You Can Do It! How to Write Programs for Inventor

Brian Ekins

Technical Evangelist

Class summary

A “taste” of what it’s like to use Inventor’s API.

Key learning objectives

At the end of this class, you will be able to:

- Learn how to create a simple Visual Basic .NET program.
- Discover the basic concepts of Inventor’s programming interface.
- Learn how to create a program that works with parameters.
- Learn how to create a program that works with iProperties.

Choosing a Language

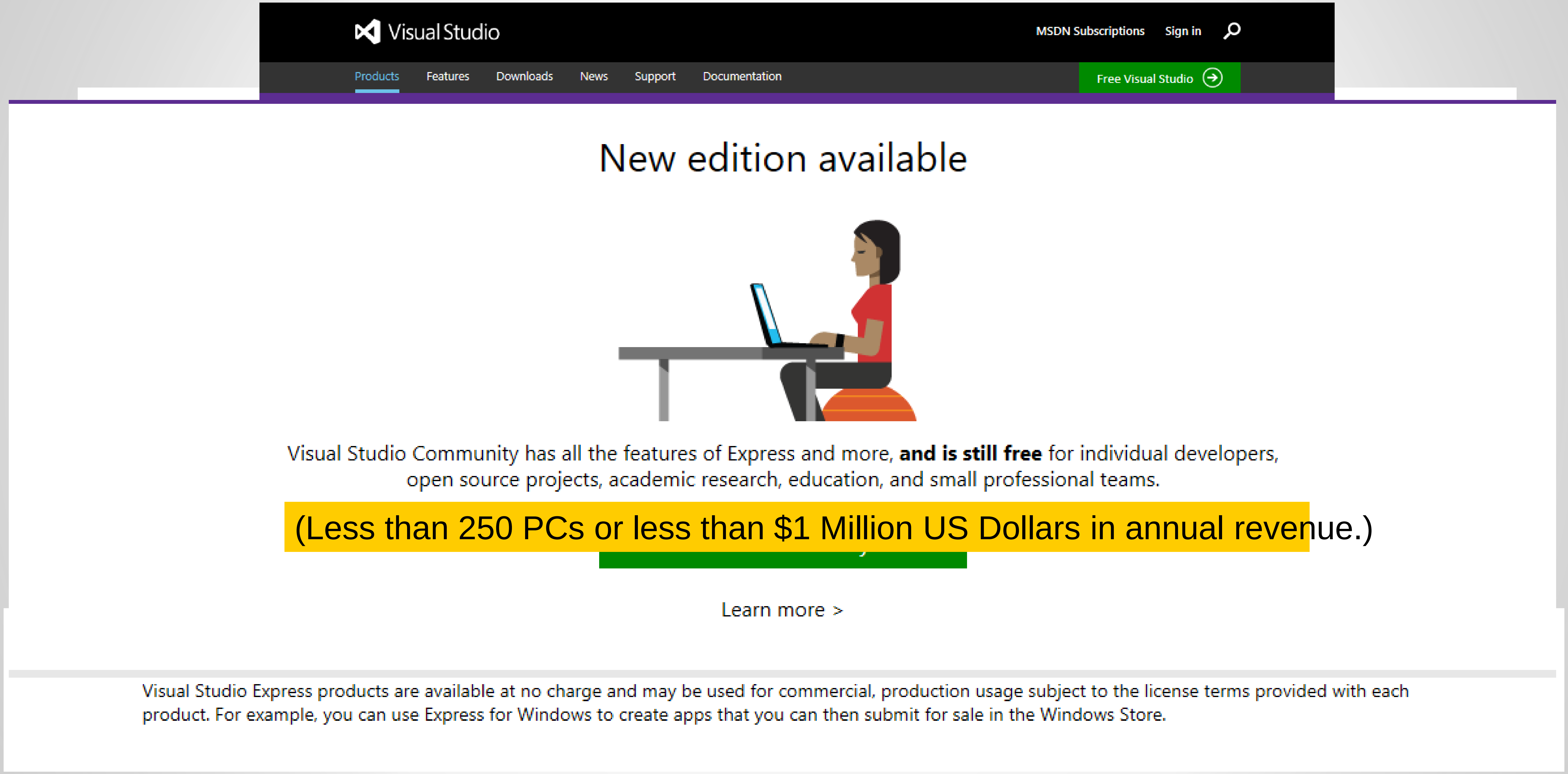
VBA (Visual Basic for Applications)

- Free
- Comes with Inventor
- Best for Inventor RAD (Rapid Application Development)
- Best for debugging Inventor code
- Old technology
- Limited to macros (<u>can’t create an add-in</u>)
- Difficult to share programs
- Limited system functionality
- Custom dialogs are very limited

Choosing a Language

- Visual Basic (VB.Net)
  - New language that is similar to VBA
  - Better language features
  - Very powerful library (.NET)
  - Creates exe's and dll's (needed to create add-ins)
  - Easy to share programs
  - Useful for programs besides customizing Inventor
  - Excellent custom dialogs
- C#
- C++

Visual Studio



New edition available

Visual Studio Community has all the features of Express and more, **and is still free** for individual developers, open source projects, academic research, education, and small professional teams.

(Less than 250 PCs or less than $1 Million US Dollars in annual revenue.)

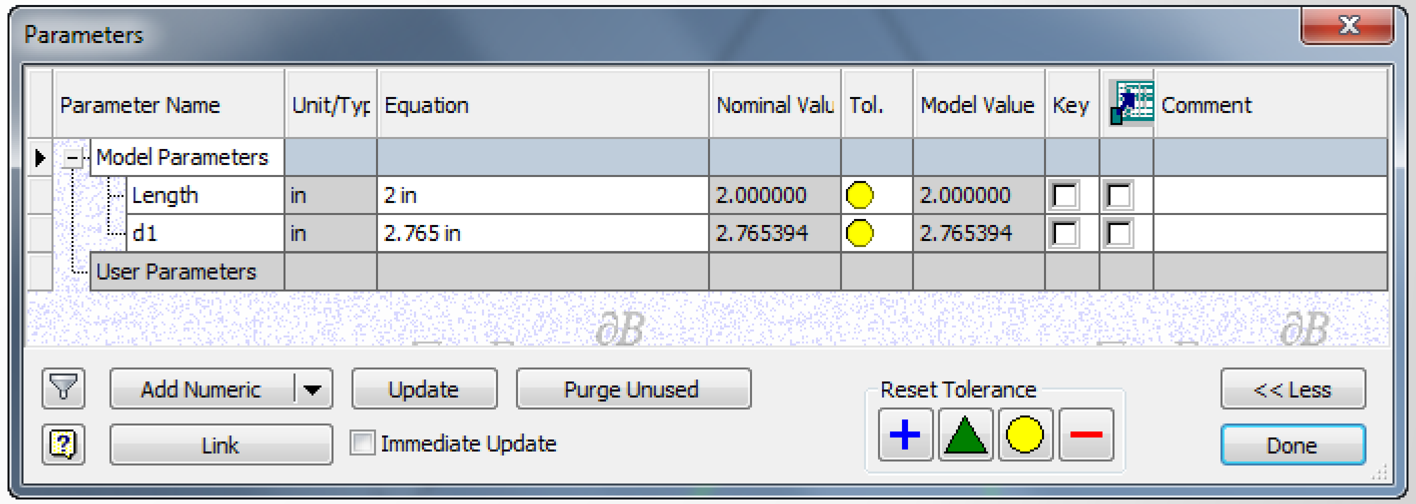
Learn more >

Visual Studio Express products are available at no charge and may be used for commercial, production usage subject to the license terms provided with each product. For example, you can use Express for Windows to create apps that you can then submit for sale in the Windows Store.

