```
{
MARKET INTERNALS SMART CODE (version AUG 2018)

************************************************************************************
*********************************
MARKET INTERNALS PRO COURSE - THIS CODE IS A PART OF AN EDUCATIONAL PRODUCT.
THIS CODE IS FOR STUDYING PURPOSES ONLY.

DISCLAIMER: Futures trading systems and commodity trading bear a high degree of
risk. People can and do lose money.
Hypothetical results have many inherent limitations. Past performance does not
guarantee future results.

2018(c)Tomas Nesnidal
www.SystemsOnTheRoad.com
www.BetterSystemTrader.com

Copyrighted material. Sharing without author´s approval is strictly prohibited and
against the law.
************************************************************************************
*********************************
}

Inputs:
MI_Switch(1),             //1-32
MI_Parameter(5),          //1-15
Entry_Exit_both(1),         //1 - apply MI for ENTRIES only, 2 - apply MI for EXITS
only, 3 - apply MI for BOTH (Entries and Exits)
ResetOnDayBeginning(0),     //0-1   0=false, 1=true
ResetAfterExit(0);          //0-1   0=false, 1=true


//**********************************************************************************
*********************
//Here insert INPUTS OF YOUR OWN TRADING STRATEGY
//Note: Preferably change them to VARS







//**********************************************************************************
*********************


Vars:
modeLong(false),
modeShort(false),
myTickLOD(0),
myTickHOD(0),
myTickHighest100(0),
myTickLowest100(0),
myZoneLong(0),
myZoneShort(0),
cnt_zona(0),
appliedZones(0),
myDiff(0),
myDiffAvg(0),
myDiffAvgBig(0),
myUvolDvolRatio(0),
MP(0),
RatioAVG(0),
RatioSTDVplus(0),
RatioSTDVminus(0),
maxRatioLong(0),
maxRatioShort(0),
myLOOP(0),
myVIXhighest(0),
myVIXlowest(0),
myRatioLong(0),
myRatioShort(0);

//**********************************************************************************
*********************
//Here insert VARIABLES OF YOUR OWN TRADING STRATEGY






//**********************************************************************************
*********************

Arrays:
ZoneLong[6](0),
ZoneShort[6](0);

//**********************************************************************
//PREPARATION BEGINNING
//**********************************************************************
//Reset at the beginning of a new day (IF SWITCH ResetOnDayBeginning = TRUE)
If ResetOnDayBeginning = 1 then begin

       if Date <> date[1] then begin
              if marketposition = 0 then begin
                     modeLONG = false;
                     modeSHORT = false;
              End else begin
                     If Entry_Exit_both <> 1 then begin   //If exit based on Market-
Internals = TRUE (Entry_Exit_both = 2 or 3)...
                            If marketposition > 0 then modeLONG = false;   //...
then we CAN´T reset modeSHORT if in LONG position as it can be a potential exit
signal on the beginning of the day
                            If marketposition < 0 then modeSHORT = false;  ////...
