

Welcome To VTL Course



VertexFX Trading Language Course Hybrid Solutions

www.hybrid-solutions.com

Contents



Hot Tips



Introduction



Programming structure

.

.



VTL Client Script



VTL Server Script

Hot Tip

 If you think that creating scripts is too complicated or beyond your capacity, don't worry, this course is intended for those of us who are just starting out in programming, so it starts from zero.

Hot Tip

 These lessons have lots of examples, and we've simplified them as much as possible so that it's easy to learn VTL.

Hot Tip

 We recommend that you work your way through the lessons in the order in which they are provided, because at each level you will need the skills acquired in the previous ones.

Introduction

 VTL (VertexFX Trading Language) is a language related to Visual Basic that can only run through a host application (VTL Editor, in our case).

Using VTL , we can do almost anything imaginable with VertexFX ...

VTL Editor

To open the editor click on "Technical Analysis" tab then choose VTL Editor 10 :

≤ ≤ Q × Q Q ≈ Q	÷	-	100000 [SUPPORT] - SUPP	ORT@HS Dev	1754						x
Home View Charts Te	chnical Analysis Tick Chart	Plugins Help									O	otions *
D1 Dally Chart Period	bonacci Studies Ad	Drawings Tawings Symbols and Text +	Edit / Delete Objects	Oscillators Trends V	Volumes Custom dd Indicator	Edit / Delete Indicators	Display Darvas Boxes Darvas Boxes	Templates Templates	Full Screen Chart Full Screen	VTL VTL Editor 10 Strategy	2	
Navigator	₽ × AUD/USD-D1			₽ × Market	Watch							Ψ×
Account	AUD/USD D1 1.0441 1.0	456 1.0393 1.0398		Symbo	Bid Bid	Ask	k	Hiah	Low	Time	Expiry Date	-
Hs Dev Testing System				1.0499 → AUD 1.0497 → EUR 1.0457 → SILV 1.0457 → SILV 1.0457 → SILV 1.0456 → EUR 1.0415 → GBP 1.0415 → GBP 1.0415 → GBP 1.0373 → USD 1.0373 → USD 1.0373 → EUR 1.0373 → EUR 1.0373 → EUR	//USD 1.0398 //USD 1.2805 //USD 1.5131 //ER 28.48 //CHF 1.2196 //JPY 120.77 //GBP 0.8461 /JPY 142.69 D 1596.7 //CAD 1.0162 //AUD 1.2311 //CAD 1.3014	1.0 1.0 1.2 1.5 28. 1.2 1.2 0.8 142 0.8 0 155 1.0 1.2 1.3 0.1 1.2 0.8 142 0.8 0 155 1.0 1.2 0.8 0.8 1.2 0.8 0.8 1.2 0.8 0.8 1.2 0.8 0.8 1.2 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8	x 1 1403 1807 134 201 0.82 1463 2.76 1364 97.30 1167 1313 1016 02	1.0457 1.0457 1.2824 1.5176 28.84 1.2202 120.95 0.8476 142.94 0.8385 1607.80 1.0174 1.2317 1.3036 00.20	1.0397 1.2756 1.5112 28.47 1.2176 119.75 0.8416 142.21 1.08352 1595.39 1.0149 1.2225 1.2971	09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22 09:22		
	18 Mar 00:00 21 M	lar 00:00	26 Mar 00:00	Hark								_
Balance: 14,698.00 Flt P/L: 0.00 Equits	y: 14,698.00 Used Margin: 0.00 Incel Selected Orders) Free Margin: 14,69	8.00 Margin Lev	el: 100.00%								
Ticket Date/Time	Type	Amount	Symbol	Open Price	Current Price	S/L	T/P		Comm	Interest	P/L	
P											0.00	
2013-02-27 07:47:57	Buy Limit	1.5	USD/CHF	0.5000	0.9527							_
# Trade (0/1) Net Trade History (0) J	ournal (Mailbox (SUPPORT) (Alert	s_vStore				1.4				(
Dev plattorm This is our public with any private news Private	c dealing room news for Cl News: Thanks for J	ients Dea using our platform	nling Room Nev We w	vs: We vill inform you b	Icome to HS De	ev platform	Thi	s is our pu ate News:	blic dealing r Tha	oom news for Clie nks for using our p	nts De latform	aling F We s
Filvate news Filvate	ine ins. Interior interior	asing our platform		you i	iere with any pi	wate news	- 110	ate news.	1114	ma for using our p		

VTL Editor

(VII)) 🖡 🥥		-	5 💭 (•										VTL I	ditor .		-			-									L	- 0	23
	Home	Tool	s He	lp																												
+ New	Open	Make VTX	Save All	Save File	Save As	Close	e Prin Previ	int F	Print	Login Conr	Logout	Copy) of Cut	t Pas	te U	ndo I	Redo	97 Find	Replace	Select	t Stop		/L ompile ebug	Run	E) xit xit						
																										Nav	igator					₽ ×
																											VIL Scrip ↓ Loca ↓	IVTL Scri IVTL Scri .ocal VTL .ocal VTX er VTL Sc /TL Serve Saved Loc	pts Scripts Files r ally			
Errors l	.ist																															ą ×
Error N	0		E	ror Desci	iption																								Line Nu	mber		
Errors	List vSt	ore																														
						1			•	Γο	ope	en	the	e h	elr	5 F	Pre	SS	; (F	-1)		28/03	/2013 :	3:18:48 F	PM	Line: 1	Col: 1	CA	NUM	SCRL (Connect	ted

VTL script types

• Client VTL Scripts

Scripts saved only and work in client computer .