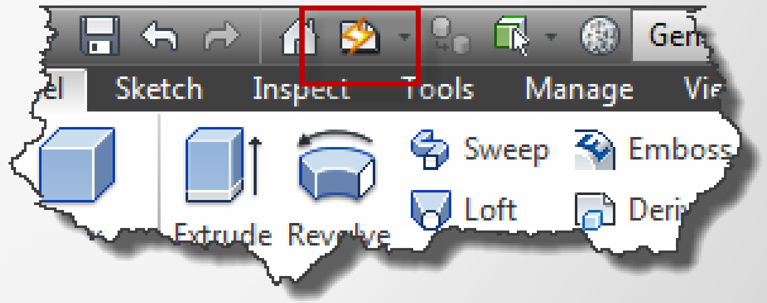
Exercise One

- What we're going to do:
  - Change the value of a parameter.
- Changing a parameter value in the user interface.

1. Start Inventor
2. Open the document
3. Run the Parameters command
4. Find the parameter in the dialog
5. Edit the value
6. Dismiss the dialog
7. Update the document

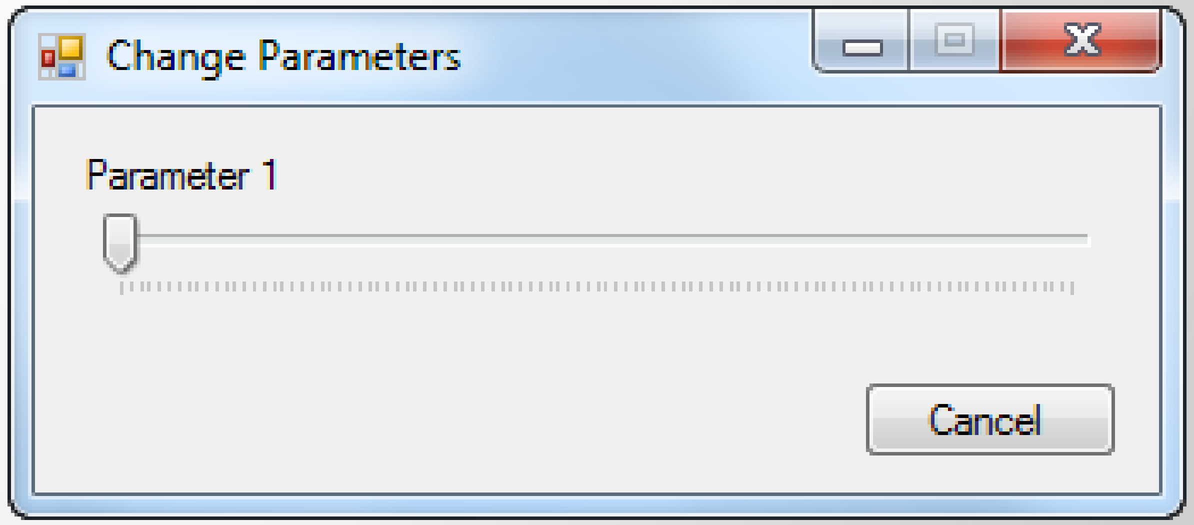


Parameter Name	Unit/Typ	Equation	Nominal Valu	Tol.	Model Value	Key		Comment
Model Parameters								
Length	in	2 in	2.000000	●	2.000000	☐	☐	
d1	in	2.765 in	2.765394	●	2.765394	☐	☐	
User Parameters								

