Hello Mark,

This is not the logical mathematical document. It is just a brief summary of the project. Knowledge of stocks, candle charts, IEX API, SQL database, Cloud Computing and Mathematical logical operation in Python are **MUST**. Please realize, I’m not a coder and this script is not yet tested and may require on-going fine-tuning, until scanner is producing results that I can verify visually on chart. Off course, for this phase, the document that I will send you determine the scope of work. Any additional changes after that will be phase two at additional remuneration.

I have been trading for 8 years and this system is called price action and is widely used by many traders. It is difficult to open each stock manually and visually check to find these level. Therefore, I need an automated program to scan the stocks for me. It doesn’t enter trades or anything, just outputs a table by comparing multiple stocks across multiple timeframes to find price patterns and location on the chart compared to current price. Therefore, it is actually simple but has multiple unique comparative requirements.

Finally, although the program is party complete, it was never fully working and therefore not tested. It might be that you may prefer to recode the entire program from scratch. In any case, I have spent a lot of time over 18 months with two programmers and still not be able to get a working prototype. I think, its challenging in its own way.

Therefore, this time I will only consider working with someone who is fully committed, has time and also encouraged by the challenge. I will also not pay anything upfront since I have found many freelancers leaving work mid-way, and I lose money while my project is still incomplete. This final point is mainly the reason, many programmers reject this project and I understand if you too don’t want to do it. However, if you want experience in Trading bots, and like the challenge, I can send you very detailed documents after awarding it to you. However, the motivation needs to be learning, gaining experience and ability to expand on your skillset, not just money. $500 is lot of money for a simple yet interesting scanner such as this and you will benefit tremendously from it, I assure you that. If you are still keen, we will need to connect on a skype call (My skype ID is KazaaModo), maybe even show you my screen and explain exactly, what I do and learn about your coding level and stock market knowledge.

A few examples of kind of logical comparison and operations are: Compare current price of a stock to the open price, close price, high price and low prices of multiple previous candles in each timeframe data table. Finding exact setup of 3 -5 consecutive candles, compare the location of this setup with current price, find other similar set ups on the same chart and measure distances. Multiple comparison of ratios between the body of the candle to the full range of the candle to find the largest or smallest price movements across time. In essence, the program will make thousands of comparisons within each chart, and across charts, across hundreds of stock, across multiple timeframes and find few special conditions for output. The program should be capable of doing this across as many stocks as in the database continuously and automatically across multiple scan queries. The API work is largely done, with some minor tweaks and features I might add. The main part is analysis on the data.

I’m excited and hope you will find this interesting too. Looking forward to a reply.

Regards,   
Aamir Fakih

PRICE ACTION TRADING ZONE IDENTIFICATION

Project Deliverable   
Using mathematical comparative rules on prices of individual stocks across multiple time frames to identify and shortlist from list of symbols/tickers in a given database. Each stock/ticker/symbol will have THREE timeframes that the analysis will process through – Higher, Intermediate and lower timeframe. Intermediate is optional, and only if the GUI field is checked Yes, it is used, otherwise, it is not used. The database to analyze is MySQL located on Google Cloud Computing Servers and updates at *user defined frequency.*

Project Language: Python 3.7+

Project Timeline: 30 days (plus debugging until the deliverable is fully functional)

Current Database structure  
6 Tables – Weekly, Daily, 60 Mins, 30 Mins, 15 Mins, 1 Min. Tables below daily timeframe are created using 1 min data. The database tables will require functions that notify me, if data import fails since the integrity of the scanner is only as good as the data itself. This is currently not available.

**Current Total Tickers is each table:** 498 individual tickers. Subject to change, can be thousands more and the code should run the algorithm on every unique ticker it finds in the database.

**Current Table Structure:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date/Candle/Timestamp** | **Ticker** | **Open** | **High** | **Low** | **Close** | **Volume** | **SMA5** | **SMA 20** | **EMA5** | **EMA20** |
| YYYY-MM-DD HH:MM:SS.000000 | XYZ | float | float | float | float | Int | float | float | float | float |
| YYYY-MM-DD HH:MM:SS.000000 | ABC | float | float | float | float | Int | float | float | float | float |
| YYYY-MM2-DD2-HH2:MM:SS.11.11 | XYZ | float | float | float | float | Int | float | float | float | float |

*\*Ticker can be repeated in each intraday table. For ex: A 60-min hourly table will have 7 entries, one for each hour, for the same ticker. However, timestamp/date/candles will be unique. The EMA and SMA are calculated using the data in the table and thus any number of periods can be created to compare. For eg: EMA 15 or SMA 25 etc.*

Background:(*Read carefully for full understanding of what is required)*

The purpose of this project is to find price behavior of each stock in the past on three different timeframes. We will use Open/High/Low/Close of each candle for each timeframe per stock. This needs to be done mathematically for multiple stocks, based on rules and values entered by the user in Graphical User interface. The methodology requires a Two to Three step process, which can be broken down into several smaller chunks of logical operation to arrive at the final list of recommendation/shortlist as an output. (I’ll include that later)

