

RELATED FILES

VBTab.zip

Description

Visual basic files related to this tutorial

INTRODUCTION

THIS TUTORIAL EXPLAINS HOW TO CREATE AND EMBED YOUR OWN ACTIVE X COMPONENT IN 3DCREATE USING VISUAL BASIC.

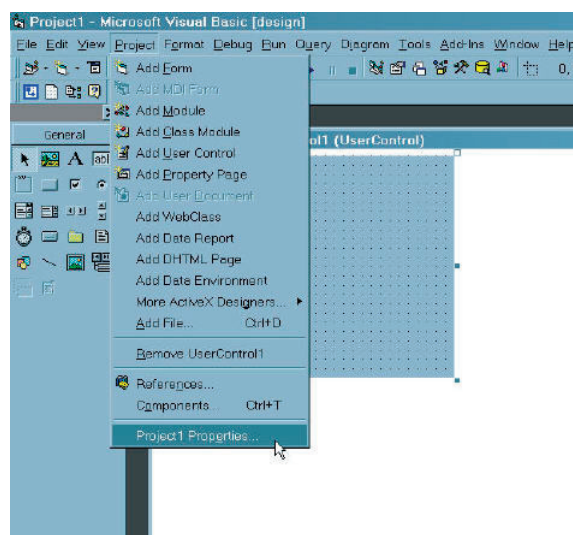
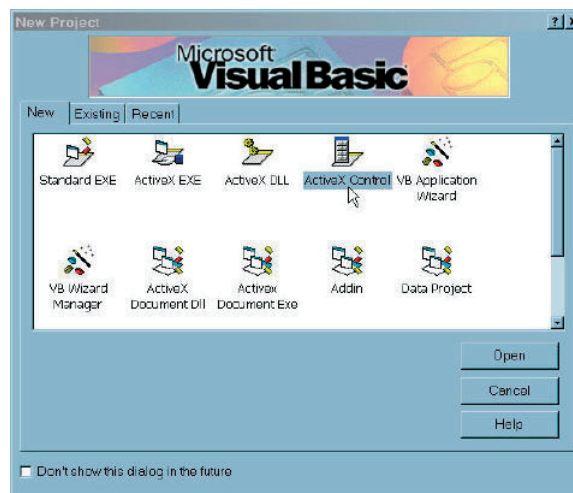
OBJECTIVES

THE OBJECTIVE OF THIS TUTORIAL IS TO CREATE A SIMPLE ACTIVE X COMPONENT AND EMBED IT INTO 3DCREATE.

CREATING THE ACTIVE X COMPONENT

1. Create the ActiveX component and give it a unique name

1.1 Start Visual Basic and create a new ActiveX control project.

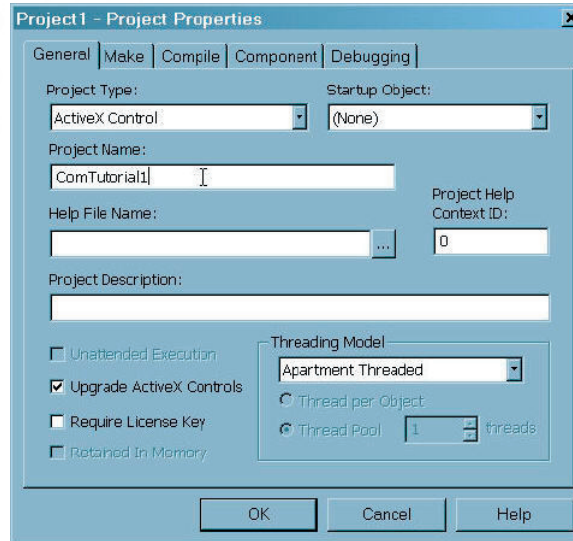


SUPPORT@VISUALCOMPONENTS.COM

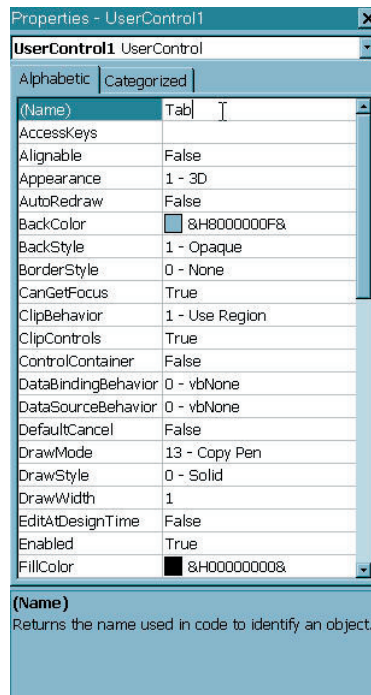
WWW.VISUALCOMPONENTS.COM

The Visual Components logo 3DCreate, 3DRealize and 3DVideo are registered trademarks of Visual Components Oy, Inc., registered in the USA, Europe and/or other countries. All other brand names, product names, or trademarks belong to their respective holders.

