeted their mutual funds down to 10-50 funds that they would like to take a look at. This model cannot act as a stand-alone tool for targeting mutual funds. It must be used in conjunction with Morningstar. It is also good to note that this model will only work on-campus. This is to protect Morningstar from unwarranted use by those outside of BYU. I have included a subprocedure that can log on through BYU's network and then the model would be able to access Morningstar and begin with the process.

Walk-Through

Below is a screen shot of the opening of this Model. This is very user friendly as there is include a set of instructions for using the Model.


Enter Tickers
Find Information
|-------------Enter Tickers Here or Click 'Set Up' Button-------------|

Selecting Mutual Funds for your portfolios

Input Ticker Below: --

#	Mutual Fund	Ticker
1		MADVX
2		NFJEX
3		AMAGX
4		KYSMX
5		DPIGX
6		PTTRX
7		PTY
8		PCK
9		PGP

Instruction for Using the Mutual Fund Filtering Program

This Model was created to help user identify Mutual Funds that are consistent with their personal financial goals. In order to use this Model, you must be on BYU's Campus on their network. You have also must have identified which mutual funds (10-50 Funds) you would like to filter. Once you have entered your Tickers, this Model will aggregate Mutual Fund onto the 'Mutual Fund Information' tab so that you may make a more well-informed decision.

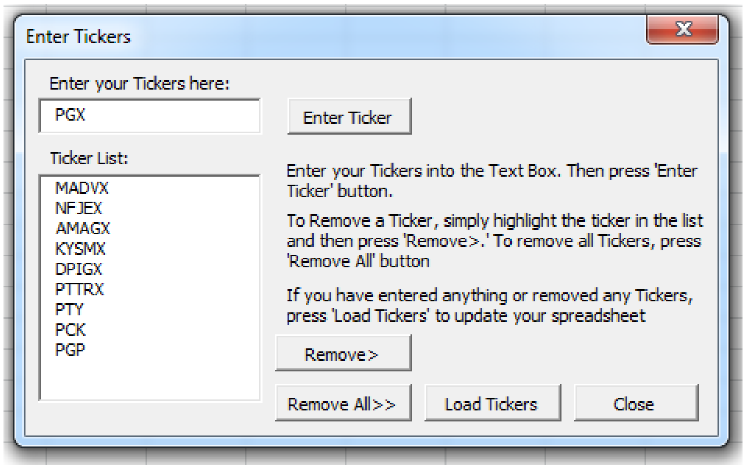
Step 1: Entering your Tickers

To enter your Tickers, please press the **'Enter Tickers'** or you may manually enter them into the brigh green range to your left. If you press the button, a userform will show. Enter your ticker in the Text Box, and then press **'Enter Ticker.'** This will populate the list below with you ticker. If you would like to remove a ticker, please highlight the ticker in the list box and then press the 'Remove' button. If you would like to remove all the tickers, press the 'Remove All' button.

IMPORTANT: After you have made all the changes within the user form, press the **'Load Tickers'** button. This will update all of your changes on the user form onto the spreadsheet. To exit the user form, press **'Close.'**

Step 2: Pulling Data from Morningstar

My Program starts out by opening to the main tab 'Input Data.' This tab is where the user inserts his/her Mutual Funds into the spreadsheet. This can be done by either of two ways. First, you may be able to enter in your tickers manually in the range that is colored Bright Green. Or, you may press the 'Enter Tickers' button located next to the Step 1: instructions box. If you choose to enter by clicking the button, a user form (included below) will pop up.



Enter your Tickers here: PGX

Ticker List:

MADVX
NFJEX
AMAGX
KYSMX
DPIGX
PTTRX
PTY
PCK
PGP

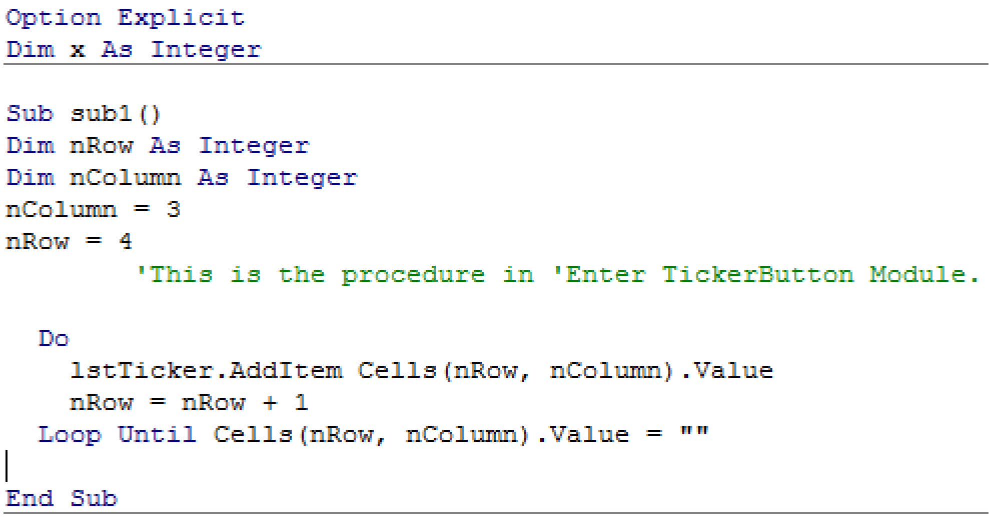
Enter your Tickers into the Text Box. Then press 'Enter Ticker' button.