then we CAN´T reset modeLONG if in SHORT position as it can be a potential exit
signal on the beginning of the day
                     End else begin
                            modeLONG = false;
                            modeSHORT = false;
                     end;
              end;
       end;

End;


//**********************************************************************
//PREPARATION END
//**********************************************************************

//Variables
MP = MarketPosition;

//****MarketInternal Condition SWITCHES******
```

```
Switch (MI_Switch) begin
//In this mode we WILL NOT use MI condition
       Case 0: Begin
              modeLONG = True;
              modeSHORT = True;
       End;

//****** PART 1: CONDITIONS FOR TICK/TICKY MARKET INTERNAL *******

       Case 1: Begin //Basic condition - absolute value - version 1
              If High of data3 > 100 * MI_Parameter then begin
                     modeSHORT = false; modeLong = true;
              end;
              If Low of data3 < -100 * MI_Parameter then begin
                     modeLONG = false; modeSHORT = true;
              end;
       end;

       Case 2: Begin //Basic condition - absolute value - version 1 - REVERSE
              If High of data3 > 100 * MI_Parameter then begin
                     modeLONG = false; modeSHORT = true;
              end;
              If Low of data3 < -100 * MI_Parameter then begin
                     modeSHORT = false; modeLong = true;
              end;
       end;

       Case 3: Begin //Basic condition - absolute value - version 2
              modeLong =  High of data3 > 100 * MI_Parameter;
              modeSHORT = Low of data3 < -100 * MI_Parameter;
       end;

       Case 4: Begin //Basic condition - absolute value - version 2 - REVERSE
              Modelong =  High of data3 < 100 * MI_Parameter and low of data3 > -100
* MI_Parameter;
              ModeShort = ModeLong;
       end;

       Case 5: begin //LEVELLING - 3 LEVELS BASED ON LAST 100 DAYS
              MyTickHighest100 = Highest(High of data6, 100);
              myTickLowest100 = Lowest(Low of data6, 100);

                     myZoneLong = myTickHighest100 / 3;
                     myZoneShort = myTickLowest100 / 3;

              ZoneLong[1] = 0; ZoneShort[1] = 0;
              For cnt_zona = 2 to 3
              begin
                     ZoneLong[cnt_zona] = myZoneLong * (cnt_zona - 1);
                     ZoneShort[cnt_zona] = myZoneShort * (cnt_zona - 1);
              end;

              appliedZones = maxlist(1, floor(MI_Parameter));
              If MI_Parameter > 3 then appliedZones = 3;
              modeLONG =  C of data3 >= ZoneLong[appliedZones];
              modeSHORT = C of data3 <= ZoneShort[appliedZones];

       end;

       Case 6: Begin //BOLLINGER BANDS EXTREMES
              if h of data3 > bollingerBand(c of data3, 10*MI_Parameter, 2) then
begin
                     modeLONG = true; modeSHORT = false;
              end;
```

```
             if l of data3 < bollingerBand(c of data3, 10*MI_Parameter, -2) then
begin
                    modeSHORT = true; modeLong = false;
             end;
      end;

      Case 7: Begin //BOLLINGER BANDS EXTREMES - REVERSE
             if h of data3 > bollingerBand(c of data3, 10*MI_Parameter, 2) then
begin
                    modeLONG = false; modeSHORT = true;
             end;
             if l of data3 < bollingerBand(c of data3, 10*MI_Parameter, -2) then
begin
                    modeSHORT = false; modeLong = true;
             end;
      end;

      Case 8: Begin //TWO BOLLINGER BANDS (COMPARATION)
             if bollingerBand(c of data3, 3*MI_Parameter, 2) > bollingerBand(c of
data3, 10*MI_Parameter, 2) then begin
                    modeLONG = true; modeSHORT = false;
             end;
             if bollingerBand(c of data3, 3*MI_Parameter, -2) < bollingerBand(c of
data3, 10*MI_Parameter, -2) then begin
                    modeSHORT = true; modeLong = false;
             end;
      end;

      Case 9: Begin //TWO BOLLINGER BANDS (COMPARATION) -- REVERSE
             if bollingerBand(c of data3, 3*MI_Parameter, 2) < bollingerBand(c of
data3, 10*MI_Parameter, 2) then begin
