Use of tradingview

TradingView

Pros

- Good dashboard with lots of indicators already available
- Support custom indicators (pinescript)
- Good community support
- Support backtesting (pinescript)
- Support alerting with email and its own apps

Cons

- Difficult to perform parameter optimization in backtesting
- No native support of telegram alerting
- No native support of auto order execution with brokers
- No data analytics support
- Need license fee for more comprehensive features

Python

Pros

- Flexible automation and data processing tasks available (e.g. stock/currency pair screening)
- Very flexible in performing backtesting with parameter optimization
- Good data analytics support
- Good support with brokers such as IB and FUTU
- Open source

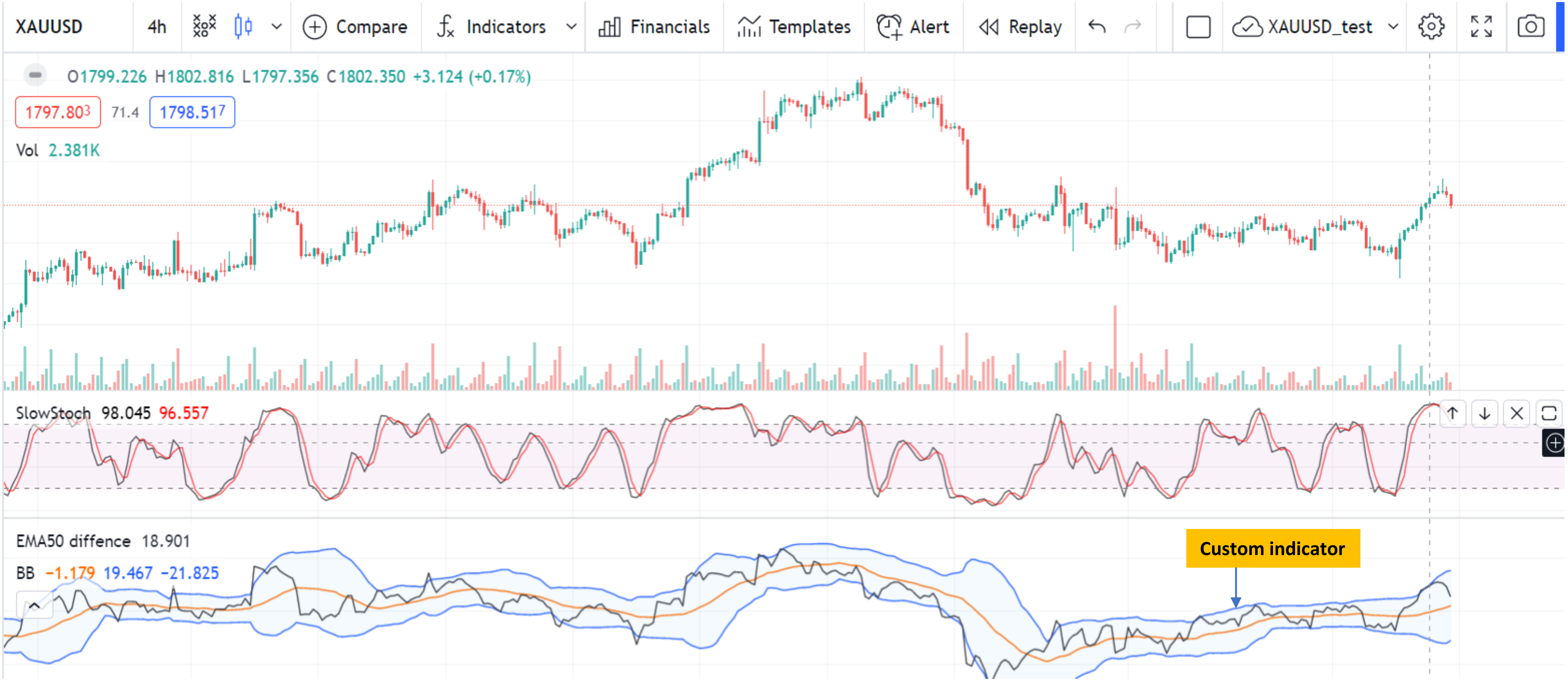
Cons

- Relatively challenging learning path

Tradingview-Main features

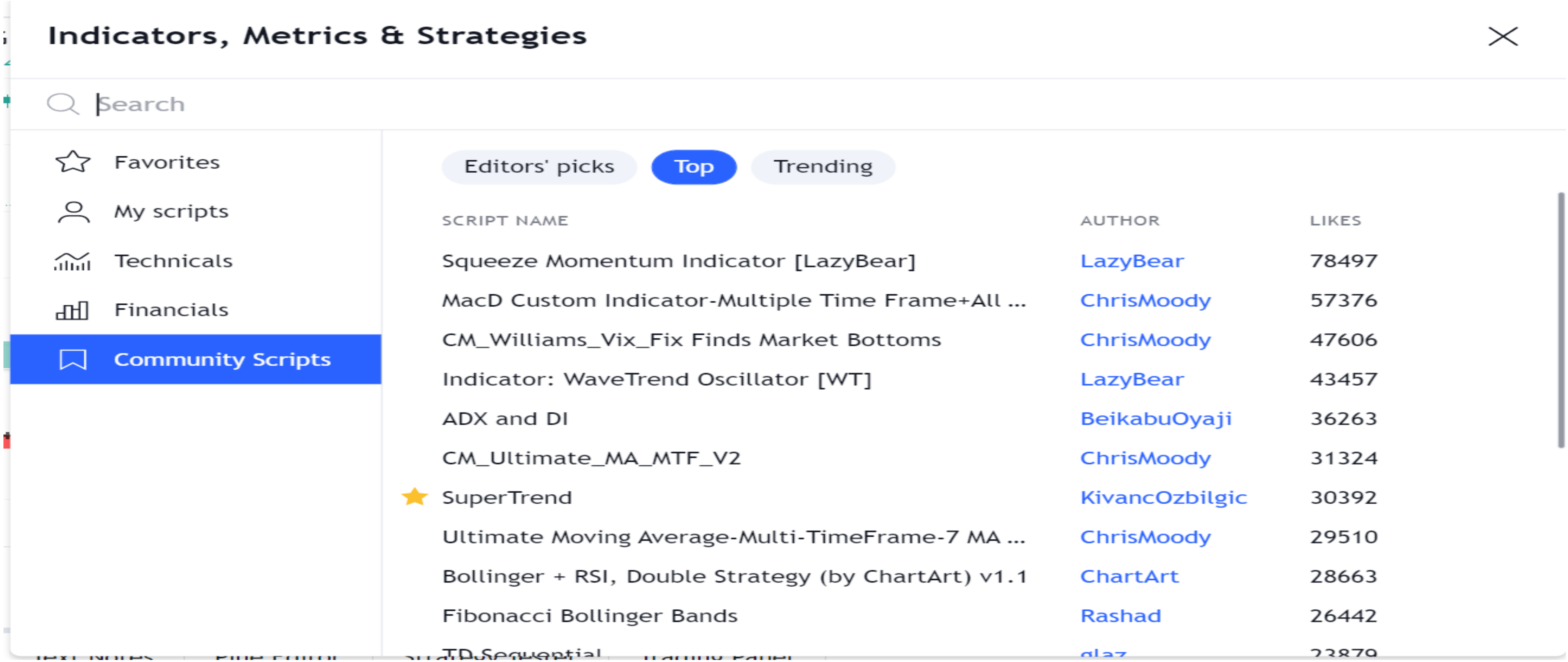
- Indicators (Built in or custom)
- Alerting
- Multiple sub-charts
- Simple back testing
- Sharing charts with others
- Replay