To Remove a Ticker, simply highlight the ticker in the list and then press 'Remove>.' To remove all Tickers, press 'Remove All' button

If you have entered anything or removed any Tickers, press 'Load Tickers' to update your spreadsheet

The user will then enter in their tickers, one by one, into the first text box. To add the ticker, you must press the ‘Enter Ticker’ button to the right of the text box. Once that is pressed, it will be added to the list box below. To remove a ticker, simply highlight the ticker in the list box and then press ‘Remove>’ button. To remove all the tickers, simply press the button ‘Remove All>>’. In order to update any of the information entered into the user form, you must press ‘Load Tickers’ before closing. This ensures that the tickers are uploaded to the spreadsheet. A list of directions on how to use the user form is included in text directly on the user form.

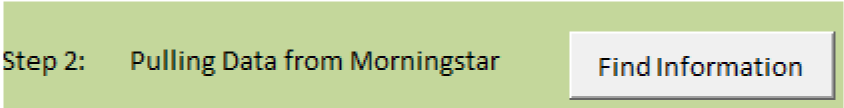
Below is a few lines of codes used within the user form:



```
Option Explicit
Dim x As Integer

Sub sub1()
Dim nRow As Integer
Dim nColumn As Integer
nColumn = 3
nRow = 4
        'This is the procedure in 'Enter TickerButton Module.

  Do
    lstTicker.AddItem Cells(nRow, nColumn).Value
    nRow = nRow + 1
  Loop Until Cells(nRow, nColumn).Value = ""

End Sub
```

I included this line of code, because it took me the longest to perfect, or at least run. This procedure is executed when the user form ‘frmTicker’ is initiated. This procedure helps populate the list box at the beginning. If users upload tickers several times during the Project, I wanted to be able to populate the existing tickers to the List Box, without having the user having to retype them. This block of code took a great while, which is the primary reason I included it. All other code relating to the user form was relatively self-explanatory.

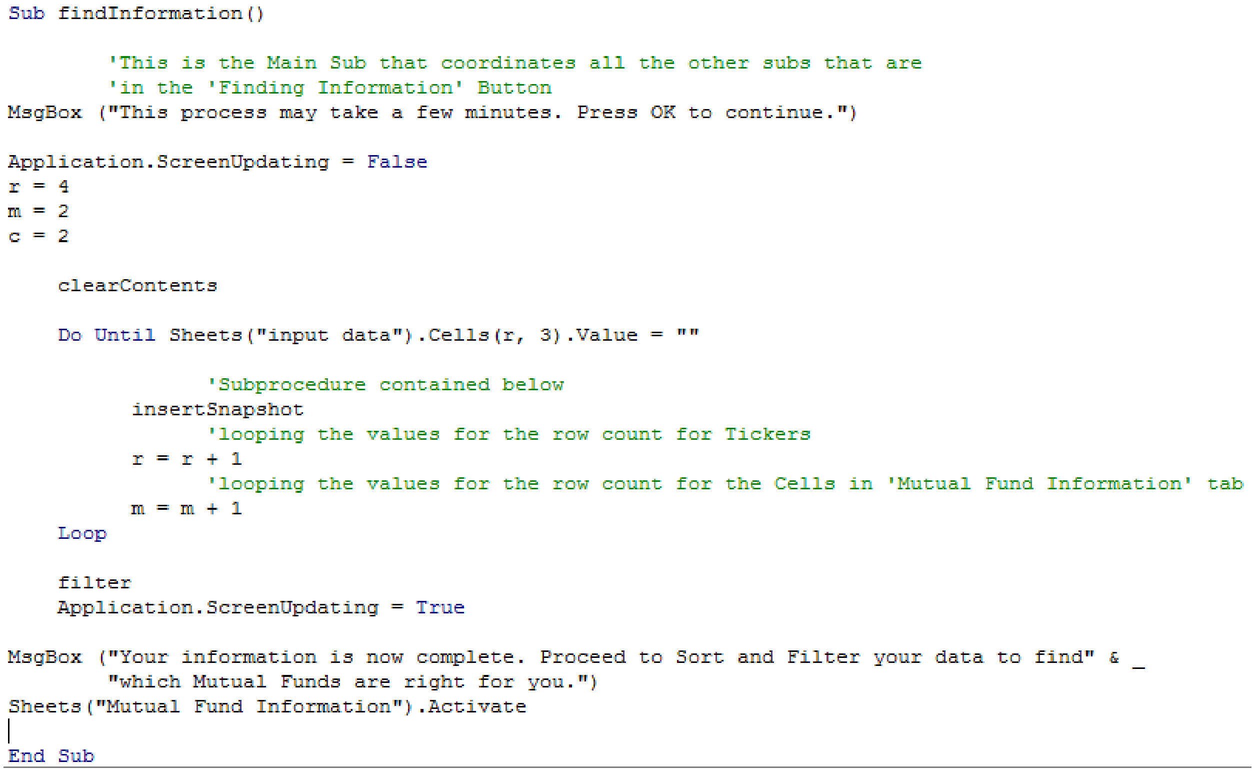
After the Tickers are all entered into the Input Data page, then the user is prompted to select the ‘Find Information’ button. This will begin the sequence of a series of sub procedures that will download data and place it in the ‘Mutual Fund Information’ tab.



