Electronic Market Making with Angel One Smart API

Introduction

Hello, traders. Welcome to Algorithmic Trading with Angel Smart API. In this course, we will look at an electronic market making strategy. This introductory lecture is going to explain what the course is all about. And I will give a brief course outline to begin with.

So we have eight sections in this course. Section one, which is the introduction that is this lecture video. We will talk briefly about what the course is covering, who is it going to benefit, what the course contains, what it does not contain, and a brief outline of the trading strategy and the methodologies and the code that we are going to use in this course. So that is section one, which is the introduction. Section two is a very basic stock lookup code. So we have basically a little bit of code whereby we can access the angel broking API and the servers and it will return the exact token number and the stock code for whichever stock we are interested in. Section seven is just a little bit of information on how you can save intraday data and do some backtesting and trade simulation. So one of the points that I will be making constantly across during this course is that there is no perfect algo that will give you a profit every time or every day.

A trading algorithm might in general be, you know, it might be good, it might be able to capture profits, but it requires constant tweaking and monitoring and modifications on a day to day basis. And based on the latest data, we have to keep on testing and editing the algorithm and the parameters used in the algorithm. So Section seven is all about doing some basic back testing and trade simulation. So based on the simulation or the data for today, we try to generate parameters for the next day. That's the idea of Section seven. And finally, in section eight, we will have a discussion and some conclusions to talk about, some do's and don'ts, some good practices, some bad practices, which we must avoid. And basically how the course can be, you know, made more useful on a longer term basis for those of you who are interested in electronic market making as a trading strategy.

So let's start with the introductory lecture. So what is this course exactly? This is an algorithmic trading course. It is meant for intermediate to advanced levels of difficulty. And we are using the Angel Smart API to place our trades. So this is key. It is an intermediate to advanced level of difficulty, so you should have a working knowledge of Angel Smart API. Uh, and this course is basically showing, explaining or demonstrating an electronic market making strategy. So what do we mean by electronic market making? It means that we are trading continuously in the order book. So we are sending automated bid and ask orders to capture the spread. So at any given time we will have an order to buy or to sell or both. And we want to just buy and sell continuously and try and earn the spread. That is the difference between the buy and the sell prices, the bid and ask. So that is what we call electronic market making or automated market making. And in this course, as I said earlier, we will look at four different variations, a basic strategy for electronic market making on one stock, a basic strategy for multiple stocks, the advanced strategy for single stock and the advanced strategy for multiple stocks.

So what is this course not including or this course is not a quick and guaranteed way to make profits in the market. As I said before, there are no guaranteed known trading algorithms that will always make you profits. This is not a quick and guaranteed way to make profits in the market.
It is not a perfect algo that will make a profit all the time. No, it requires constant monitoring, tweaking, and at best you may find that it is somewhat profitable. So you have to really test it out for yourself and see whether it works for you. And profitability will depend on a lot of things, not just on the choice of your trading parameters, which we will cover, but also on your cost structure, your brokerage level, your other costs. All of these matter. So I repeat, this is not a course where you can make a quick and guaranteed way to make profits in the market. This is not a perfect algorithm that will make a profit all the time. And this course is not a support forum for Angel Smart API. So what exactly is the electronic market making? As I said before, this is a trading strategy that offers continuous limit orders to buy and sell. So in other words, we are providing liquidity to the order book. So we are sending orders to buy and sell at the same time and we want to capture the spread. We hope to gain the spread as a profit. What is the spread? The ask minus the bid price. So if you look at the best five buy and sell orders, the order book or what we call the market microstructure. You can see that the buy price and the sell price are 84.8 and 84.9. So the best bid is to 84.8 and the best ask is to 84.9 and the spread is basically 10 paisa. So an electronic market maker is trying to buy at 84.8 and sell at 84.9 and earn that 10 paisa spread. So the idea is that it is automated, so a computer can do it several times a day over and over again. At the end of the day, we hope that it is a sizable amount of profit.

Okay, so what are the risks of electronic market making? There are quite a few risks for electronic market making strategies. Number one, it is very sensitive to brokerage costs, so you need to have very low brokerages for this to work. You can't be paying, you know, high brokerages. And it will depend, you know, it will depend on your choice of the stock. You know, the margin or the spread that you decide to capture, the amount of the spread you decide to capture, all of these will come into play. But the key point I'm making is it is sensitive to brokerage and costs. So the lower your brokerage, the better it is. Then the other risk is that the algo has to maintain a balanced book.
So we don't want to be adding on to our long or only adding on to our short position. We don't want to become directional, right? So we want to be what if we buy ten stocks, then we sell ten stocks, then we sell ten stocks, then we buy ten stocks. We don't want to keep buying. And then we become directional to the long side. So the algo has to maintain a balanced book. So when we buy, then we want to sell. If we sell, then we want to buy. We don't add to our position or we don't average.
We avoid averaging or very limited scenarios where we try or where we agree to average or up to a predefined limit, basically. Overall, the algo has to maintain a balanced book. You can't become directional. The performance of any algo like this has to be monitored for malfunctions. Sometimes, you know, if there's a code error, it can start to trade and it can just instead of, you know, buying at the lower bid and the selling at the higher ask, if you do the opposite, it will keep trading. It will go into what is called a doom loop. And then within very quickly, you can get wiped out. You know, there are famous cases where this has happened to very well established companies.

