Ravi Day Open Strategy:

Indicators: Tradingview Open Close Daily Line (Optional)

Time Frame: 1/3/5/15Min

Instruments:

NSE: BankNifty / Nifty / MidCPNifty/Fin Nifty

BSE: SENSEX / BANKEX

Trade will be taken in options based on Spot Index Strikes

Trades should be MIS / NRML

No of Lots to be defined Trade setting file

**Trade will start @ 09:15, Day Exit @ 15:28 ( Day Exit)**

**Trades to be taken in 2 Lots, 100 Points TP1 and trail SL to entry for 2nd Lot and 2nd Lot TP2 at 200 points from Entry**

BUY CE:

Entry: On Candle Close

1. When Candle Crosses and Closes above Day Open Value in Spot
2. 1st Lot Exit @ 100 Points ( NOT on Candle Close) 🡪 TP1
3. 2nd Lot will Trail to Cost and will Exit @ 200 points from Entry 🡪 TP2

SL: when Candle closes below Day Open Value in Spot

BUY PE:

Entry: On Candle Close

1. When Candle Crosses and Closes below Day Open Value in Spot
2. 1st Lot Exit @ 100 Points ( NOT on Candle Close) 🡪 TP1
3. 2nd Lot will Trail to Cost and will Exit @ 200 points from Entry 🡪 TP2

SL: when Candle closes above Day Open Value in Spot

Trade Setting File:

Strike Price for every instrument to be chosen based in Delta Option Greeks

Current Week or Next week should be defined in Trade Settings. For Every Instrument, Check the day of the week, If Expiry is on Present Day, Chose Present / next week strike based on Delta Option Greeks ( > 50,>60>,>70,>80,>90 etc to be chosen from Trade Setting File)

Option to chose 1/3/5/15Min in Trade Settings

Lots to be defined in trade setting file

Strike Price to be chosen based on Delta

Strike Price Difference:

BANKNIFTY: 100

NIFTY: 50

FINNIFTY: 50

MIDCPNIFTY: 25

SENSEX: 100

BANKEX: 100

For MCX, Trades should be taken on Commodity Futures

Trade will be taken in Commodity Futures

1/3/5/15Min TF

Trades should be MIS ( Intraday)

Trade to be taken in single Lot ..Qty / Lotsize will be defined in Trade setting file

**Trade will start @ 15:30, Day Exit @ 23:28 ( Day Exit)**

BUY

Entry: On Candle Close

1. When Candle Crosses and Closes above Day Open Value in Spot
2. TP1 exit @ 40 Points ( NOT on Candle Close) 🡪 TP1

SL: when Candle closes below Day Open Value in Spot

SELL

Entry: On Candle Close

1. When Candle Crosses and Closes below Day Open Value in Spot
2. TP1 exit @ 40 Points ( NOT on Candle Close) 🡪 TP1

SL: when Candle closes above Day Open Value in Spot

Trade Setting File:

Qty / Lotsize will be defined in Trade setting file

Option to chose Current Month or Next Month Expiry

Option to chose 1/3/5/15Min in Trade Settings

Instruments: all MCX Instruments

Pine Code from Tradingview:

study("Open/Close Daily Line", overlay=true)

odl = input(true, title="Open Daily Line")

dopen = security(tickerid, 'D', open)

dcolor = close < dopen ? red : green

plot(odl and dopen ? dopen :na , title="Daily\_Open",style=circles, color=dcolor, linewidth=3)

cdl = input(true, title="Previous Closing Daily Line")

dclose = security(tickerid, 'D', close[1])

dcolor2 = close < dclose ? red : green

plot(cdl and dclose ? dclose :na , title="Daily\_Close",style=circles, color=dcolor2, linewidth=3)

Python Code:

import pandas as pd

import matplotlib.pyplot as plt

import yfinance as yf

# Fetch historical data for a specific ticker

ticker = 'AAPL' # Change to your desired ticker

data = yf.download(ticker, period='1mo', interval='1d')

# Calculate the daily open and previous close

data['Daily\_Open'] = data['Open']

data['Previous\_Close'] = data['Close'].shift(1)

# Set colors based on the current close price compared to open and previous close

data['Open\_Color'] = data['Close'].apply(lambda x: 'red' if x < data['Daily\_Open'] else 'green')

data['Close\_Color'] = data['Close'].apply(lambda x: 'red' if x < data['Previous\_Close'] else 'green')

# Plotting

plt.figure(figsize=(12, 6))

plt.plot(data.index, data['Daily\_Open'], marker='o', linestyle='', label='Daily Open', color='gray')

plt.plot(data.index, data['Previous\_Close'], marker='o', linestyle='', label='Previous Close', color='gray')

# Add colored markers

for idx, row in data.iterrows():

plt.scatter(idx, row['Daily\_Open'], color=row['Open\_Color'], s=100)

plt.scatter(idx, row['Previous\_Close'], color=row['Close\_Color'], s=100)

# Formatting the plot

plt.title(f'{ticker} Daily Open and Previous Close')

plt.xlabel('Date')

plt.ylabel('Price')

plt.xticks(rotation=45)

plt.legend()

plt.grid()

plt.tight\_layout()

plt.show()