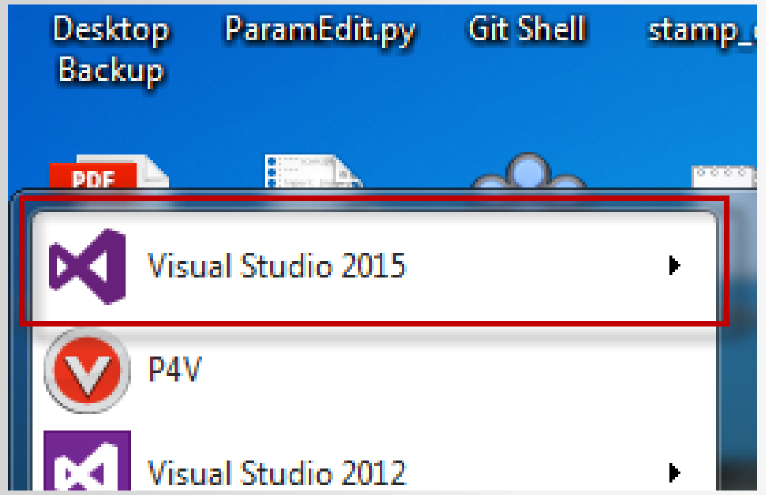


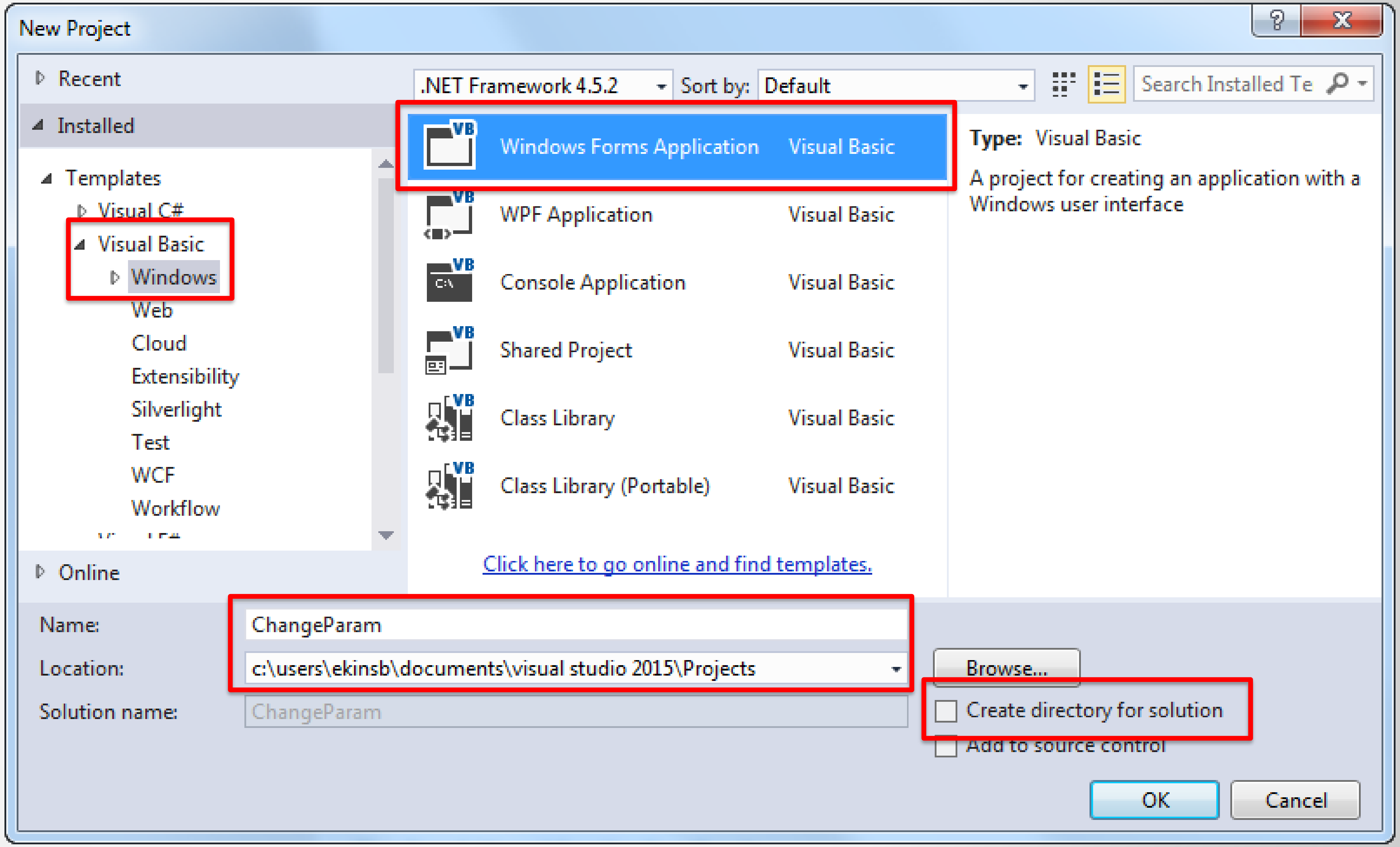
Exercise One

1. Connect to Inventor.
2. Get the active part document.
3. Watch for and react to the track bar being moved.
4. Change the value of the related parameter and update the model.
5. Dismiss when Cancel is clicked.



Step 1: Create a Project





Windows Forms Application	Visual Basic
WPF Application	Visual Basic
Console Application	Visual Basic
Shared Project	Visual Basic
Class Library	Visual Basic
Class Library (Portable)	Visual Basic

Type: Visual Basic

A project for creating an application with a Windows user interface

Click here to go online and find templates.