So, you know, retail traders like us are at a very high risk because we have to know exactly what the code is doing and we have to monitor it If if there's some error in the code or if it malfunctions for some reason, then you are at risk. You know, so you have to be careful of the code and the performance of the algo. And market conditions are different every day. So the same algo with the same parameters will not work on all the days. It needs to be constantly tweaked. The choice of the stocks is the number one decision. The trade quantity, the position sizing is another choice. You know, the spread or the margin of the profit margin we hope to make per trade.

These are all parameters that have to be defined and they will have to be constantly modified or tweaked on a day to day basis. So that is very much needed to, you know, if you ever hope to make this successful, you will have to keep tweaking the algo every day. So because market conditions are different every day. Now, what are some ideal features of electronic market making strategies? Number one, trades are closed very quickly. The best trade is the fast rate. If you can buy and sell within a few seconds, that is the ultimate trade. You're in and out. You took your small profit, you're in, you're out. So at least in this course, we are looking at time frames of less than one minute. So in other words, we want to be in and out of a trade within one minute. That's the basic idea that we're trying to do. The trade sizes are small. So whenever we trade, we're not trading in huge quantities. We are trading in small quantities, but the number of trades are high. So we might be buying just ten stocks and selling it for 1 rupees profit. But we hope to do that over and over again several hundred times a day. And therefore, at the end of the day, it becomes a sizable position or a sizable profit or outcome. The trade sizes are small. We're not risking too much on any single trade, but the number of trades are high. That is the ideal situation for electronic market making. And there are no overnight positions in the strategy.

So you have to square off at the end of the day, whatever is your result, profit or loss, you have to square off at the end of the day. And this is basically purely an intra day strategy. Okay, So what is the basic outline of the strategy? Essentially, just to give a brief intro of what kind of strategy this is. Recall that I said that we are continuously trying to send a buy and a sell order to the market and we are trying to get it executed on the buy side and the sell side and gain the spread. Right? So in very simplistic terms, what we can say the strategy is, is basically we check the last traded price and then we send a buy and sell order. So the buy order will be slightly less than the last traded price and the sell order will be slightly higher than the last traded price. And that difference is the spread that we are trying to make. Buy low, sell high. So the last traded price was, let's say, 100 rupees for any particular stock. We might decide to send the buy order at 99.75 and the sell order at 100.25. Within that, you know, within the time frame that we give to the trade. Once we send these to buy and sell orders, we check for the trade execution. And if it does get executed, we send the take profit order right at some predefined margin. So let's say our margin was only 30 paisa and the buy order for 99.75 got executed immediately. We send the sell order for 30 paisa. So 100 .05 is the take profit order price. So it's a very simple strategy. This is a very simple explanation of the strategy, but that is basically what the algo is going to do. And then we repeat the cycle every minute. We keep on repeating the cycle every minute. It's just a matter of in many ways you can think of it as a brute force kind of a trading strategy where we are constantly ready to buy and sell at.

We are constantly ready to buy slightly below the last traded price and we are constantly ready to sell at slightly above the last traded price. Right? Purely we are offering orders, limit orders to the market. And as I said, we cover four variations of the trading strategy in this course. So the basic strategy on one stock, basic strategy on multiple stocks. The advanced strategy on one stock and the advanced strategy on multiple stocks. So four different variations and each of them has its own advantages, has its own disadvantages, which we will get into in more detail. But essentially, this is the sort of the scope of this course. That is the basic strategy on one stock. A basic strategy on multiple stocks. Advanced strategy on one stock and the advanced strategy on multiple stocks.

