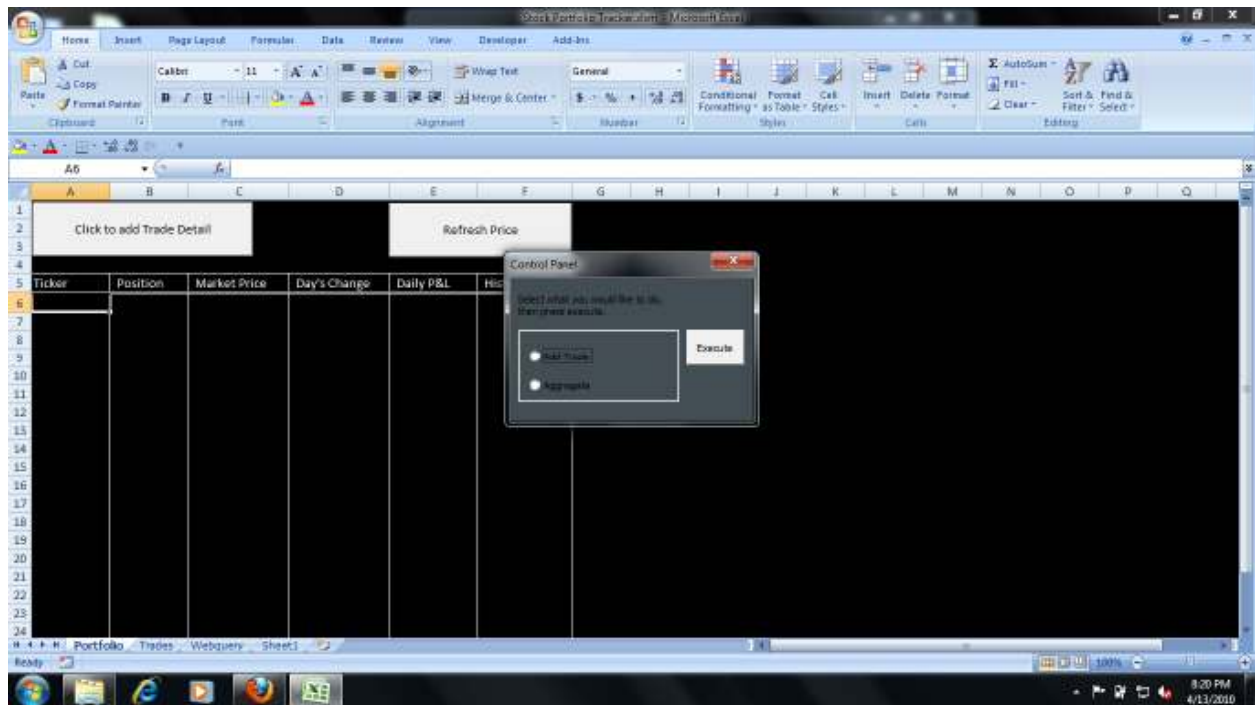


Portfolio Tracking Tool

Executive Summary

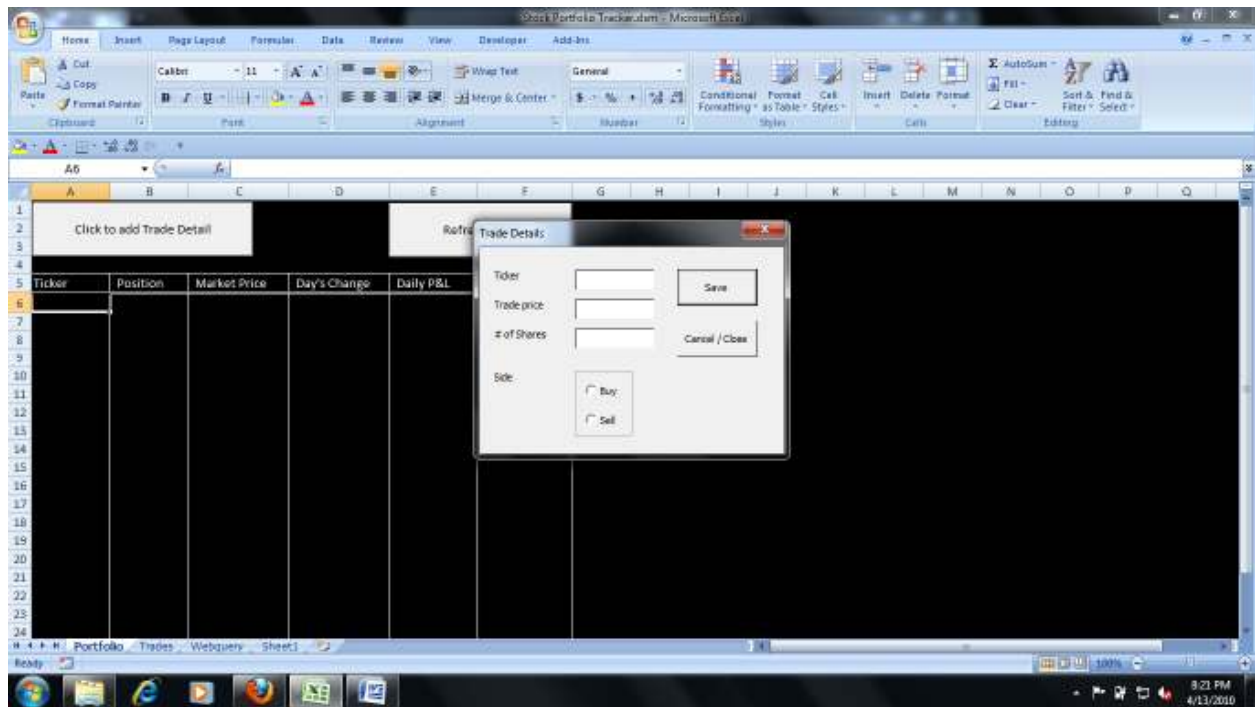
I thought it would be interesting to try to develop an Excel-based portfolio tracking tool. This tool will allow the user to input trade details, aggregate the trade positions, and track the performance of the portfolio via intra-day pricing being pulled from a web-source. With its user-friendly interface screens, adding new trades is very simple and easy. Buttons allow for market prices to be refreshed at the user's command. Note: it is not the purpose of this application to execute trades in the real world (although additional coding likely would allow for this) and is only designed to be used for tracking purposes – either tracking of a real or fictitious portfolio.

Upon opening the program, the "Control Panel" box opens with option (radio) buttons giving the user the option to either "Add Trade" or to "Aggregate" all added trades based on trade ticker.



If the user had closed the Control Panel window previously when opening the application, clicking the “Click to add Trade Detail” button on the Portfolio tab will call the window open again.

Selecting the “Add Trade” option and clicking “Execute” will pull up the “Trade Details” window that allows for the user to input transaction details including: trade ticker, trade price, number of shares, and buttons to select either a buy or sell trade.