Step 2: Pulling Data from Morningstar

Pulling data from Morningstar was definitely the most difficult. Some data had to come from a saved page from the website, some had to be redirected to another website. I wanted to walk you through the process of how I built this Module.

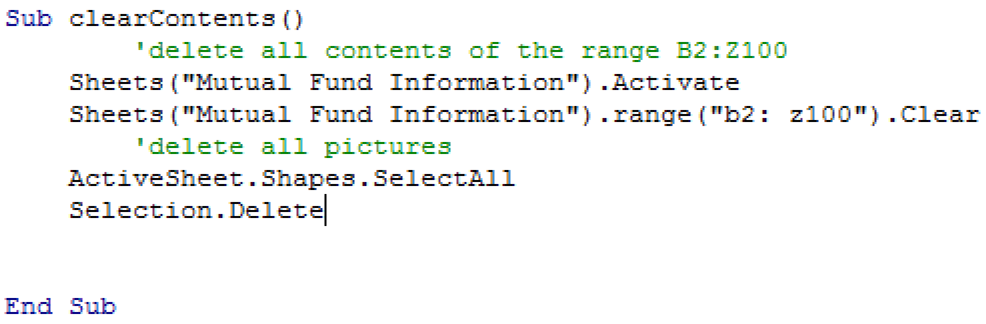
First, the main sub-procedure that called in all other helping sub-procedures:



```
Sub findInformation()

        'This is the Main Sub that coordinates all the other subs that are
        'in the 'Finding Information' Button
MsgBox ("This process may take a few minutes. Press OK to continue.")

Application.ScreenUpdating = False
r = 4
m = 2
c = 2

    clearContents

    Do Until Sheets("input data").Cells(r, 3).Value = ""

                'Subprocedure contained below
          insertSnapshot
                'looping the values for the row count for Tickers
          r = r + 1
                'looping the values for the row count for the Cells in 'Mutual Fund Information' tab
          m = m + 1
    Loop

    filter
    Application.ScreenUpdating = True

MsgBox ("Your information is now complete. Proceed to Sort and Filter your data to find" & _
        "which Mutual Funds are right for you.")
Sheets("Mutual Fund Information").Activate

End Sub
```

The sub-procedure ‘findInformation’ starts out by letting the user know that this will procedure will take a few minutes, via a message box. The application will then not update the screen during the process. I included a few variables that will be used in this and in later procedures.

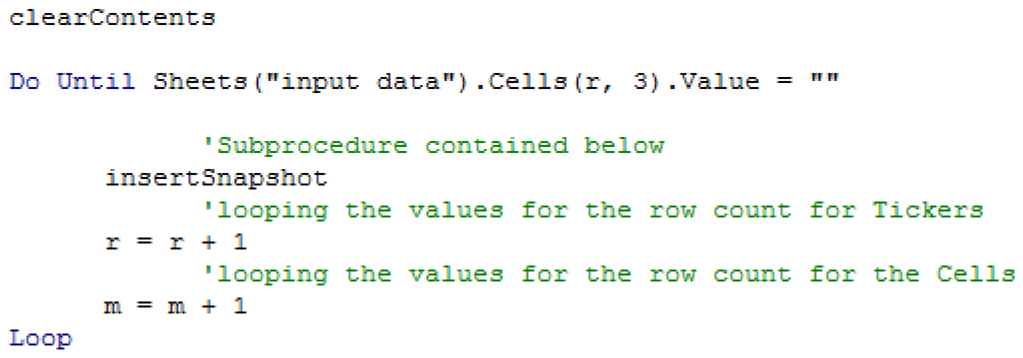
The next line is calling a sub-procedure call ‘clearContents’. The line of codes directing that is included below:



```
Sub clearContents()
        'delete all contents of the range B2:Z100
    Sheets("Mutual Fund Information").Activate
    Sheets("Mutual Fund Information").range("b2: z100").Clear
        'delete all pictures
    ActiveSheet.Shapes.SelectAll
    Selection.Delete


End Sub
```