So coming to the end of this introductory lecture, final thoughts? I must caution you once again, do not under any circumstances blindly copy and start trading with this code. Right. It can have very bad outcomes. You can make serious losses very quickly because this is a strategy that sends many, many orders very, very quickly. So you need to be careful. So don't just copy the code and start, you know, press play and see what happens. Don't do that. It's very risky. So understand the logic and try it out incrementally. So do the first step. Make sure you're clear.
Then do the second step. Make sure you're clear like that incrementally, you need to test your knowledge. That's the key thing here. So understand the logic and try it out incrementally, which means use small trade quantities and trade for a few cycles. By cycles I mean iterations or a few minutes, you know, run it for a little bit. With very small quantities. Check the price and quantity of the sent orders are actually doing what you expected it to do. So check the price and quantity of the sent orders and if the orders are sent as expected. So you need to make sure that the orders are going as you expected it to go, You know, so it is buying at below the last traded price. It is trying to sell at above the last traded price, not the opposite, because the opposite you will lose the spread, you will keep losing. So these are very simple mistakes. So you have to be careful. Then repeat this again to double check for edge cases like partial execution. Sometimes if an order gets partially executed, these are edge cases. So check that it is still okay. That it is. I mean, we have the code will address these edge cases as well, but make sure that it is doing what you expect it to do and you are happy that you want it to do. Then very important, after you have done a turnover of maybe 1 or 2 lakhs, you need to stop and wait for the daily trade statement. Right. So just pause it for that day and check your trade statement. Why? Because you need to check that the cost of trading is within the expected margins. If not, you have to increase the margin or your profit margin or your spread because that spread has to cover your cost structure and your, you know, a component which includes your profit margin as well. Right. So if you've got high brokerages and high cost, that spread will become wider and wider and then it becomes it's not ideal for this strategy. So the best way to know what your cost structure is, is either you trade 1 or 2 lakhs and then see what the trade statement looks like. And that will give you a better idea of, you know, of how the algo will work in your case because it will be different for each person based on their algo, based on their brokerage and your cost structure. If the costs are very high, then you will need to change the trade parameters. So in other words, you have to tweak the margins so that you are covering your costs or you have to change the trade quantity so that the order is becoming more substantial to cover, you know. The trade size is big enough to make a profit eventually. And this algo will require daily monitoring and tweaking. It's never optimal for long, you know, because markets are changing every day.

You will have to keep tweaking the algo tweaking by tweaking, I mean, you will have to, you know, the decision on which stocks to trade. That is one decision that has to be reviewed every day. The decision on what your profit margin or your spread margin should be, that's another decision that has to be reviewed every day, at the end of the day. The trade quantity. Are your positions sizing? That's another decision that has to be reviewed every day. And to some extent, from what time to what time are you going to run this algo? That's another decision that you have to review almost every day.

Stock Lookup and Trade Strategy

Stock Lookup

So welcome to section two of our course. And of course, as you know, in this course, we are looking at an electronic market making strategy. Section two is basically we are only going to cover two topics here. The first lecture video in this section is going to cover this code, which is essentially how we can look up the token number and the token symbols, right? So because the API requires the token number for each stock as well as the ticker symbols for each stock while placing an order. So this is just a little bit of code to, you know, to quickly get that information whenever we want to. Get into electronic market making in the next sections when we want to run the strategy and when we are choosing stocks, this code will be useful to confirming the, you know, the details of the stock. By details, I mean the token number and the ticker symbols for each exchange.

So basically it's fairly straightforward code. The file is called stock lookup, so we import the URL library and the JSON library. This instrument is an important one. This is from the Angel, as you know, from the Angel website. This is where they maintain all the symbols and the tokens for every stock that is offered for trading. So if you go to that particular link, this is what you will see. You are basically going to see this JSON file, which is a list of dictionaries. So we have all of these different instruments and stocks that are available for trading. So we have NSE, there could be equity symbols like so this is an equity. Then you could have various different futures and options. You could have mutual funds, you could have debt markets. So there are different instruments or stocks which are being traded here, right?

So from this we have to get the correct token number and the symbol for the stocks we are interested in. Right. It's not easy to manually search through this. It's not ideal. So we want something which can programmatically give us the correct tokens, right? So let's go back to our code. And as you can see here. We define the instrument as this link that is provided by Angel, where they maintain the latest information on instruments that are traded.nAnd then we define response when we open the said URL. And then we create instrument list, a JSON loaded response and we read it. Right now we are for the moment interested only in the equity market, so NSE and BSE.

So we define exchange as a list of BSE and NSE. Then we are defining a function which is called stock lookup, which takes the arguments of the ticker, the instrument list and the exchange.
We define a list, an empty list called data. And then we basically have a for loop which is iterating through the instrument list. And if the instrument name is equal to the ticker that we have specified and the instrument exchange segment equals the first element in exchange. So that is BSE, or if the instrument exchange segment equals the second element of exchange, that is NSE. And the instrument symbol when we split it by the hyphen. The last element equals equity.

Right. So that's a little bit of, you know, precise coding there because we just want the equity tokens and symbols from NSE. NSE has several different exchange segments, right? So we just want the equity part. If these conditions are met. Append to the data list, the instrument exchange segment, the instrument symbol and the instrument token and return that data. So basically, let's try to run this function. So stock details equals stock lookup. And we want to find out the details of the INFY instrument list and exchange. So basically we want to print the stock details of it.

So when I run this code, it will take just a minute to run because it has to go through that rather big JSON file and then pull up the relevant symbols and tokens that match the conditions that we have specified. In this case, we want BSE where the instrument name is equal to the ticker and the NSE where the instrument, you know, has an EQ as the end of the string on the ticker symbol. Of course you could edit this code for your own purposes and that is the idea of this course. You can edit the condition to return whatever you wish to get. Uh, for the moment, as I said before, in this course, we are only looking at the equity segment. So we are doing all of the trading and the testing on the equity segment. So that's why we are limited to the NSE and BSE equity segment.

