Twitter Thread by Indian Quant ■■



Indian Quant ■■
@indian_quant



Nano Course On Python For Trading

==========

Module 4

In this post, I will attempt to teach you how to write a trading strategy in Equity Segment that runs on your PC and create a Telegram bot that sends you buy/sell signals with Stop Loss.

> Nano Course On Python For Trading
> ==========
> Module 1
>
> Python makes it very easy to analyze and visualize time series data when you\u2019re a beginner. It's easier when you don't have to install python on your PC (that's why it's a nano course, you'll learn python...
>
> — Indian Quant \U0001f1ee\U0001f1f3 (@indian_quant) December 13, 2021

Prerequisite: If you hadn't gone through the earlier modules, I strongly recommend you go through them all. Module 2: https://t.co/pciDOJXyVI

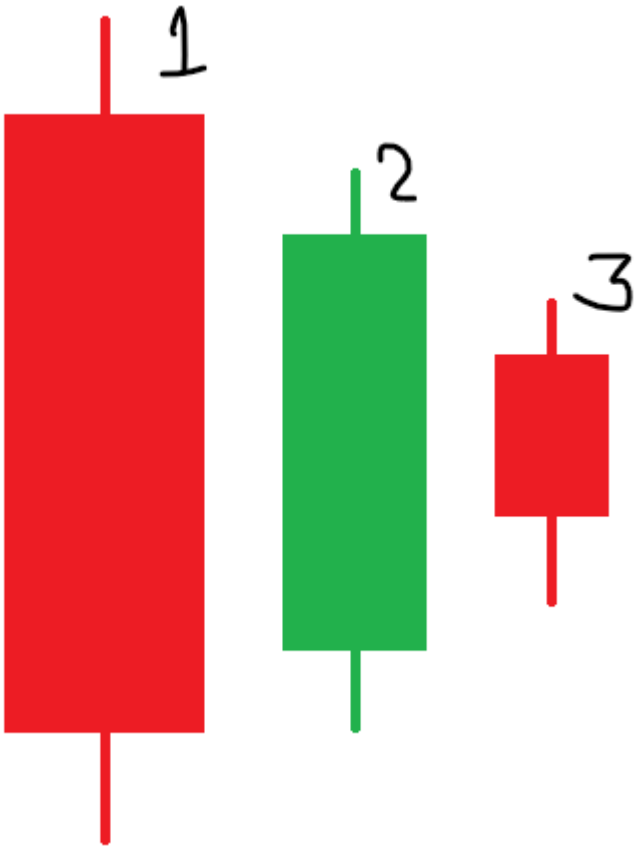
Note:
If you liked my content, you can donate, tip and support me on this link (any amount you prefer) https://t.co/ZgBLBmXK1p

+

> Nano Course On Python For Trading
> ==========
> Module 1
>
> Python makes it very easy to analyze and visualize time series data when you\u2019re a beginner. It's easier when you don't have to install python on your PC (that's why it's a nano course, you'll learn python...
>
> — Indian Quant \U0001f1ee\U0001f1f3 (@indian_quant) December 13, 2021

We are going to implement below strategy:

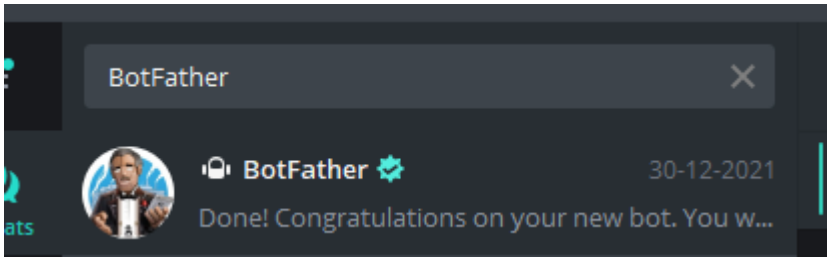
Rules: There should be three candles -> high of candle 1 > high of candle 2 > high of candle 3 and low of candle 1 < low of candle 2 < low of candle 3, where candle 3 is T-1 day, candle 2 is T-2 day and candle 1 is of T-3 Day, T = today.



If on today(day=T), the stock crosses yesterday high(candle 3), then send a buy signal to your telegram handle with candle 3 low as SL.