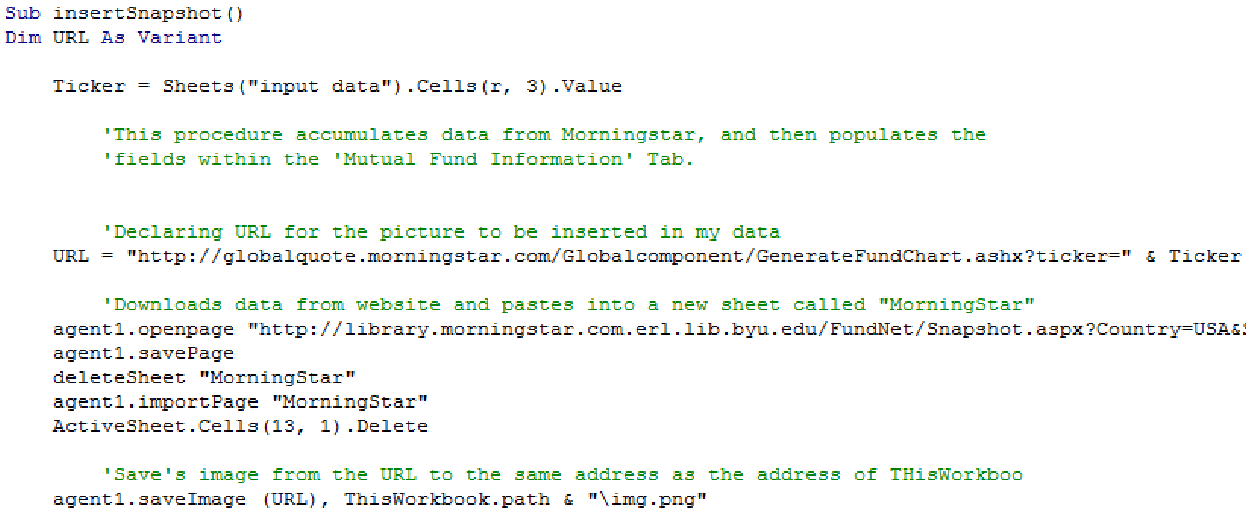
This code basically activates the ‘Mutual Fund Information’ tab and clears all contents within it, including all contents within the cell range (b2:z100) and all picture files, which will be pulled from Morningstar as well. This procedure makes sure there is not remaining data left from running code from prior use.

After the clearContents procedure, findInformation procedure will direct you to a loop that pulls data in another sub-procedure named ‘insertSnapshot’ (Snapshot, because that is the tab that all the information on Morningstar is located).



```
clearContents

Do Until Sheets("input data").Cells(r, 3).Value = ""

            'Subprocedure contained below
      insertSnapshot
            'looping the values for the row count for Tickers
      r = r + 1
            'looping the values for the row count for the Cells
      m = m + 1
Loop
```