Embedded in the code are safeguards to ensure that proper characters are entered in the text boxes. As an example, numbers aren't allowed in the Ticker field, or letters in the shares or trade price fields. Some sample code from the project is listed below:

```

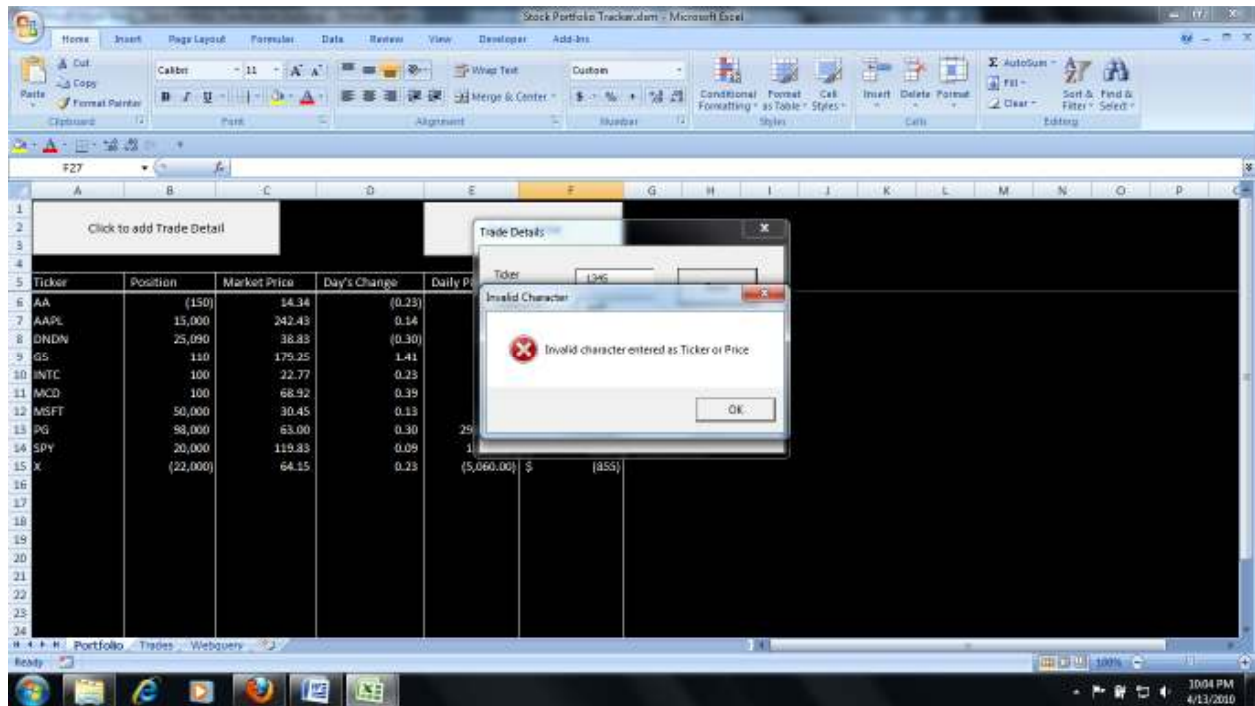
If txtTicker.Text = "" Or txtPrice.Text = "" Or txtShares.Text = "" Then
    MsgBox "You must fill in all information.", vbInformation, "Incomplete Information"
Exit Sub

ElseIf optBuy.Value = False And optSell.Value = False Then
    MsgBox "Please select the side.", vbInformation, "Select side"
Exit Sub

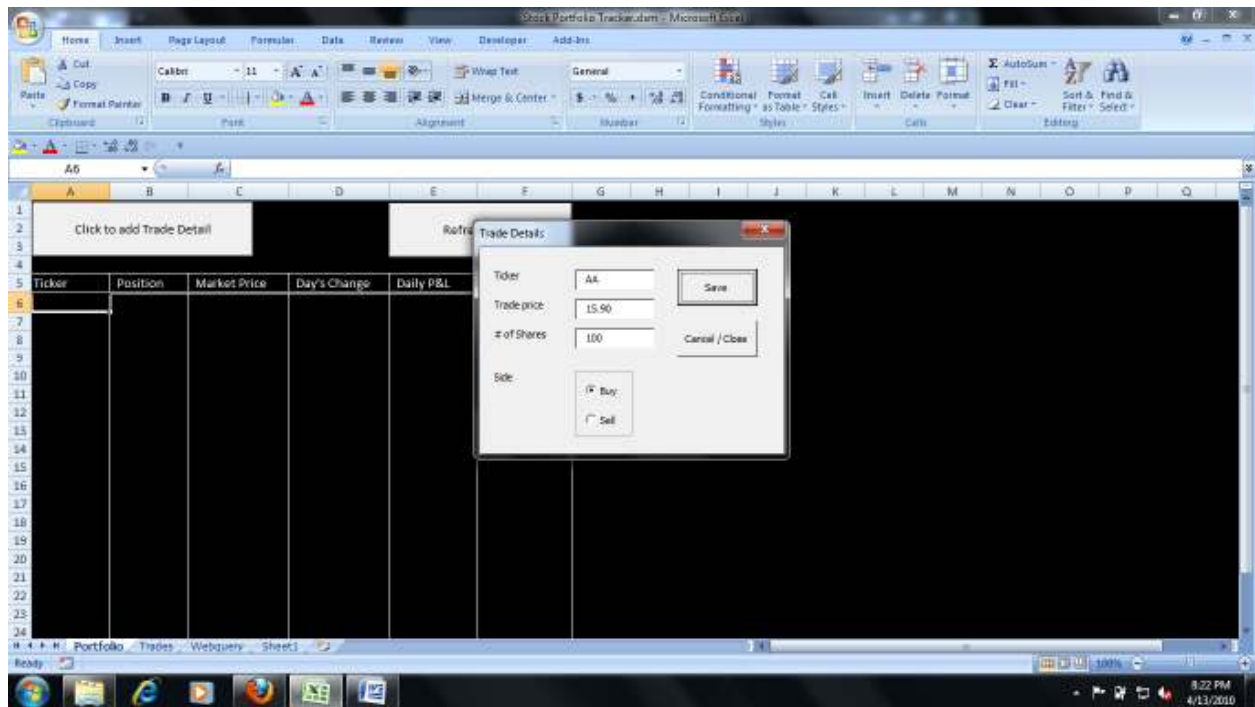
ElseIf IsNumeric(txtTicker) = True Or IsNumeric(txtPrice) = False Then
    MsgBox "Invalid character entered as Ticker or Price", vbCritical, "Invalid Character"
Exit Sub

