

The need for simplicity... thru ease—

Automated Trading System
Currency futures trading / speculation
Golden Ratio
Java API for Interactive Brokers (IBKR)
Gentle application to Long short-term memory (LSTM) neural network
Adding Machine learning to IBKR's java API
Tesla-3-6-9

This collaboration is open and conducted purely under open-source principles. Not only are you free, but encouraged to post/share any and all the techniques, formulas, codes, blueprint, and documents with wider public. After all, it's philosophy of open-source implementations that applications progressively undergo iterations, improvements and optimization in that striving for eventual perfection. Too, an increase in market participants can result in greater liquidity and thereby overall market stability, as opposed to volatility, and it's hoped the strategy would work better with greater volume of participants.

To that end, please note, these codes and materials will be published in Github and others forums related to the TWS API (Trader work station) like following URLs.

<https://groups.io/g/twsapi>

<https://groups.io/g/ibcalpha>

<https://stackoverflow.com/search?q=interactive+brokers&s=85e0b476-3da9-4d6f-a4d5-eb0822d7011b>

TWS is an interface application for connecting to Interactive Brokers LLC (www.ibkr.com) for trading and performing other tasks, account management, while custom strategies can be implemented in any widely used languages and systems such as Java, VB, .NET (C#), C++, Python, or Microsoft Excel (for ActiveX or DDE). There even exists MATLAB based interface.

<https://interactivebrokers.github.io/tws-api/>

Question--Is it possible to train a time-series neural network with instances of successful (profitable) trades and then go live with hopes of transacting only those profitable executions if the opportunity arises? Because a signal indicator will always indicate "buy" or "sell", however the key lies in transacting only on those trades that would be profitable. Following image is of countless signals... but few are worth executing. So how does one determine which signal to execute—to buy or sell?

5-MIN (Presentation) - Microsoft Excel non-commercial use

B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
YYYYMMDD_HHMMSS	Open	High	Low	Close	Volume	WAP	Count	Minute	Tesla3	Tesla6	Tesla9	Decision	EXECUTE							
20200623 10:30:00	0.6968	0.6971	0.6964	0.697	653	0.69675	168	30		0.000735	0.002622	-0.000765	NO	NO						
20200623 10:35:00	0.697	0.697	0.6968	0.6969	434	0.69694	134	35		0.001089	0.002442	-0.000243	NO	NO						
20200623 10:40:00	0.6969	0.697	0.6963	0.6963	477	0.69677	144	40	0.001068	0.00099	-0.00054	SELL	EXECUTE							
20200623 10:45:00	0.6963	0.6966	0.6959	0.696	456	0.69615	155	45		0.000453	-0.001176	-0.002547	SELL	NO						
20200623 10:50:00	0.6961	0.6963	0.6958	0.6961	551	0.69603	157	50		0.000447	-0.002238	-0.002241	SELL	NO						
20200623 10:55:00	0.696	0.6963	0.6959	0.6961	471	0.69612	152	55		0.000237	-0.00213	-0.001269	NO	NO						
20200623 11:00:00	0.6961	0.6966	0.6958	0.6965	549	0.69614	196	0		-0.000261	-0.001872	-0.000459	NO	NO						
20200623 11:05:00	0.6966	0.6967	0.6964	0.6966	225	0.69652	73	5		-0.000144	-0.00081	0.001476	NO	NO						
20200623 11:10:00	0.6966	0.6967	0.6965	0.6966	269	0.69661	79	10		-0.000423	-0.000102	0.001467	BUY	NO						
20200623 11:15:00	0.6965	0.6966	0.6964	0.6964	244	0.69648	66	15		-0.000633	1.8E-05	0.000477	BUY	NO						
20200623 11:20:00	0.6964	0.6965	0.6962	0.6965	244	0.69639	82	20		-0.000636	0.000294	-0.000171	NO	NO						
20200623 11:25:00	0.6965	0.6965	0.6959	0.696	311	0.69609	102	25		-0.000921	-4.8E-05	-0.001476	NO	NO						
20200623 11:30:00	0.696	0.6961	0.6959	0.6959	340	0.69595	107	30		-0.000843	-0.000606	-0.001593	NO	NO						
20200623 11:35:00	0.6959	0.6959	0.6956	0.6957	322	0.69579	119	35		-0.000819	-0.001338	-0.001575	SELL	NO						
20200623 11:40:00	0.6957	0.6959	0.6957	0.6959	201	0.6958	65	40		-0.000561	-0.00153	-0.000918	NO	NO						
20200623 11:45:00	0.6959	0.6959	0.6955	0.6958	224	0.69568	65	45		-0.000516	-0.001968	-0.000819	NO	NO						
20200623 11:50:00	0.6957	0.6958	0.6951	0.6952	439	0.69542	165	50		-0.000939	-0.002334	-0.001386	NO	NO						
20200623 11:55:00	0.6952	0.6954	0.695	0.6954	443	0.69522	144	55		-0.001353	-0.002418	-0.001629	NO	NO						
20200623 12:00:00	0.6954	0.6955	0.6954	0.6954	166	0.69545	55	0		-0.001185	-0.001452	-0.000288	NO	NO						
20200623 12:05:00	0.6954	0.6955	0.6953	0.6953	92	0.69541	33	5		-0.001308	-0.001194	-0.000117	BUY	NO						
20200623 12:10:00	0.6952	0.6955	0.6952	0.6954	163	0.6954	44	10		-0.001191	-0.000972	9E-05	BUY	NO						
20200623 12:15:00	0.6954	0.6957	0.6954	0.6956	130	0.6955	43	15		-0.000873	-0.000414	0.000468	BUY	NO						
20200623 12:20:00	0.6955	0.6955	0.6953	0.6955	153	0.69544	54	20		-0.000738	-0.000216	-1.9984E-16	BUY	NO						
20200623 12:25:00	0.6954	0.6955	0.6953	0.6953	82	0.6954	34	25		-0.000609	-0.000102	-0.000135	NO	NO						
20200623 12:30:00	0.6954	0.6955	0.6952	0.6953	117	0.69532	44	30		-0.000582	-0.000132	-0.000414	NO	NO						
20200623 12:35:00	0.6952	0.6952	0.6947	0.6948	321	0.695	106	35		-0.000807	-0.000888	-0.001494	SELL	NO						
20200623 12:40:00	0.6948	0.695	0.6945	0.6947	194	0.6947	83	40		-0.000996	-0.001776	-0.002124	SELL	EXECUTE						
20200623 12:45:00	0.6947	0.6948	0.6945	0.6948	205	0.69458	65	45		-0.001035	-0.002058	-0.00189	NO	NO						
20200623 12:50:00	0.6948	0.6949	0.6946	0.6946	91	0.69477	36	50		-0.000726	-0.001596	-0.000468	NO	NO						
20200623 12:55:00	0.6946	0.6946	0.6943	0.6944	230	0.69447	65	55		-0.00108	-0.002268	-0.001053	NO	NO						
20200623 13:00:00	0.6944	0.6947	0.6944	0.6947	180	0.69454	51	0		-0.00108	-0.000324	-0.000324	NO	NO						