Dashboard with tradingview



Custom indicator

Basic charting and standard indicators



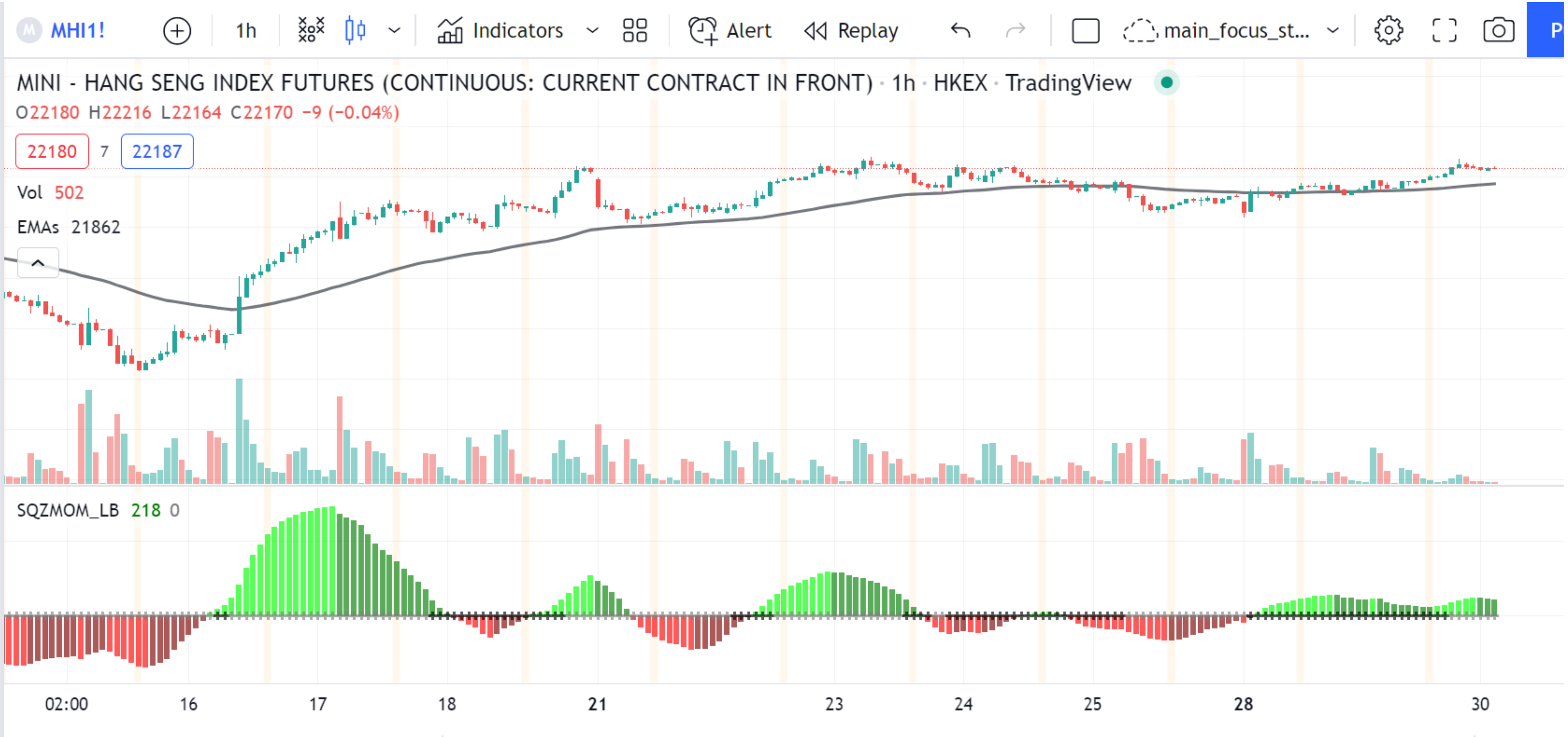
Indicators, Metrics & Strategies

Search

SCRIPT NAME	AUTHOR	LIKES
Squeeze Momentum Indicator [LazyBear]	LazyBear	78497
MacD Custom Indicator-Multiple Time Frame+All ...	ChrisMoody	57376
CM_Williams_Vix_Fix Finds Market Bottoms	ChrisMoody	47606
Indicator: WaveTrend Oscillator [WT]	LazyBear	43457
ADX and DI	BeikabuOyaji	36263
CM_Ultimate_MA_MTF_V2	ChrisMoody	31324
★ SuperTrend	KivancOzbilgic	30392
Ultimate Moving Average-Multi-TimeFrame-7 MA ...	ChrisMoody	29510
Bollinger + RSI, Double Strategy (by ChartArt) v1.1	ChartArt	28663
Fibonacci Bollinger Bands	Rashad	26442
TD Sequential	glaz	23879

Example of standard indicators

Question: Any hypothesis of trading strategy from here ?


MHI1!
1h
Indicators
Alert
Replay
main_focus_st...
P
MINI - HANG SENG INDEX FUTURES (CONTINUOUS: CURRENT CONTRACT IN FRONT) · 1h · HKEX · TradingView
O22180 H22216 L22164 C22170 -9 (-0.04%)
22180
7
22187
Vol 502
EMAs 21862
SQZMOM_LB 218 0
02:00
16
17
18
21
23
24
25
28
30

Pinescript

- For version 4 of pinescript
- Custom indicator (**study()**)
  - Add custom chart
  - Define alerts
- Strategy (for backtesting and forward testing)

Create custom indicator & Alerts

Pinescript

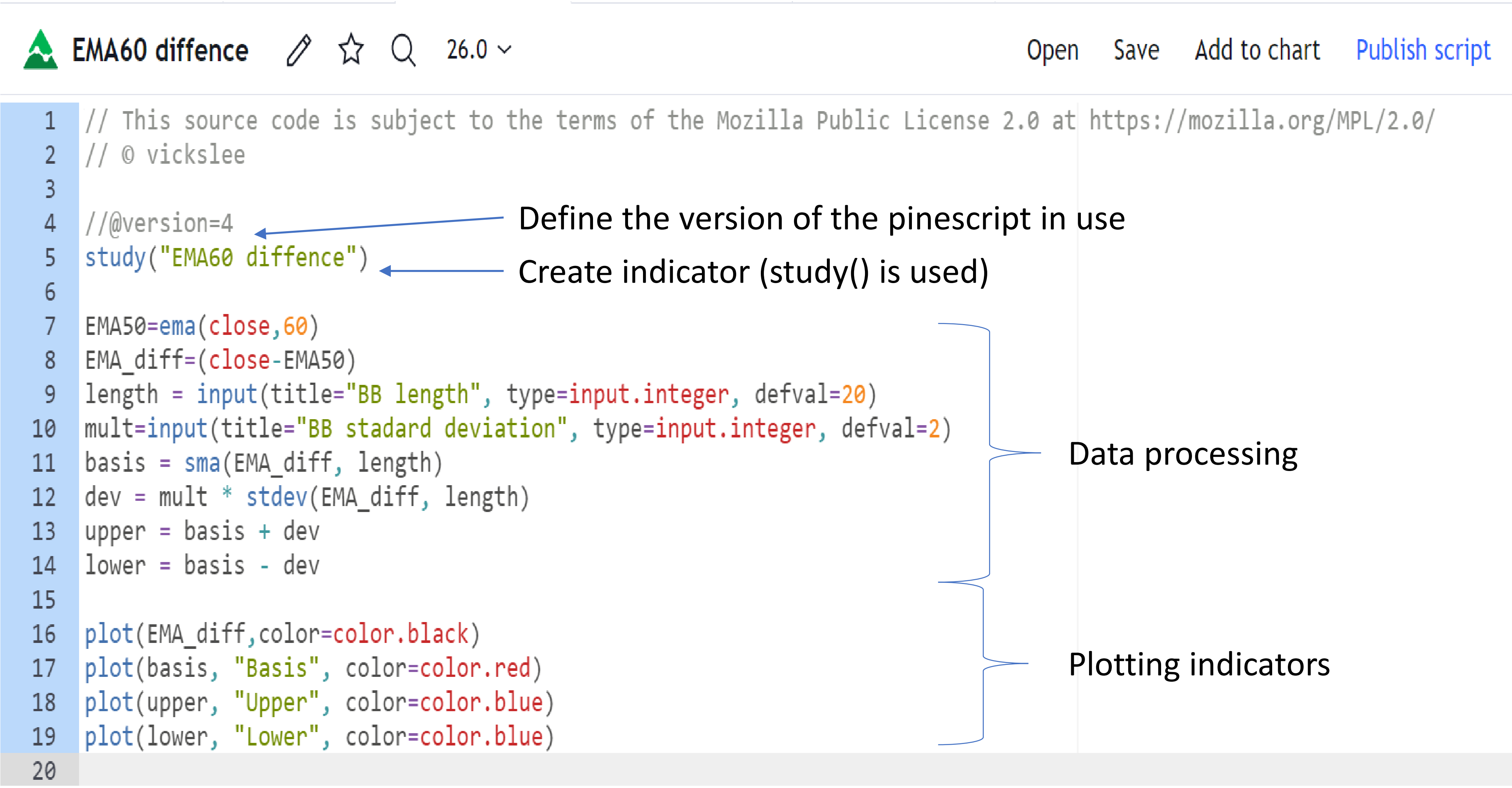
Program logic-Create custom indicator

1. Define version of pinescript used (e.g. version 4)
2. Define the inputs of the indicators
3. Data processing (e.g. calculate EMA10 and EMA20 and also crossover and crossunder conditions)
4. Define the custom indicator
5. Plot the custom indicator
    - plot()
    - plotshape()

Example: custom indicator with pinescript

```
// This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
//@version=4
study("EMA50 diffence")
EMA50=ema(close,50)
EMA_diff=(close-EMA50)
plot(EMA_diff,color=color.black)
```

Sample-Custom indicator



```
// This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
// © vickslee

//@version=4
study("EMA60 diffence")

EMA50=ema(close,60)
EMA_diff=(close-EMA50)
length = input(title="BB length", type=input.integer, defval=20)
mult=input(title="BB stadard deviation", type=input.integer, defval=2)
basis = sma(EMA_diff, length)
dev = mult * stdev(EMA_diff, length)
upper = basis + dev
lower = basis - dev

plot(EMA_diff,color=color.black)
plot(basis, "Basis", color=color.red)
plot(upper, "Upper", color=color.blue)
plot(lower, "Lower", color=color.blue)