- Work only while client terminal is running.
- Visual Basic like Syntax and technology.

Server VTL Scripts

- Scripts saved and run in server side.
- Also it's still running even if you closed client terminal.
- Visual Basic.Net (VB.Net) like Syntax and technology.

VTL script creation

• Select "Home" tap , then press New.

VIC) 🖡 🚄 🖻	N #	- db -	5 JI	0)						-		-			VTL	Editor @	[HASSA	N]			12				_							
\square	Home	Tools	He	lp																													
+ New	Open N	Vake VTX	Save All	Save	Save As	Clos	se F	Print Preview	Print	it I	Login	Logout	Co	ру	Cut	Paste	Und	Redo	Fin	d Repla	ice S	Select	Stop	Comp	ile Ru	>	Exit						
				File							Conne	ection					Cli	pboard						Debu	g		Exit						
U	•																												ator L Scripts Local V Loc Loc Server N Server N Sav	TL Scrip Sal VTL S Sal VTX F VTL Scri L Server ved Loca	ts iccripts pts ally		4 >
Errors L	ist																																д >
Error No)		En	or Desc	ription																										Line Nur	mber	
		1																															
Errors	List vStore	/																					28	3/03/20	13 3:58:0	6 PM	Line	:1C	ol: 1	CAP	NUM S	CRL Con	nected

VTL script creation

• Type file name then select type that would you like with click on specified button .

Create New Co	de	le l	×
Parent Node			-
File Name	/TL Document		
New Local	VTL Script	New Server VTL Sc	ript

VTL Editor

	VIL Editor @ [HASSAN] - [VIL <loc>*]</loc>										
Home Tools Help											
📄 🗐 🛃 🐒 🐌 🔔 🔊 🕯	/ 🚨 🗳 🍫 👘 🍎 🦰 🧠 🎭 🖺 🔲 🕫										
New Open Make Save Save Save Close Print Print VTX All As All Preview	Login Logout Copy Cut Paste Undo Redo Find Replace Select Stop Compile	Run Exit									
File	Connection Clipboard Debug	Exit									
V L VTL <loc>*</loc>		↓ ▷ × Navigator ₽ ×									
Main		VIL VTL Scripts									
Main Wain											
Errors List		₽ ×									
Error No Error Description		Line Number									
Errors List vStore	Errors List vStore										
Welcome to VTL Editor	31/03/2013	11:06:02 AM Line: 5 Col: 89 CAP NUM SCRL Connected									

• Remember ,To open the help Press (F1).

Client VTL script initial sub procedure and events

- main()
 - The main function is the entry point for any Script program.

OnTick(SymbolName)

- The function is generated when a new tick is received for any symbol.
- Onlnit()
 - The function is generated when initializing script.
- OnDelnit()
 - The function is generated when deinitializing script.
- OnTimer()
 - The OnTimer function called when the timer event occurs.

Client VTL script Initial sub procedure and events

- OnCalculate(symbol, symbolPeriod, openVal, highVal, lowVal, closeVal)
 - The OnCalculate function called when a new candle received

AllDataLoaded()

• Raise when all data are loaded after Login

– AccountSelected()

- Raise when account selected
- OnOrderTrade(actionType ,orderID , returnValue)
 - Raise when the server returns order result
- OnPositionTrade (actionType ,ticketID)
 - Raise when the server returns position result

Variables

Variables make it possible to store all sorts of information.

Declaring these variables is not absolutely necessary, but it is recommended. It makes it easier to find them, can help resolve problems, etc.
In short, it's a good idea to get in the habit of declaring variables correctly.

Variables

 A variables type indicates the nature of its contents (text, numbers, etc.).

 If a variable is declared at the beginning of a procedure (Sub), it can only be used within this same procedure ,The value of the variable will not be maintained after the execution of the procedure.

Sub variables()

Dim my_variable
 'Assigning a value to the variable
 my_variable = 12
 'Displaying the value of my_variable in a MsgBox
 MsgBox my_variable
 End Sub

- This first line of code declares the variable (this is generally placed at the beginning of the procedure).
- Dim : declares the variable
- my_variable : the name chosen for this variable (no spaces allowed)

Arrays

 While variables only allow us to store one value each, arrays make it possible to store many values and they work almost exactly the same way.

- Important : the first cell in an array is numbered 0.