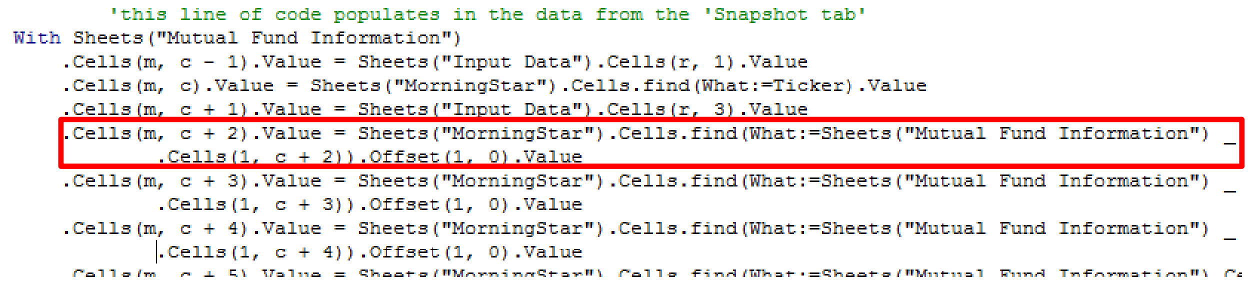
This code will run above will run a loop pulling data from Morningstar’s website for every ticker. It will update row by row, and then exit once there are not tickers. The sub-procedure ‘insertSnapshot’ is given below.



```
Sub insertSnapshot()
Dim URL As Variant

    Ticker = Sheets("input data").Cells(r, 3).Value

        'This procedure accumulates data from Morningstar, and then populates the
        'fields within the 'Mutual Fund Information' Tab.


        'Declaring URL for the picture to be inserted in my data
    URL = "http://globalquote.morningstar.com/Globalcomponent/GenerateFundChart.ashx?ticker=" & Ticker

        'Downloads data from website and pastes into a new sheet called "MorningStar"
    agent1.openpage "http://library.morningstar.com.erl.lib.byu.edu/FundNet/Snapshot.aspx?Country=USA&
    agent1.savePage
    deleteSheet "MorningStar"
    agent1.importPage "MorningStar"
    ActiveSheet.Cells(13, 1).Delete

        'Save's image from the URL to the same address as the address of THisWorkboo
    agent1.saveImage (URL), ThisWorkbook.path & "\img.png"
```

This code will run by taking the cells and assigning them to a Integer named “Ticker.” It will then create another variable named ‘URL’ by taking the URL of a picture of a historical price graph, the URL is basically the URL in quotes plus the Ticker at the end. This line of code is helping to insert a graph of historical prices for the fund signifying their performance.

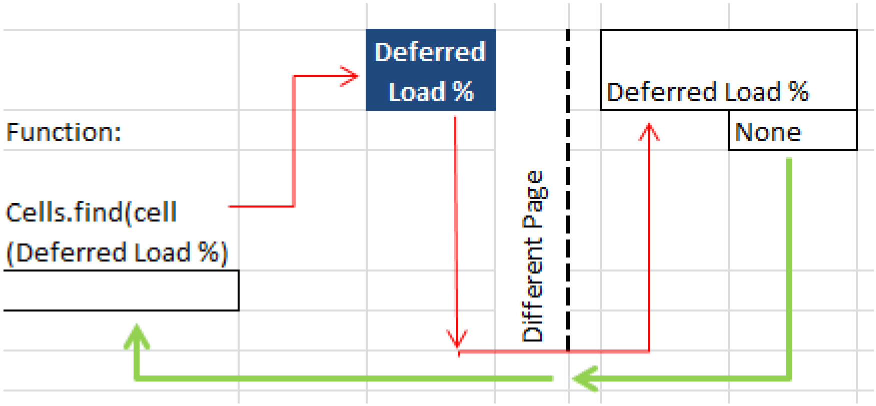
After that, agent1 will open the main page where the data will be pulled from. Save that page to a new tab called ‘Morningstar.’ It will then save the image in the same address as the currently saved Workbook. Continuing on with the code, a snapshot contained below is where the data obtained from Morningstar will be placed:



```
            'this line of code populates in the data from the 'Snapshot tab'
 With Sheets("Mutual Fund Information")
      .Cells(m, c - 1).Value = Sheets("Input Data").Cells(r, 1).Value
      .Cells(m, c).Value = Sheets("MorningStar").Cells.find(What:=Ticker).Value
      .Cells(m, c + 1).Value = Sheets("Input Data").Cells(r, 3).Value
      .Cells(m, c + 2).Value = Sheets("MorningStar").Cells.find(What:=Sheets("Mutual Fund Information") _
                .Cells(1, c + 2)).Offset(1, 0).Value
      .Cells(m, c + 3).Value = Sheets("MorningStar").Cells.find(What:=Sheets("Mutual Fund Information") _
                .Cells(1, c + 3)).Offset(1, 0).Value
      .Cells(m, c + 4).Value = Sheets("MorningStar").Cells.find(What:=Sheets("Mutual Fund Information") _
                .Cells(1, c + 4)).Offset(1, 0).Value
      .Cells(m, c + 5).Value = Sheets("MorningStar").Cells.find(What:=Sheets("Mutual Fund Information")
```