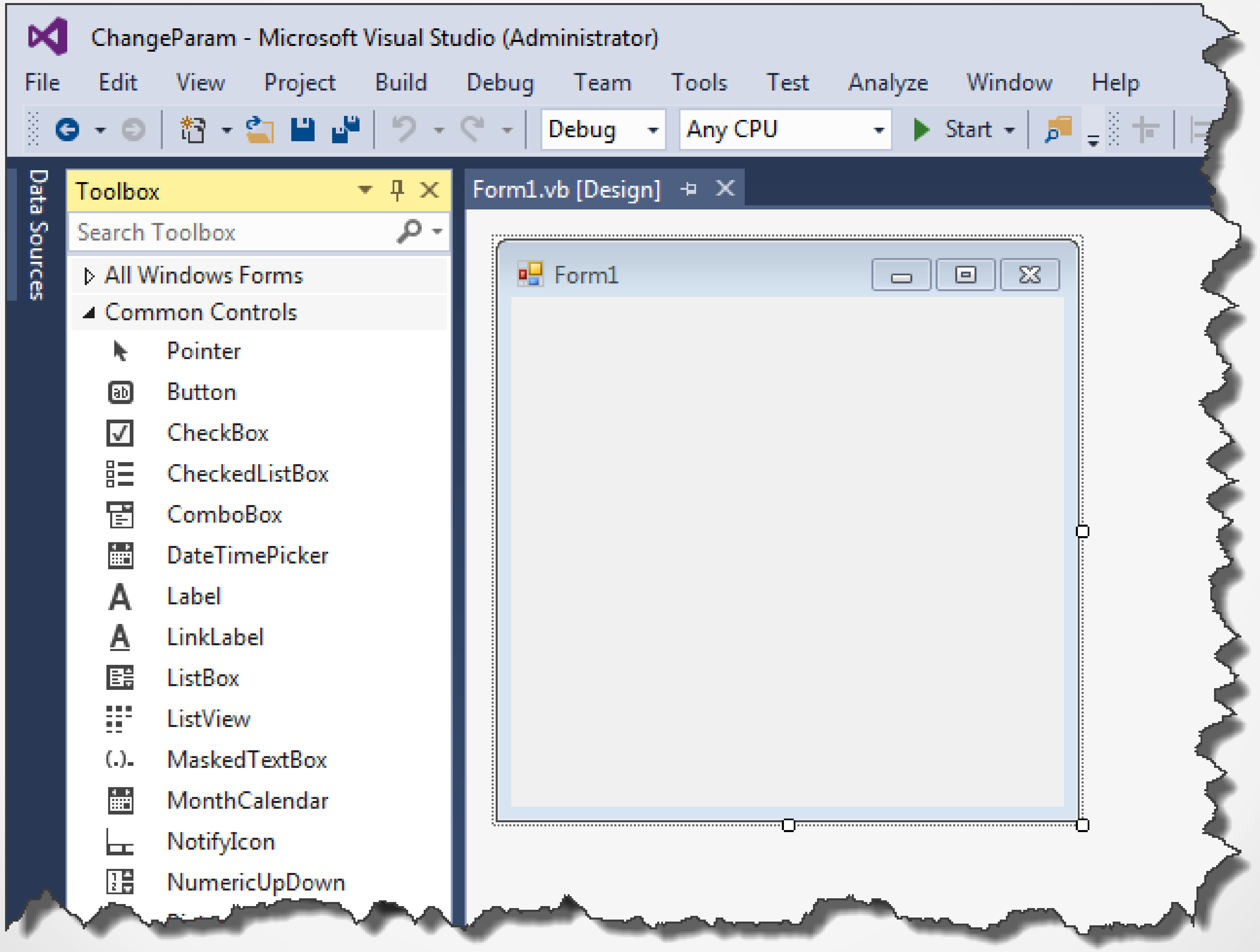
Name: ChangeParam

Location: c:\users\ekinsb\documents\visual studio 2015\Projects

Solution name: ChangeParam

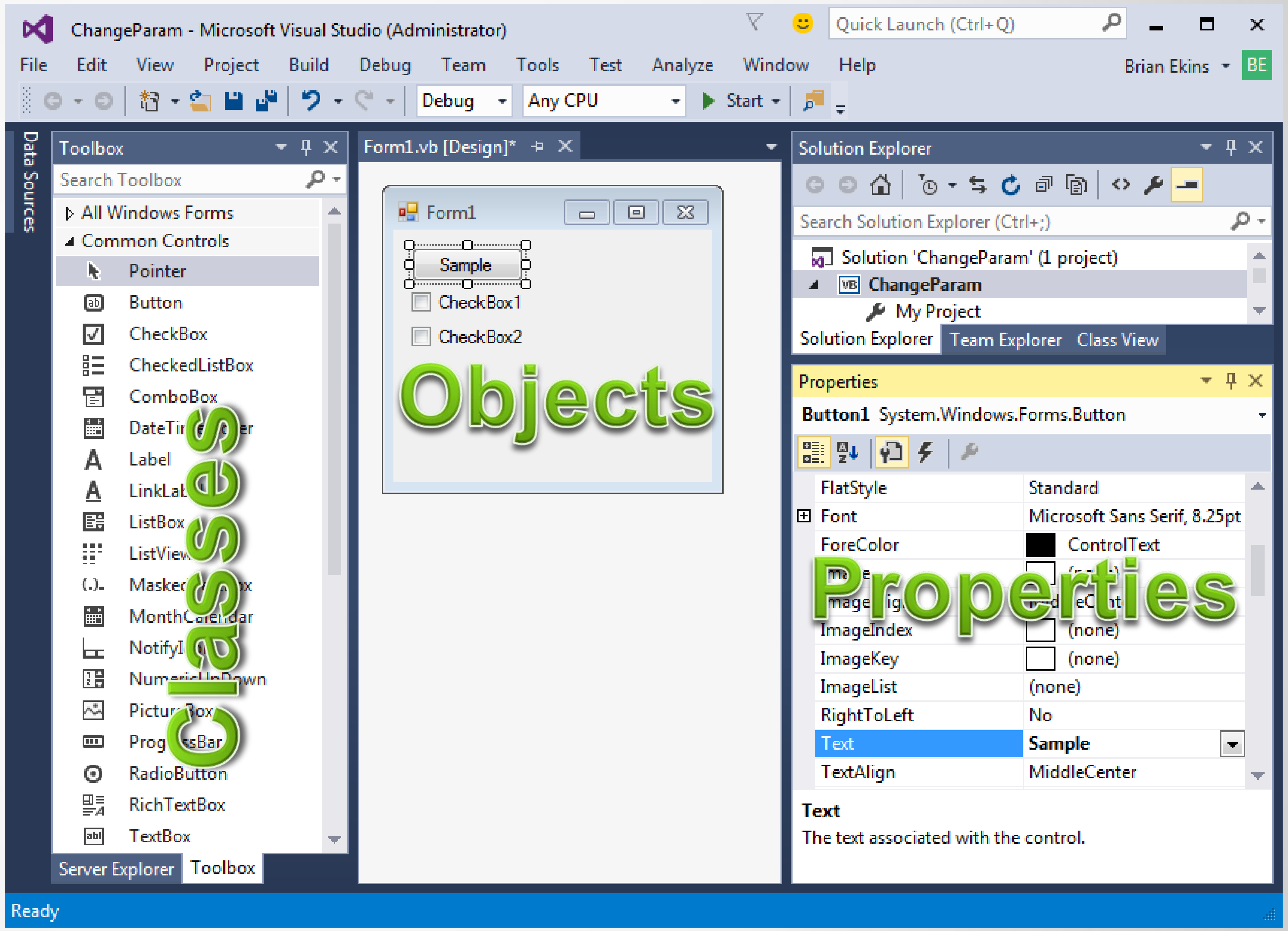
[ ] Create directory for solution

The “Visual” Part of Visual Basic



Classes, Objects, Methods, Properties, and Events

- Classes
- Objects
- Properties
  - Name, Text, etc.
- Methods
  - Refresh, BringToFront, etc.
- Events
  - Click, etc.

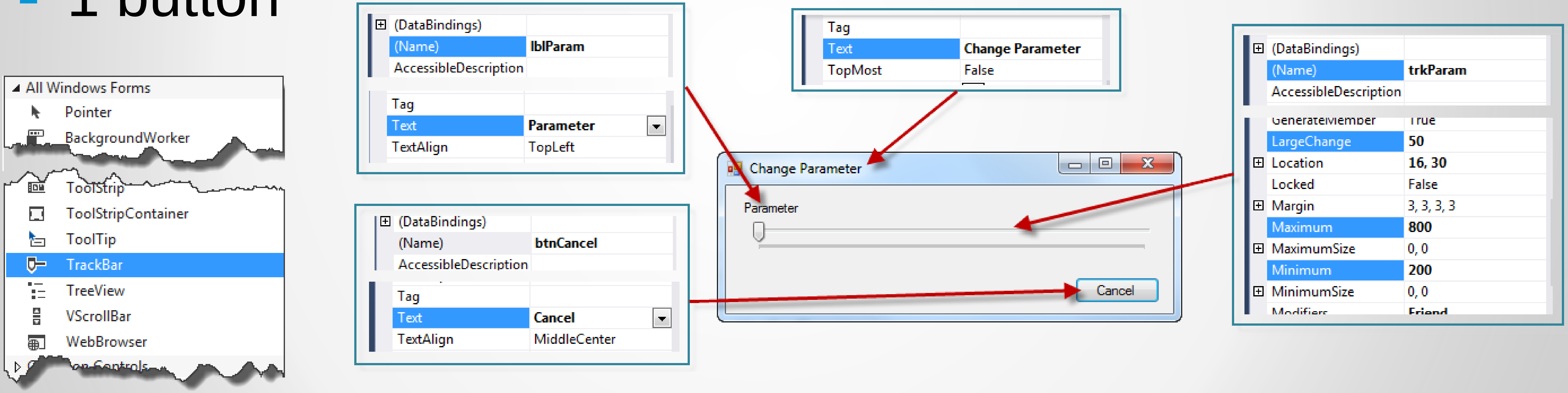


FlatStyle	Standard
Font	Microsoft Sans Serif, 8.25pt
ForeColor	ControlText
[illegible]	[illegible]
[illegible]	[illegible]
ImageIndex	(none)
ImageKey	(none)
ImageList	(none)
RightToLeft	No
Text	Sample
TextAlign	MiddleCenter

Text
The text associated with the control.

Step 2: Design your Dialog (The "Visual" Part of Programming)

