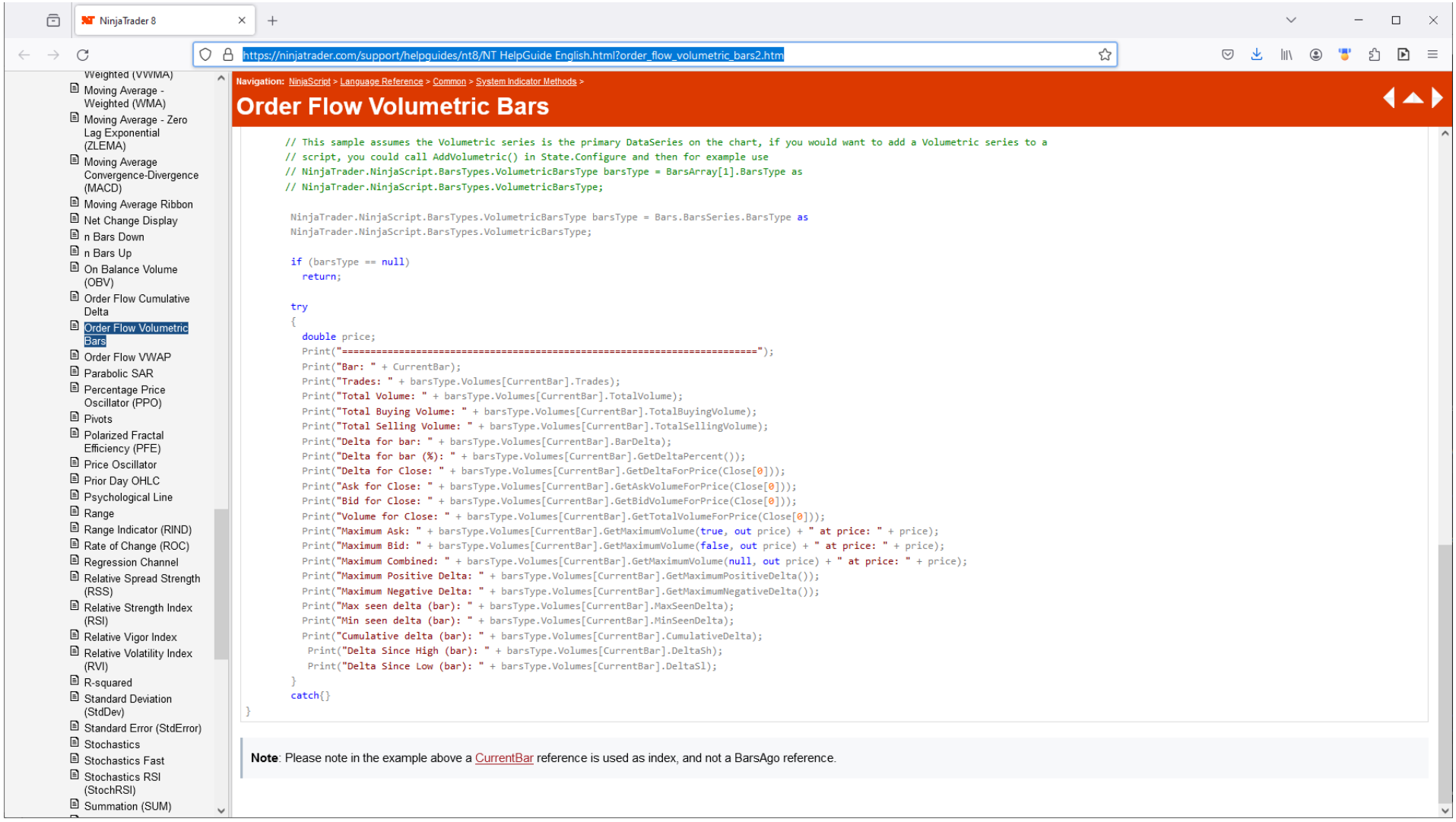
NinjaTrader 8
https://ninjatrader.com/support/helpguides/nt8/NT HelpGuide English.html?order_flow_volumetric_bars2.htm
Weighted (VWMA)
Moving Average - Weighted (WMA)
Moving Average - Zero Lag Exponential (ZLEMA)
Moving Average Convergence-Divergence (MACD)
Moving Average Ribbon
Net Change Display
n Bars Down
n Bars Up
On Balance Volume (OBV)
Order Flow Cumulative Delta
Order Flow Volumetric Bars
Order Flow VWAP
Parabolic SAR
Percentage Price Oscillator (PPO)
Pivots
Polarized Fractal Efficiency (PFE)
Price Oscillator
Prior Day OHLC
Psychological Line
Range
Range Indicator (RIND)
Rate of Change (ROC)
Regression Channel
Relative Spread Strength (RSS)
Relative Strength Index (RSI)
Relative Vigor Index
Relative Volatility Index (RVI)
R-squared
Standard Deviation (StdDev)
Standard Error (StdError)
Stochastics
Stochastics Fast
Stochastics RSI (StochRSI)
Summation (SUM)
Navigation: NinjaScript > Language Reference > Common > System Indicator Methods >
Order Flow Volumetric Bars
// This sample assumes the Volumetric series is the primary DataSeries on the chart, if you would want to add a Volumetric series to a
// script, you could call AddVolumetric() in State.Configure and then for example use
// NinjaTrader.NinjaScript.BarsTypes.VolumetricBarsType barsType = BarsArray[1].BarsType as
// NinjaTrader.NinjaScript.BarsTypes.VolumetricBarsType;
NinjaTrader.NinjaScript.BarsTypes.VolumetricBarsType barsType = Bars.BarsSeries.BarsType as
NinjaTrader.NinjaScript.BarsTypes.VolumetricBarsType;
if (barsType == null)
return;
try
{
double price;
Print("================================================================================");
Print("Bar: " + CurrentBar);
Print("Trades: " + barsType.Volumes[CurrentBar].Trades);