```

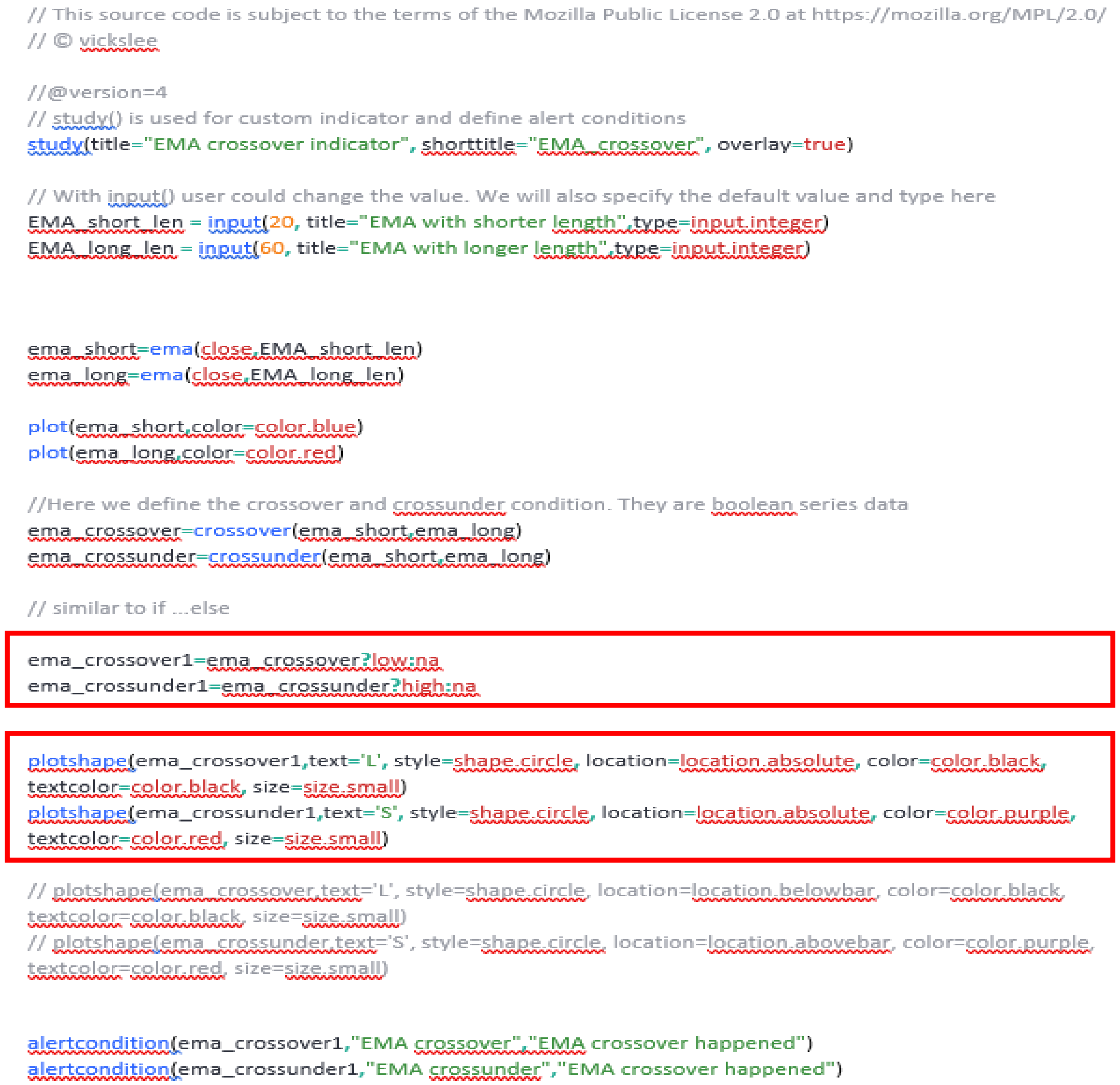
Define the version of the pinescript in use

Create indicator (study() is used)

Data processing

Plotting indicators

Plotshape



```
// This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
// © vickslee

//@version=4
// study() is used for custom indicator and define alert conditions
study(title="EMA crossover indicator", shorttitle="EMA_crossover", overlay=true)

// With input() user could change the value. We will also specify the default value and type here
EMA_short_len = input(20, title="EMA with shorter length",type=input.integer)
EMA_long_len = input(60, title="EMA with longer length",type=input.integer)



ema_short=ema(close,EMA_short_len)
ema_long=ema(close,EMA_long_len)

plot(ema_short,color=color.blue)
plot(ema_long,color=color.red)

//Here we define the crossover and crossunder condition. They are boolean series data
ema_crossover=crossover(ema_short,ema_long)
ema_crossunder=crossunder(ema_short,ema_long)

// similar to if ...else

ema_crossover1=ema_crossover?low:na
ema_crossunder1=ema_crossunder?high:na

plotshape(ema_crossover1,text='L', style=shape.circle, location=location.absolute, color=color.black,
textcolor=color.black, size=size.small)
plotshape(ema_crossunder1,text='S', style=shape.circle, location=location.absolute, color=color.purple,
textcolor=color.red, size=size.small)

// plotshape(ema_crossover,text='L', style=shape.circle, location=location.belowbar, color=color.black,
textcolor=color.black, size=size.small)
// plotshape(ema_crossunder,text='S', style=shape.circle, location=location.abovebar, color=color.purple,
textcolor=color.red, size=size.small)


alertcondition(ema_crossover1,"EMA crossover","EMA crossover happened")
alertcondition(ema_crossunder1,"EMA crossunder","EMA crossover happened")
```

Indicate the points with circle when crossover or Crossunder happen

Another example (EMA in multiple timeframe)

- // This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
- // © vickslee

- //@version=4

- study("multiple_timeframe_study")
- close_D1=security(syminfo.tickerid, "D", close)
- ema_d1_10=ema(close_D1,10)
- ema_d1_20=ema(close_D1,20)
- ema_d1_60=ema(close_D1,60)

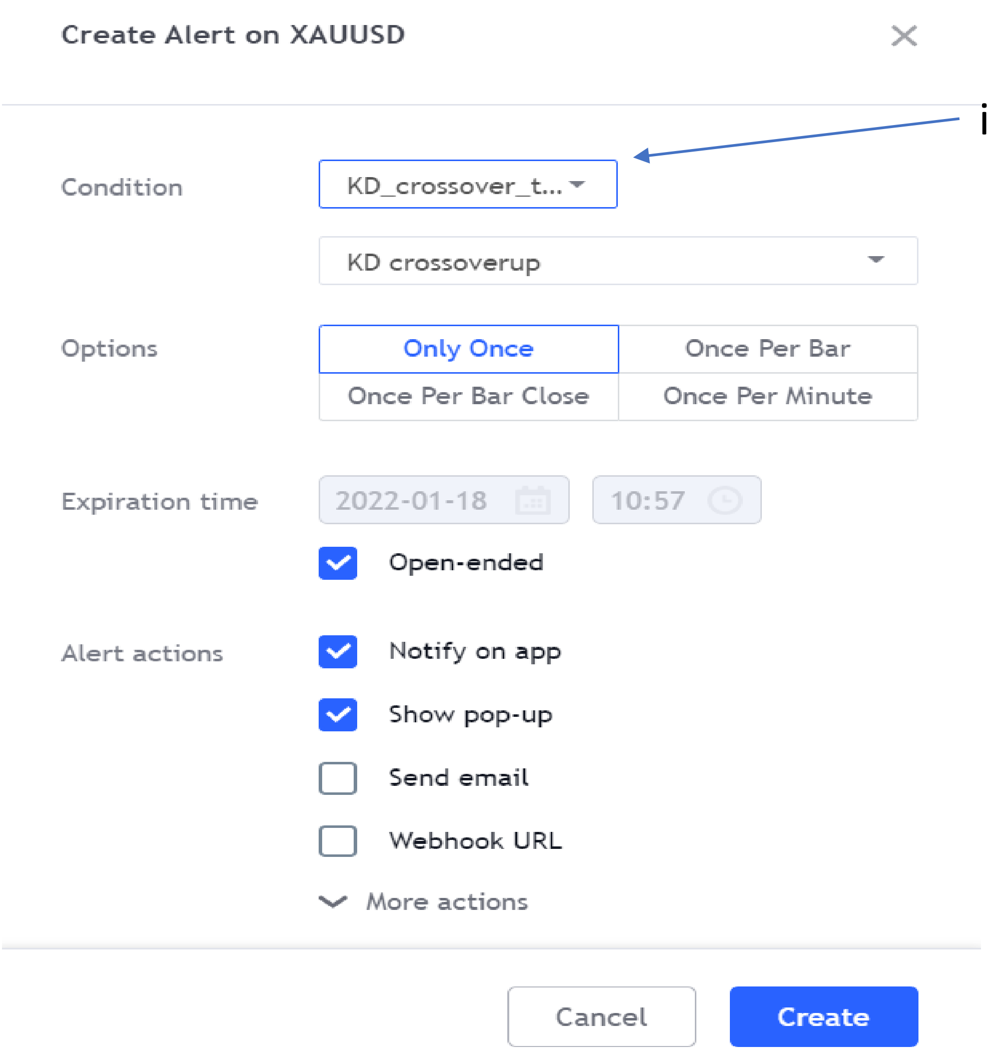
- plot(ema_d1_10,color=color.blue)
- plot(ema_d1_20,color=color.red)
- plot(ema_d1_60,color=color.black)

Define alert condition with pinescript

```
// This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
//@version=4
study("KD_crossover_test1")
length = input(14, minval=1)
OverBought = input(80)
OverSold = input(20)
smoothK = 3
smoothD = 3
k = sma(stoch(close, high, low, length), smoothK)
d = sma(k, smoothD)
co = crossover(k,d)
cu = crossunder(k,d)
alertcondition(co,"KD crossoverup","KD crossover happened")
plot(k,color=color.black)
plot(d,color=color.red)
```

Define the alert condition

Set up alert condition



Create Alert on XAUUSD

Condition: KD_crossover_t...