1.2 Change the project name to ComTutorial1, through 'Projects > Project1 Properties'.



1.3 Change the ActiveX component name to Tab, through the Properties window.



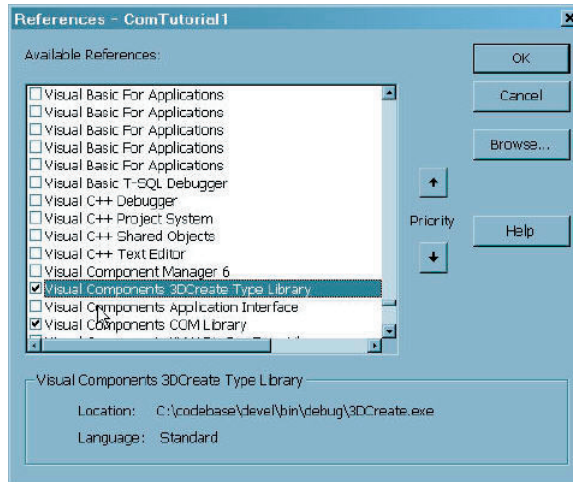
Now the component has a unique name, which will be used to identify it later.

2. Import the Visual Components COM Interface to your project

This step imports the *Visual Components COM interfaces* to your project and implements *IvcClient*, which is necessary for embedding ActiveX components to 3DCreate.

2.1 Import the Visual Components COM Interface type libraries to your project, using 'Project > References....' You need to check the following type libraries:

- Visual Components COM Library
- Visual Components 3DCreate Typ Library



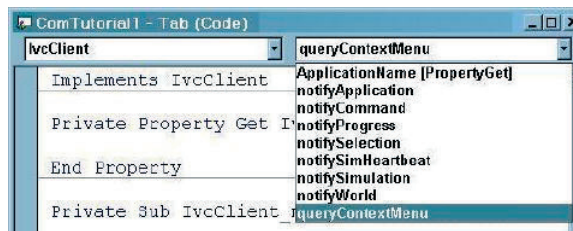
After this, you can access all the Visual Components COM interfaces in your project.

2.2 Open the code editor by selecting 'View > code' from the menu.

2.3 Make your ActiveX component implement *IvcClient* by adding the following line to the top of the code

```
Implements IvcClient
```

2.4 Now make sure to instantiate all methods in *IvcClient*. Select *IvcClient* from the left hand combo box at the top of the code window. Then select, in turn, all options from the right hand combo box.



After this, your code should look something like the following:

```

Implements IvcClient

Private Property Get IvcClient_ApplicationName() As String
End Property

Private Sub IvcClient_notifyApplication(ByVal AppReady As Boolean)
End Sub

Private Sub IvcClient_notifyCommand(command As vcCOM.IvcCommand,
    ByVal State As Long)
End Sub

Private Sub IvcClient_notifyProgress(ByVal Progress As Double)
End Sub

Private Sub IvcClient_notifySelection(selection As vcCOM.IvcSelection,
    ByVal SelectionTypeChange As Long)
End Sub

Private Sub IvcClient_notifySimHeartbeat(ByVal SimTime As Double)
End Sub

Private Sub IvcClient_notifySimulation(ByVal State As Long)
End Sub

Private Sub IvcClient_notifyWorld(Component As vcCOM.IvcComponent,
    ByVal Added As Boolean)
End Sub

Private Function IvcClient_queryContextMenu() As Boolean
End Function
    
```

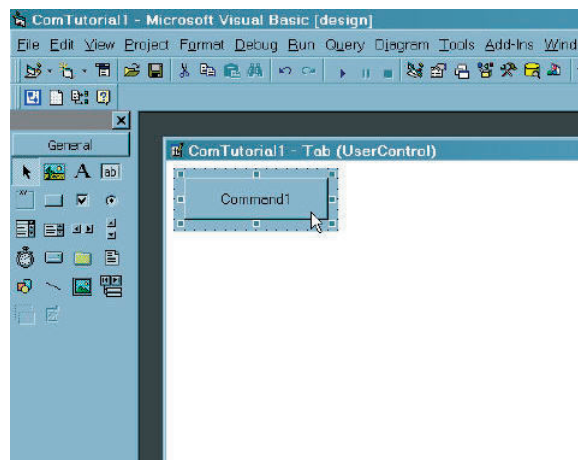
2.5 Add a string to be displayed as the title for your embedded ActiveX control. Insert the following line into *IvcClient_ApplicationName*.

```
IvcClient_ApplicationName = "COM Tutorial 1"
```

3. Add some functionality to the tab

In the *Object window*, add a button that creates a simple block component in the layout.

3.1 Create a button using the *CommandButton* control and double-click it to add an event handler.



3.1 Add the following code to the handler procedure:

```
Dim vcapp As IvcApplication
Dim cmp As IvcComponent
Dim feature As IvcFeature

Set vcapp = New vcc3DCreate
Set cmp = vcapp.createComponent("ComTutorial1")
Set feature = cmp.RootNode.RootFeature.createFeature("Block")
feature.setProperty "Width", 1000
feature.setProperty "Height", 1000
feature.setProperty "Length", 1000

cmp.rebuild
vcapp.render
```

This button will now create a component and add a 1000x1000x1000 unit box to it.

4. Save and compile the component

- 4.1 Select 'File > Save Project As...' to save the project, then select 'File > Make ComTutorial1.ocx...' to build and register your ActiveX component. After this, you may exit Visual Basic.

EMBEDDING THE ACTIVEX COMPONENT INTO 3DCREATE

To embed the ActiveX component into 3DCreate the *3DCreate.ini* file must be edited.

- Open up *3DCreate.ini* in a text editor. *3DCreate.ini* is in the same directory as *3DCreate.exe*.
- Modify the *.ini* file.
- Add the following line somewhere after the first line:

```
TabPanel15=ComTutorial1.Tab
```

- Save the file.
- The ActiveX component will be embedded the next time 3DCreate is started.

NOTE! The format of the component name is "Project name.Component name". Additional ActiveX components can be embedded simply by increasing the number after *TabPanel*.

