All the fields in brackets can be edited by user. All mentioned values are for examples purpose.

Example:

Let's consider opening price to be 100.

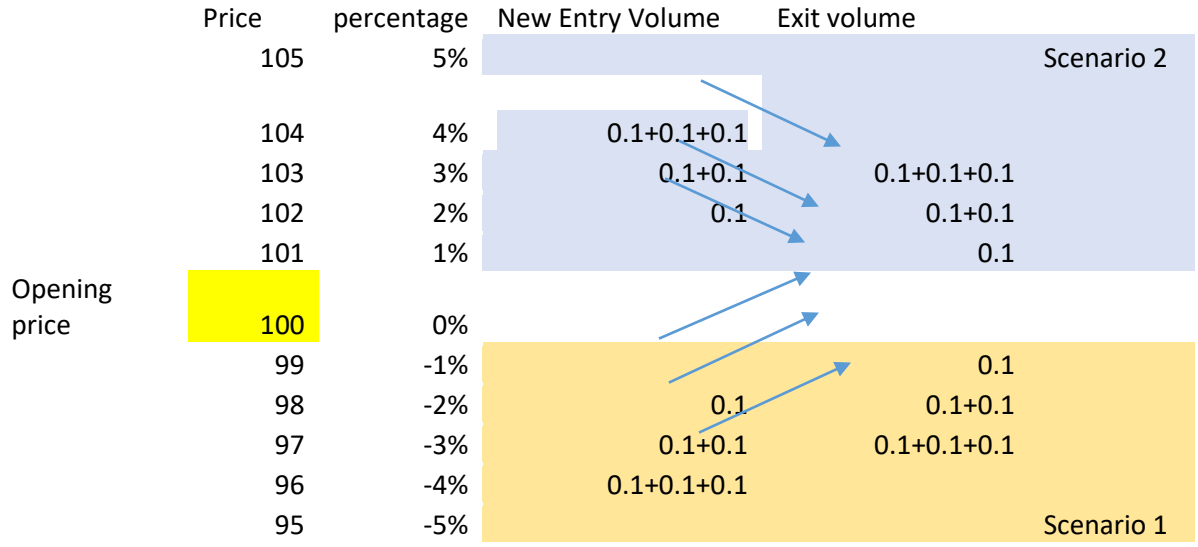
Scenario 1 (for shares fallen below opening price)

If shares fall by a certain percentage (Low_entry_percent = 2%), then buy (entry_vol = 0.1) lot.

The robot will place buy order at 98 with 0.1 lot.

Once buy order is executed, place sell order with price higher than buying price by (Low_exit_percent = 1%) and same volume i.e., Place sell order at 99 with 0.1 lot.

If the shares fall **further** by a certain percentage (Inc_entry_percent = 1%), place another buy order with increased volume (Inc_entry_lot=0.1+0.1). If executed, place sell order with increased price(Low_exit_percent = 1%) and same volume.



User should be able to enable Scenario 1, or scenario 2 or both

Scenario 2 (for shares risen above opening price):

If shares rise by a certain percentage (high_entry_percent = 2%), then sell (entry_vol = 0.1) lot.

The robot will place sell order at 102 with 0.1 lot.

Once sell order is executed, place buy order with price lower than selling price by (high_exit_percent = 1%) and same volume i.e., Place buy order at 101 with 0.1 lot.

If the shares rise **further** by a certain percentage (Inc_entry_percent = 1%), place another sell order with increased volume (Inc_entry_lot=0.1+0.1). If executed, place buy order with decreased price (high_exit_percent = 1%) and same volume.

For all open positions, pending orders to be carried forward to next day until position closes by code or by manually user.