The goal of this task is to implement a Long short-term memory (LSTM) neural network using **customized** time-series data on a machine learning trading algorithm in Java for commodity trading, modified and hugely forked or borrowed on version of Rezaul Karim's stock trading implementation specifically found at GitHub URL below—chapter 7 of his book.

<https://github.com/PacktPublishing/Java-Deep-Learning-Projects/tree/master/Chapter07>

Author: **Md. Rezaul Karim**

Book: **Java Deep Learning Projects: Implement 10 real-world deep learning applications using Deeplearning4j and open source APIs**

<https://github.com/rezacsedu>

Packt Publishing sells the book with option for PDF version.

https://www.packtpub.com/product/java-deep-learning-projects/9781788997454?_ga=2.61244076.1886860056.1659036504-969600299.1659036504

Book is also available at Amazon.

<https://www.amazon.com/Java-Deep-Learning-Projects-Deeplearning4j-ebook/dp/B07F6YKKVC>

Both the paper hardback and PDF versions offer freely available working codes in downloadable format.

Rezaul has his implementation set up for 448 stocks (securities) over 1439 individual customers' account. Those figures are derived from one of the provided Microsoft Excel files —specifically

Name	Type	Compressed size	Password ...	Size	Ratio	Date modified
fundamentals	Microsoft Excel Comma Separated Values File	342 KB	No	1,382 KB	76%	6/5/2018 11:52 PM
prices	Microsoft Excel Comma Separated Values File	13,803 KB	No	50,457 KB	73%	6/5/2018 11:52 PM
prices-split-adjusted	Microsoft Excel Comma Separated Values File	14,593 KB	No	51,467 KB	72%	6/5/2018 11:52 PM
securities	Microsoft Excel Comma Separated Values File	15 KB	No	60 KB	76%	6/5/2018 11:52 PM
StockPriceLSTM_ALL	Compressed (zipped) Folder	3,703 KB	No	3,702 KB	0%	6/10/2018 3:42 PM
StockPriceLSTM_CLOSE	Compressed (zipped) Folder	3,661 KB	No	3,660 KB	0%	6/6/2018 11:07 AM
StockPriceLSTM_OPEN	Compressed (zipped) Folder	3,701 KB	No	3,700 KB	0%	6/10/2018 4:00 PM

Rezaul's implementation appears targeted for brokerage firms, trading houses and big funds transacting in multiple securities; whereas this implementation we seek is for a sole player or individual trader.

Thus, we will simplify as follows.

- Do away with the “fundamentals.csv” file.
- Do away with the “prices-split-adjusted.csv” file.
- We don’t need the “Securities.csv” also.
- We will not use 3 files above.

Only One.

Instead, we will work with the “prices.csv” file only. Although Rezaul’s file has 851,264 data points (or rows in Excel), once we count and remove duplicates, we see in actuality it only has 501 unique securities (stock symbols).

Ok.