And there we go. We get some output. So these are the parameters we would use when we want to trade in the NSE equity segment or the BSE equity segment for Infosys. Just for testing purposes. Let's try another stock. Let's get the information for Reliance again. When we run it, it will take a minute to run. And we should get the result in a similar fashion. Now, obviously, as you will see in section three, four, five and six, when we get into the actual trading, these aspects are done much before the market opens. So all of this housekeeping and, you know, testing to check that everything is correct has to be done way before the market even opens. Right. So yeah, there we go. We get the tokens for Reliance at NSE and BSE and this should work for any stock that you wish to trade.

Of course, we are interested in a certain kind of stock, which we will talk about in the next lecture. Uh, is, you know, we will go into the trading strategy and the different parameters that need to
be defined, in the next lecture. So this lecture was just a simple bit of code whereby we can essentially go through this JSON file, which is maintained by Angel, to get the correct symbol, uh, ticker symbols and the token numbers for whichever stocks we are interested in.

Trade Strategy

In this video, we will look at a detailed explanation of the electronic market making trading strategy and what are the parameters that we have to select when we run the strategy and how do we select those parameters? It's basically an explanation of all those steps. Okay.
So we will first start off by explaining the basic strategy and how to select the parameters for a basic strategy on a single stock. And then we will look at the advanced strategy and how to select the trade values for multiple stocks. And we will also talk about the limits.
And lastly, we will also look at how exactly we select the stocks that we want to run our strategy on. So what are the possible conditions that make a suitable stock for our particular strategy in this case.

So to begin with, let's look at the Excel file that we have here. Basically, we will first explain the electronic market making the basic strategy first. Now, assume you have a stock called Reliance. Of course, we know Reliance is very well known. The last traded price of the stock is 2346. The token is 2885. And this is our order book. So this is the present order book. Market is open at one particular point in time. This is the order book when the strategy starts. The best buy price or the bid price is 2345.5 and the best ask price is 2345.75. And we can see the best five bids and offers on the market microstructure or the market depth picture. Now, for this strategy, we have decided to use a trade quantity of 15 and a trade margin of 1.9 rupees. So there is a certain decision making on how we got to these values, which we will look at in the next part of this video.

But for the moment, assume that the trade quantity is 15 and the trade margin is 1.9.
So ₹1.9 is the profit margin that we are looking to get. And for every trade we do, the quantity is 15 shares. Okay. So in that case, let us start with the basic logic of the trading algorithm. The first step of what we call step one is to send order 1A and 1B simultaneously and then repeat that every 45 seconds. So in step one, we are sending order 1A and 1B simultaneously, and we repeat it every 45 seconds. So what is order 1A and order 1B? Order 1A is to send a buy order at 2344.1 for quantity of 15. How did we get to 2344.1? Well, that is the last traded price minus the trade margin. Right. So 23498 -1.9 that is 234.1 and quantity is 15. So order 1A is to send a buy order at last traded price minus margin and order 1B is to send a sell order at last traded price plus
margin. So 2346 + 1.9 that is 2347.9.

Okay. So both those orders are sent. Then we come to step two. If both order 1A and 1B get executed. Remember, this is repeating every 45 seconds. So in that 45 seconds, if both order 1A and 1B get executed, then we return back to step one. We just go back to step one and we restart the procedure. But it may be that both don't get executed. Only one gets executed. So step 2.1 is basically if order 1A gets executed, we cancel order 1B and we send the take profit sell order for quantity 15 at a price of 2346. Again, how did we get 2346? Well 2344.1 was the buy price plus the trade margin. That is my take profit sell order price. And it might be that instead of 1A getting executed, 1B was executed. In that case, we go to step 2.2. If order 1B is executed, cancel order 1A, send the take profit buy order for quantity 15 at a price of 2346. So how did we get to 2346 here? Well.
The initial order was 2347.9. That was a sell order. That price minus the trade margin, which gives us the price of 2346. Right. So we have executed the first buy order or the first sell order, and we have sent the take profit order to the market. Then step 3.1 we are sending a sell order at last

trading price plus margin. So that means whatever the last traded price is, in case this 1A is executed, then it might be that order 2.1 is pending, but it hasn't been executed yet. But we don't wait in perpetuity until it is executed. It might be that the order will not get executed if the price never reaches there. So that is why we go to step 3.1, which is we send a sell order at the last traded price plus margin. So now we have a new last traded price. It is not 2346. It could be 2346. But when the cycle restarts, we are going to check the new last traded price plus the margin. And then step 3.1.1. So that's like a sub step of 3.1. Order 3.1 and order 2.1 one of them is executed, 3.1 or 2.1 is executed and then return to step one and then we cancel the, uh, the other one, which is not executed, basically. Similarly, it could be that when order 1B was executed and Order 2.2 was sent to the market and it is pending in the market, but it hasn't been executed yet. So we have a take profit buy order pending. We send a secondary buy order after the 45 second has passed and that is at the last traded price minus the margin. Right. So then what happens is step 3.2.1, either order 3.2 or order 2.2 is executed, one of them gets executed, cancel the other one and return to step one. Right? So that is the basic logic of the electronic market making strategy.