- 1 TrackBar control
- 1 Label control
- 1 button



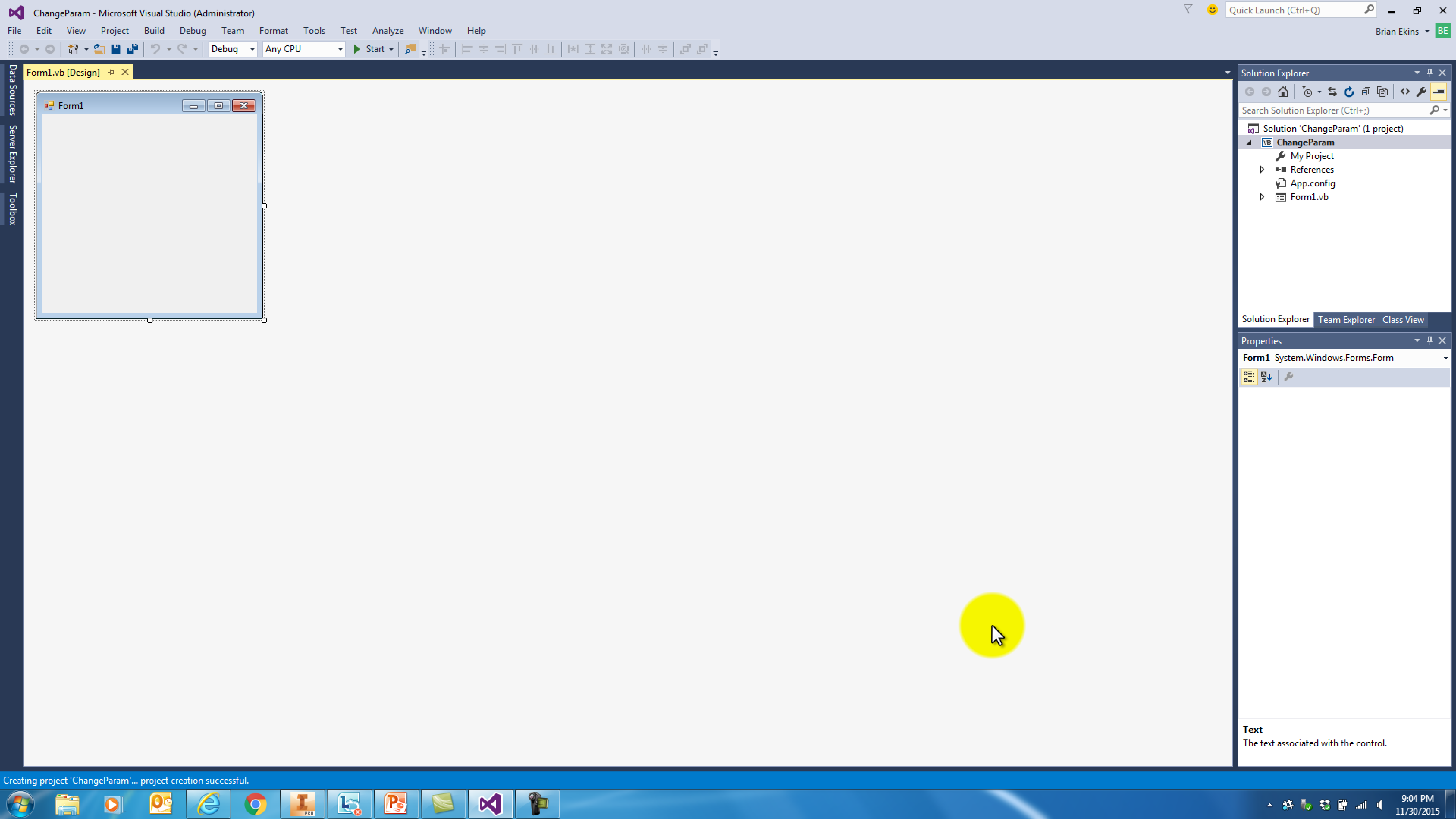
All Windows Forms
- Pointer
- BackgroundWorker
- ToolStrip
- ToolStripContainer
- ToolTip
- TrackBar
- TreeView
- VScrollBar
- WebBrowser

(DataBindings)	
(Name)	lblParam
AccessibleDescription	
Tag	
Text	Parameter
TextAlign	TopLeft

(DataBindings)	
(Name)	btnCancel
AccessibleDescription	
Tag	
Text	Cancel
TextAlign	MiddleCenter

Tag	
Text	Change Parameter
TopMost	False

(DataBindings)	
(Name)	trkParam
AccessibleDescription	
GenerateMember	True
LargeChange	50
Location	16, 30
Locked	False
Margin	3, 3, 3, 3
Maximum	800
MaximumSize	0, 0
Minimum	200
MinimumSize	0, 0
Modifiers	Friend



Creating project 'ChangeParam'... project creation successful.

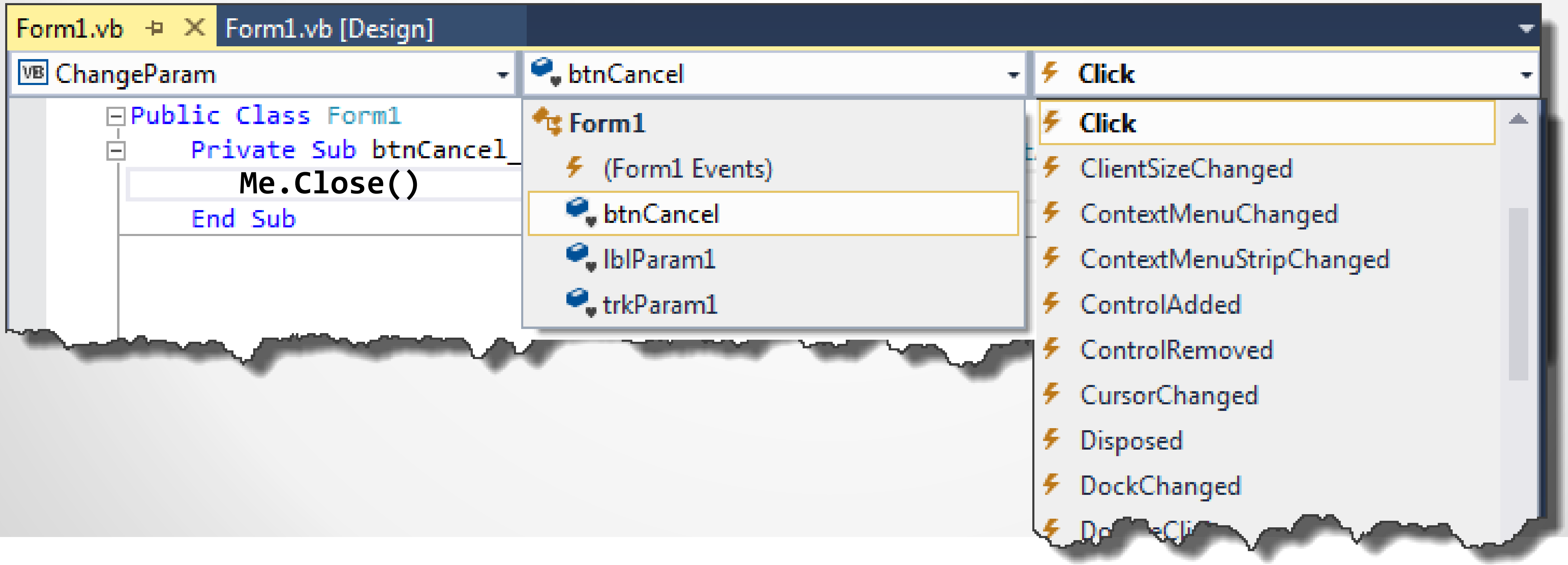
Step 3: Write the Code

- Where to put your code
  - Event Handlers
  - Functions and Subs
- How to use objects
  - Calling properties:
    - ObjectName.*PropertyName* = value

      lblParam.Text = "Some New Text"
    - value = Object.*PropertyName*
  - Calling methods:
    - ObjectName.*MethodName*(*argument1, argument2*)