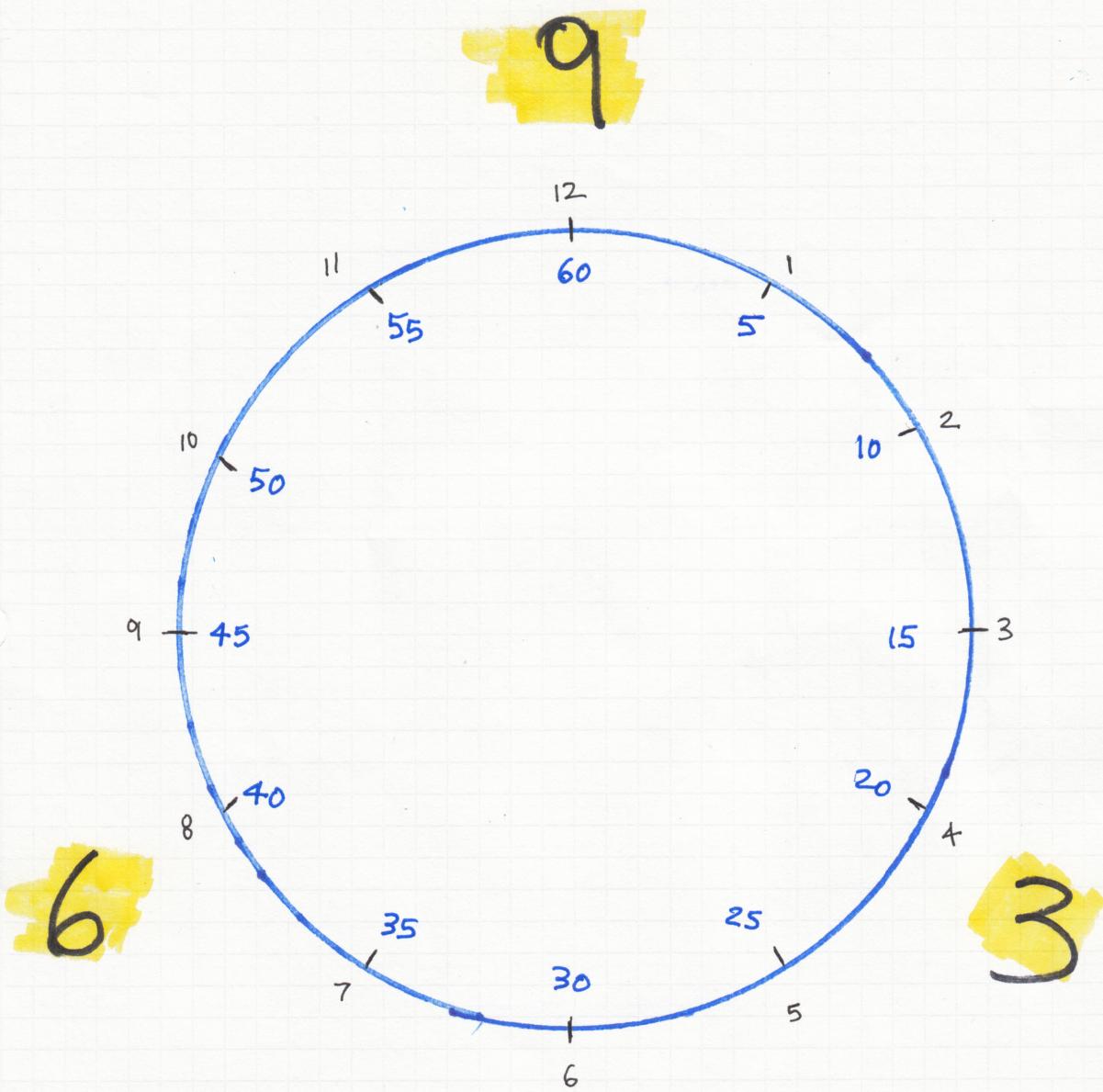
Now onto our strategy combining the mathematical mysteries: tesla369 and golden ratio. Of course, you are free to choose any secret or “personal” strategy—as long as you can parley it (present it) onto a Microsoft Excel CSV file in order for it to serve as the training data of successful (profitable) trades.

On the face of it, this strategy combines the famous **Tesla—3—6—9** code and the **Golden Ratio**. (*While Rezaul’s implementation is for stocks, hence needing numerous securities and possibly higher volume transaction, our attempt is for futures, needing much less volume due to the privilege and danger afforded us by leverage.*)

It’s not necessary to go into mathematical derivation or expounding of above famous mysteries further, suffice to say, if we superimpose Tesla-3-6-9 onto a clock face, we know

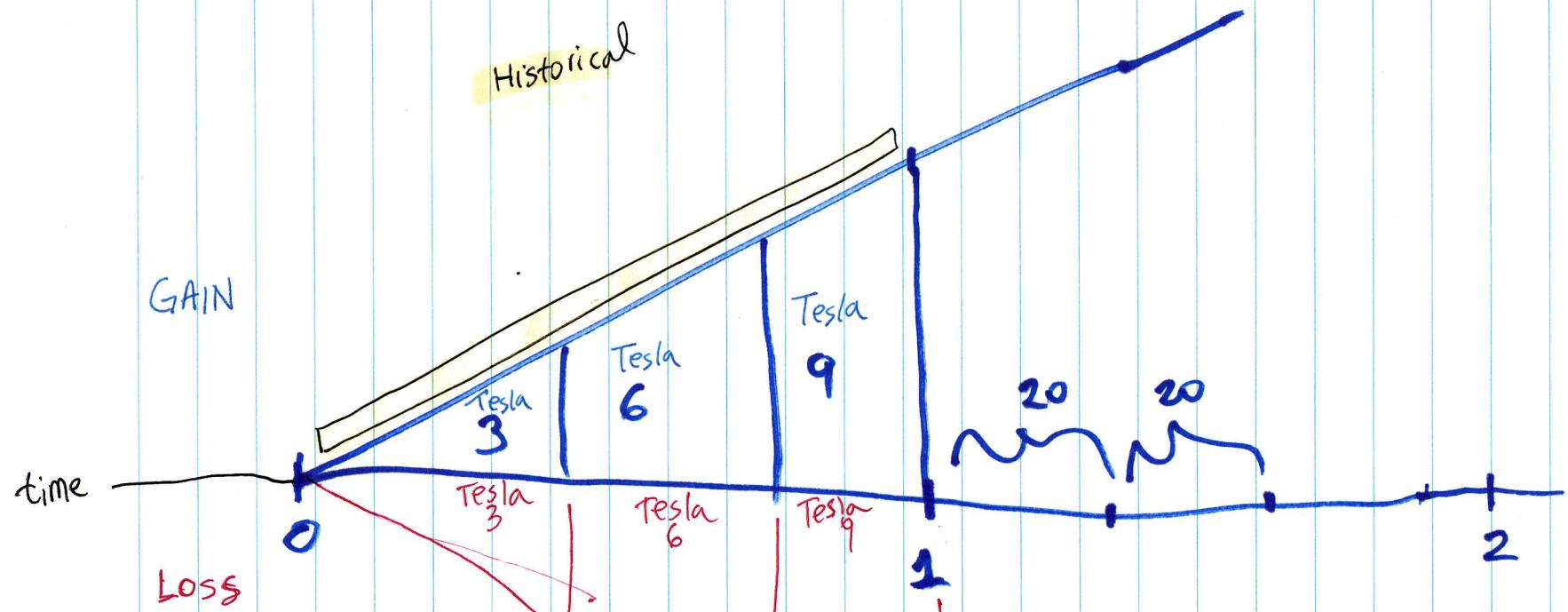
- the 3 corresponds to the 4 o’clock hour or 20-minute mark.
- the 6 corresponds to the 8 o’clock hour or 40-minute mark.
- the 9 corresponds to the 12 o’clock hour or 60-minute mark.