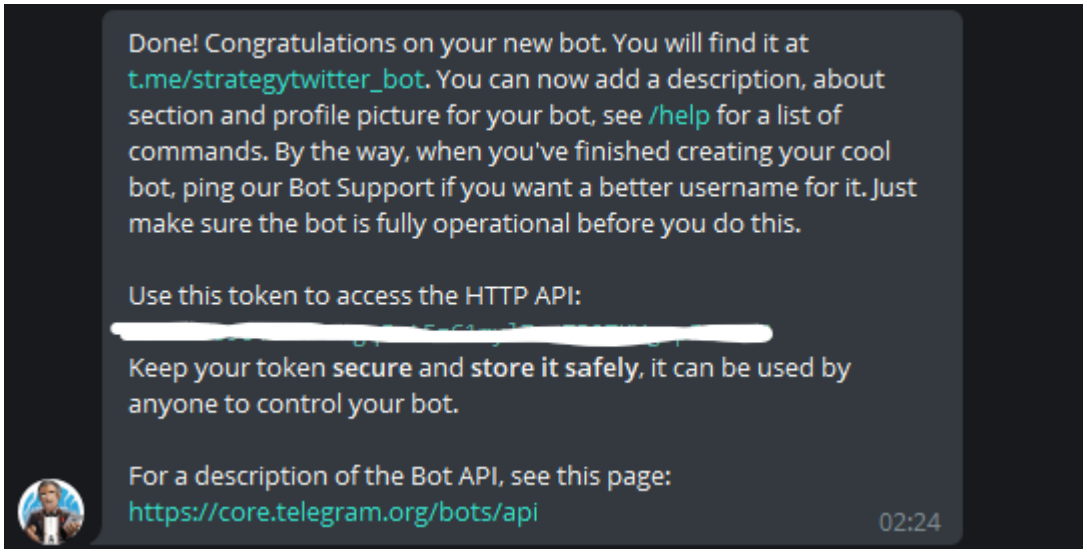
Before we get started with code, let's create a telegram bot using BotFather.

Step 1: Search BotFather in the telegram.
Step 2: type /newbot and then give the name to your bot. Refer to the second image as an example



Step 3: Save the token to access the Telegram API. Remember, don't share it with anyone. I have market white for my token. Let's call it .
Step 4: Send a Hi to your bot.

Step 5: Search for the telegram handle "@userinfobot" and save the id field (white tick)



This id field is unique for you and will be used by the code so that the bot can send you the message to your telegram ID.

Summary: you should have these two values saved,
1. token from the bot father for your bot
2. user "id" field when you interact with the userinfobot

So, this is about the theory. Let's get started with the code.

First, we import certain python libraries. All these libraries have been used before.

```
import os
import pandas as pd
import numpy as np
from nsepy import get_history
from datetime import date
from datetime import timedelta
import requests
```

Then we define the function to get stock data and name it getData(). We use the library NsePy used in Module 2. The function is self-explanatory as it fetches the last 10-day OHLC data for any stock.
In this strategy, we will scan among the list of Nifty 500 stocks.

```
def getData(ticker):
    # get latest last 10 day daily timeframe O-H-L-C data
    s =  date.today()-timedelta(days=10)
    e =  date.today()
    data = get_history(symbol=ticker, start=s, end=e)
    data100 = pd.DataFrame(data[-9:])
    return data100
```

Then we define the function telegram_bot_sendtext which takes the message and sends the message to your telegram. There are 2 variables:
tg_api_token and tg_chat_id. I have already explained above what they are(since confidential info so hidden in the image)

```
#use the one given by the BotFather
tg_api_token = '[redacted]'

#it's not 123456789, use the id you got from @userinfobot
tg_chat_id = '[redacted]'

def telegram_bot_sendtext(bot_message):

    send_text = 'https://api.telegram.org/bot' + tg_api_token + '/sendMessage?chat_id='
    + tg_chat_id + '&parse_mode=Markdown&text=' + bot_message

    response = requests.get(send_text)
```

Then we define the function StrategyOne, which takes OHLC data and ticker name as input and sends the message to your telegram id.

Rules: There should be three candles -> high of candle 1 > high of candle 2 > high of candle 3 and low of candle 1 < low of candle 2 < low of

+

candle 3, where candle 3 is T-1 day, candle 2 is T-2 day and candle 1 is of T-3 Day, T = today. If on today(day=T), the stock crosses yesterday high(candle 3), then send a buy signal to your telegram handle with candle 3 low as SL.

```
def StrategyOne(ohlc_1day,ticker):

    try:
        if ohlc_1day["High"][-3]>ohlc_1day["High"][-2] and ohlc_1day["High"][-2]>ohlc_1day["High"][-1]:
            if ohlc_1day["Low"][-3]<ohlc_1day["Low"][-2] and ohlc_1day["Low"][-2]<ohlc_1day["Low"][-1]:

                message = "Strategy 1 Buy:" + ticker + " at price : " + str(ohlc_1day['High'][-1]) + " with SL: "+ str(ohlc_
                print(message)
                telegram_bot_sendtext(message)


    except:
        pass
```

Finally, we iterate through the list of nifty 500 stocks, and in the for loop, we fetch the data for this ticker and pass it to the StrategyOne function.