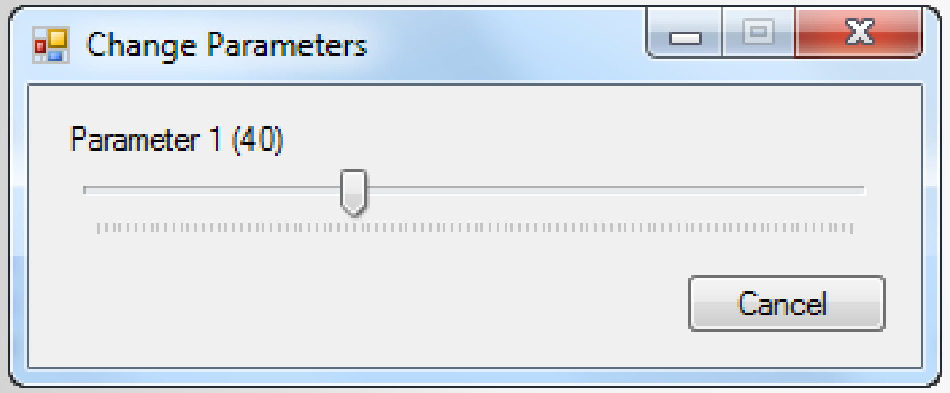
Control Event Handlers

- A Sub that VB calls when a certain action occurs.
- Double-click Cancel button to create handler.
- Handling the Cancel button click.
- Use drop-downs in code window to choose other events.



```
Public Class Form1
    Private Sub btnCancel_
        Me.Close()
    End Sub
```

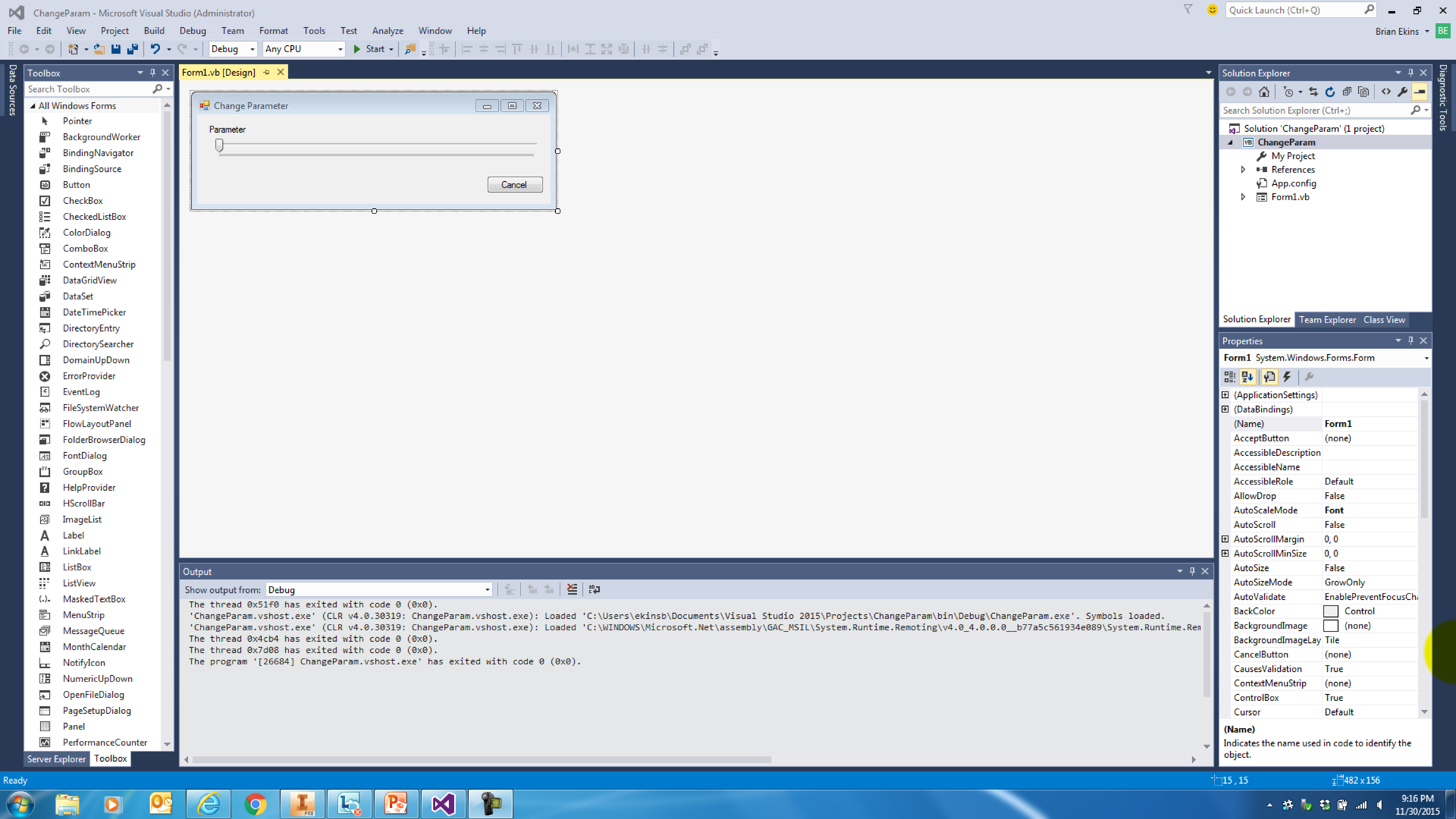
Handling the TrackBar Scroll event.



```
Private Sub trkParam_Scroll(sender As Object, e As EventArgs) Handles trkParam.Scroll
    lblParam.Text = "Parameter (" & trkParam.Value & ")"
End Sub
```

```
Private Sub trkParam_Scroll(sender As Object, e As EventArgs) Handles trkParam.Scroll
    Dim newText As String
    newText = "Parameter (" & trkParam.Value & ")"
    lblParam.Text = newText
End Sub
```

```
Private Sub trkParam_Scroll(sender As Object, e As EventArgs) Handles trkParam.Scroll
    Dim newText As String = "Parameter (" & trkParam.Value & ")"
    lblParam.Text = newText
End Sub
```



Show output from: Debug

```
The thread 0x51f0 has exited with code 0 (0x0).
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\Users\ekinsb\Documents\Visual Studio 2015\Projects\ChangeParam\bin\Debug\ChangeParam.exe'. Symbols loaded.
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Runtime.Remoting\v4.0_4.0.0.0__b77a5c561934e089\System.Runtime.Rem
The thread 0x4cb4 has exited with code 0 (0x0).
The thread 0x7d08 has exited with code 0 (0x0).
The program '[26684] ChangeParam.vshost.exe' has exited with code 0 (0x0).
```