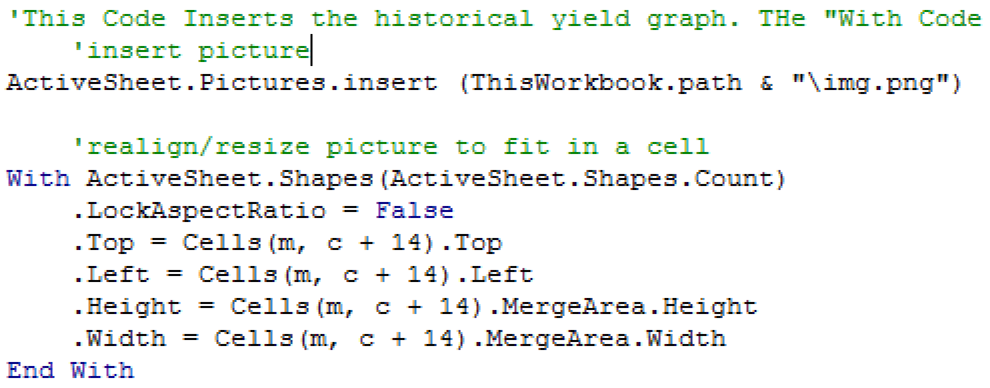
This is a rather long line of code, so I did not include all of the lines. The outline in red is a line of code I am particularly proud of. It searches for cells on the new tab created, ‘Morningstar’, by finding cells named at the top of the column to match those pulled from Morningstar, and then offsets down one row to get the value of the cell. The value of the cell is then pasted in the column of where the name that was originally called from the .Find function.

A diagram of this code is included below.



Inserting Pictures onto the ‘Mutual Fund Information’ Tab

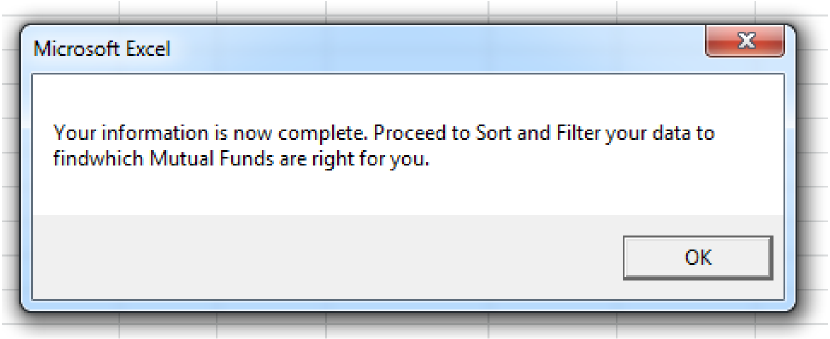
The hardest part about the code was getting to paste the PNG picture files of the Mutual Fund historical performance into a cell, and then resize to fit. It took me the longest until I realized that range and cell have different properties.



```
'This Code Inserts the historical yield graph. THe "With Code
    'insert picture
ActiveSheet.Pictures.insert (ThisWorkbook.path & "\img.png")

    'realign/resize picture to fit in a cell
With ActiveSheet.Shapes(ActiveSheet.Shapes.Count)
    .LockAspectRatio = False
    .Top = Cells(m, c + 14).Top
    .Left = Cells(m, c + 14).Left
    .Height = Cells(m, c + 14).MergeArea.Height
    .Width = Cells(m, c + 14).MergeArea.Width
End With
```

The easy part of this block of code is inserting the pictures onto the active sheet. I ran into some real trouble trying to resize it to a particular cell. If I hadn't of resized it to a particular cell, there you be graphs all over the spreadsheet without any indication of which went to what mutual fund. I kept on getting errors when I originally set my cells as "range(m, c + 14).Top/.Left/MergeArea.Height /MergeArea.Width. It was a global error because I used range. My guess after researching the solution is that you can't put in variables into a range. You can only do that with the Cell object. This caused me a lot of grief from working hours trying to fix it.