                    modeLONG = true; modeSHORT = false;
             end;
             if bollingerBand(c of data3, 3*MI_Parameter, -2) > bollingerBand(c of
data3, 10*MI_Parameter, -2) then begin
                    modeSHORT = true; modeLong = false;
             end;
      end;

      Case 10: Begin //NEW HIGHEST-HIGH or NEW LOWEST-LOW
             if High of data3 = highest(high, 3* MI_Parameter) of data3 then begin
                    modeLONG = true; modeSHORT = false;
             end;
             if Low of data3 = lowest (low, 3* MI_Parameter) of data3 then begin
                    modeSHORT = true; modeLONG = false;
             end;
      end;

      Case 11: Begin //NEW HIGHEST-HIGH or NEW LOWEST-LOW -- FINER VERSIION (uses
STEP 1 instead of STEP 3)
             if High of data3 = highest(high, MI_Parameter) of data3 then begin
                    modeLONG = true; modeSHORT = false;
             end;
             if Low of data3 = lowest (low, MI_Parameter) of data3 then begin
                    modeSHORT = true; modeLONG = false;
             end;
      end;

      Case 12: Begin  //Highest High Of day (paremeterless)
             If date<>date[1] then begin
                    myTickLOD = L of data3;
                    myTickHOD = H of data3;
             end;
```

```
                If L of data3 < myTickLOD then myTickLOD = L of data3;
                If H of data3 > myTickHOD then myTickHOD = H of data3;

                If myTICKhod > myTickHod[1] then begin
                        modeLONG = TRUE; modeSHORT = false;
                end;
                If myTICKlod < myTickLod[1] then begin
                        modeSHORT = true; ModeLONG = false;
                end;
        end;

        Case 13: Begin      //Highest High Of day (paremeterless) - REVERSE
                If date<>date[1] then begin
                        myTickLOD = L of data3;
                        myTickHOD = H of data3;
                end;

                If L of data3 < myTickLOD then myTickLOD = L of data3;
                If H of data3 > myTickHOD then myTickHOD = H of data3;

                If myTICKhod > myTickHod[1] then begin
                        modeSHORT = true; ModeLONG = false;
                end;
                If myTICKlod < myTickLod[1] then begin
                        modeLONG = TRUE; modeSHORT = false;
                end;
        end;

//****** PART 2: CONDITIONS FOR UVOL-DVOL MARKET INTERNALS *******

        Case 14: Begin //Basic use - absolute value
                modeLONG = Close of data4 - Close of data5 > 20000 * MI_Parameter;
                modeSHORT = Close of data5 - Close of data4 > 20000 * MI_Parameter;
        end;

        Case 15: Begin //UVOL-DVOL difference and Moving Average of the Difference -
ADVANCED version
                myDiff = Close of data4 - Close of data5;
                myDiffAvg = average(myDiff, 10 * MI_Parameter);
                //Entry condition
                If myDiff > 0 and myDiff > myDiffAvg and myDiffAvg > 0 then begin
                        modeLONG = TRUE; modeSHORT = false;
                end;
                If myDiff < 0 and myDiff < myDiffAvg and myDiffAvg < 0 then begin
                        modeSHORT = true; ModeLONG = false;
                end;
                //Cancel condition
                If modeLONG = True and ((myDiff < 0)or (myDiffAvg < 0)) then modeLong
= false;
                If modeSHORT = True and  ((myDiff > 0)or (myDiffAvg > 0)) then
modeSHORT = false;
        end;

        Case 16: Begin //UVOL-DVOL difference and Moving Average of the Difference -
SIMPLE version, different AVG step
                myDiff = Close of data4 - Close of data5;
                myDiffAvg = average(myDiff, 5 * MI_Parameter);
                //Entry condition
                If myDiff > 0 and myDiff > myDiffAvg  then begin
                        modeLONG = TRUE; modeSHORT = false;
                end;
                If myDiff < 0 and myDiff < myDiffAvg  then begin
                        modeSHORT = true; ModeLONG = false;
                end;
```

```
             //cancel condition
             If modeLONG = True and (myDiff < 0) then modeLong = false;