When you run all the cells, you will get the message on Telegram as seen in the Image with the buy price and SL price.

```
nifty500list = pd.read_csv("nifty500list.csv")
tickerNifty500 = []

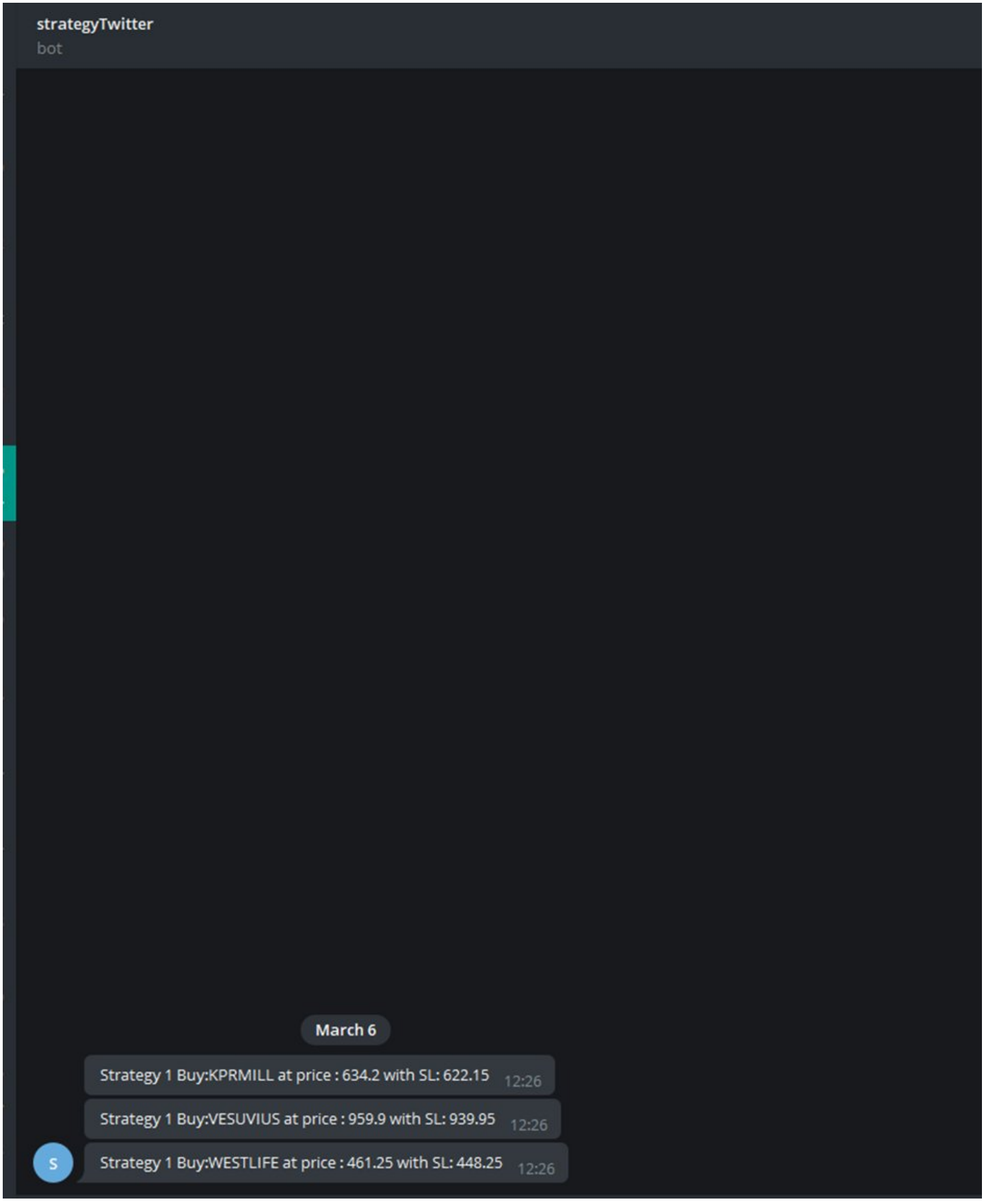
for i in range(500):
    tickerNifty500.append(nifty500list["Symbol"][i])


for ticker in tickerNifty500:
    try:
        #fetch ohlc data for current stock
        ohlcData = getData(ticker)

        #pass it to the strategyOne function to find if it is a buy signal?
        StrategyOne(ohlcData,ticker)

    except Exception as e: print(e)
```

Messages as seen on your telegram.



This is about writing code to write a trading strategy and using telegram to receive buy sell signals. In Module 5, we will understand how to schedule it on PC using a task scheduler, and one step ahead, we will use AWS to automate it to receive these signals daily ■.

External Links
Google Colab: https://t.co/ImmqZAMVg4

Donation Link: https://t.co/ZgBLBmXK1p