Properties

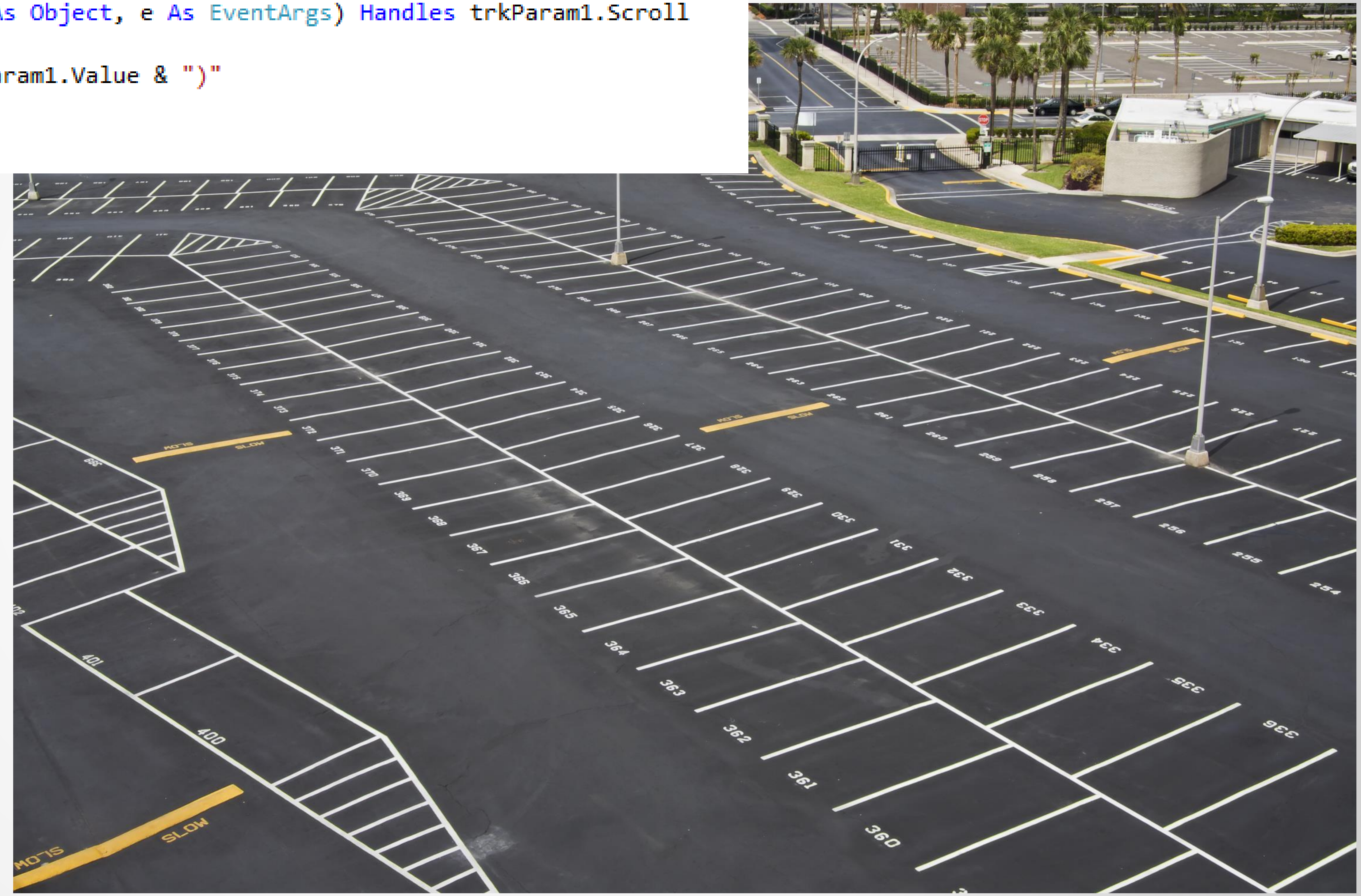
Form1 System.Windows.Forms.Form

(ApplicationSettings)	
(DataBindings)	
(Name)	Form1
AcceptButton	(none)
AccessibleDescription	
AccessibleName	
AccessibleRole	Default
AllowDrop	False
AutoScaleMode	Font
AutoScroll	False
AutoScrollMargin	0, 0
AutoScrollMinSize	0, 0
AutoSize	False
AutoSizeMode	GrowOnly
AutoValidate	EnablePreventFocusCha
BackColor	Control
BackgroundImage	(none)
BackgroundImageLay	Tile
CancelButton	(none)
CausesValidation	True
ContextMenuStrip	(none)
ControlBox	True
Cursor	Default

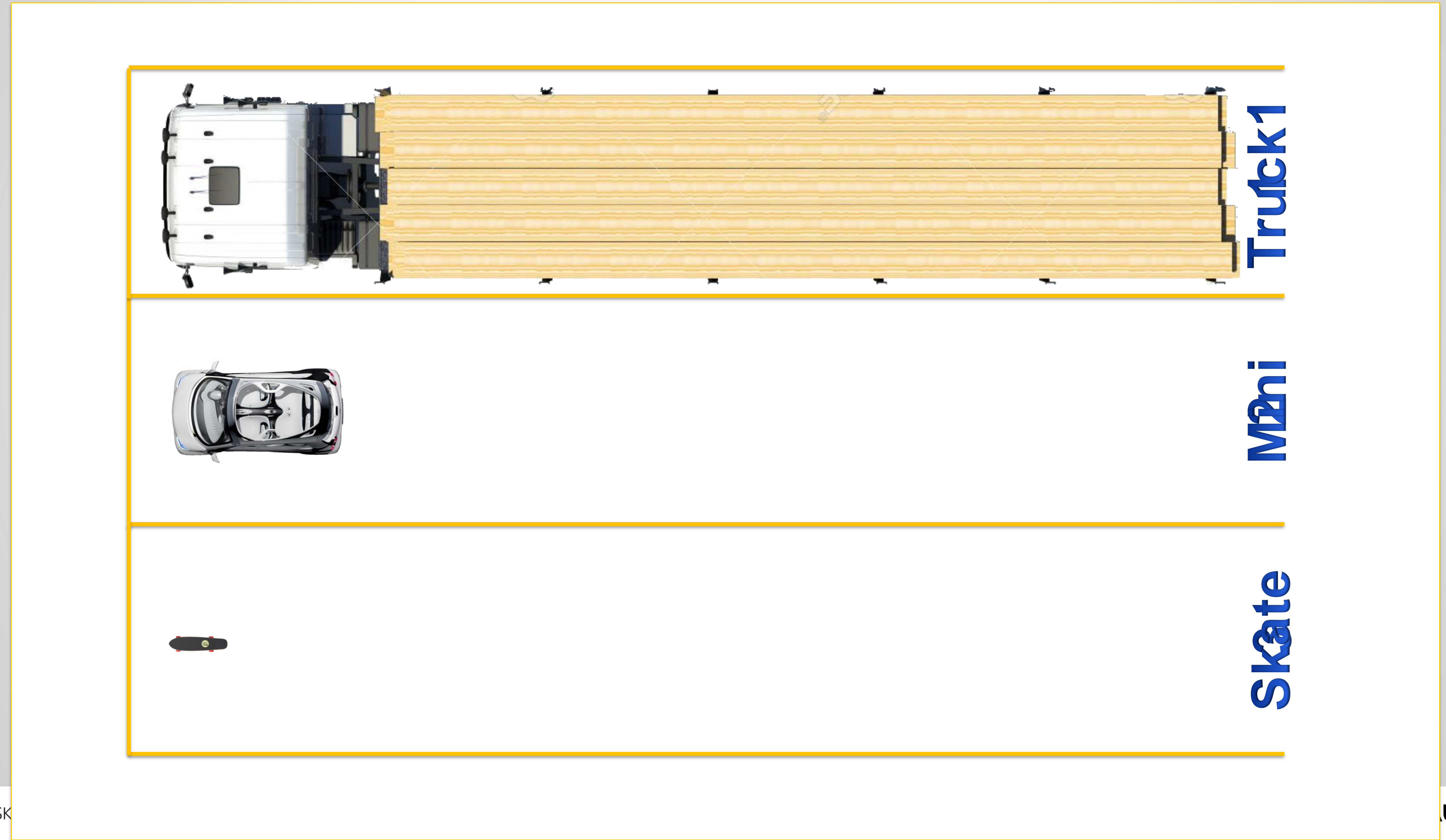
(Name)
Indicates the name used in code to identify the object.