KD crossoverup

Options: [✓ Only Once] Once Per Bar Once Per Bar Close Once Per Minute

Expiration time: 2022-01-18 10:57

☑ Open-ended

Alert actions:

☑ Notify on app

☑ Show pop-up

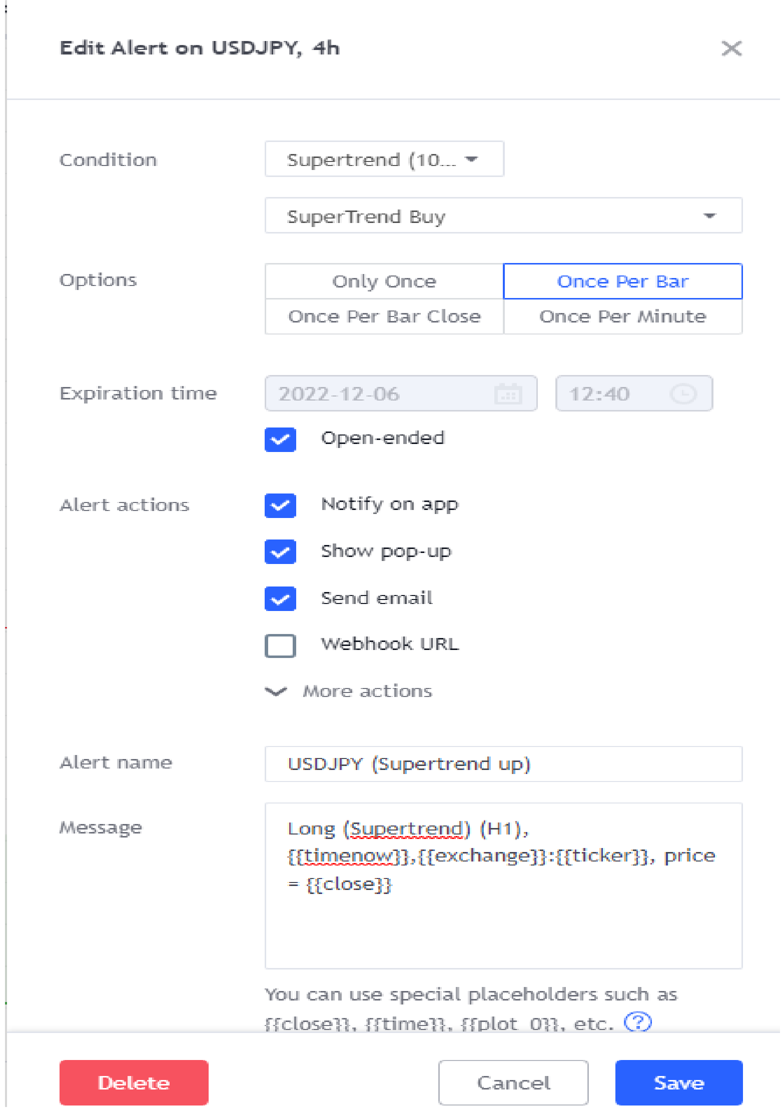
□ Send email

□ Webhook URL

More actions

indicator name

Example: Define alert condition with custom message



Edit Alert on USDJPY, 4h

Condition: Supertrend (10... / SuperTrend Buy

Options: Only Once [✓ Once Per Bar] Once Per Bar Close Once Per Minute

Expiration time: 2022-12-06 12:40

☑ Open-ended

Alert actions:

☑ Notify on app

☑ Show pop-up

☑ Send email

☐ Webhook URL

More actions

Alert name: USDJPY (Supertrend up)

Message:

```
Long (Supertrend) (H1),
{{timenow}},{{exchange}}:{{ticker}}, price
= {{close}}
```

You can use special placeholders such as {{close}}, {{time}}, {{plot_0}}, etc.

Backtesting

Pinescipt

Backtesting strategies

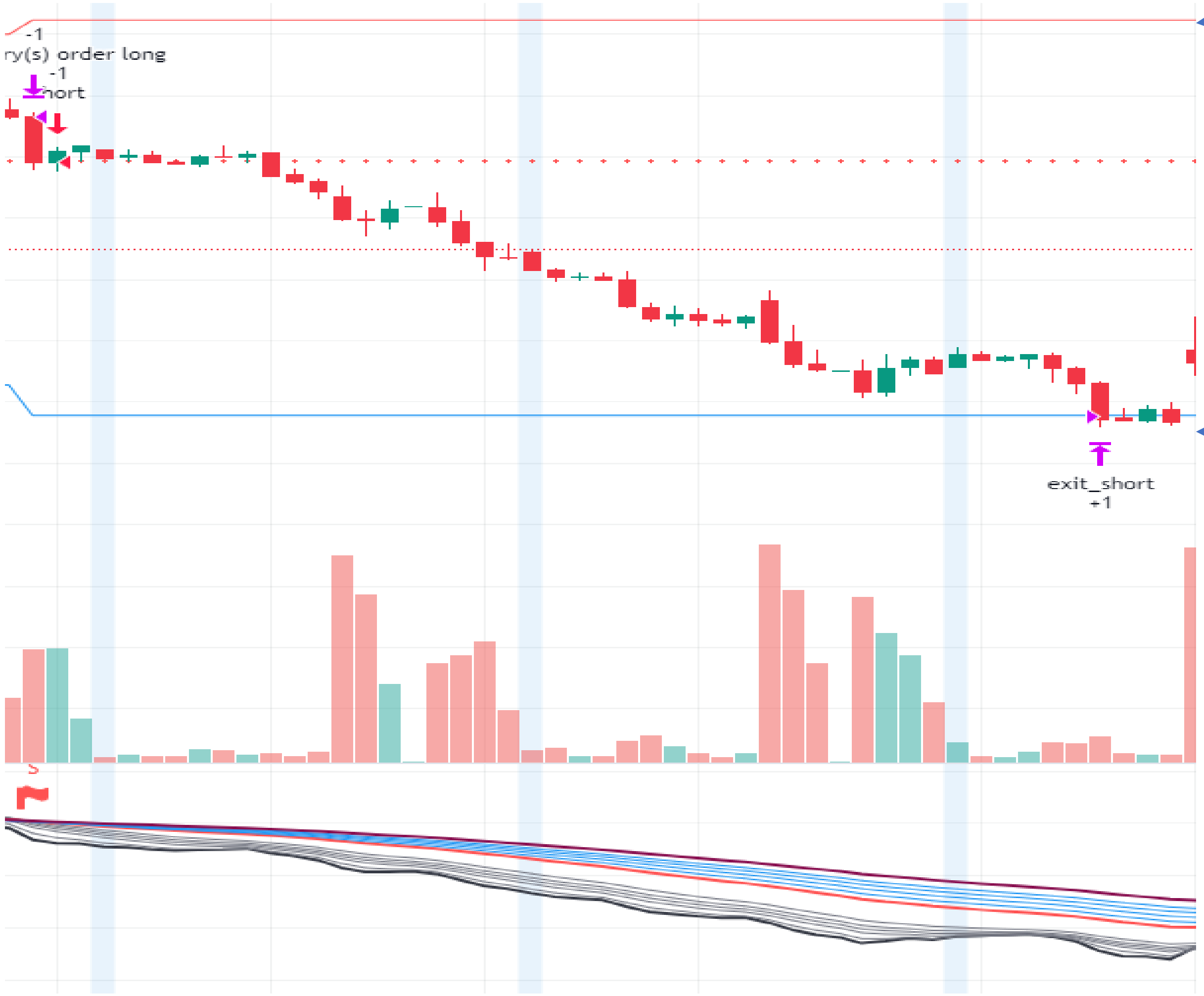
- Simple stop loss and profit target
- Trailing stop setup
- Backtesting time duration setup
- Time duration for position entry

Examples of backtesting strategies

- Simple target profit & stop loss strategy
- Target profit with trailing stop loss strategy