We send a buy order at the last traded price minus the trade margin and a sell order at the last traded price plus the trade margin. If both get executed, return step one, if one of them gets executed, we send a take profit order for that one. After 45 seconds after the first iteration is over in the next iteration, we only send the opposite side. So if the first order of buy order was executed, then in the second iteration we only send the sell order at the last traded price plus margin. So that means the original take profit order and the sell order will be pending. One of them gets executed, the other is canceled, and we return to step one and restart the whole algorithm again. So ideally, we are hoping to finish each cycle within one minute or the next minute. So within, you know, a few minutes we should be in and out of a trade. That is the idea of this algorithm.

Now, let us look at the logic behind how do we get to these trade quantity 15 and the trade margin 1.9. Right. So recall from the previous lecture, we know how to get the token, the ticker symbol and the token number with the stock lookup code. Right. So we went to the stock lookup code and we got the token number 2885. The symbol is Reliance-EQ for NSE. Same way for BSE. The symbol was Reliance and the token was 500325. Right, That's fine. Now take the close price of the previous day. And then we estimate our costs. Now, this formula will change depending on what your particular brokerage structure and cost structure is. So if you have a brokerage, which is very low and let's say we are at very, very low brokerage, this is the cost structure that you can get in my experience. So whatever the turnover, it is 0.02%. That is the entire cost, including your brokerage as well as your STT, your exchange transaction fees, your SEBI fees, GST, everything. Right? So whatever is the turnover multiplied by 0.02%, which is 0.0002, right? That is the level of cost that you're at.
If that is the case, then when your close price or when the stock price is 2350, then a buy and sell. Turnover means buy and sell. So basically we are looking at the close price multiplied by two times the cost function, which is 0.02%. That gives me a cost structure of 95 paisa. So whenever I buy and sell Reliance stock, which is about 2350 in price, I'm expecting 95 paisa of cost. Anything above this, it should give me a profit. So I want a healthy trade margin because there will be times when I make a loss. So in other words, what this is telling me is if I buy a 2350 and I sell a 2350, I will still lose 94 paisa.

Right. So if I don't make 95 paisa on every trade, I'm actually losing money at 94 Paisa, I'm breaking even. And above 94 Paisa, I am going to make a final profit. So for that, again, this is a broad rule of thumb. I am saying keep it at the trade margin should be double of my cost. So 94 paisa times two and I'm rounding it to the nearest five paisa. That is how I get to 1.9 double of my cost basis. So that is the cost plus a 100% margin. So cell number C6, which is my cost, 0.94 paisa times two rounded in five paisa because tick size is a five paisa in the market, right? So that is the trade margin that I am intending to use. Again, I'm stressing the point here. These will be different based on each person's brokerage structure. Okay? So you have to try it out for yourself and find out what works for you. If you reduce this margin instead of double and you make it only 1.5, you will get higher execution. You're traded. You know, the quantity of the traded turnover every day will go up much higher because it is easier to get executed at 1.5 margin than it is at two. You will get more and more execution more frequently. But your risk of making a profit or your probability of a profitable outcome starts to go down because now your margin is only about 50 paisa per trade, whereas earlier you were looking for a margin of 95 paisa per trade, right? So that is the trade off that you have now. How did we get to the quantity of 15? Well, basically I am assuming that I have a trading capital of about 35,000. So if I have 35,000 in traded in tradable capital, I am willing to divide it in one trade. I am willing to invest 35,000. That means I can trade for 15 quantities, 35,000 divided by the last traded price. That is 2350 rounded with zero decimal and I am getting a quantity of 15. So which means with just 15 quantities, I am running my algorithm on a single stock. That means at any given time, I will not have a net position more than 15.

Right. The net position will never be more than 15. You understand that if I buy for 15, then my quantity will be 15. Then the next step is to sell for 15. Right. So then I go back to net quantity zero. Then I may buy or sell simultaneously. Right. So at any given time, my net quantity may be either plus 15 or -15. It doesn't go more than that. That is why we say the risk is limited to just the amount of net quantity that is open at any given time, which is set at 15. Now, you can set this according to your wish. You can set this according to your wish. If you want to trade with double the quantity or let's say you have 70,000 or 1,00,000 in trading capital that you want to put on just Reliance, then you could be trading at a quantity of, let's say three times that. You know, you just change the formula according to what you want, right? Just change the 35,000 to 1,00,000 and then it will tell you to trade 45 quantities. If you have 1,01,050 in capital, you could be trading at a quantity of 43. Right. So the point is, you can use exactly the value that you want because it is algorithmic. You just have to tell it whether it is 43 or 45 or 41 or 40. It's you know, it's. There is no difficulty for the computer to process whether it is an even number or an odd number, etcetera. So we would like to make it as precise as possible.