See following image—



GAIN
LOSS

Depends on position whether LONG, SHORT



Fair
Speculation
Not
Approximation

ALL IF CONDITIONS ARE MET.

Our strategy calls for pulling 1-hour historical data—essentially exactly 1 hour after market open—do a brief calculation to determine the parameters Tesla_3, Tesla_6, Tesla_9, and the derived signal Decision (from if statements) whether “BUY”, “SELL”, or “Do nothing”. The only thing we seek from the neural network is whether to execute or not.

Currency	Futures contract at play
YYYYMMDD_HHMMSS	The bar's date and time (either as a yyyyymmss hh:mm:ss formatted string or as system time according to the request).
Open	The bar's open price.
High	The bar's high price.
Low	The bar's low price.
Close	The bar's close price.
Volume	The bar's traded volume if available (only available for TRADES)
Count	The number of trades during the bar's timespan (only available for TRADES)
WAP	The bar's Weighted Average Price (only available for TRADES)
Minute	Minute easily derived using Java script.
Tesla3, Tesla6, Tesla9	Almost nearly custom calculated values performed on the previous 1 hour historical data returned from broker and held in the bar class. On the Excel sheet (a XLSM- macro enabled file where the training data is created), the values are calculated using both formulas and very simple VBA macros within the Excel sheet. Both Excel files, the macro-enabled xlsm and csv (extension) will be provided separately.
Decision	Buy or Sell or "do nothing" recommendation from signal. Signals "do nothing" are not executed.
EXECUTE	This is only outcome we want returned--whether to execute order or not.

"Decision" is a calculated field from our signal.

Tesla3, 6 and 9 are calculated fields on Excel.

EXECUTE is field whose return we want from the machine learning.

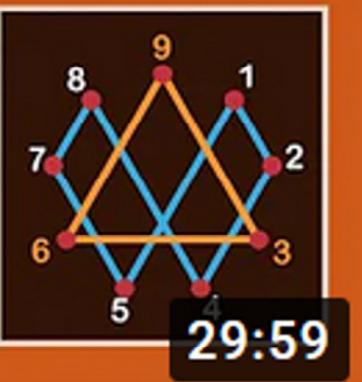
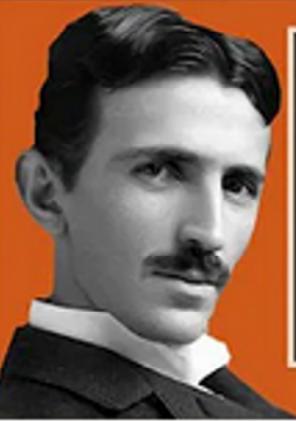
All other fields are API return values/data from broker.

For in-depth explanation..See IBKR's API Reference Guide.

<https://interactivebrokers.github.io/tws-api/>

https://interactivebrokers.github.io/tws-api/classIBApi_1_1Bar.html

MATHOLOGER



Tesla's 3-6-9 and Vortex Math: Is this really the key to the...

Mathologer

1.7M views • 5 months ago

Tesla's 3-6-9 and Vortex Math: Is this really the key to the universe?

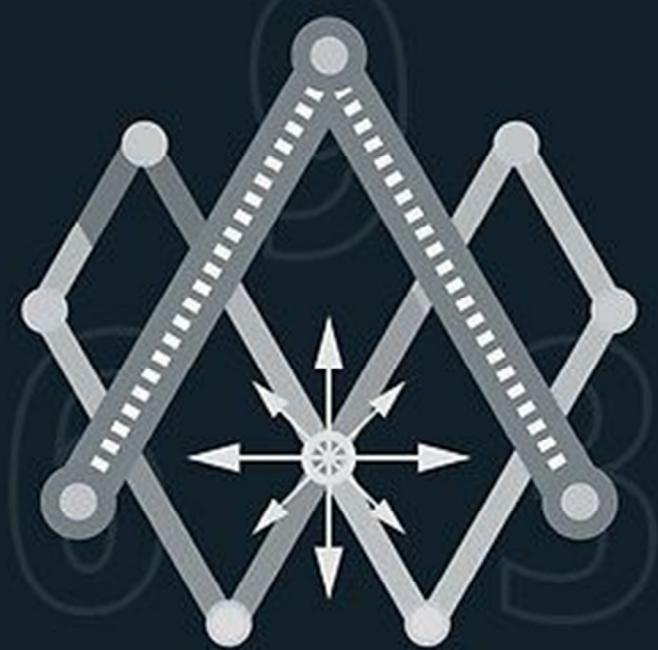
1,789,617 views Feb 19, 2022 Today, a long overdue foray into the realm of VORTEX MATHEMATICS :)



