

## EASLANGUAGE CODE FOR MONTHLY AND WEEKLY CYCLES

```

Day of Month

{
  DayOfMonthStudy
  Data1 - Security for study / daily bars
}

const:
  maxdaysinmonth( 31 );

inputs:
  MultBy1000( 0 ),
  MonthToTest( 0 ), // 0 for all months
  AbovePosBelowNegAverage( 0 ),
  AverageLength( 200 );

variables:
  TDM( 0 ),
  Counter( 0 ),
  Initialized( False ),
  AvgVolume( 0 ),
  VMaxDaysInMonth( 0 ),
  Avg( 0 );

arrays:
  DayCount[maxdaysinmonth](0),
  PointMove[maxdaysinmonth](0), // close - open
  TruePointMove[maxdaysinmonth](0), // close - close[1]
  TotalUpMove[maxdaysinmonth](0), // close - low
  TotalDownMove[maxdaysinmonth](0), // high - close
  UpCount[maxdaysinmonth](0), // number of days where close > open
  DownCount[maxdaysinmonth](0), // number of days where close < open
  EqualCount[maxdaysinmonth](0), // number of days where close = open
  TrueUpCount[maxdaysinmonth](0), // number of days where close > close[1]
  TotalTrueUpMove[maxdaysinmonth](0),
  TrueDownCount[maxdaysinmonth](0), // number of days where close < close[1]
  TotalTrueDownMove[maxdaysinmonth](0),
  TrueEqualCount[maxdaysinmonth](0); // number of days where close = close[1]

if ( Month( date ) <> Month( date[1] ) ) then begin
  TDM = 0;
  Initialized = True;
end;

if Initialized = True Then Begin
  Avg = Average( Close, AverageLength );
  AvgVolume = Average( Volume, 50 );
  //If Volume > ( AvgVolume * 0.40 ) Then Begin
  If ( AbovePosBelowNegAverage = 0 or
    ( AbovePosBelowNegAverage > 0 and Close > Avg ) or
    ( AbovePosBelowNegAverage < 0 and Close < Avg ) ) Then Begin
    TDM = TDM + 1;

    if ( ( MonthToTest = 0 ) or ( MonthToTest = Month( date ) ) ) then begin
      DayCount[TDM] = DayCount[TDM] + 1;
      PointMove[TDM] = PointMove[TDM] + Close - Open;
      TruePointMove[TDM] = TruePointMove[TDM] + Close - Close[1];
      TotalUpMove[TDM] = TotalUpMove[TDM] + Close - Low;
      TotalDownMove[TDM] = TotalDownMove[TDM] + High - Close;
      if ( Close > Open ) then UpCount[TDM] = UpCount[TDM] + 1;
      if ( Close < Open ) then DownCount[TDM] = DownCount[TDM] + 1;
      if ( Close = Open ) then EqualCount[TDM] = EqualCount[TDM] + 1;
      if ( Close > Close[1] ) then begin
        TrueUpCount[TDM] = TrueUpCount[TDM] + 1;
        TotalTrueUpMove[TDM] = TotalTrueUpMove[TDM] + Close - Close[1];
      end;
      if ( Close < Close[1] ) then begin
        TrueDownCount[TDM] = TrueDownCount[TDM] + 1;
        TotalTrueDownMove[TDM] = TotalTrueDownMove[TDM] + Close - Close[1];
      end;
      if ( Close = Close[1] ) then TrueEqualCount[TDM] = TrueEqualCount[TDM] + 1;
    end;

    If TDM > VMaxDaysInMonth Then
      VMaxDaysInMonth = VMaxDaysInMonth + 1;
  end;
End;

if ( LastBarOnChart = TRUE ) then begin
  print( "Symbol: ", SymbolName, " Date Run: ", date, " Day of Month Study" );
  print( "" );

  for Counter = 1 to VMaxDaysInMonth begin
    If DayCount[Counter] > 0 Then Begin
      If MultBy1000 = 1 Then Begin

```

```

      print( "TDM: ", Counter, " Num Days: ", DayCount[Counter], " Up
Days: ", UpCount[Counter], " Down Days: ", DownCount[Counter], " Equal Days: ",
EqualCount[Counter] );
      print( "TDM: ", Counter, " PointMove: ", PointMove[Counter] * 1000, " Average
Move: ", PointMove[Counter] * 1000 / DayCount[Counter] );
      print( "TDM: ", Counter, " True Up Days: ", TrueUpCount[Counter], " True Down
Days: ", TrueDownCount[Counter], " True Equal Days: ", TrueEqualCount[Counter] );
      print( "TDM: ", Counter, " TruePointMove: ", TruePointMove[Counter] * 1000,
" Average True Move: ", TruePointMove[Counter] * 1000 / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalUpMove: ", TotalUpMove[Counter] * 1000, "
Average Up Move: ", TotalUpMove[Counter] * 1000 / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalDownMove: ", TotalDownMove[Counter] * 1000,
" Average Down Move: ", TotalDownMove[Counter] * 1000 / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalTrueUpMove: ", TotalTrueUpMove[Counter] *
1000, " Average True Up Move: ", TotalTrueUpMove[Counter] * 1000 / DayCount[Counter]
);
      print( "TDM: ", Counter, " TotalTrueDownMove: ", TotalTrueDownMove[Counter] *
1000, " Average True Down Move: ", TotalTrueDownMove[Counter] * 1000 / DayCount[Counter]
);
      print( "-----" );
    End Else Begin
      print( "TDM: ", Counter, " Num Days: ", DayCount[Counter], " Up
Days: ", UpCount[Counter], " Down Days: ", DownCount[Counter], " Equal Days: ",
EqualCount[Counter] );
      print( "TDM: ", Counter, " PointMove: ", PointMove[Counter], " Average Move:
", PointMove[Counter] / DayCount[Counter] );
      print( "TDM: ", Counter, " True Up Days: ", TrueUpCount[Counter], " True Down
Days: ", TrueDownCount[Counter], " True Equal Days: ", TrueEqualCount[Counter] );
      print( "TDM: ", Counter, " TruePointMove: ", TruePointMove[Counter], " Average
True Move: ", TruePointMove[Counter] / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalUpMove: ", TotalUpMove[Counter], " Average
Up Move: ", TotalUpMove[Counter] / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalDownMove: ", TotalDownMove[Counter], "
Average Down Move: ", TotalDownMove[Counter] / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalTrueUpMove: ", TotalTrueUpMove[Counter], "
Average True Up Move: ", TotalTrueUpMove[Counter] / DayCount[Counter] );
      print( "TDM: ", Counter, " TotalTrueDownMove: ", TotalTrueDownMove[Counter],
" Average True Down Move: ", TotalTrueDownMove[Counter] / DayCount[Counter] );
      print( "-----" );
    End;
  End;
end;
end;
end;

