

```
{
FILE:      GLOBAL RECEIVER

COMMENTS:  Demonstrates how to RECEIVE/GET Global Data and then process and/or print to other windows

AUTHOR:    Frederick Braden
EMAIL:     fbraden@yahoo.com
RELEASED:  07/23/2011  Version 1.0
```

You must format this indicator -

- right click THIS window,
- then select [Properties...],
- then select [Scaling] tab,
- set Axis [Scale On:] to "Same Axis as Underlying Data"
- then click [Ok]

USAGE: Add to any chart as an indicator. (Will print data in realtime to window)

Created from information available at the following link:
<https://www.tradestation.com/wiki/display/EasyLanguage/GlobalDictionary+Class>

```
}
{ code will use classes from these namespaces }
using elsystem.collections;           // Needed for creating Global Dictionary

inputs:
    TextColor(WHITE),                 // Default Text Color
    VertTextLocation(85);             // Vertical Text Position - valid setting is 0 to 95 percent from bottom of window

variables:
    TxtStr(""),                       // Text String
    TxtID(Text_New(date,time,last, " ")), // Text Object Reference
    // *** Global Variables ***
    Intrabarpersist double myTickVal(0), // Declared TICK and DATA TYPE
    Intrabarpersist double myTrinVal(0), // Declared TRIN and DATA TYPE
    string GDictName("myGDictName"),     // Declared Name must match SENDER
    GlobalDictionary myGDict(NULL);       // Declared Global Dictionary

    // -----
    // *** Retrieve Sender's global data ***
    // -----
    Method void ItemChg(elsystem.Object Sender, ItemProcessedEventArgs Args) // ItemChanged Event Handler
    Begin
        Switch(Args.Key) // Global dictionary KEY that was updated
        Begin
            Case "TickVal" : // Retrieve "TickVal" data
                myTickVal = myGDict.Items[Args.Key] astype double; // Get "TickVal" data ($TICK) and set DATA TYPE

            Case "TrinVal" : // Retrieve "TrinVal" data
                myTrinVal = myGDict.Items[Args.Key] astype double; // Get "TrinVal" data ($TRIN) and set DATA TYPE
        End;
        PlotOutput(); // Process data anytime Sender updates global dictionary
    End;
    // -----
```

```

// *****
// Start processing/calculating/analyzing/plotting or whatever makes you happy here...
// *****
Method void PlotOutput()                                // Process all data...
Begin
    If time > 829 AND time < 1501 Then                    // Only display during normal trading hours
        TxtStr = "TICK: " + NumToStr(myTickVal, 0) + " TRIN: " + NumToStr(myTrinVal, 2) // Build Text string
    Else
        TxtStr = "Non-Market Hours";                      // Build alternate Text sting

    // Set Text Location from right of chart
    Value0 = Text_FloatLocation(TxtID, VertTextLocation); // An easier to use custom fuction for locating text
    Text_SetString(TxtID, TxtStr);                        // Display Text to window
    Text_SetColor(TxtID, TextColor);                      // Set Text Color
End;
// *****

// -----
// *** Initialize Global Dictionary ***                  // This code should always be placed below all Methods
// -----
Once Begin
    myGDICT = GlobalDictionary.Create(TRUE, GDICTName); // Create Named Instance (Name must match SENDER)
    myGDICT.ItemChanged += ItemChg;                    // Assigns Method ItemChg() to Event Handler
End;
// -----

```