Mathologer

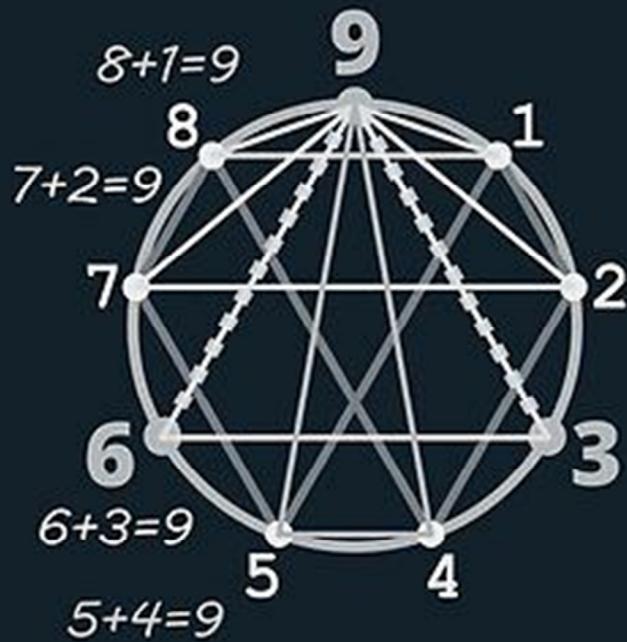
811K subscribers

<https://www.youtube.com/watch?v=6ZrO90AI0c8>



KEY TO THE UNIVERSE

tuzlay



Tuzlay Art - Pinterest

<https://www.pinterest.com/pin/537335799299052850/>



3 triangles :

$$1.4.7 = 12 = 3 \quad 8+1=9 \quad 7+2=9 \quad 6+3=9 \quad 5+4 = 9$$

$$2.5.8 = 15 = 6 \quad 1+2+4+8+7+5 = 18 = 9$$

$$3.6.9 = 18 = 9 \quad 1+2+3+4+5+6+7+8+9=45=9$$

<https://www.pinterest.com/pin/853291460622409845/>

<https://matrixdisclosure.com/nikola-tesla-3-6-9/>



Golden ratio



All

Images

Books

Videos

News

More

Tools

About 255,000,000 results (0.62 seconds)

Golden ratio :

Overview

Formula

Examples

Calculators

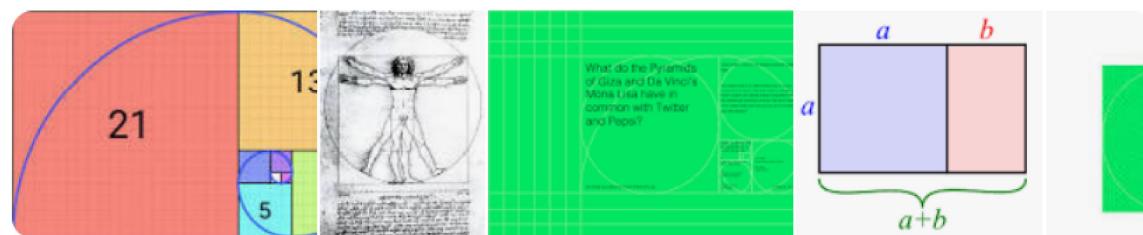
Videos

Practice problems

This search may be relevant to recent activity:

Search History | Feedback

[Golden ratio song](#)



golden ratio, also known as the golden section, golden mean, or divine proportion, in mathematics, **the irrational number $(1 + \sqrt{5})/2$** , often denoted by the Greek letter ϕ or τ , which is approximately equal to 1.618.

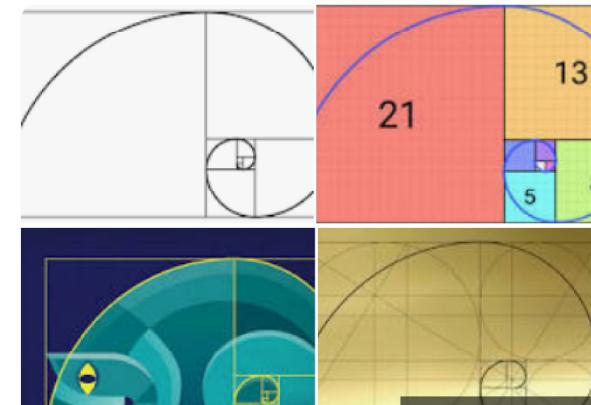
<https://www.britannica.com> › Science › Mathematics

[golden ratio | Examples, Definition, & Facts | Britannica](#)

[About featured snippets](#) • [Feedback](#)

Formula

$$\phi = \frac{1 + \sqrt{5}}{2}$$





WIKIPEDIA
The Free Encyclopedia

Main page

Contents

Current events

Random article

About Wikipedia

Contact us

Donate

Contribute

Help

Learn to edit

Community portal

Recent changes

Upload file

Tools

What links here

Related changes

Special pages

Permanent link

Page information

Cite this page

Wikidata item

Print/export

Download as PDF

Printable version



https://en.wikipedia.org/wiki/Golden_ratio



Article

Talk

Read

View source

View history

Not logged in Talk Contributions Create account Log in

Search Wikipedia



Golden ratio

From Wikipedia, the free encyclopedia

This article is about the number. For the Ace of Base album, see [The Golden Ratio \(album\)](#). For the calendar dates, see [Golden number \(time\)](#).