```

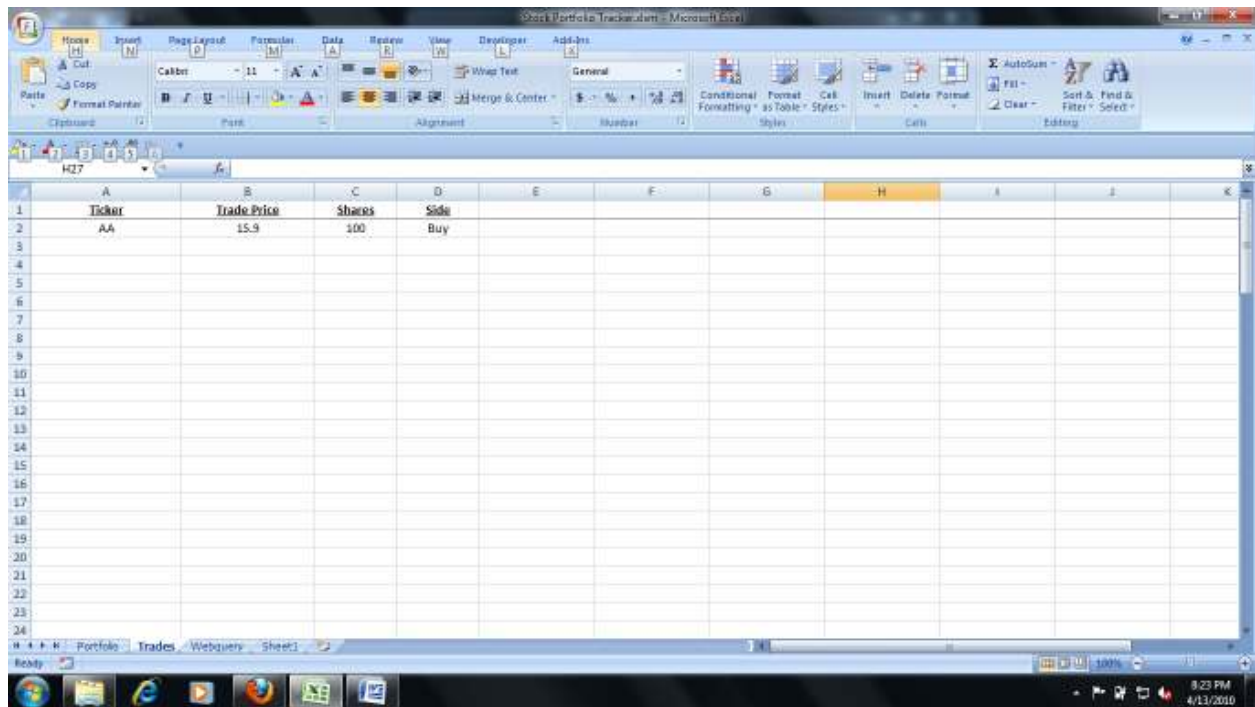
Entering invalid characters or not selecting a buy or sell radio button will bring up an error message and prevent the transaction details from being recorded. A screenshot of the error message can be found below:



The below screenshot is an example transaction with the fields populated. In this case, the user has bought 100 shares of Alcoa (ticker AA). The user will then be able to save the trade details by clicking the “Save” button. If the user chooses not to save it, pressing the “Cancel/Other” button will close the Trade Detail screen and the Control Panel window will again be in view.