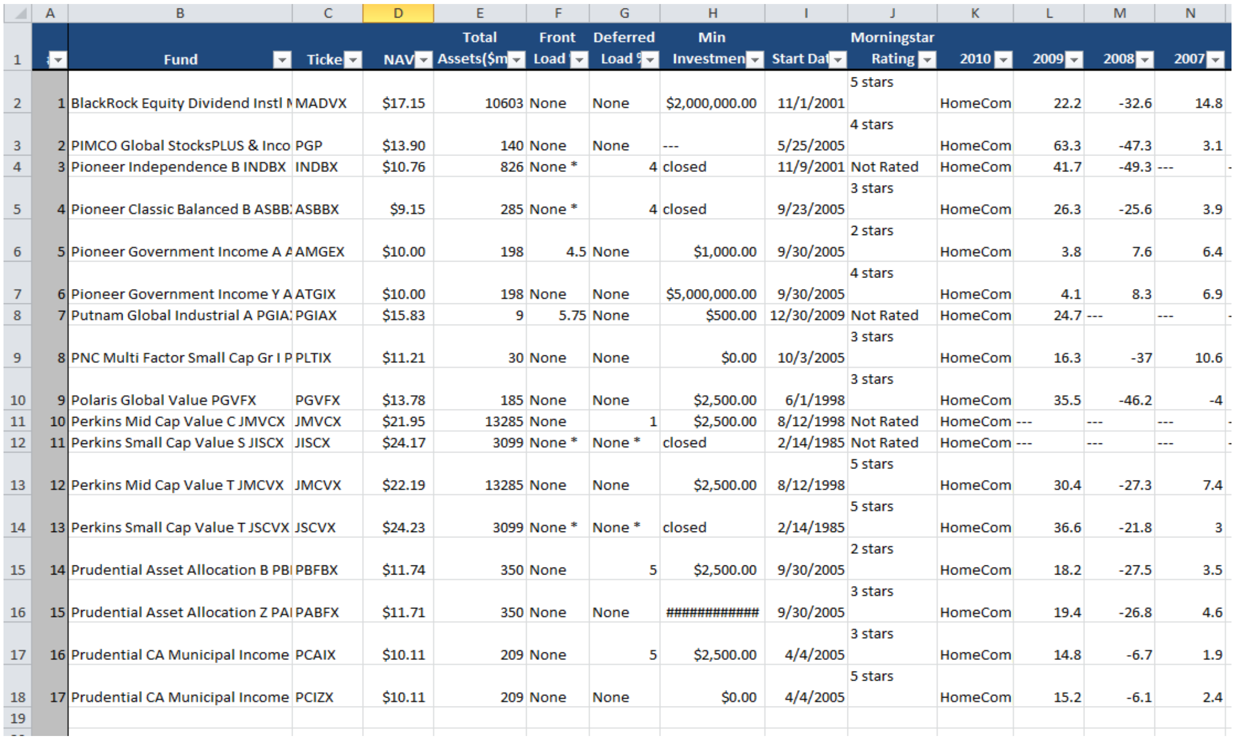
Once the code is all done with pasting the information on the tab, a message box saying it is finished will appear.



Your information is now complete. Proceed to Sort and Filter your data to findwhich Mutual Funds are right for you.

The Finished Product

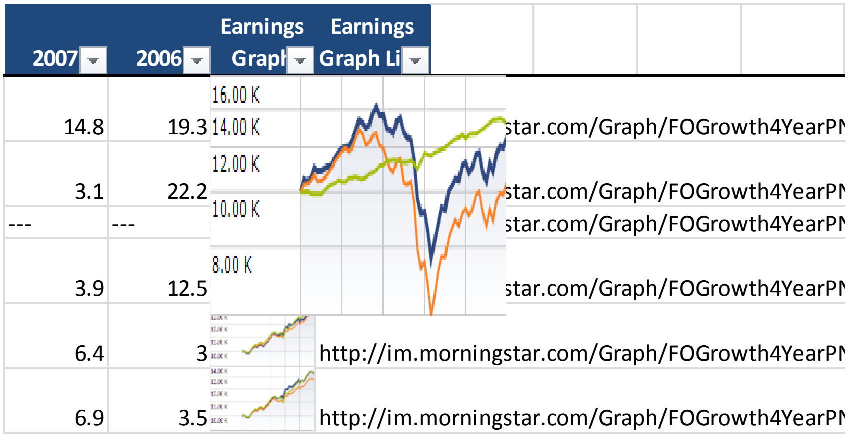
This is a screenshot of the Mutual Fund Information tab where BYU students can have all the data on the various mutual funds in one place. They can then sort and filter or delete themselves the funds that don't meet their criteria.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		Fund	Ticke	NAV	Total Assets($m	Front Load	Deferred Load	Min Investmen	Start Dat	Morningstar Rating	2010	2009	2008	2007
2	1	BlackRock Equity Dividend Instl M	MADVX	$17.15	10603	None	None	$2,000,000.00	11/1/2001	5 stars	HomeCom	22.2	-32.6	14.8
3	2	PIMCO Global StocksPLUS & Inco	PGP	$13.90	140	None	None	---	5/25/2005	4 stars	HomeCom	63.3	-47.3	3.1
4	3	Pioneer Independence B INDBX	INDBX	$10.76	826	None *	4	closed	11/9/2001	Not Rated	HomeCom	41.7	-49.3	---
5	4	Pioneer Classic Balanced B ASBB	ASBBX	$9.15	285	None *	4	closed	9/23/2005	3 stars	HomeCom	26.3	-25.6	3.9
6	5	Pioneer Government Income A A	AMGEX	$10.00	198	4.5	None	$1,000.00	9/30/2005	2 stars	HomeCom	3.8	7.6	6.4
7	6	Pioneer Government Income Y A	ATGIX	$10.00	198	None	None	$5,000,000.00	9/30/2005	4 stars	HomeCom	4.1	8.3	6.9
8	7	Putnam Global Industrial A PGIA	PGIAX	$15.83	9	5.75	None	$500.00	12/30/2009	Not Rated	HomeCom	24.7	---	---
9	8	PNC Multi Factor Small Cap Gr I P	PLTIX	$11.21	30	None	None	$0.00	10/3/2005	3 stars	HomeCom	16.3	-37	10.6
10	9	Polaris Global Value PGVFX	PGVFX	$13.78	185	None	None	$2,500.00	6/1/1998	3 stars	HomeCom	35.5	-46.2	-4
11	10	Perkins Mid Cap Value C JMVCX	JMVCX	$21.95	13285	None	1	$2,500.00	8/12/1998	Not Rated	HomeCom	---	---	---
12	11	Perkins Small Cap Value S JISCX	JISCX	$24.17	3099	None *	None *	closed	2/14/1985	Not Rated	HomeCom	---	---	---
13	12	Perkins Mid Cap Value T JMCVX	JMCVX	$22.19	13285	None	None	$2,500.00	8/12/1998	5 stars	HomeCom	30.4	-27.3	7.4
14	13	Perkins Small Cap Value T JSCVX	JSCVX	$24.23	3099	None *	None *	closed	2/14/1985	5 stars	HomeCom	36.6	-21.8	3
15	14	Prudential Asset Allocation B PB	PBFBX	$11.74	350	None	5	$2,500.00	9/30/2005	2 stars	HomeCom	18.2	-27.5	3.5
16	15	Prudential Asset Allocation Z PA	PABFX	$11.71	350	None	None	############	9/30/2005	3 stars	HomeCom	19.4	-26.8	4.6
17	16	Prudential CA Municipal Income	PCAIX	$10.11	209	None	5	$2,500.00	4/4/2005	3 stars	HomeCom	14.8	-6.7	1.9
18	17	Prudential CA Municipal Income	PCIZX	$10.11	209	None	None	$0.00	4/4/2005	5 stars	HomeCom	15.2	-6.1	2.4
19														