So if that is clear, then we now go to the next step, which is. The advanced strategy. Okay. So given a capital of about 35,000 and this is without using any leverage. So Angel does give some intraday leverage, but for the moment, we are not using it. As long as you have 35,000 in capital, you can buy and sell 15 quantities of Reliance, and the net quantity at any given time will not go over 15 either 15 long or 15 short. That's the point I'm making, right? Again, I stress you have to set your cost structure to what it is and you have to set your margin to what you feel is the best option. So these are variables or parameters that need to be tweaked and basically optimized. Okay, So that is clear.

Let us go to the next part of the lecture. Now we go to the advanced strategy. Right. So advanced strategy again, very similar to the basic strategy. We have a stock Reliance. The last traded price is2346. The token is 2885 on NSE. Now we are saying that the total trade quantity is 6 and the trade margin is 1.9. Again, we will come to how to calculate this in the next step of this video. But first, let us explain what the video is, what the basic strategy steps are here. We also have something called an order multiplier of 3, which means instead of sending one order, we are sending three orders. And these are volume weighted orders. Okay. And a margin multiplier of 0.5. Now, I'll explain what that is in just a moment. Okay. Keep in mind, the trade margin is 1.9 and the trade quantity is 6. The total trade quantity is 6, not the single trade quantity, but the total trade quantity. Order multiplier is 3 and the margin multiplier is 0.5. Okay, so what exactly does all that mean? Let's look at the trade strategy.

Step one, send orders 1A and 1B simultaneously. Repeat every 45 seconds. So what is order 1A? A, We are sending three buy orders because the order multiplier is 3. That means instead of the basic strategy where we send 1 order, here we are sending 3 buy orders. And how are they divided? That is basically the order is the buy order is one quantity at 2344.1, quantity two at 2343.15 and quantity three at 2342.2. Now what does all those actually mean? So look at the quantities first. We have three buy orders, one quantity, two quantity and three quantity. That means one plus two plus three total buy is 6. Right? That is how we get to this total trade quantity.

So we have 6 is the total quantity that we intend to buy and sell, but we are splitting it up into three orders. That is why we say the order multiplier is 3 and they are volume weighted. So the closer you are to the last weighted price, you are trading less quantity. The further away from the last traded price you are willing to trade more quantity, right? So we have volume weighting in an averaging sort of pyramid, right? So one quantity, at 2344.1, two quantities at 2344 .15. three quantities at 2342.2. So the quantity is divided one, two, three. That is from the higher closest to the last traded price is one quantity, little further away from the last traded price is two quantity or higher quantity and further away from the last traded price is even more quantity. One, two, three are the quantities we are trading. Now, how did we get to these prices? So remember that the last traded price was 2346. That minus the trade margin is the first price here. Then 2346 - 1.5 times the trade margin. That gives me 2343.15. And then the third one is 2346 is my last traded price. Margin is 1.9, but now we have two steps away. So two times the margin, right? So the first one is that the trade margin, this is at 1.5 times the margin. This is at 2 times the margin. You see that? So we are increasing the quantity in one quantity, two quantity, three quantity and the price is set as margin times.

So the trade margin or the margin multiplier is moving in 0.5 or increments of 0.5. So this is last traded price -1.9. That is the first price. That is 2344.1. Then it is actually the last traded price 2346 -1.9 -0.95, which is where we get to 2343.15. And then the third order is 2342.2. That is actually 1.9 times two. So those three orders will go and simultaneously the sell orders. Three sell orders will also go. One quantity, two quantity , three quantity. For last traded price plus trade margin, last traded price plus trade margin times 1.5, last traded price plus trade margin times two. Right. So it is very symmetric. The entire algorithm is symmetric on the buy side and the sell side.

So both of these order sets go. So here, order 1A and order 1B are actually a set of orders of 3 orders each. Right? So again, you could make this multiplier of five orders. You know, you can tweak these values as you wish. Order margin multiplier may be 0.25, not increments of half your margin. But you could, you know, change it in various ways and tweak it in different ways.

So come to step two. If the order is 1A and 1B, both get executed. Return to step one. Right. And if order 1A is executed. So suppose the entire set of order one is executed, that is, you know, quantity one, quantity two and quantity three at these prices. Then cancel orders 1B and send the take profit order for net quantity at volume weighted average price plus margin. So what do we mean here? Because when all of these three get executed, we will have a volume weighted average price. The volume weighted average price we have to calculate, that is what the algo will do. There's a specific function for that. It might even be that only order one and order to get executed or only the order one get executed. So whatever it is, the volume weighted average price plus margin, that is where we will try to sell, take the sell order. And same way, if order 1B is executed, then cancel orders 1A and we send a take profit, buy order for the net quantity. So whatever. So if all three are executed, then the quantity will be six. If only the one and two are executed, then the quantity is three. If only the first order is executed, quantity is one and we send the take profit buy order at the net, quantity at the volume weighted average price minus the margin because we want to buy at below the volume weighted average price.