Simple target profit and stop loss strategy

Simple stoploss and target profit

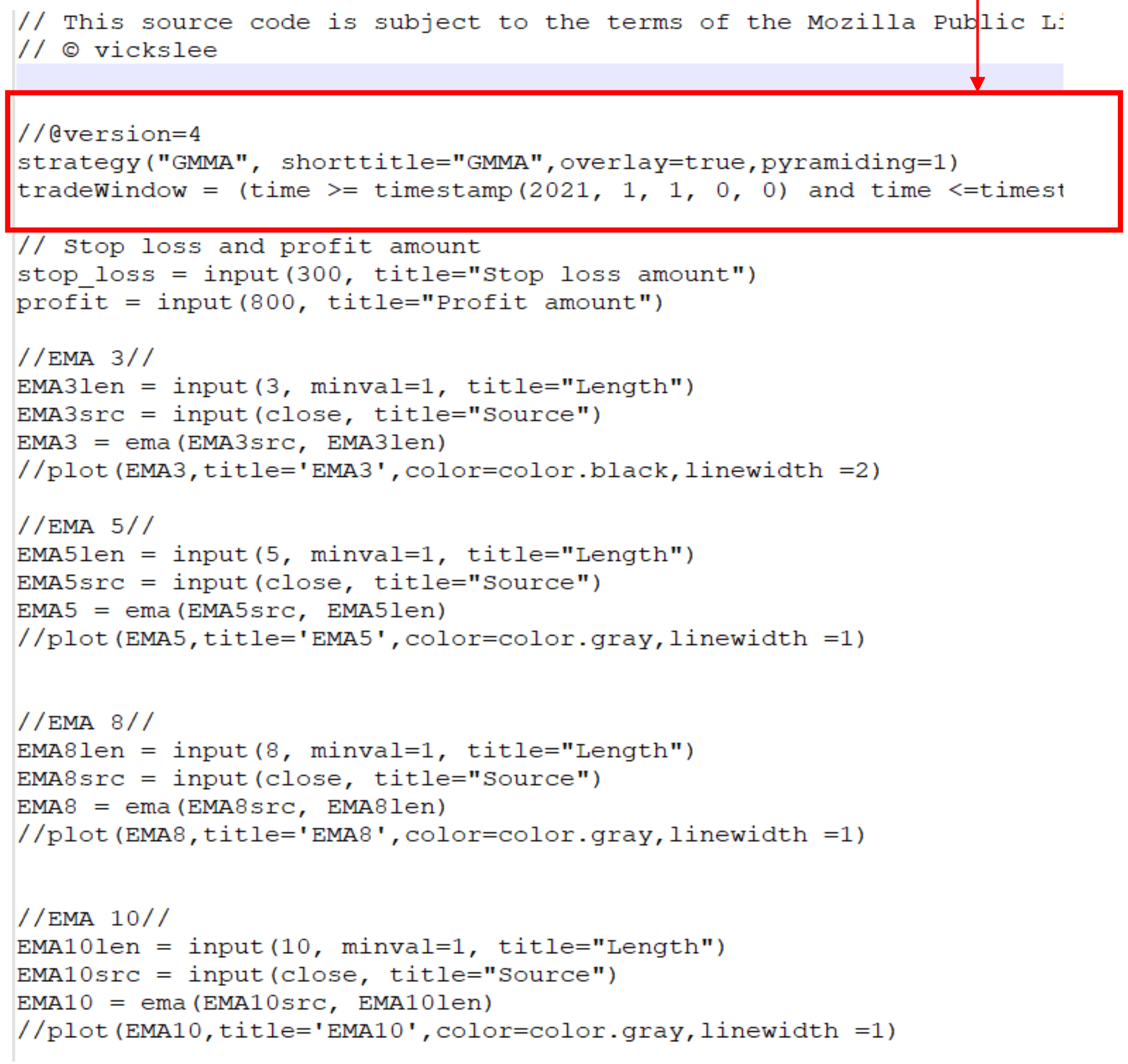


stoploss

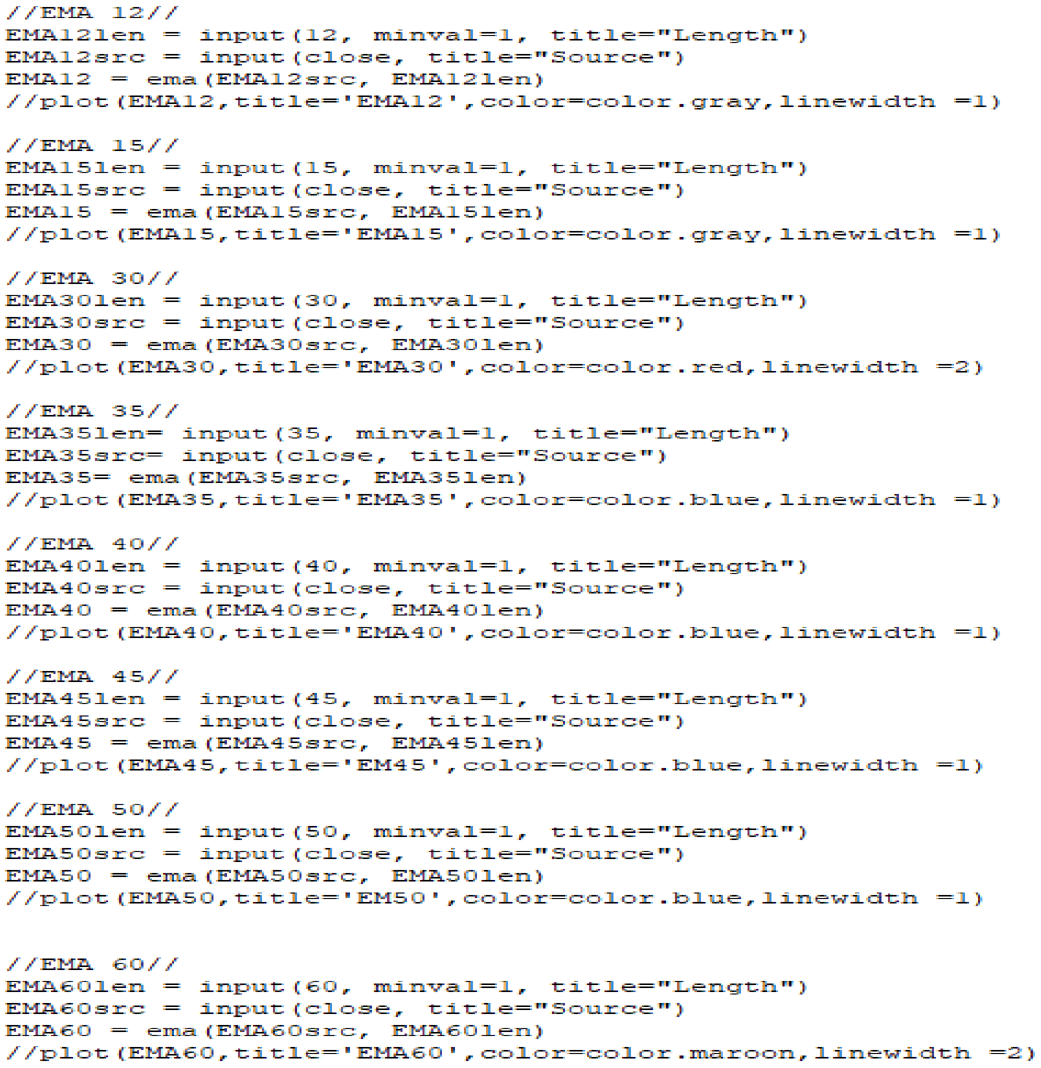
Target profit

Backtesting of trading strategy(Sample)-1/3

Define the time duration used by the trading strategy



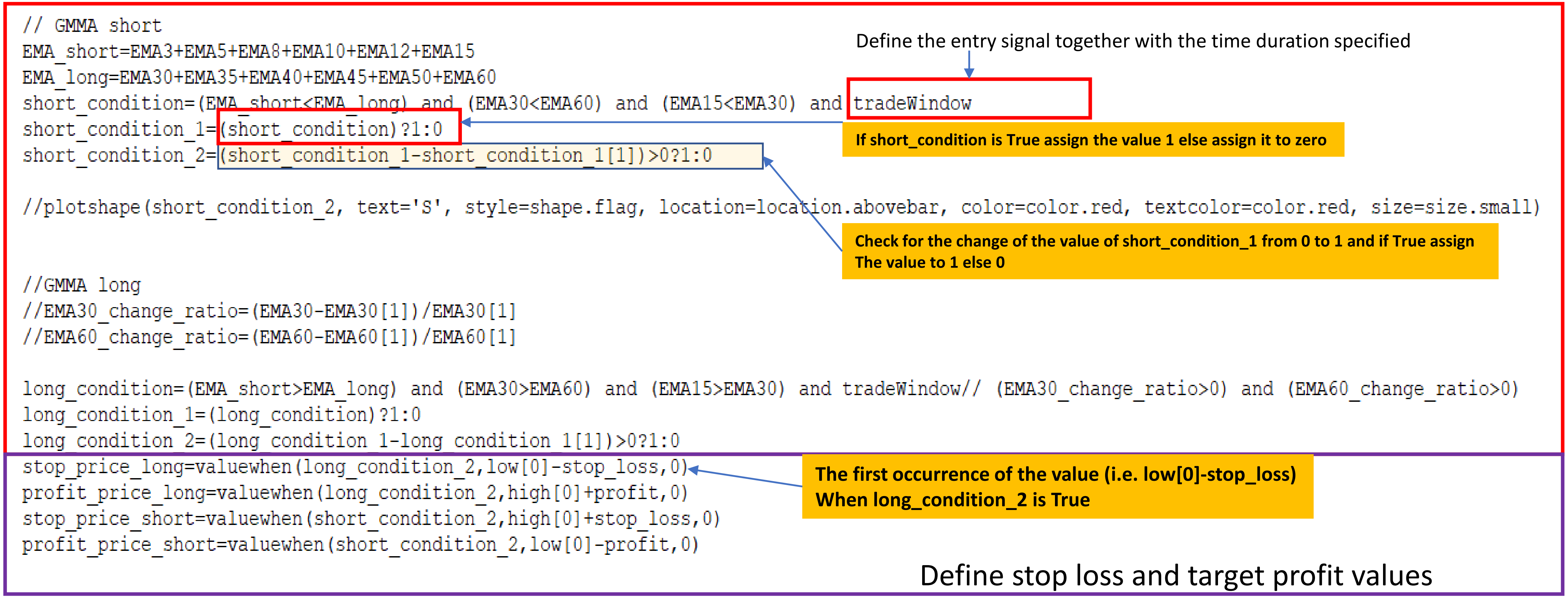
```
// This source code is subject to the terms of the Mozilla Public Li
// © vickslee

//@version=4
strategy("GMMA", shorttitle="GMMA",overlay=true,pyramiding=1)
tradeWindow = (time >= timestamp(2021, 1, 1, 0, 0) and time <=timest
// Stop loss and profit amount
stop_loss = input(300, title="Stop loss amount")
profit = input(800, title="Profit amount")

//EMA 3//
EMA3len = input(3, minval=1, title="Length")
EMA3src = input(close, title="Source")
EMA3 = ema(EMA3src, EMA3len)
//plot(EMA3,title='EMA3',color=color.black,linewidth =2)

//EMA 5//
EMA5len = input(5, minval=1, title="Length")
EMA5src = input(close, title="Source")
EMA5 = ema(EMA5src, EMA5len)
//plot(EMA5,title='EMA5',color=color.gray,linewidth =1)


//EMA 8//
EMA8len = input(8, minval=1, title="Length")
EMA8src = input(close, title="Source")
EMA8 = ema(EMA8src, EMA8len)
//plot(EMA8,title='EMA8',color=color.gray,linewidth =1)


//EMA 10//
EMA10len = input(10, minval=1, title="Length")
EMA10src = input(close, title="Source")
EMA10 = ema(EMA10src, EMA10len)
//plot(EMA10,title='EMA10',color=color.gray,linewidth =1)
```



```
//EMA 12//
EMA12len = input(12, minval=1, title="Length")
EMA12src = input(close, title="Source")
EMA12 = ema(EMA12src, EMA12len)
//plot(EMA12,title='EMA12',color=color.gray,linewidth =1)

//EMA 15//
EMA15len = input(15, minval=1, title="Length")
EMA15src = input(close, title="Source")
EMA15 = ema(EMA15src, EMA15len)
//plot(EMA15,title='EMA15',color=color.gray,linewidth =1)

//EMA 30//
EMA30len = input(30, minval=1, title="Length")
EMA30src = input(close, title="Source")
EMA30 = ema(EMA30src, EMA30len)
//plot(EMA30,title='EMA30',color=color.red,linewidth =2)

//EMA 35//
EMA35len= input(35, minval=1, title="Length")
EMA35src= input(close, title="Source")
EMA35= ema(EMA35src, EMA35len)
//plot(EMA35,title='EMA35',color=color.blue,linewidth =1)

//EMA 40//
EMA40len = input(40, minval=1, title="Length")
EMA40src = input(close, title="Source")
EMA40 = ema(EMA40src, EMA40len)
//plot(EMA40,title='EMA40',color=color.blue,linewidth =1)

//EMA 45//
EMA45len = input(45, minval=1, title="Length")
EMA45src = input(close, title="Source")
EMA45 = ema(EMA45src, EMA45len)
//plot(EMA45,title='EM45',color=color.blue,linewidth =1)

//EMA 50//
EMA50len = input(50, minval=1, title="Length")
EMA50src = input(close, title="Source")
EMA50 = ema(EMA50src, EMA50len)
//plot(EMA50,title='EM50',color=color.blue,linewidth =1)


//EMA 60//
EMA60len = input(60, minval=1, title="Length")
EMA60src = input(close, title="Source")
EMA60 = ema(EMA60src, EMA60len)
//plot(EMA60,title='EMA60',color=color.maroon,linewidth =2)
```