In mathematics, two quantities are in the **golden ratio** if their **ratio** is the same as the ratio of their **sum** to the larger of the two quantities. Expressed algebraically, for quantities a and b with $a > b > 0$,

$$\frac{a+b}{a} = \frac{a}{b} =: \varphi$$

where the Greek letter **phi** (φ or ϕ) represents the golden ratio.^[a] It is an **irrational number** that is a solution to the quadratic equation $x^2 - x - 1 = 0$, with a value of^{[2][1]}

$$\varphi = \frac{1 + \sqrt{5}}{2} = 1.618\ 033\ 988\ 749\dots \text{(OEIS: A001622)}$$

The golden ratio is also called the **golden mean** or **golden section** (Latin: *sectio aurea*).^{[3][4]} Other names include **extreme and mean ratio**,^[5] **medial section**, **divine proportion** (Latin: *proportio divina*),^[6] **divine section** (Latin: *sectio divina*), **golden proportion**, **golden cut**,^[7] and **golden number**.^{[8][9][10]}

Mathematicians since Euclid have studied the properties of the golden ratio, including its appearance in the dimensions of a **regular pentagon** and in a **golden rectangle**, which may be cut into a square and a smaller rectangle with the same **aspect ratio**. The golden ratio has also been used to analyze the proportions of natural objects as well as man-made systems such as **financial markets**, in some cases based on dubious fits to data.^[11] The golden ratio appears in some **patterns in nature**, including the **spiral arrangement** of leaves and other parts of vegetation.

Some 20th-century **artists** and **architects**, including Le Corbusier and Salvador Dalí, have proportioned their works to approximate the golden ratio, believing this to be **aesthetically** pleasing. These often appear in the form of the **golden rectangle**, in which the ratio of the longer side to the shorter is the golden ratio.

Contents [hide]

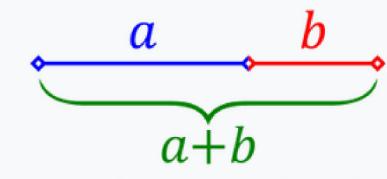
1 Calculation

2 History

3 Mathematics

... 4 External links

Golden ratio



$a+b$ is to a as a is to b

Line segments in the golden ratio

Representations

Decimal $1.618\ 033\ 988\ 749\ 894\dots$ ^[1]

Algebraic $\frac{1 + \sqrt{5}}{2}$

Continued fraction $1 + \cfrac{1}{1 + \cfrac{1}{1 + \cfrac{1}{1 + \cfrac{1}{1 + \ddots}}}}$

Binary $1.1001\ 1110\ 0011\ 0111\ 0111\dots$

Hexadecimal $1.9E37\ 79B9\ 7F4A\ 7C15\dots$



— PLANTS —

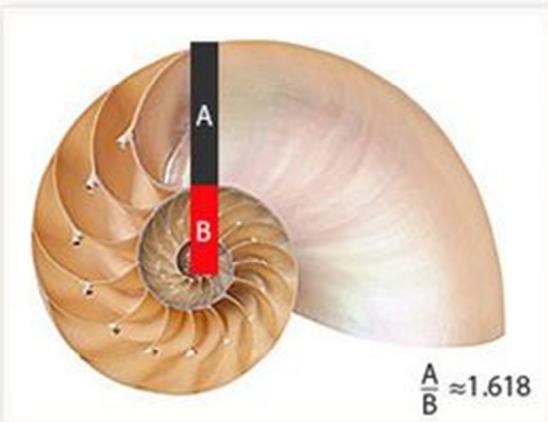


Sunflower

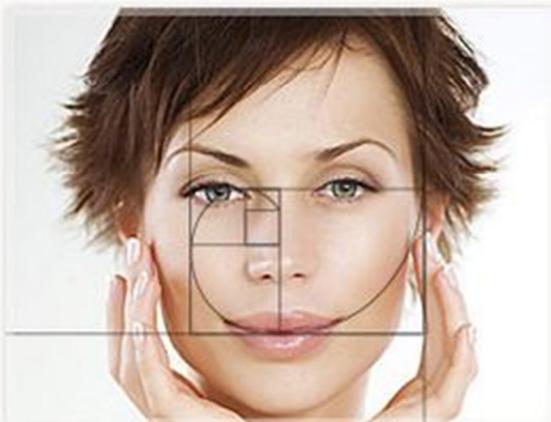


Spiral Aloe

— ANIMALS —

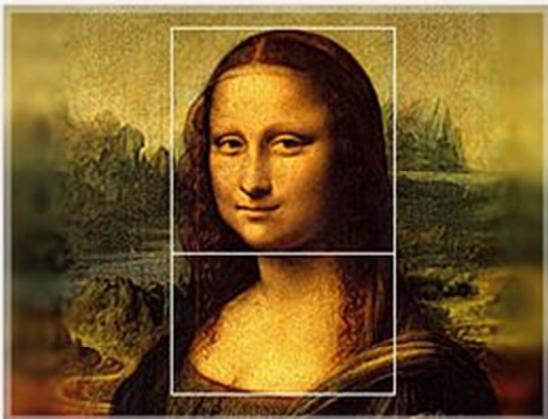


Nautilus Shell



Human Face

— ART —

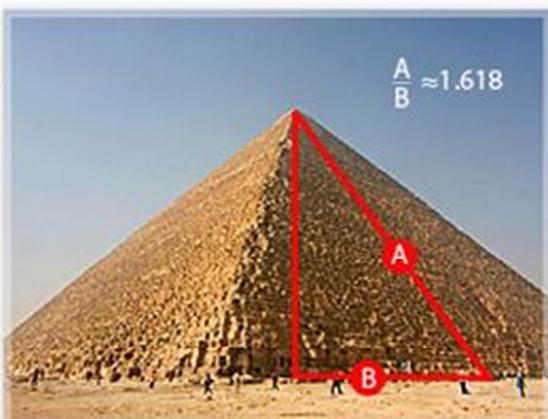


Da Vinci's Mona Lisa



Dali's Sacrament of the Last Supper

— ARCHITECTURE —



Great Pyramid of Giza



Parthenon

The need for simplicity... thru ease

Target audiences of this project are established day-time individuals/professionals in whatever respective field, be it finance, accounting or any business arena. Hence our objective is to gently introduce neural networks and machine-learning to add onto an already existing automated trading system for these prospective traders and speculators, therefore we must appreciate since they're already somewhat set career wise, they have no ambition of pursing advanced certification.