So whatever the average price of all the trades were, that minus margin we are willing to buy, we want to buy slightly cheaper than the average traded price for these orders 1A and 1B. Then we come to step 3.1. We repeat the cycle 45 seconds over. Now, if we have executed 1A, that means we have a long position. Then we send sell orders at the last traded price plus margin. Right?
So we check the new last traded price. Because order 1A is executed, that means we have a long position. So we send the sell orders at last traded price plus margin. Similarly, if order 1B was executed, then we have a short position. Therefore, we will now only send the buy orders at last traded price minus margin. Right. So last traded price minus margin is what we are doing, right?
It's a new last traded price. We check it and then we send the three orders to buy at the last traded price minus margin, minus margin times 1.5, minus margin times 2. Now for 3.1 and 3.2. There are two possible outcomes. So let's look at the first one. Step 3.1.1. Order 3.1 or order 2.1 is executed. Whichever one is executed, cancel the other one and we return to step one. Very simple.
In step 3.2.1, either order 3.2 or order 2.2 of them will get executed. Cancel the other one. Return to step one. That's it. This is the advanced strategy.

The key difference between the basic strategy and the advanced strategy is in the basic strategy, we are only sending one order every time we send a buy or sell. It is a single order. In the advanced strategy, that single order is split up into a volume weighted, uh, set of multiple orders. In this case, we have volume weighting it by half. Increments of 0.5 of the trade margin and we are multiplying the number of orders by three. So instead of the basic strategy where we send one order in the advanced strategy, we are sending three orders and each of those three orders, the first one will be at the trade margin. The second one with the higher quantity will be at trade margin times 1.5, and the third one with even higher order will be at trade margin times two. Right. So that is a volume weighted order that orders that we are sending in the advanced strategy right now. Both of these strategies have their own, you know, advantages and disadvantages. It might be better in some cases in different cases. So how we use it is the basic challenge. Okay. That is where a lot of the, you know, the success will depend on how exactly we use it. On the choice of these parameters, so on and so forth. Okay.

Now let's look at how exactly we calculate these total trade quantities 6 and the trade margin of 1.9 when we have multiple stocks, Right. So similar to what we did when we calculated a single stock, now we have, let's say, a multiple stock. Now I'm assuming that we have a capital of about 1.2 lakhs to trade, right? That is what we will need if we want to run this strategy at these values. So basically we have using the stock lookup code, we can find out the ticker symbols and the token numbers for NSE as well as the ticker symbols and the token numbers for BSE. Right. And we take the close price for each of these stocks and we put it into the Excel system, putting the close prices there. And that's what we need. Okay. Now, based on the close price, we are able to calculate the cost of the trade. So if we were to buy one share and sell one share, so a single stock of Axis Bank, if we buy for 856.95 and we sell it at 856 .95, basically assuming a cost structure of 0.02%, which is 0.0002, we know that our cost is 34 or 35 paisa we can say. Right. Similarly, HDFC Bank has a cost of 65 paisa, ICICI 35 paisa, Infosys 60 paisa, ITC 15 paisa or Reliance is maybe 94 to 95 Paisa, SBI is about 23 paisa, Tata motors about 17 to 18 paisa of cost. Now what is our trading margin? Well, as we did before, we just use double of our cost structure and we rounded to the nearest five paisa, right? So we use this function MROUND, take the cost basis, multiply it by two rounded to the nearest five paisa.

So Axis Bank, we are taking 0.7 as our margin, HDFC Bank 1.3, ICICI Bank 0.7, and so on (refer to the Excel file). Right. So that is the trade margin we are going to use. Now, assuming we have 1.2 lakhs in total capital that we can use to trade, then what is the quantity we can trade? Well, we have eight stocks here, right? And that means we can have a capital we can allocate 15,000 to each of these stocks. Right. So 15,000 to each of these stocks is what we can allocate. So basically, that's what we are doing here. You see here we have 15,000 divided by the closing price and rounded to zero digits. That is the total quantity that we can trade for Axis Bank. So Axis Bank, at the most, we can have a quantity of 18, HDFC at the most we can have a quantity of 9, ICICI at most 17, and so on (refer to the Excel file). Now for Reliance the quantity of 6 is maximum. We can have a position in Reliance. That is what we have used here. Total trade quantity of six. That six is divided into three different orders of one quantity, two quantity and three quantity. One plus two plus three is six. Right?
So you will have to specify what the trade value for each of these stocks are. So reliance can be one plus two plus three. That gives me 6. Axis bank can be three plus six plus nine. That gives me 18. HDFC Bank is a little tricky. You might have to do one plus three plus five, right? So you'll have to use a little bit of logic there to sort of tweak it and get them into if your order multipliers are three, that means you have to divide it into three. You know, the order multiplier. If it's three, then we have to divide it into three.

