We need to build a web based lightweight stock data chart system that can be run on a local machine, for very specific scenarios. There is no need to write charting components – we can buy chart libraries as needed and just write the application on top.

When the web page loads, we need to have 2 charts on the page:

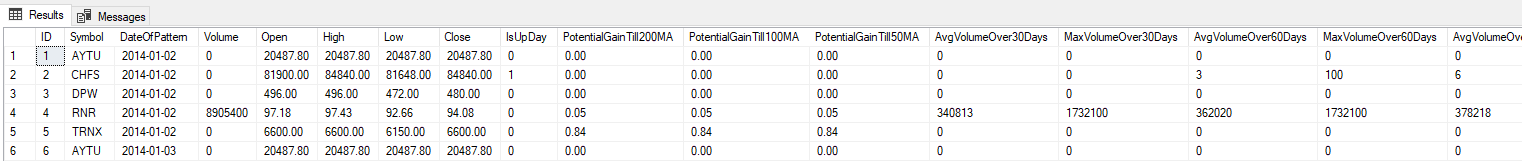
1. The top chart is a candlestick chart for a given stock symbol, from a given date and on. On the candlestick chart, we also want to see some line indicators like moving averages or such.
2. The bottom chart is a volume chart.



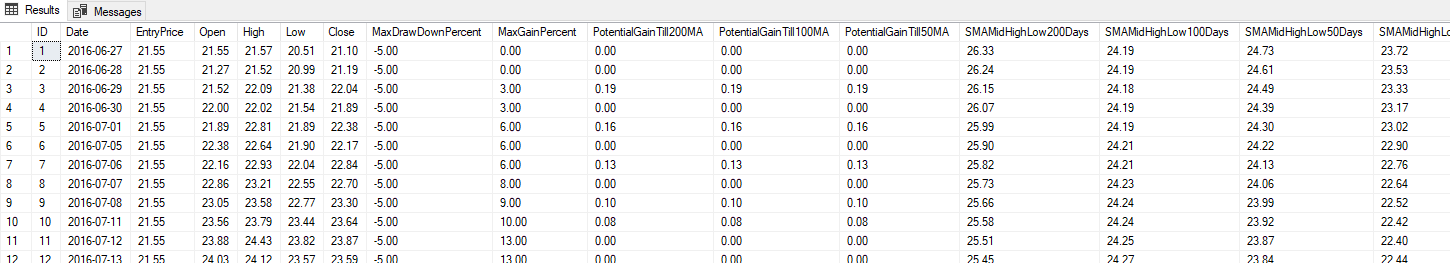
Resources:

We have a Microsoft SQL Server holding table data. The javascript code (or whatever language you decide to build this in) will be calling one or more stored procedures to get the data it needs in the desired format. The stored procedures will be interacting with several tables:

dbo.Trades – a table that has information about a stock symbol and related information, from a certain date and on. The table has one record per trade.



dbo.TST001\_[Symbol] – this table is specific for each symbol. For example, TST001\_AMZN for Amazon stock data. After each trade happens in dbo.Trades for a given symbol and date, the table dbo.TST001\_[Symbol] has daily data of what happened to the trade, for that symbol and from that date on. The table can hold data for 30 days after the trade, or 60 days, or more.



The chart – in more detail:

In the top left of the chart there should be a text box for the stock symbol and for the date of trade, and 2 buttons – Next and Go. Something like this:



The following scenarios should be supported:

1. When the chart webpage loads for the first time, it should call a proc to get the first set of data. The stored procedure will return the data for the first (oldest) trade in the table dbo.Trades, for the first symbol (first = in alphabetical order), and the daily data for however many days from dbo.TST001\_[Symbol]. Then, the webpage will display the candlestick chart with moving average indicators in the top chart, and the volume chart in the chart below, all in one page.

Note: The value of the moving averages and everything else is in the database. There is no need to calculate it in the web page code.

1. If the user clicks Next, the web page will get the symbol and date of trade that is showing on the current chart, and then call a stored procedure and pass the symbol and date of trade to the stored procedure. In return, the stored procedure will get the next trade from dbo.Trades and the daily data from the relevant dbo.TST001\_[Symbol] table. The web page will then show the data for the next trade. This will happen each time the user hits Next.
2. The user can enter a symbol and click Go. In this case, the web page will call a stored procedure and pass only the symbol name. The stored procedure will return the first (oldest) trade for this symbol from dbo.Trades, and the daily data to be shown in the charts from dbo.TST001\_[Symbol] for that trade (i.e., for that symbol and from the trade date an on).
3. The user can enter a symbol and a date and click Go. Here, the web page will call a stored procedure that will return the data for the charts for that symbol and trade date.

I’m happy to write the stored procedures and give access to the SQL Server.

Note: There will have to be a place where the user can change the connection string for the SQL Server, since the server will likely move in the future.