             If modeSHORT = True and  (myDiff > 0) then modeSHORT = false;
       end;

       Case 17: Begin //Ratio and absolute values
        modeLONG = IFF(Close of data5 <> 0, Close of data4 / Close of data5, 0) >
(MI_Parameter/3);
        modeSHORT = IFF(close of data4 <> 0, Close of data5 / Close of data4, 0) >
(MI_Parameter/3);
       end;

       Case 18: Begin //Ratio and absolute values -- REVERSE
        modeLONG = IFF(Close of data5 <> 0, Close of data4 / Close of data5, 0) <
(MI_Parameter/3);
        modeSHORT = IFF(Close of data4 <> 0, Close of data5 / Close of data4, 0) <
(MI_Parameter/3);
       end;

       Case 19: begin //Extremes with the use of Statistical Arbitrage principles -
RATIO model
             If C of data4 > C of data5 then myUvolDvolRatio = C of data4 / C of
data5;
             If C of data4 < C of data5 then myUvolDvolRatio = -1*(C of data5 / C
of data4);
             If C of data4 = C of data5 then myUvolDvolRatio = 0;
             RatioAVG = average(myUvolDvolRatio, 5* MI_Parameter);
             RatioSTDVplus = RatioAVG + standardDev(myUvolDvolRatio, 5*
MI_Parameter, 2);
             RatioSTDVminus = RatioAVG - standardDev(myUvolDvolRatio, 5*
MI_Parameter, 2);
             //Establish trade mode
             modeLONG = myUvolDvolRatio > RatioSTDVplus and myUvolDvolRatio > 0;
             modeSHORT = myUvolDvolRatio < RatioSTDVminus and myUvolDvolRatio < 0;
       end;

       Case 20 : begin //Extremes with the use of Statistical Arbitrage principles
- DIFFERENCE model
             myDiff = Close of data4 - Close of data5;
             MyDiffAVG = average(myDiff, 5* MI_Parameter);
             RatioSTDVplus = MyDiffAVG + standardDev(myDiff, 5* MI_Parameter, 2);
             RatioSTDVminus = MyDiffAVG - standardDev(myDiff, 5* MI_Parameter, 2);
             //Establish trade mode
             modeLONG = myDiff > RatioSTDVplus and myDiff > 0;
             modeSHORT = myDiff < RatioSTDVminus and myDiff < 0;
       end;

       Case 21: begin //LEVELLING of RATIO - Highest and Lowest Ratio of last 100
days (complex MI condition) - version PERMANENT -- 3 ZONES
             maxRatioLong = 0;
             maxRatioShort = 0;
             //Find the highest and lowest values for the ratio in last 100 days
             For myLOOP = 1 to 100 begin
                    if Close[myLOOP] of data7 - Close[myLOOP] of data8 > 0 then
myRatioLong = Close[myLOOP] of data7 / Close[myLOOP] of data8;
                    if Close[myLOOP] of data7 - Close[myLOOP] of data8 < 0 then
myRatioShort = Close[myLOOP] of data8 / Close[myLOOP] of data7;
                    If myRatioLong > maxRatioLong then maxRatioLong = myRatioLong;
                    If myRatioShort < maxRatioShort then maxRatioShort = myRati-
oShort;
             end;
             //Establish 3 levels (zones)
             MyZoneLong = maxRatioLong / 3;
             MyZoneShort = maxRatioShort / 3;
```

```
              For cnt_zona = 1 to 3
              begin
                     ZoneLong[cnt_zona] = myZoneLong * cnt_zona;
                     ZoneShort[cnt_zona] = myZoneShort * cnt_zona;
              end;
              //final condition
              appliedZones = maxlist(1, floor(MI_Parameter));
              If MI_Parameter > 3 then appliedZones = 3;
              If ( ( (Close of data4 - Close of data5) > 0 ) and (Close of data4 /
Close of data5) > ZoneLong[appliedZones]) then begin
                     modeLong = true; ModeShort = False;
              end;
              If ((Close of data4 - Close of data5) < 0 and (Close of data5 / Close
of data4) > ZoneShort[appliedZones]) then begin
                     modeShort = true; ModeLong = false;
              end;
       end;

       Case 22: begin //LEVELLING of RATIO - Highest and Lowest Ratio of last 100
days (complex MI condition) - version PERMANENT -- 3 ZONES -- REVERSE
              maxRatioLong = 0;
              maxRatioShort = 0;
              //Find the highest and lowest values for the ratio in last 100 days
              For myLOOP = 1 to 100 begin
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 > 0 then
myRatioLong = Close[myLOOP] of data7 / Close[myLOOP] of data8;