Backtesting of trading strategy(Sample)-2/3

Define enter conditions (Long and short)



```
// GMMA short
EMA_short=EMA3+EMA5+EMA8+EMA10+EMA12+EMA15
EMA_long=EMA30+EMA35+EMA40+EMA45+EMA50+EMA60
short_condition=(EMA_short<EMA_long) and (EMA30<EMA60) and (EMA15<EMA30) and tradeWindow
short_condition_1=(short_condition)?1:0
short_condition_2=(short_condition_1-short_condition_1[1])>0?1:0

//plotshape(short_condition_2, text='S', style=shape.flag, location=location.abovebar, color=color.red, textcolor=color.red, size=size.small)


//GMMA long
//EMA30_change_ratio=(EMA30-EMA30[1])/EMA30[1]
//EMA60_change_ratio=(EMA60-EMA60[1])/EMA60[1]

long_condition=(EMA_short>EMA_long) and (EMA30>EMA60) and (EMA15>EMA30) and tradeWindow// (EMA30_change_ratio>0) and (EMA60_change_ratio>0)
long_condition_1=(long_condition)?1:0
long_condition_2=(long_condition_1-long_condition_1[1])>0?1:0
stop_price_long=valuewhen(long_condition_2,low[0]-stop_loss,0)
profit_price_long=valuewhen(long_condition_2,high[0]+profit,0)
stop_price_short=valuewhen(short_condition_2,high[0]+stop_loss,0)
profit_price_short=valuewhen(short_condition_2,low[0]-profit,0)
```

Define the entry signal together with the time duration specified

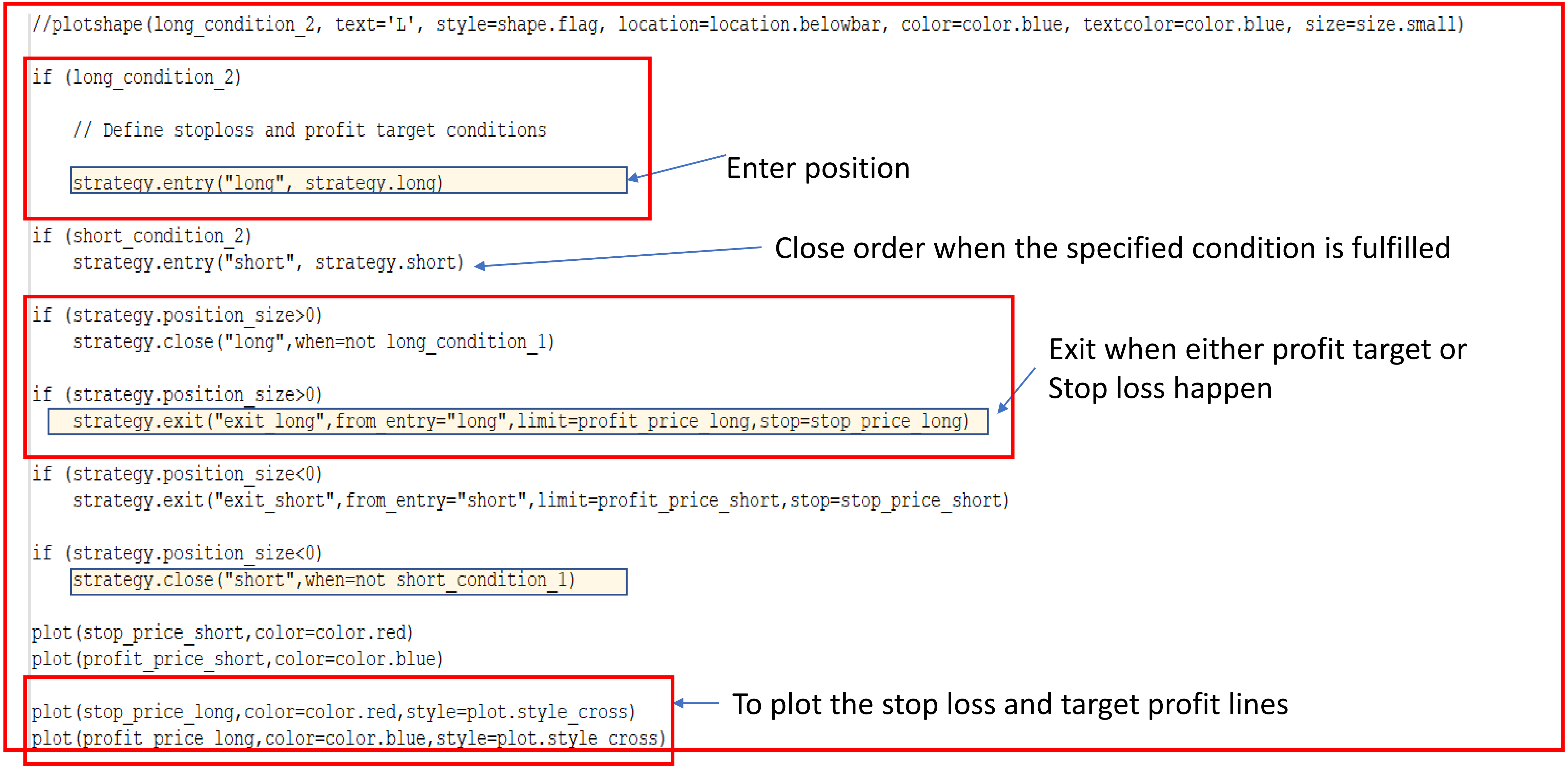
If short_condition is True assign the value 1 else assign it to zero

Check for the change of the value of short_condition_1 from 0 to 1 and if True assign The value to 1 else 0

The first occurrence of the value (i.e. low[0]-stop_loss) When long_condition_2 is True

Define stop loss and target profit values

Backtesting of trading strategy(Sample)-3/3



```
//plotshape(long_condition_2, text='L', style=shape.flag, location=location.belowbar, color=color.blue, textcolor=color.blue, size=size.small)

if (long_condition_2)

    // Define stoploss and profit target conditions

    strategy.entry("long", strategy.long)

if (short_condition_2)
    strategy.entry("short", strategy.short)

if (strategy.position_size>0)
    strategy.close("long",when=not long_condition_1)

if (strategy.position_size>0)
    strategy.exit("exit_long",from_entry="long",limit=profit_price_long,stop=stop_price_long)

if (strategy.position_size<0)
    strategy.exit("exit_short",from_entry="short",limit=profit_price_short,stop=stop_price_short)

if (strategy.position_size<0)
    strategy.close("short",when=not short_condition_1)

plot(stop_price_short,color=color.red)
plot(profit_price_short,color=color.blue)

plot(stop_price_long,color=color.red,style=plot.style_cross)
plot(profit_price_long,color=color.blue,style=plot.style_cross)
```