Print("Total Volume: " + barsType.Volumes[CurrentBar].TotalVolume);
Print("Total Buying Volume: " + barsType.Volumes[CurrentBar].TotalBuyingVolume);
Print("Total Selling Volume: " + barsType.Volumes[CurrentBar].TotalSellingVolume);
Print("Delta for bar: " + barsType.Volumes[CurrentBar].BarDelta);
Print("Delta for bar (%): " + barsType.Volumes[CurrentBar].GetDeltaPercent());
Print("Delta for Close: " + barsType.Volumes[CurrentBar].GetDeltaForPrice(Close[0]));
Print("Ask for Close: " + barsType.Volumes[CurrentBar].GetAskVolumeForPrice(Close[0]));
Print("Bid for Close: " + barsType.Volumes[CurrentBar].GetBidVolumeForPrice(Close[0]));
Print("Volume for Close: " + barsType.Volumes[CurrentBar].GetTotalVolumeForPrice(Close[0]));
Print("Maximum Ask: " + barsType.Volumes[CurrentBar].GetMaximumVolume(true, out price) + " at price: " + price);
Print("Maximum Bid: " + barsType.Volumes[CurrentBar].GetMaximumVolume(false, out price) + " at price: " + price);
Print("Maximum Combined: " + barsType.Volumes[CurrentBar].GetMaximumVolume(null, out price) + " at price: " + price);
Print("Maximum Positive Delta: " + barsType.Volumes[CurrentBar].GetMaximumPositiveDelta());
Print("Maximum Negative Delta: " + barsType.Volumes[CurrentBar].GetMaximumNegativeDelta());
Print("Max seen delta (bar): " + barsType.Volumes[CurrentBar].MaxSeenDelta);
Print("Min seen delta (bar): " + barsType.Volumes[CurrentBar].MinSeenDelta);
Print("Cumulative delta (bar): " + barsType.Volumes[CurrentBar].CumulativeDelta);
Print("Delta Since High (bar): " + barsType.Volumes[CurrentBar].DeltaSh);
Print("Delta Since Low (bar): " + barsType.Volumes[CurrentBar].DeltaSl);
}
catch{}
}
Note: Please note in the example above a CurrentBar reference is used as index, and not a BarsAgo reference.