* The first Input is to choose an *asset class* to run this algorithm. The choices are:   
  1) Stocks 2) Options Contracts 3) Foreign Exchange Currency Pairs 4) Futures Contracts. Depending on the choice, there will be specific set of data tables to analyze using the script that is being coded here. *Currently, this project is only for* ***Stocks****. However, the Graphical User Interface/Configuration File must be ready to eventually expand to other asset classes listed above.*
* The second input is to choose **two** or **three** different timeframes that the scanner/program will run through. Choices are: **Monthly, Weekly, Daily, 60 mins, 30 mins, 15 mins, 5 mins and 1 min**. The analysis should run on **multiple, specified timeframes** (explained in detail below)  
   *The potential examples of three different timeframes to analyze for each single stock, options contracts, futures commodity or Foreign currency pair in the future are:*
* The direction of analysis moves in three steps from exactly in this order – First is HTF, then ITF (optional) and Finally LTF.

|  |  |  |
| --- | --- | --- |
| Higher Time Frame (HTF) (Required) | Intermediate Time Frame (ITF) (OPTIONAL) | Lower Time Frame (LTF) (Required) |
| Monthly Price Chart | Weekly Price Chart | Daily price Chart |
| Weekly Price Chart | Daily Price Chart | Hourly Price Chart |
| Daily Price Chart | Hourly Price Chart | 15 Minute Price Chart |
| Hourly Price Chart | 15 Minute Price Chart | 5 Minute Price Chart |
| 30 Min Price Chart | 5 Minute Price Chart | 1 Min Price Chart |

* The third input is to assign a *trade direction* before running the code. The direction of trades can be **Long(buying) OR Short(selling).**
* Once values for each timeframe is entered into the GUI (detailed below), the code will **execute continuously** on a set schedule until stopped manually on **ALL** the stocks in the chosen database.
* **Multiple instances** of the code/script with different values should be allowed to be executed independent of previous active scripts running.

Schedule for automatic script query execution:

* When Higher Time frame is ***M*onthly*,*** the script analysis should run *once per day* at {user defined} time of day. Default: 20:00 ET Mon-Fri. For HTF **Weekly**, it should run hourly, on the hour, every hour from 10:00 to 16:00. Default is M – F: 1000 – 1600 ET. For HTF **Daily,** it should run every 15 minutes and for HTF **Hourly**, it should run every 5 minutes. In other words, the Lower time frame, is how often the script should run once executed. This is true for automatic continuous execution. Manual executions should run on most current/real time data and is not required to be run again.
* Once the script determines that a ticker/stock has been found where all conditions/values are true, a report with a **score** is generated and **notifies via email/alert** each time there is a new buy or sell zone found based on the conditions described.

INPUT FIELDS (GUI interface):  
  
GENERAL FIELDS

*GUI field 1:* **“Asset Class”** Stocks [other choices: Options, Foreign Exchange, Futures]. Default is Stocks. Field is required.

Stocks, Futures, Options, Forex

*GUI field 2: “***Timeframe”:** [Choices are:Monthly, Weekly, Daily, 60 Min, 30 Min, 15 Min, 5 Min, 1 Min.]. This list must co-ordinate with the tables in the database. If a new timeframe is added to the database, it should also appear here as a choice.   
*\* Make sure to test that Higher time frame is always larger than intermediate and lower time frames and that ITF is larger than LTF, when selecting then below. For Example, LTF cannot be Daily, when Higher time frame is Hourly, or intermediate time frame cannot be Weekly, when higher time frame is Daily.*

2A - HTF (required), Choose 1: 30 Mins, Hourly, Daily, Weekly, Monthly

2B – ITF (Optional): 5 Min, 15 Min, Hourly, Daily, Weekly

2C – LTF (required), Select 1: 1 Min, 5 Min, 15 Min, Hourly, Daily

*GUI field 3:* **Enter Trade Direction:** Long [other choice: Short]. Default is Long. Field is required.

Long/ Short

*GUI field 4:* **Enter Analysis Start Date:** MM/DD/YYYY. Default date is always Today/most current. This field must also take date from the past for back testing.

ENTER Date/Calendar  
*(check for past setup starting from this date)*

HIGHER TIME FRAME (HTF) FIELDS

*HTF GUI field 5:* **Enter Total Number of unique past candles/timestamps in HTF to compare from ‘Trade Start Date’:** 0000. *(Integer field). Default 100. Max: till end of data table. (This is important because it allows control how far back the price setup that I’m looking for in each stock. I may want to restrict only to see if a specific price behavior in the past X number of days or weeks or hours)*

Integer  
*(look how many candle back from the date above)*

*HTF GUI field 6:* **HTF****Leg-in and Leg Out candle’s Open and Close range is within top** **X** of the total past number *(HTF GUI 5)* of candles. Default is 5. Max: 10. Min 1

6B. HTF Leg out value  
*(compared to other candles, how long is this candle)*

6A. HTF Leg in value  
*(compared to other candles, how long is this candle)*

*HTF GUI field 7:* **HTF****Leg-in and Leg Out candle’s {Open and Close} price range RATIO compared to {Highest and Lowest} price of that same candle. I.E. Ratio of Body to Full Candle. Body must be X% of the full range.** Default is 70%. Max: 100%. Min 60%.

