



Tutorial 250319-01	Autodesk Inventor	Addin	Vb.net	Arthur Knoors
--------------------	-------------------	-------	--------	---------------

Creating an Autodesk Inventor Add-in using Visual Studio Vb.net 2022 (Context Menu Add Finish Cancel Sketch.

Adding Buttons and controldefinitions:

```
Public Class StandardAddInServer
```

```
Implements Inventor.ApplicationAddInServer
```

```
Private m_inventorApp As Inventor.Application
```

```
Private WithEvents m_sampleButtonFin As ButtonDefinition
```

```
Private WithEvents m_sampleButtonCancel As ButtonDefinition
```

```
Dim NativeContrDefFin As ControlDefinition
```

```
Dim NativeContrDefCancel As ControlDefinition
```

```
Private guid As String = Nothing
```

```
Private WithEvents uie As UserInputEvents
```

Create two new control definitions and two new button definitions:

```
Public Sub Activate(AddInSiteObject As ApplicationAddInSite, FirstTime As Boolean) Implements ApplicationAddInServer.Activate
```

```
MsgBox("My First Add In Activated")
```

```
m_inventorApp = AddInSiteObject.Application
```

```
MsgBox(m_inventorApp.UserName)
```

```
uie = m_inventorApp.CommandManager.UserInputEvents
```

```
Try
```

```
Dim t As Type = GetType(MyInventorAddin.StandardAddInServer)
```

```
Dim customAttributes() As Object = t.GetCustomAttributes(GetType(GuidAttribute), False)
```

```
Dim guidAttribute As GuidAttribute = CType(customAttributes(0), GuidAttribute)
```

```
guid = "{" + guidAttribute.Value.ToString() + "}"
```

```
Catch
```

```
End Try
```



Tutorial 250319-01	Autodesk Inventor	Addin	Vb.net	Arthur Knoors
--------------------	-------------------	-------	--------	---------------

```
NativeContrDefFin = m_inventorApp.CommandManager.ControlDefinitions.Item("Finish2dSketchCmd")
```

```
NativeContrDefCancel = m_inventorApp.CommandManager.ControlDefinitions.Item("Cancel2dSketchCmd")
```

```
m_sampleButtonFin =
```

```
m_inventorApp.CommandManager.ControlDefinitions.AddButtonDefinition(NativeContrDefFin.DisplayName,  
"MyFirstButton",
```

```
inventor.CommandTypesEnum.kShapeEditCmdType,
```

```
guid, "My Description", "My Tool Tip Text",
```

```
NativeContrDefFin.StandardIcon, NativeContrDefFin.LargeIcon,  
ButtonDisplayEnum.kDisplayTextInLearningMode)
```

```
m_sampleButtonCancel =
```

```
m_inventorApp.CommandManager.ControlDefinitions.AddButtonDefinition(NativeContrDefCancel.DisplayName,  
"MyFirstButtonCancel",
```

```
inventor.CommandTypesEnum.kShapeEditCmdType,
```

```
guid, "My Description", "My Tool Tip Text",
```

```
NativeContrDefCancel.StandardIcon, NativeContrDefCancel.LargeIcon,  
ButtonDisplayEnum.kDisplayTextInLearningMode)
```

If FirstTime Then

AddToUserInterface()

End If

End Sub

Create the on execute sub for both the buttons.

```

StandardAddInServer.vb*  What's New?  Command.vb  ClsGetProperties.vb  MyInventorAddIn.addin  MyInventorAddin
VB MyInventorAddin
13 Dim NativeContrDefFin As ControlDefinition
14 Dim NativeContrDefCancel As ControlDefinition
15 Private guid As String = Nothing
16 Private WithEvents uie As UserInputEvents
17 Public Sub Activate(AddInSiteObject As ApplicationAddInSite, FirstTime As Boolean) Implements ApplicationAddInServer.Activate
18     MsgBox("My First Add In Activated")
19     m_inventorApp = AddInSiteObject.Application
20     MsgBox(m_inventorApp.UserName)
21     uie = m_inventorApp.CommandManager.UserInputEvents
    
```

```

StandardAddInServer.vb*  What's New?  Command.vb  ClsGetProperties.vb  MyInventorAddIn.addin  MyInventorAddin
VB MyInventorAddin
13 Dim NativeContrDefFin As ControlDefinition
14 Dim NativeContrDefCancel As ControlDefinition
15 Private guid As String = Nothing
16 Private WithEvents uie As UserInputEvents
17 Public Sub Activate(AddInSiteObject As ApplicationAddInSite, FirstTime As Boolean) Implements ApplicationAddInServer.Activate
18     MsgBox("My First Add In Activated")
    
```