INFY For example, if the total quantity traded should not be more than ten, then we should divide it as, let's say, one, three and six, for example. That gives me a total quantity of ten. So there's some tweaking at these levels, which you'll have to specify here. That's what I'm trying to tell you.
But what I'm trying to show is when our traded quantity, maximum traded quantity that we allow for any stock, is these values (refer to the Excel file). When these are the total trade quantities, then our position value will be very close to 15,000. So in and around 15,000 is what each counter we are willing to have. That is the exposure we are willing to take. This should be equal. It has to be uniformly distributed. Otherwise we are waiting more to a particular stock which will skew our results. Right. So if you are trading, you know, if the traded quantity is 12 in Reliance, then actually we are giving double the weight to Reliance. So Reliance performs well, the strategy will do very well, but if it does badly, then it will do doubly badly. It will double the bad impact, you know.
So ideally, these should be uniformly distributed. The probability distribution of the weightage on these should be equal, or you should have a very objective reason why you are not keeping it equal and why it is weighted, you know, so that something which is data driven there. So in this file, it is very much equally distributed. So that's how we get to the margin and the trade quantities for multiple stocks when we use.

Now, keep in mind, we are assuming that we have about 1.2 lakhs in traded capital to run this strategy without leverage. So if Angel is giving you leverage with leverage, you should be able to buy and sell for 1.2 lakhs, you know, safely. I would suggest. Don't. I mean, that's up to you. Individual risk preference, whether you want to use leverage or not. Yeah. Okay. But essentially you should be able to have a buying power or traded margin should be 1.2 lakhs or more to run this. Otherwise, if some orders get rejected, then your strategy has gone wrong because the orders have to go.
You know, you have to get in and out.So if orders get rejected because of lack of margin, then you're running it wrong. You're not operating the strategy properly. Okay. So keep that in mind.

Now this is the API limits for Angel. So as you know, the important ones here is you can only place ten orders per second. Okay. So that's something to keep in mind. That's why we are basically keeping the stocks as eight or maximum ten. We don't want to hit that API limit. It might get, you know, there might be an error and then the algo will stop. Same way modify or cancel orders or get the last traded price. This is all only ten requests per second. So that is a key thing to keep in mind. Same way the get position that is we update our position or the net position, we are updating it by the limit of one per second. Therefore we run a multi thread operation to update the positions every two seconds once every two seconds. So we are within the limit of one per second. We are doing one per two seconds. So that's what the algo in the next videos you will see that multithreading to update the positions is being run once every two seconds and orders are limited to at most less than ten per second. Otherwise you will have an error. Okay. So that is fine. That much is. I hope that was clear.

Now let's talk a little bit about what is the logic in choosing the stock? So how exactly do we select a particular stock? Right. So let's go to the National Stock Exchange website. Right.
So what exactly is the logic of choosing the stock or how do we know which stocks might be suitable? Of course, all of you would have seen the NSE website. Basically, what we want for a

strategy like this is a stock which is being traded very well. There's good liquidity and there is no special, you know, information or events on that stock. Right. So we go to the stock NSE website. We look at most active securities. And we basically look at the most active securities and we go to view all. Okay, here we go. Go to the most active security. So we want stocks which are heavily traded where there is good liquidity already. That is what will allow us to get in and get out quickly because there are a lot of people buying and selling the stock. Right. So if something is not liquid, if something is not traded very well, this strategy is not suitable. This strategy works. Electronic market making works best when there is already a lot of liquidity in the market. Ironically. Right.

Although the purpose of providing electronic market making is to provide liquidity, it actually works better when there is already a good amount of liquidity anyway. We can decide which stocks to trade based on what suits us. So basically we look at the most active securities for the day and sort it by value, right? So which ones have been traded the most value? Right. So you see here, these are the best, you know, and choose the first eight of these. But when you choose the first eight, make sure that you don't include any where there is any earnings announcement on that day or the previous day. Right. Or in the last few days, there shouldn't be any earnings announcements or any sort of news like something happened and, you know, some major event happening.
Avoid any special events in that week, then avoid that stock as well. That's the basic idea that I'm suggesting. Right? So basically what I'm telling you is a stock which is a high traded quantity.
On a normal day. That is the stock we are interested in. Right. So because on an earnings day, the traded quantity will be very high for that particular stock. Okay. Earnings came out. A lot of people are trading it. Maybe some news came out okay. A lot of people are trading it. Avoid those with that kind of liquidity. We want stocks which are anyway they are the best you know traded counters in the market so ICICI, HDFC, Axis Bank, INFY, SBI, Reliance, Tata motors these are very commonly found as the best traded or the highest traded stocks on. On basically every day of the market.
They are the ones which are the most traded in the market. So that is the logic which we use.
Okay. So basically we will stop this lecture here. I hope that was clear. And in the next section we will look at each of the strategies.