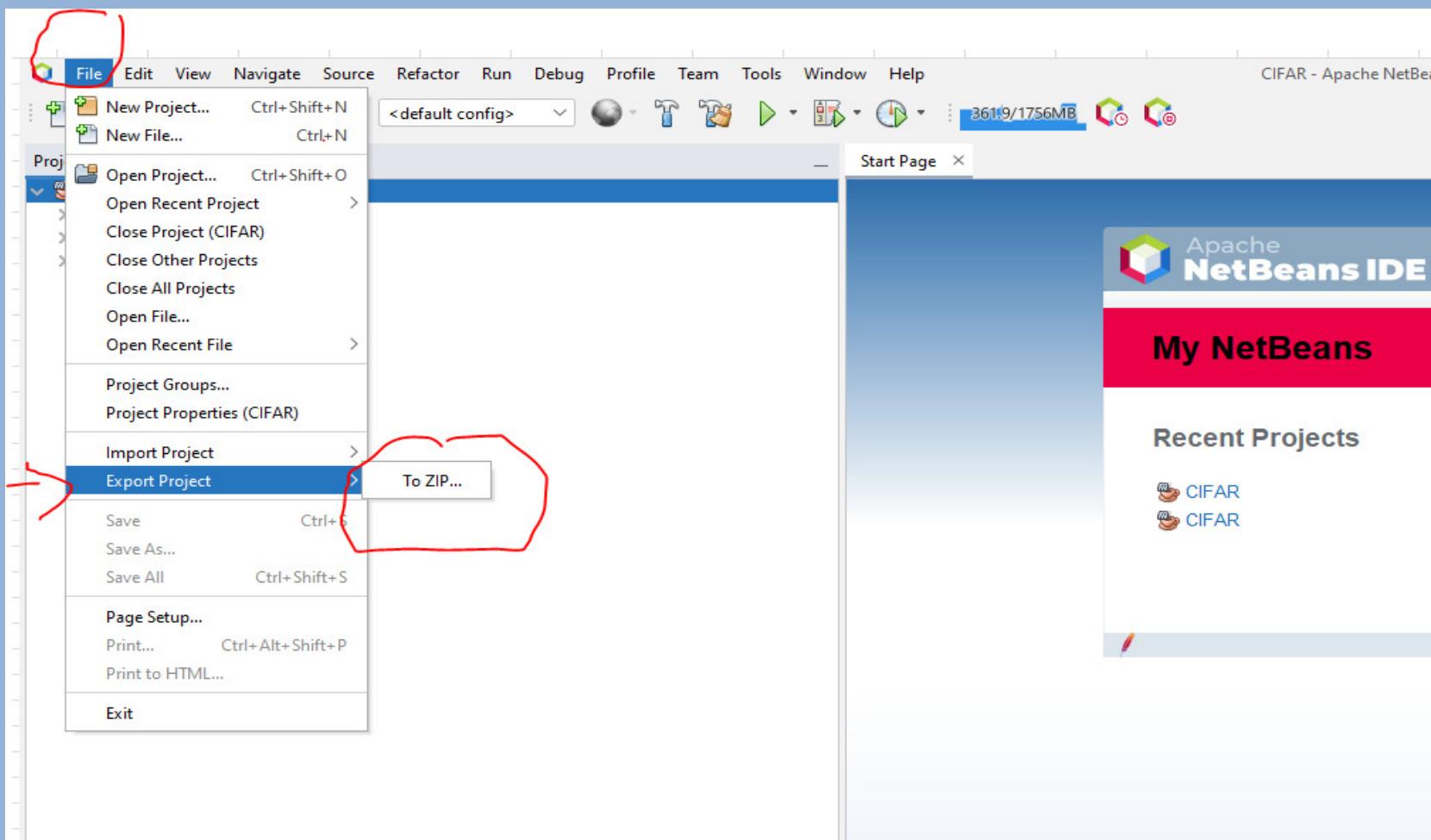
Our thinking... inspired by caveman could go as follows. For instance, borrowing from a Cro-Magnon man approach—a car/automobile can travel from point A to point B without windshield wipers, windshield wiper fluid, glove compartment, carpet, both driver side and passenger side mirror, etc. All the car needs are the motor, transmission and seat. This type of brutal caveman simplicity is what we're seeking in this world where applications (and IDE's) increasingly become complicated, bloated with each update, upgrades and revision. We're not appreciating the spiritual message of the COVID “pause” where we need to pause on a societal, cultural, technological level, slowdown and think...instead of rushing. One can make money without travelling the HFT road.