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 < 0 then
myRatioShort = Close[myLOOP] of data8 / Close[myLOOP] of data7;
                     If myRatioLong > maxRatioLong then maxRatioLong = myRatioLong;
                     If myRatioShort < maxRatioShort then maxRatioShort = myRati-
oShort;
              end;
              //Establish 3 levels (zones)
              MyZoneLong = maxRatioLong / 3;
              MyZoneShort = maxRatioShort / 3;
              For cnt_zona = 1 to 3
              begin
                     ZoneLong[cnt_zona] = myZoneLong * cnt_zona;
                     ZoneShort[cnt_zona] = myZoneShort * cnt_zona;
              end;
              //final condition
              appliedZones = maxlist(1, floor(MI_Parameter));
              If MI_Parameter > 3 then appliedZones = 3;
              If ( ( (Close of data4 - Close of data5) > 0 ) and (Close of data4 /
Close of data5) < ZoneLong[appliedZones]) then begin
                     modeLong = true; ModeShort = False;
              end;
              If ((Close of data4 - Close of data5) < 0 and (Close of data5 / Close
of data4) < ZoneShort[appliedZones]) then begin
                     modeShort = true; ModeLong = false;
              end;
       end;

       Case 23 : begin //LEVELLING of RATIO - Highest and Lowest Ratio of last 100
days (complex MI condition) - version TEMPORARY
              maxRatioLong = 0;
              maxRatioShort = 0;
              //Find the highest and lowest values for the ratio in last 100 days
              For myLOOP = 1 to 100 begin
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 > 0 then
myRatioLong = Close[myLOOP] of data7 / Close[myLOOP] of data8;
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 < 0 then
myRatioShort = Close[myLOOP] of data8 / Close[myLOOP] of data7;
                     If myRatioLong > maxRatioLong then maxRatioLong = myRatioLong;
                     If myRatioShort < maxRatioShort then maxRatioShort = myRatioShort;
              end;
              //Establish 3 levels (zones)
              MyZoneLong = maxRatioLong / 3;
              MyZoneShort = maxRatioShort / 3;
              For cnt_zona = 1 to 3
              begin
                     ZoneLong[cnt_zona] = myZoneLong * cnt_zona;
                     ZoneShort[cnt_zona] = myZoneShort * cnt_zona;
              end;
              //final condition
              appliedZones = maxlist(1, floor(MI_Parameter));
              If MI_Parameter > 3 then appliedZones = 3;
              modeLong =  ((Close of data4 - Close of data5) > 0 and (Close of data4
/ Close of data5) > ZoneLong[appliedZones]);
              modeShort = ((Close of data4 - Close of data5) < 0 and (Close of data5
/ Close of data4) > ZoneShort[appliedZones]);
       end;

       Case 24 : begin //LEVELLING of RATIO - Highest and Lowest Ratio of last 100
days (complex MI condition) - version TEMPORARY -- REVERSE
              maxRatioLong = 0;
              maxRatioShort = 0;
              //Find the highest and lowest values for the ratio in last 100 days
              For myLOOP = 1 to 100 begin
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 > 0 then
myRatioLong = Close[myLOOP] of data7 / Close[myLOOP] of data8;
                     if Close[myLOOP] of data7 - Close[myLOOP] of data8 < 0 then
myRatioShort = Close[myLOOP] of data8 / Close[myLOOP] of data7;
                     If myRatioLong > maxRatioLong then maxRatioLong = myRatioLong;
                     If myRatioShort < maxRatioShort then maxRatioShort = myRatioShort;
              end;
              //Establish 3 levels (zones)
              MyZoneLong = maxRatioLong / 3;
              MyZoneShort = maxRatioShort / 3;
              For cnt_zona = 1 to 3
              begin
                     ZoneLong[cnt_zona] = myZoneLong * cnt_zona;
                     ZoneShort[cnt_zona] = myZoneShort * cnt_zona;
              end;
              //final condition
              appliedZones = maxlist(1, floor(MI_Parameter));
              If MI_Parameter > 3 then appliedZones = 3;
              modeLong =  ((Close of data4 - Close of data5) > 0 and (Close of data4
/ Close of data5) < ZoneLong[appliedZones]);