- Example :

'Sample 1 dimensional array declaration **Dim** array1(4)

'Assigning values to the 5 cells array1(0) = "Value of cell 0" array1(1) = "Value of cell 1" array1(2) = "Value of cell 2" array1(3) = "Value of cell 3"

Conditions

 Conditions are very useful in programming because they allow us to execute actions based on specific criteria.

But first, have a look at this list of comparison operators :

list of comparison operators :

- = is equal to
- <> is different than
 - < is less than
 - is less than or equal to
- > is greater than
- >= is greater than or equal to

And these other useful operators :

– AND

[condition1] AND [condition2] The two conditions must be true.

– OR

[condition1] OR [condition2] At least 1 of the 2 conditions must be true.

– NOT

NOT [condition1] The condition should be false.

• If

The most important conditional function is **If**, and now we'll take a look at how it works :

If [CONDITION HERE] Then ' => IF condition is validated, THEN
 'Instructions if true
Else ' => OTHERWISE
 'Instructions if false
End If

• Select

There is an alternative to using **If** with lots of **Elself** instructions the **Select** command, which is better suited to these sorts of situations.

select case [Parameter] case 1 MsgBox "One" case 2 MsgBox "Two" end Select



Loops

Loops make it possible to repeat instructions a number of times, which can save a lot of time.

– While

As long as the condition is true, the instructions in the loop will continue to be executed (careful not to create an infinite loop).

While [condition] 'Instructions Wend



Loops

– Do loop

In this case, the conditions can also be placed at the end of the **Do Loop** loop, which means that the instructions will definitely be executed at least once :

Do While [condition] 'Instructions **Loop**



Loops

– For

The For loop will be repeated here 5 times. For i = 1 To 5 'Instructions Next

It's possible to exit a **For** loop early by using the following instruction :

- Exit For

Exit a For loop



The other Exit instructions :

Exit Do
 Exit a Do Loop loop

Exit Sub
 Exit a procedure

Exit Function
 Exit a function



Procedures and Functions

 The main difference between Sub and Function is the value returned by the function.

Arguments

Arguments make it possible to use values from a procedure in a sub procedure (remember that by default, variables are only accessible from the procedure in which they are declared).

• ByRef

means that if a variable is submitted as an argument, its reference will be transmitted. In other words, if the variable is modified in the sub procedure, it will also be modified in the procedure that called the sub procedure.

• ByVal

Unlike **ByRef**, which transmits the reference (shortcut), **ByVal** transmits the value, which means that the value submitted as an argument will not be modified.

MsgBox

Until now, we have only used **MsgBox** dialog boxes to display information ;In this case, **MsgBox** is called with only one argument.

See code below :

MsgBox " Hello VTL ! "

We will now create a dialog box that asks us to confirm the deletion before the two instructions are carried out. Here are the three arguments that we will use : MsgBox([TEXT], [BUTTONS], [TITLE])

- Text : the dialog box text
- Buttons : choice of buttons (yes, no, cancel, etc.) + other options
- **Title** : dialog box title
 - Sub delete_B2()

If MsgBox("Do you wish to display chart symbol ?", vbYesNo, "Confirm") = 1 Then MsgBox ChartSymbol(0)

End If End Sub

• The different ways to use the second argument with MsgBox

Constant	Numerical Value	Description
vbOKOnly	0	ОК
vbOKCancel	1	OK Cancel
vbAbortRetryIgnore	2	Abort Retry Ignore
vbYesNoCancel	3	Yes No Cancel
vbYesNo	4	Yes No
vbRetryCancel	5	Retry Cancel
vbCritical	16	8
vbQuestion	32	0
vbExclamation	48	<u>A</u>
vbInformation	64	0
vbDefaultButton1	0	Default button : Button 1
vbDefaultButton2	256	Default button : Button 2
vbDefaultButton3	512	Default button : Button 3
vbApplicationModal	0	Forces the user to provide an answer before continuing to use Excel
vbSystemModal	4096	Forces the user to provide an answer before continuing to use any program on the computer (dialog box in foreground)

The values returned by MsgBox

Constant	Numerical Value	Button that corresponds to the numerical value
vbOK	1	ОК
vbCancel	2	Cancel
vbAbort	3	Abort
vbRetry	4	Retry
vbIgnore	5	Ignore
vbYes	6	Yes
vbNo	7	No

InputBox

InputBox asks the user to enter a value in a dialog box, for example :

Sub example() Dim result

result = InputBox("Text ?", "Title") 'The variable is assigned the value entered in the InputBox

If result <> "" Then 'If the value anything but "" the result is displayed MsgBox result End If End Sub

VTL script types

Remember ?

- Client VTL Scripts
 - Scripts saved only and work in your computer .
 - Work only while client terminal is running.
 - Visual Basic like Syntax and technology.

Server VTL Scripts

- Scripts saved and run in server side.
- Also it's still running even if you closed client terminal.
- Visual Basic.Net (VB.Net) like Syntax and technology.

Client VTL Scripts Docs

Please Click here

Server VTL Scripts Docs

Please Click here

Server VTL Scripts

Please Click here



To Read More Please Visit us at www.hybrid-solutions.com