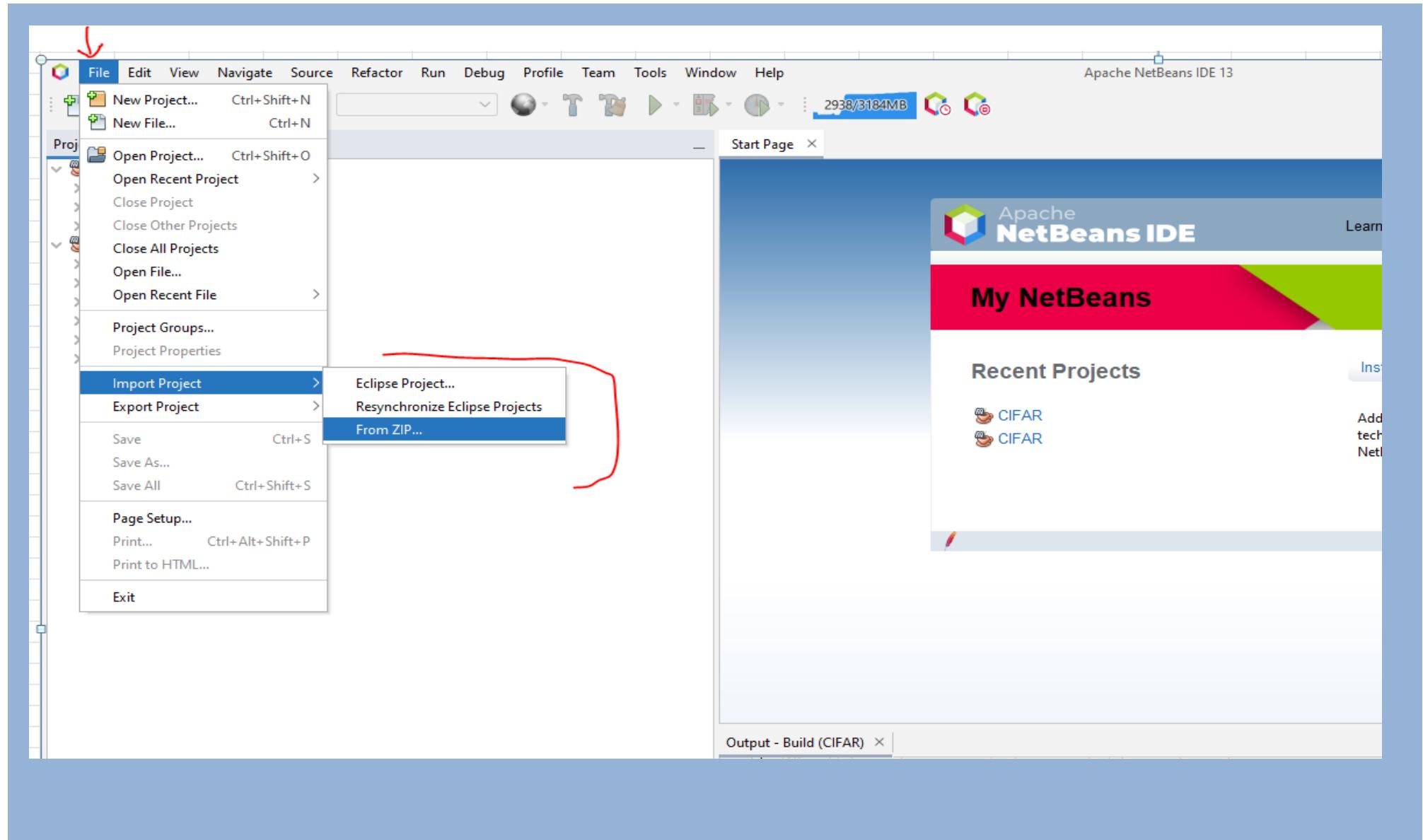
Not only referring to Rezaul Karim's implementation, but almost majority of advanced applications nowadays seem unnecessarily(?) complicated to a novice given the bundling numerous artifacts in the POM file. To a novice....

Hence to address this concern, as a humble novice programmer with no intention of pursing high-level java certification, I respectfully ask for simplifying this request to make as bare minimum and as lightweight as possible. I ask to do away with logging, and the majority of the POM libraries which would not be used otherwise.

For this reason, we ask this implementation be completed using Apache NetBeans IDE and not IntelliJ. Once completed and compiled, we ask if we can obtain a ZIP file exported from NetBeans. (See following figures). Exports from NetBeans are easier to work with since the code is already successfully compiled and ran. For our part, upon receipt, we would only need to import the ZIP file to our NetBeans and run. Painful experience has shown me this approach results in greatest ease saving time configuring issues related to POM, maven, artifacts, JAR files and libraries.

Because for our objective the main training data is the time-series custom file in Microsoft Excel CSV file, this approach can potentially reach greater audience because more people know Microsoft Excel than Java. Greater potential for reachability exists as well as user can employ/represent any hard to code strategies—such as Japanese





harami candle stick patterns (with volume) onto a Microsoft CSV file—where it's easier to populate the Excel cells using VBA than otherwise having to employ Java on the previous 1-hour historical data. [Underlying VBA codes/scripts are provided showing how we created the file].

I would need all the libraries and source code----everything--in a ZIP file for returning it to me. All I need would be run open project and run hopefully minimizing configuration issues.

See image: Apache NetBeans, Export Project --> to ZIP.

And then I would simply Import the Project from ZIP from NetBeans. This very easy and would save both us time configuring issues.

Thus, to recap, we ask all programmers upon successful completion and compiling of this task if they would please supply us with a ZIP file directly exported from Apache NetBeans. For our part, we would only need to import the ZIP file to our NetBeans and run. Thank you.

DISCLAIMER: This PDF booklet is authored by coderofjava888 (email robertvanquish@yahoo.com)

DISCLAIMER: Rezaul Karim, Ph.D. is not associated with this publication in any manner. Instead, Rezaul Karim is the Ph.D. whose work help we seek to implement the machine-learning aspect or portions of the code.

CoderOfJava

<https://github.com/CoderOfJava8888>