              modeShort = ((Close of data4 - Close of data5) < 0 and (Close of data5
/ Close of data4) < ZoneShort[appliedZones]);
       end;


//****** PART 3 : VOLATILITY MI CONDITIONS (WITH WIX) ********

       Case 25 : begin //Current volatility above average volatility
              if Close[1] of data9 > Average(Close, 10 * MI_Parameter) of data9 then
begin
                     modeLong = true; modeShort = true;
              end else begin
                     modeLong = false; modeShort = false;
              end;
       end;

       Case 26 : begin //Current volatility below average volatility
```

```
              if Close[1] of data9 < Average(Close, 10 * MI_Parameter) of data9 then
begin
                    modeLong = true; modeShort = true;
              end else begin
                    modeLong = false; modeShort = false;
              end;
       end;

       Case 27 : begin //short term average volatility above average volatility
              if     Average(Close, 2 * MI_Parameter) of data9 > Average(Close, 10 *
MI_Parameter) of data9 then begin
                    modeLong = true; modeShort = true;
              end else begin
                    modeLong = false; modeShort = false;
              end;
       end;

       Case 28 : begin //short term average volatility below average volatility
              if Average(Close, 2 * MI_Parameter) of data9 < Average(Close, 10 *
MI_Parameter) of data9 then begin
                    modeLong = true; modeShort = true;
              end else begin
                    modeLong = false; modeShort = false;
              end;
       end;

       Case 29 : begin //Levelling - 3 zones (levels) according to last 100 days -
version 1
              myVIXhighest = Highest(C of data9, 100);
              myVIXlowest = Lowest(C of data9, 100);

           myZoneLong = (myVIXhighest-myVIXlowest) / 3;  //calculate the range for
one level

              //according to MI_Parameter input get the actual zone and allow trades
only in such a zone (level)
              if MI_Parameter = 1 then begin
                    if  (Close[1] of data9 <= myVIXlowest + myZoneLong) then begin
                           modeLong = true; modeShort = true;
                    end else begin
                           modeLong = false; modeShort = false;
                    end;
              end;

              if MI_Parameter = 2 then begin
                    if (Close[1] of data9 <= myVIXlowest + 2*myZoneLong) and
(Close[1] of data9 > myVIXlowest + myZoneLong) then begin
                           modeLong = true; modeShort = true;
                    end else begin
                           modeLong = false; modeShort = false;
                    end;
              end;

              if MI_Parameter = 3 then begin
                    if (Close[1] of data9 > myVIXlowest + 2*myZoneLong)  then begin
                           modeLong = true; modeShort = true;
                    end else begin
                           modeLong = false; modeShort = false;
                    end;
              end;

       end;
       Case 30: begin //Levelling - 3 zones (levels) according to last 100 days -
version 2
              myVIXhighest = Highest(H of data9, 100);
              myVIXlowest = Lowest(L of data9, 100);

           myZoneLong = (myVIXhighest-myVIXlowest) / 3;  //calculate the range for
one level

              //according to MI_Parameter input get the actual zone and allow trades
only in such a zone (level)
              if MI_Parameter = 1 then begin
                     if  (Close[1] of data9 <= myVIXlowest + myZoneLong) then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
                     end;
              end;

              if MI_Parameter = 2 then begin
                     if (Close[1] of data9 <= myVIXlowest + 2*myZoneLong) and
(Close[1] of data9 > myVIXlowest + myZoneLong) then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
                     end;
              end;

              if MI_Parameter = 3 then begin
                     if (Close[1] of data9 > myVIXlowest + 2*myZoneLong)  then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
                     end;
              end;
