

```

#region Using declarations
using System;
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;
using System.Windows.Media;
using NinjaTrader.Cbi;
using NinjaTrader.Gui;
using NinjaTrader.Gui.Chart;
using NinjaTrader.Data;
using NinjaTrader.NinjaScript;
using NinjaTrader.NinjaScript.DrawingTools;
#endregion

namespace NinjaTrader.NinjaScript.Indicators
{
    public class SessionRangeTracker : Indicator
    {
        private double currentHigh = double.MinValue;
        private double currentLow = double.MaxValue;

        protected override void OnStateChange()
        {
            if (State == State.SetDefaults)
            {
                Description           = "Tracks and plots current session High, Low, and Midpoint levels.";
                Name                   = "SessionRangeTracker";
                Calculate               = Calculate.OnBarClose;
                IsOverlay               = true;
                DisplayInDataBox       = true;
                DrawHorizontalGridLines = true;
                DrawVerticalGridLines  = true;
                PaintPriceMarkers       = true;
                ScaleJustification      = NinjaTrader.Gui.Chart.ScaleJustification.Right;
                IsSuspendedWhileInactive = true;

                AddPlot(new Stroke(Brushes.Green, 2), PlotStyle.Line, "Session High");
                AddPlot(new Stroke(Brushes.Red, 2), PlotStyle.Line, "Session Low");
                AddPlot(new Stroke(Brushes.Gold, 1), PlotStyle.Dash, "Session Mid");
            }
        }

        protected override void OnBarUpdate()
        {

```

```

if (CurrentBar < 1) return;

// Reset extremes on new trading session
if (Bars.IsFirstBarOfSession)
{
    currentHigh = High[0];
    currentLow = Low[0];
}
else
{
    currentHigh = Math.Max(currentHigh, High[0]);
    currentLow = Math.Min(currentLow, Low[0]);
}

Values[0][0] = currentHigh;
Values[1][0] = currentLow;
Values[2][0] = currentLow + ((currentHigh - currentLow) / 2.0);
}
}
}

```