7B. HTF *BELOW CURRENT PRICE*    
Leg **out** Body-Range Ratio

7A. HTF *BELOW CURRENT PRICE* Leg **in** Body-Range Ratio

DEMAND SIDE

7C. HTF *ABOVE CURRENT PRICE*    
Leg **in** Body-Range Ratio

7D. HTF *ABOVE CURRENT PRICE*    
Leg **out** Body-Range Ratio

*HTF GUI field 8:* **HTF****Basing Candles {Open and Close} price range RATIO compared to {Highest and Lowest} price of that same candle. I.E. Ratio of Body to Full Candle. Body must be X% of the full range.** Default is 30%. Max: 49%. Applies to all (up to 3) basing candles.

8B. HTF *ABOVE* CURRENT PRICE  
Basing candle body-range %

8A. HTF *BELOW* CURRENT PRICE  
Basing candle body-range %

LOWER TIME FRAME (LTF) FIELDS

*LTF GUI field 9:* **LTF****Leg-in and Leg Out candle’s Open and Close range is top** **X** of the total past number *(gui field 5)* of candles. Default is 5. Max: 10. Min 3. For eg: *If value of X is 5, it will shortlist top 5 candles compared against all the candles between (GUI field4) and (GUI field5). Leg in and leg out values are INDEPENDENT of each other.*

Enter LTF Leg-in value

Enter LTF Leg-out value

*LTF GUI field 10:* **LTF****Leg-in and Leg Out Candle’s Open and Close price range RATIO compared to High and Low of the same candle. I.E. Ratio of Body to Full Candle. Body must be X% of the full range.** Default is 80%. Max: 100%. Min 70%.

10A. LTF *BELOW CURRENT PRICE* Enter Leg **in** Body-Range Ratio

10B. LTF *BELOW CURRENT PRICE*    
Enter Leg **out** Body-Range Ratio

10C. LTF *ABOVE CURRENT PRICE*    
Enter Leg **in** Body-Range Ratio

10D. LTF *ABOVE CURRENT PRICE*    
Enter Leg **out** Body-Range Ratio

*LTF GUI field 11:* **LTF****Basing Candles {Open and Close} price range RATIO compared to {High and Low} price of that same candle. I.E. Ratio of Body to Full Candle. Body must be X% of the full range.** Default is 30%. Max 49%. Applies to all (up to 3) basing candles.

11B. LTF *ABOVE CURRENT PRICE*    
Enter Basing candle body-range %

11A. LTF *BELOW CURRENT PRICE*  
Enter Basing candle body-range %

GENERAL FIELDS II

*GUI Filed 12:* Number of candles in the past to compare for Opposing Zone, if current candle is making All-Time-High prices or All-Time-Low OR there are no valid zones found in this period. *(details on this exception is explained on a separate document, not included here.)*

Enter Value (integer)  
Total candles to look back when no valid opposing zone found.

*GUI field 13:* **Enter Frequency to execute script continuously every:** Daily [other choices: Hourly, Minutes]. (After the first execution, this should occur only if new price data/candle has been added to the underlying database. No point running the script if there is no change in the data.)

Once/Daily/Hourly/Minutes

*GUI field 14:* Choose any one type of Execution:

Manual

Automatic

*GUI field 15:*  For automatic, **Enter Time of day for** continuous scheduled execution of script**:** hh:mm:ss

Enter Time of day to execute automatically

Save and run any of the above settings to reuse the same values in the future.

Enter Query Name & SAVE

Sample Output:   
This table should be saved in a database every time it changes. Also, in the next phase of this project, I may want automatic ordering based on this table and thus it should be in a format, where broker API’s could pull from this table to place orders. However, until such time, I should be able to receive continuous alert as the output as often as new valid zones are found. Require feature: An Email alert notification **as often as new Asset/Ticker with a scored zone are discovered**, if the code is running on Automatic frequency.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Trade Direction** (from gui field 3) | **High Time Frame** (from gui field 2) | **Ticker** *(from database)* | **Last Closing price** *(from database)* | **Entry Price at LTF Zone** *(from analysis)* | **Date/Timestamp of leg-out at LTF Zone** *(from analysis)* | **ITF Trend, if app** | **Curve level** *(from analysis)* | **RRR** *(from analysis)* | **Zone Score** *(from analysis)* |
| Long | Weekly | AAPL | $000.00 | $000.00 | MM/DD/YY | 5<20 | Low | 5:1 | 8 |
| Short | Daily | XYZ | $000.00 | $000.00 | MM/DD/YY | 5EMA>20 | Very High | 3:1 | 10 |
| Long | Hourly | AAPL | $000.00 | $000.00 | MM/DD/YY | >20 | Very Low | 2.5:1 | 9 |
| Long | Weekly | AAPL | $000.00 | $000.00 | MM/DD/YY | 5EMA<20 | Low | 4.78:1 | 7 |

**The ABOVE TABLE IS WHAT THIS ENTIRE PROJECT IS ABOUT. If the logic of the code is accurate, this information is what is valuable.**

**Complete mathematical, logical operations will be sent on awarding the project.**