I configured this spreadsheet to be able to filter on all columns. If they don't like what they see in a fund, they can simply remove all those values that aren't consistent with their standards. The spreadsheet will then remove that rows with the low-performing ticker.

Here are the rows that include the snapshot of the PNG file:



2007	2006	Earnings Graph	Earnings Graph Li
14.8	19.3		star.com/Graph/FOGrowth4YearPN
3.1	22.2		star.com/Graph/FOGrowth4YearPN
---	---		star.com/Graph/FOGrowth4YearPN
3.9	12.5		star.com/Graph/FOGrowth4YearPN
6.4	3		http://im.morningstar.com/Graph/FOGrowth4YearPN
6.9	3.5		http://im.morningstar.com/Graph/FOGrowth4YearPN

What I Learned

This project has been both Heaven and Hell for me. Heaven in the sense that I got the chance to solidify and practice all of the tools I learned over the semester. I used a piece of knowledge from every assignment we turned in on this final project. It really was a great end to this class as I feel like I can really read VBA and look for ways to simplify. I feel I got a lot better in error handling. I probably ran my program upwards of 500 times. You get good at code by messing up a lot, I sure did that with this project. I feel like I learned more in the last two weeks than I did over the whole semester.

This project was very difficult for me. I really wasted a lot of time trying to fix one thing. Either, I left a decimal out, or I misspelled a sub-procedure. It was really hard to look at my code and see if it was working when I couldn't get anything to run. I wish I could have had a TA by my side whenever I needed. His two-second review of my code could have saved me, literally, tens of hours. That was a very frustrating experience, but I think I am a better VBA programmer because of it.

Things I Didn't Get to Finish or Implement

One minor detail that I couldn't do, was to use the '.findnext' function. When I try to pull data from the Column head named '2010,' the first cell it found was the actually date that the 'Morningstar' Page was pulled from. If I were to press next on the Find Function, it would pull the right "2010" from the tab and insert it in the 2010 Returns tab. This took me a while and I wasn't able to finish it. To reference what I am talking about, run the code with tickers, then lookup Column(K). You will see it is populated with "HomeCompaniesFundsETFs………"

One major part of this project I didn't think was feasible was to actually do all of the screening through excel. The timeline didn't make sense to do it, I am not sure if I could have accessed Morningstar's Mutual Fund database. I think it would have been a stellar VBA project, but my level of expertise is very limited.

Overall, I am very happy with this project. I feel I have improved ten-fold over the last two weeks over writing this code. Although, I originally was going to make the project a decision making tool in itself, I think I was over-stretching myself. If I were to grade myself on this project, I would have given myself an A+, for effort and for awesomeness.

Notes:

- Mention the Find Next in your code, how it was hard and couldn’t use it.
- Mention the Login code.