Enter position

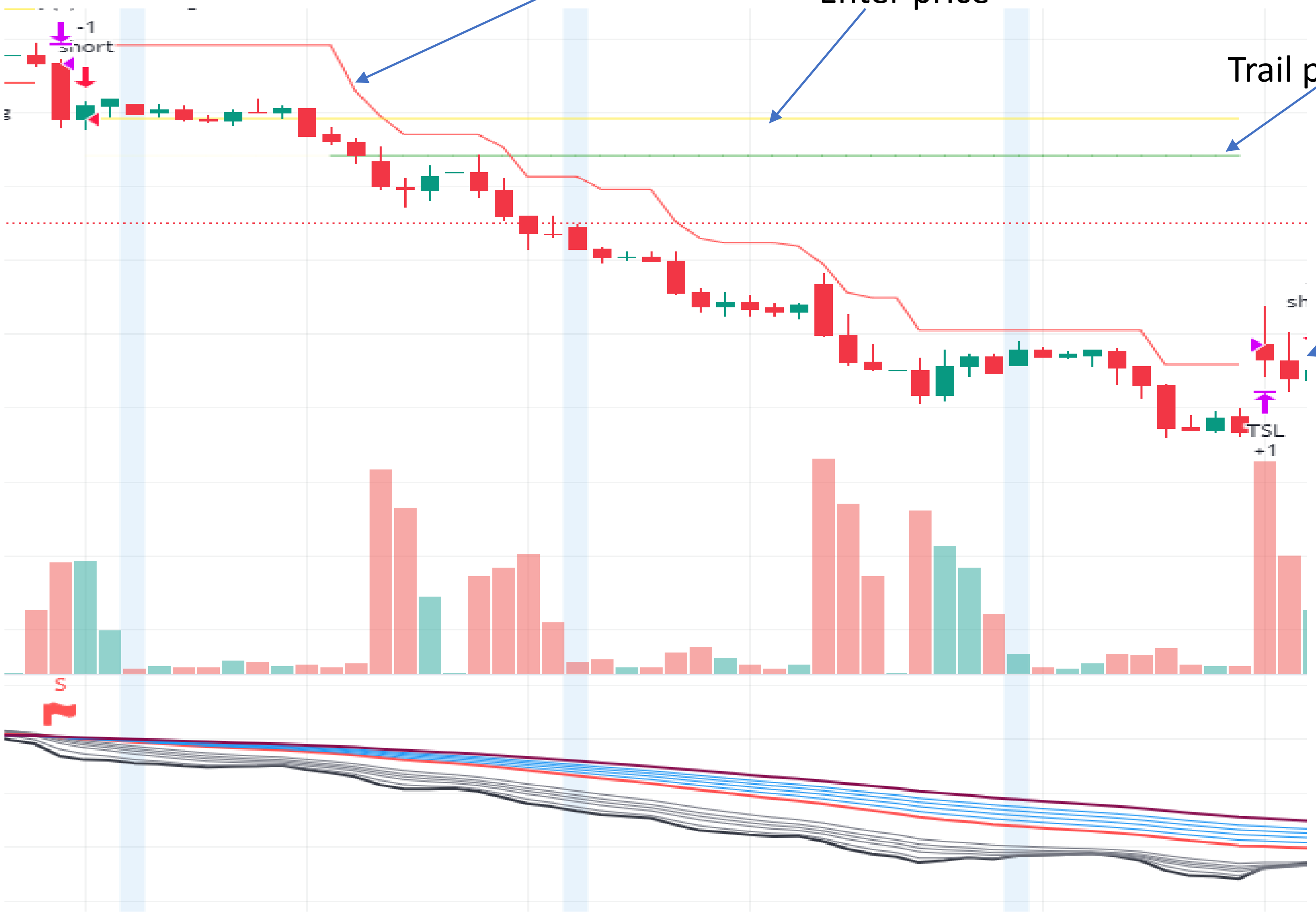
Close order when the specified condition is fulfilled

Exit when either profit target or Stop loss happen

To plot the stop loss and target profit lines

Target profit with trailing stop loss strategy

Trailing stop



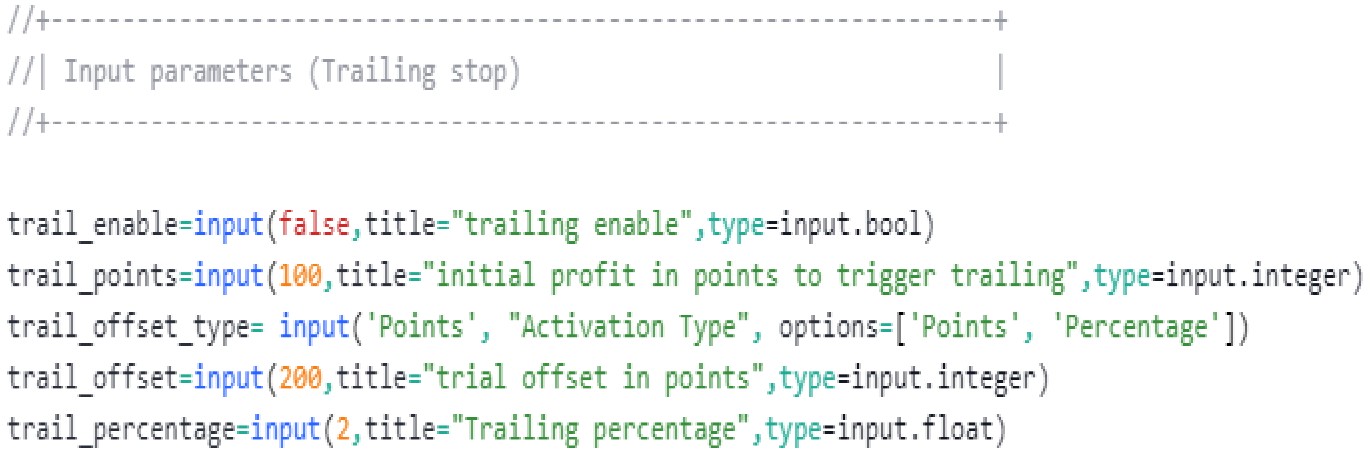
Stoploss

Enter price

Trail point

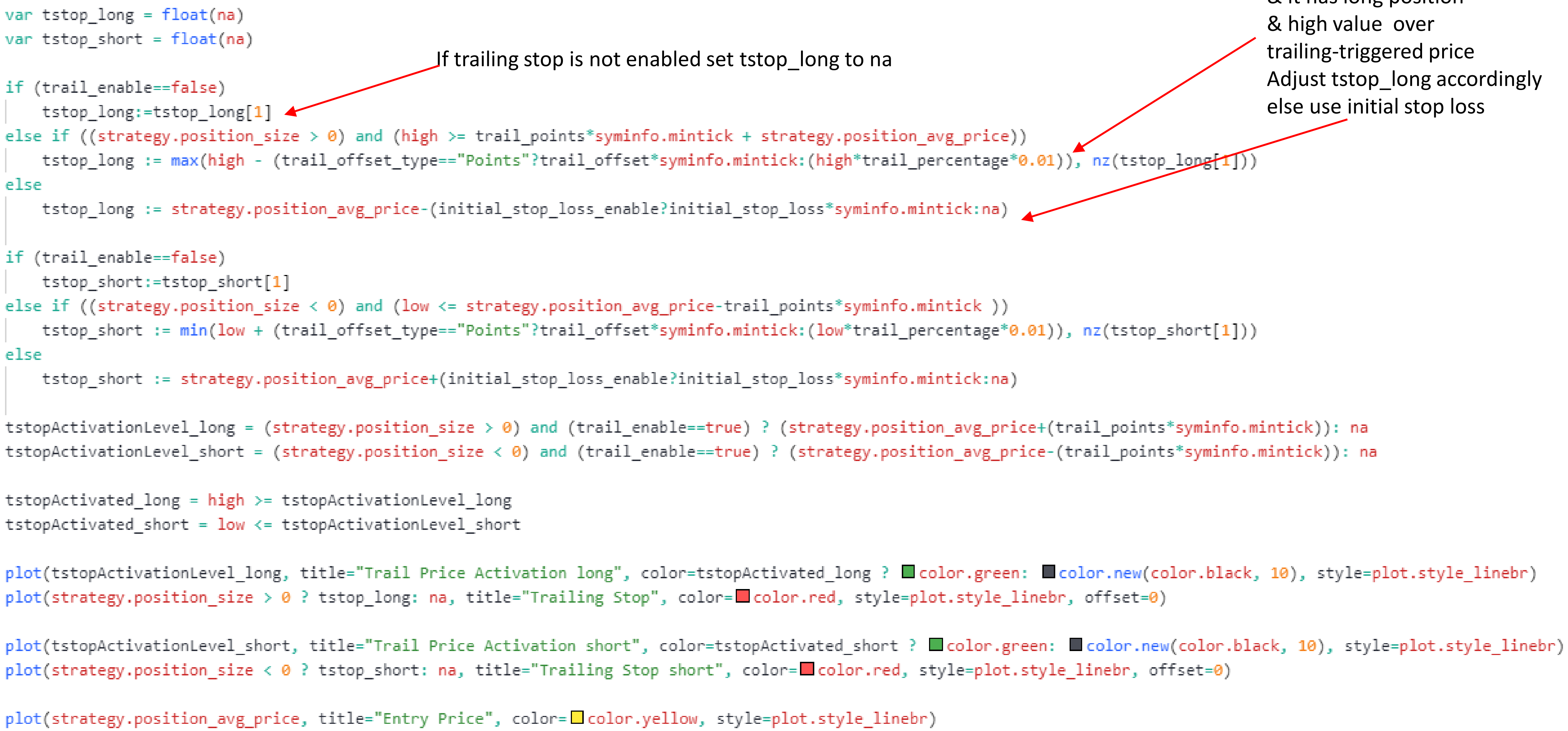
Stop out
By trailing stoploss

Trailing stoploss-1/2



```
//+------------------------------------------------------------------+
//| Input parameters (Trailing stop)                                 |
//+------------------------------------------------------------------+

trail_enable=input(false,title="trailing enable",type=input.bool)
trail_points=input(100,title="initial profit in points to trigger trailing",type=input.integer)
trail_offset_type= input('Points', "Activation Type", options=['Points', 'Percentage'])
trail_offset=input(200,title="trial offset in points",type=input.integer)
trail_percentage=input(2,title="Trailing percentage",type=input.float)
```