```

-----

Day of Week

```

{
  DayOfWeekStudy
  Data1 - Security for study / daily bars
}

```

```

inputs:
  AbovePosBelowNegAverage( 0 ),
  AverageLength( 200 );

```

```

variables:
  Avg( 0 ),
  AvgVolume( 0 ),
  Counter( 0 ),
  DayVal( 0 ),
  DayName( "" );

```

```

arrays:
  DayCount[5](0),
  PointMove[5](0), // close - open
  TruePointMove[5](0), // close - close[1]
  TotalUpMove[5](0), // close - low
  TotalDownMove[5](0), // high - close
  UpCount[5](0), // number of days where close > open
  DownCount[5](0), // number of days where close < open
  EqualCount[5](0), // number of days where close = open
  TrueUpCount[5](0), // number of days where close > close[1]
  TrueDownCount[5](0), // number of days where close < close[1]
  TrueEqualCount[5](0), // number of days where close = close[1]
  TrueUpMove[5](0),
  TrueDownMove[5](0);

```

```

Avg = Average( Close, AverageLength );
AvgVolume = Average( Volume, 50 );
If ( Volume > AvgVolume * 0.25 ) Then Begin
  If ( AbovePosBelowNegAverage = 0 or
    ( AbovePosBelowNegAverage > 0 and Close > Avg ) or
    ( AbovePosBelowNegAverage < 0 and Close < Avg ) ) Then Begin
    DayVal = DayOfWeek( date ) - 1;

```

```

DayCount[DayVal] = DayCount[DayVal] + 1;
PointMove[DayVal] = PointMove[DayVal] + Close - Open;
TruePointMove[DayVal] = TruePointMove[DayVal] + Close - Close[1];
TotalUpMove[DayVal] = TotalUpMove[DayVal] + Close - Low;
TotalDownMove[DayVal] = TotalDownMove[DayVal] + High - Close;
if ( Close > Open ) then UpCount[DayVal] = UpCount[DayVal] + 1;
if ( Close < Open ) then DownCount[DayVal] = DownCount[DayVal] + 1;
if ( Close = Open ) then EqualCount[DayVal] = EqualCount[DayVal] + 1;

if ( Close > Close[1] ) then Begin
    TrueUpCount[DayVal] = TrueUpCount[DayVal] + 1;
    TrueUpMove[DayVal] = TrueUpMove[DayVal] + Close - Close[1];
End;

if ( Close < Close[1] ) then Begin
    TrueDownCount[DayVal] = TrueDownCount[DayVal] + 1;
    TrueDownMove[DayVal] = TrueDownMove[DayVal] + Close - Close[1];
End;

if ( Close = Close[1] ) then TrueEqualCount[DayVal] = TrueEqualCount[DayVal] + 1;
End;
End;

if ( LastBarOnChart = TRUE ) then begin
    print( "Symbol: ", SymbolName, " Date Run: ", date, " Day of Week Study" );
    print( "" );

```

```

for Counter = 0 to 4 begin
    if ( Counter = 0 ) then DayName = "Monday";
    if ( Counter = 1 ) then DayName = "Tuesday";
    if ( Counter = 2 ) then DayName = "Wednesday";
    if ( Counter = 3 ) then DayName = "Thursday";
    if ( Counter = 4 ) then DayName = "Friday";

    print( DayName, " Num Days: ", DayCount[Counter], " Up Days: ", UpCount[Counter],
    " Down Days: ", DownCount[Counter], " Equal Days: ", EqualCount[Counter] );
    print( DayName, " PointMove: ", PointMove[Counter], " Average Move: ",
    PointMove[Counter] / DayCount[Counter] );
    print( DayName, " True Up Days: ", TrueUpCount[Counter], " True Down Days: ",
    TrueDownCount[Counter], " True Equal Days: ", TrueEqualCount[Counter] );
    print( DayName, " TruePointMove: ", TruePointMove[Counter], " Average True Move: ",
    TruePointMove[Counter] / DayCount[Counter] );
    print( DayName, " TrueUpMove: ", TrueUpMove[Counter], " Average True Up Move: ",
    TrueUpMove[Counter] / DayCount[Counter] );
    print( DayName, " TrueDownMove: ", TrueDownMove[Counter], " Average True
    Down Move: ", TrueDownMove[Counter] / DayCount[Counter] );
    print( DayName, " TotalUpMove: ", TotalUpMove[Counter], " Average Up Move: ",
    TotalUpMove[Counter] / DayCount[Counter] );
    print( DayName, " TotalDownMove: ", TotalDownMove[Counter], " Average Down
    Move: ", TotalDownMove[Counter] / DayCount[Counter] );
    print( "" );
    end;
end;

```