These two subs should have been created.



Tutorial 250319-01	Autodesk Inventor	Addin	Vb.net	Arthur Knoors
--------------------	-------------------	-------	--------	---------------

*Private Sub m_sampleButton_OnExecute(Context As NameValueCollection) Handles
m_sampleButtonFin.OnExecute*

NativeContrDefFin.Execute()

End Sub

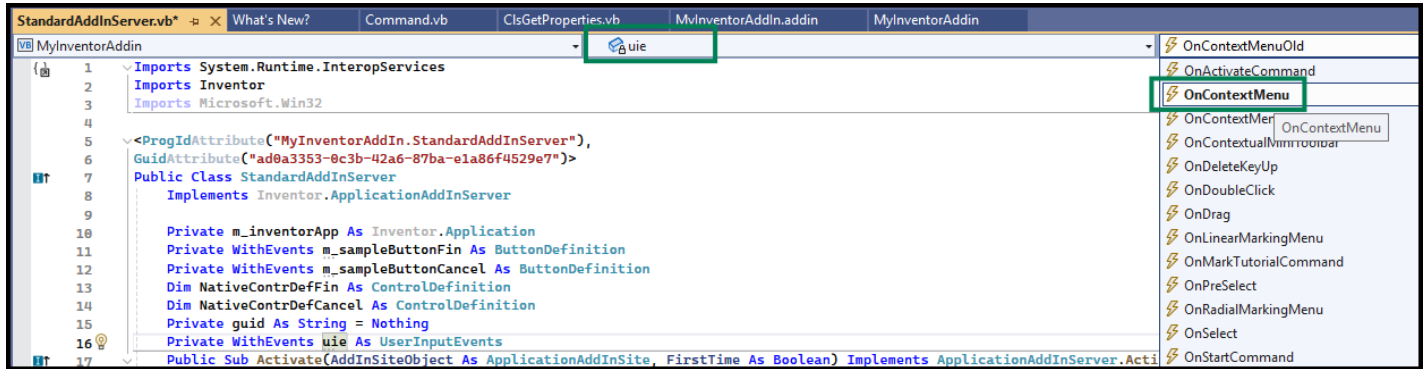
*Private Sub m_sampleButtonCancel_OnExecute(Context As NameValueCollection) Handles
m_sampleButtonCancel.OnExecute*

NativeContrDefCancel.Execute()

End Sub



Create the OnContextMenu sub:



The below sub should be created.

Put the highlighted code in it.

Private Sub uie_OnContextMenu(SelectionDevice As SelectionDeviceEnum, AdditionalInfo As NameValueCollection, CommandBar As CommandBar) Handles uie.OnContextMenu

If m_inventorApp.ActiveEnvironment.InternalName = "PMxPartSketchEnvironment" Then

If TypeOf m_inventorApp.ActiveEditObject Is Inventor.Sketch Then

'CommandBar.Controls

'm_sampleButton.DescriptionText = c.DescriptionText

'm_sampleButton.LargeIcon = c

'm_sampleButton.StandardIcon = cs

CommandBar.Controls.AddButton(m_sampleButtonCancel, 1)

CommandBar.Controls.AddButton(m_sampleButtonFin, 1)

End If

Else

'm_sampleButton.Enabled = False

End If

End Sub

In Inventor:



Congrats you have just added two existing commands or control definitions to your context menu of Inventor Part sketch.

- Finish Sketch
- Cancel Sketch