       end;

       Case 31 : begin //Filter the weakest zone - version1
              myVIXhighest = Highest(C of data9, 100);
              myVIXlowest = Lowest(C of data9, 100);

           myZoneLong = (myVIXhighest-myVIXlowest) / 3;

              if MI_Parameter = 1 then begin
                     if  (Close[1] of data9 > myVIXlowest + myZoneLong) then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
                     end;
              end;

              if MI_Parameter = 2 then begin
                     if (Close[1] of data9 < myVIXlowest + myZoneLong) or (Close[1]
of data9 > myVIXlowest + 2*myZoneLong) then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
                     end;
              end;

              if MI_Parameter = 3 then begin
                     if (Close[1] of data9 < myVIXlowest + 2*myZoneLong)  then begin
                            modeLong = true; modeShort = true;
                     end else begin
                            modeLong = false; modeShort = false;
```

TRADING INVOLVES A SUBSTANTIAL RISK OF LOSS AND IS NOT SUITABLE FOR ALL INVESTORS,
PAST PERFORMANCE IS NOT NECESSARILY INDICATIVE OF FUTURE RESULTS

```
                        end;
                end;

        end;

        Case 32 : begin //Filter the weakest zone - version2
                myVIXhighest = Highest(H of data9, 100);
                myVIXlowest = Lowest(L of data9, 100);

            myZoneLong = (myVIXhighest-myVIXlowest) / 3;

                if MI_Parameter = 1 then begin
                        if  (Close[1] of data9 > myVIXlowest + myZoneLong) then begin
                                modeLong = true; modeShort = true;
                        end else begin
                                modeLong = false; modeShort = false;
                        end;
                end;

                if MI_Parameter = 2 then begin
                        if (Close[1] of data9 < myVIXlowest + myZoneLong) or (Close[1]
of data9 > myVIXlowest + 2*myZoneLong) then begin
                                modeLong = true; modeShort = true;
                        end else begin
                                modeLong = false; modeShort = false;
                        end;
                end;

                if MI_Parameter = 3 then begin
                        if (Close[1] of data9 < myVIXlowest + 2*myZoneLong)  then begin
                                modeLong = true; modeShort = true;
                        end else begin
                                modeLong = false; modeShort = false;
                        end;
                end;
        end;

END;//END of Market Internals Switches


//*********************************************************************
//BEGINING: HERE INSERT YOUR TRADING SYSTEM AND FOLLOW THE INSTRUCTION
//*********************************************************************


//here insert the strategy WITHOUT the final entry condition intself and WITHOUT
exit conditions


{Final entry conditions - modified for MI testing}
if Entry_Exit_both <> 2 then begin

        { here insert the FINAL LONG CONDITION AND ORDER + add syntax "and ModeLong"
}
        { here insert the FINAL SHORT CONDITION AND ORDER + add syntax "and
ModeShort" }

end else begin

        { here insert the FINAL LONG CONDITION AND ORDER (dont add anything else) }
        { here insert the FINAL SHORT CONDITION AND ORDER (dont add anything else) }

end;
```

```
//here insert the EXIT conditions of your original strategy


//*******************************************************************
//YOUR ORIGINAL STRATEGY END
//*******************************************************************

{premature EXIT based on MI - if a significant change in the mood of the market as
a whole}
if Entry_Exit_both <> 1 then begin
       if MP >0 and ModeShort then begin
              Sell ("MI_ExitLNG") this bar at close;
       end;
       if MP <0 and ModeLong then begin
              BuyToCover ("MI_ExitSHRT") this bar at close;
       end;
end;

{Reset MI condition after exit}
If ResetAfterExit = 1 then begin
       If MP = 0 and MP[1] > 0 then modeLONG = false;
       If MP = 0 and MP[1] < 0 then modeSHORT = false;
end;
```