Upon pressing the “Save” button, the transaction details are then sent over to the “Trades” tab with details of the trade being populated into a spreadsheet. Trades are initially listed in the order in which they are entered. The below screenshot shows the same transaction from the example above being recorded on the trades tab.



The most difficult part of the project became how to take the list of entered individual trades listed on the “Trades” tab and aggregate the position based on ticker and have that populate the “Portfolio” tab. A two-step solution to the problem was put in place. Using the “Aggregate” function affixed to the Control Panel window, individual trades in the “Trades” tab were first sorted alphabetically by ticker. This allowed for an aggregation procedure to be written in the code that calculated a net position that comprised all of the buys and sells of a ticker and populated this into the “Portfolio” tab. Some of the code is as follows:

```
For Each cell In Sheets("trades").Range(Range("A2"), Range("A65536").End(xlUp).Address)
    If cell.Value = cell.Offset(-1, 0).Value Then
        SharesTotal = SharesTotal + CInt(cell.Offset(0, 2).Value)
        TotalCash = TotalCash + (cell.Offset(0, 1).Value * cell.Offset(0, 2).Value)
        TickerSymbol = cell.Value
    Else
        'dump variables to agg page
    If Not cell.Offset(-1, 0).Value = "Ticker" Then
    If Not SharesTotal = 0 Then
        'AggSpot = Sheets("Portfolio").Range("A65536").End(xlUp)
        Sheets("Portfolio").Range("A65536").End(xlUp).Offset(1, 0).Value = TickerSymbol
```

```
Sheets("Portfolio").Range("A65536").End(xlUp).Offset(0, 1).Value = SharesTotal  
End If
```

A web-query was then written that pulled both the current market price and the amount the current change in price for the day from the financial website www.forbes.com. These two pieces of data were then pulled into the “Porfolio” tab and are used to calculate the field “Daily P&L”. [It should be noted that the stock price information on the forbes website is on a fifteen minute delay.]

The button “Refresh Price” allows the user to update the price on demand. It wouldn’t be terribly difficult to have the refresh price function work on a continuous loop, but I decided that because of the speed limitations of the webquery in excel, it didn’t make much sense to have the program continually running and that allowing the user to choose when to refresh the pricing would be more beneficial.

The end result is displayed in the below screenshot of a portfolio with live, intra-day pricing.

Ticker	Position	Market Price	Day's Change	Daily P&L	Historical P&L
AA	(150)	14.34	(0.23)	85	\$ 290
AAPL	15,000	242.43	0.14	2,100	\$ 2,000
DNDN	25,090	38.83	(0.30)	(7,527)	\$ 300
GS	110	179.25	1.41	155	\$ 2,987
INTC	100	22.77	0.23	23	\$ 2,250
MCD	100	68.92	0.39	39	\$ 6,798
MSFT	50,000	30.45	0.13	6,500	\$ 34,375
PG	98,000	63.00	0.30	29,400.00	\$ 17,200
SPY	20,000	119.83	0.09	1,800.00	\$ 2,550
X	(22,000)	64.15	0.23	(5,060.00)	\$ (855)

Being personally not at all savvy in the workings of VBA – and considering it a foreign language that at times is very difficult to grasp – this was an interesting project. I definitely see the possibility of a number of analytical tools that could be added to a project such as this to increase the value to an investor and readily acknowledge that this project merely scratches the surface. Many other investment criteria and tracking can be added to this spreadsheet application – including listing betas, risk calculations and the like. However, with my very limited background of VBA, this project seemed a challenge enough.