Trailing stop-2/2

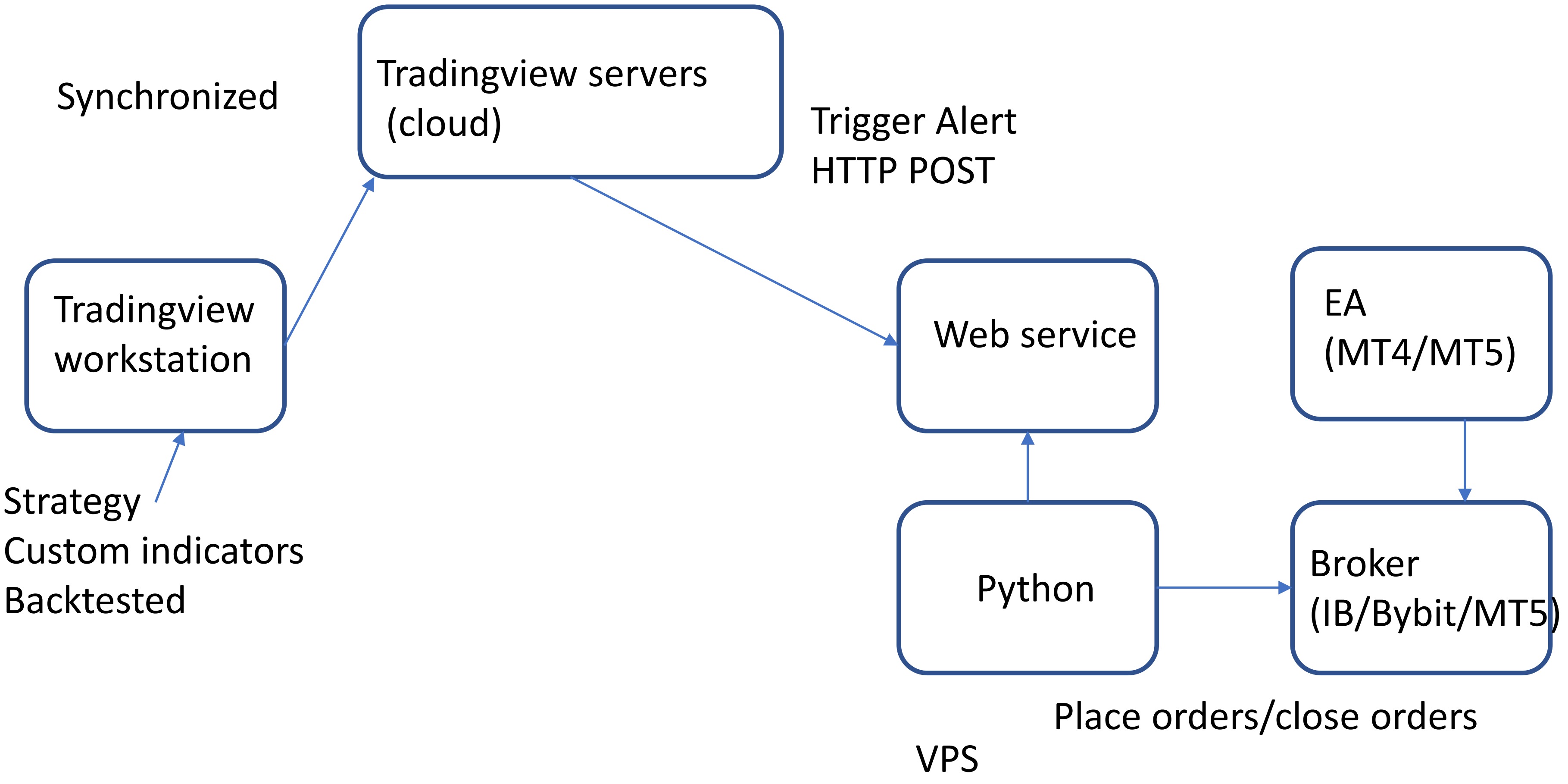


```
var tstop_long = float(na)
var tstop_short = float(na)

if (trail_enable==false)
    tstop_long:=tstop_long[1]
else if ((strategy.position_size > 0) and (high >= trail_points*syminfo.mintick + strategy.position_avg_price))
    tstop_long := max(high - (trail_offset_type=="Points"?trail_offset*syminfo.mintick:(high*trail_percentage*0.01)), nz(tstop_long[1]))
else
    tstop_long := strategy.position_avg_price-(initial_stop_loss_enable?initial_stop_loss*syminfo.mintick:na)

if (trail_enable==false)
    tstop_short:=tstop_short[1]
else if ((strategy.position_size < 0) and (low <= strategy.position_avg_price-trail_points*syminfo.mintick ))
    tstop_short := min(low + (trail_offset_type=="Points"?trail_offset*syminfo.mintick:(low*trail_percentage*0.01)), nz(tstop_short[1]))
else
    tstop_short := strategy.position_avg_price+(initial_stop_loss_enable?initial_stop_loss*syminfo.mintick:na)

tstopActivationLevel_long = (strategy.position_size > 0) and (trail_enable==true) ? (strategy.position_avg_price+(trail_points*syminfo.mintick)): na
tstopActivationLevel_short = (strategy.position_size < 0) and (trail_enable==true) ? (strategy.position_avg_price-(trail_points*syminfo.mintick)): na

tstopActivated_long = high >= tstopActivationLevel_long
tstopActivated_short = low <= tstopActivationLevel_short

plot(tstopActivationLevel_long, title="Trail Price Activation long", color=tstopActivated_long ? 🟩color.green: ⬛color.new(color.black, 10), style=plot.style_linebr)
plot(strategy.position_size > 0 ? tstop_long: na, title="Trailing Stop", color=🟥color.red, style=plot.style_linebr, offset=0)

plot(tstopActivationLevel_short, title="Trail Price Activation short", color=tstopActivated_short ? 🟩color.green: ⬛color.new(color.black, 10), style=plot.style_linebr)
plot(strategy.position_size < 0 ? tstop_short: na, title="Trailing Stop short", color=🟥color.red, style=plot.style_linebr, offset=0)

plot(strategy.position_avg_price, title="Entry Price", color=🟨color.yellow, style=plot.style_linebr)
```

If trailing stop is not enabled set tstop_long to na

If trailing stop is enabled
& it has long position
& high value over
trailing-triggered price
Adjust tstop_long accordingly
else use initial stop loss

Webhooking



Synchronized

Tradingview servers (cloud)

Trigger Alert
HTTP POST

Tradingview workstation

Strategy
Custom indicators
Backtested

Web service

EA (MT4/MT5)

Python

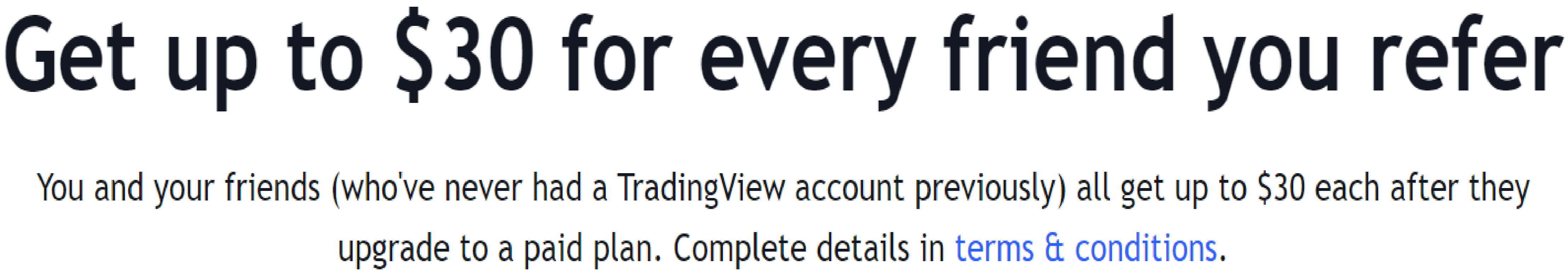
Broker (IB/Bybit/MT5)

Place orders/close orders

VPS

Referral program

- https://www.tradingview.com/gopro/?share_your_love=vickslee



Get up to $30 for every friend you refer

You and your friends (who've never had a TradingView account previously) all get up to $30 each after they upgrade to a paid plan. Complete details in terms & conditions.