Variables

```
Private Sub trkParam1_Scroll(sender As Object, e As EventArgs) Handles trkParam1.Scroll
    Dim newText As String
    newText = "Parameter 1 (" & trkParam1.Value & ")"
    lblParam1.Text = newText
End Sub
```

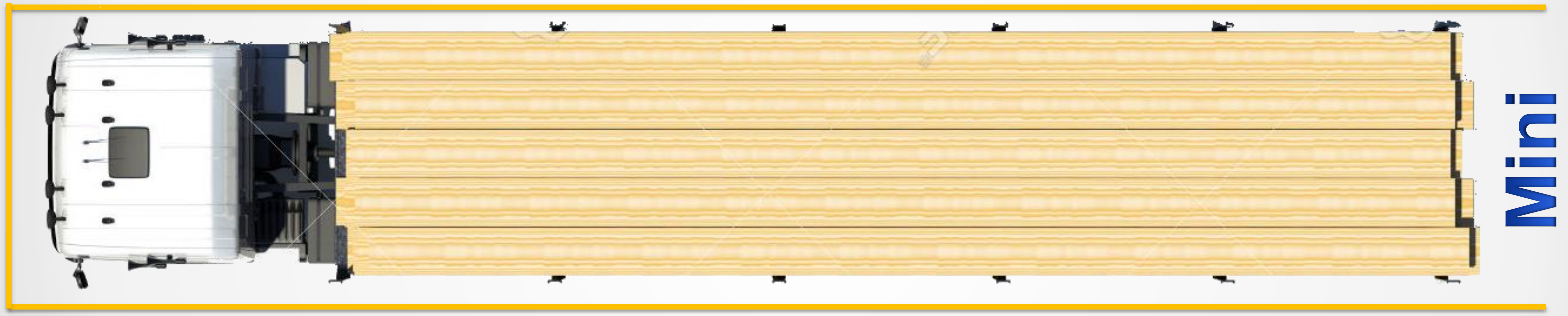


Variables



Variables

- `Dim Mini As Object`
- `Mini = CreateCar("My Car")`



- `Dim Mini As Car`
- `Mini = CreateCar("Wife's Car")`

Commonly Used Variable Types

- Double – For all floating point numbers (1.5, 3.14)

```
Dim angle As Double = 30.5
```

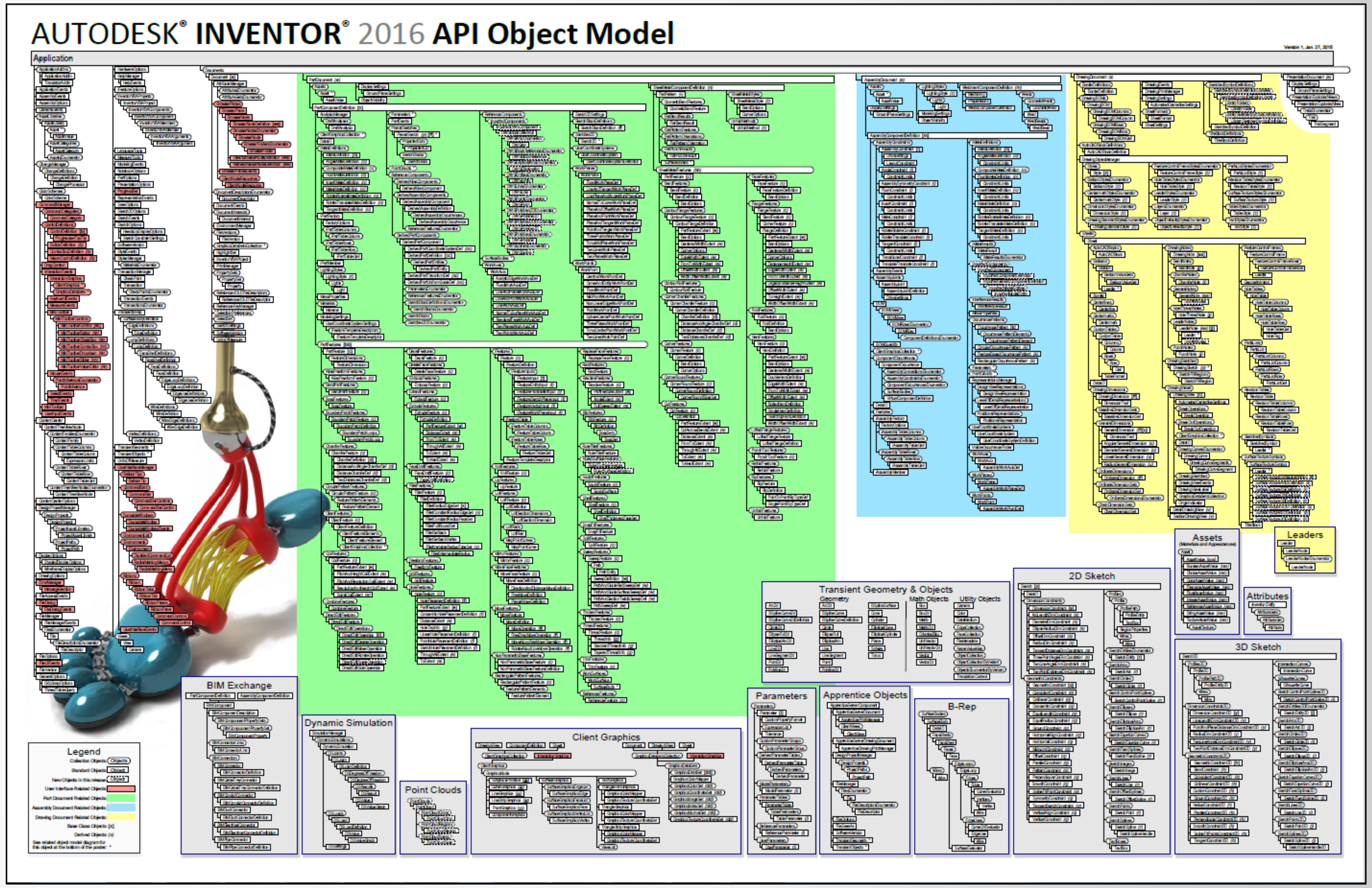
- Long and Integer – For all whole numbers (8, 345667)
  - Long – Up to 9,223,372,036,854,775,807
  - Integer – Up to 2,147,483,647
- String – Any textual information (“Hello”, “1.5”)

```
Dim input As String = "abc 123"
```

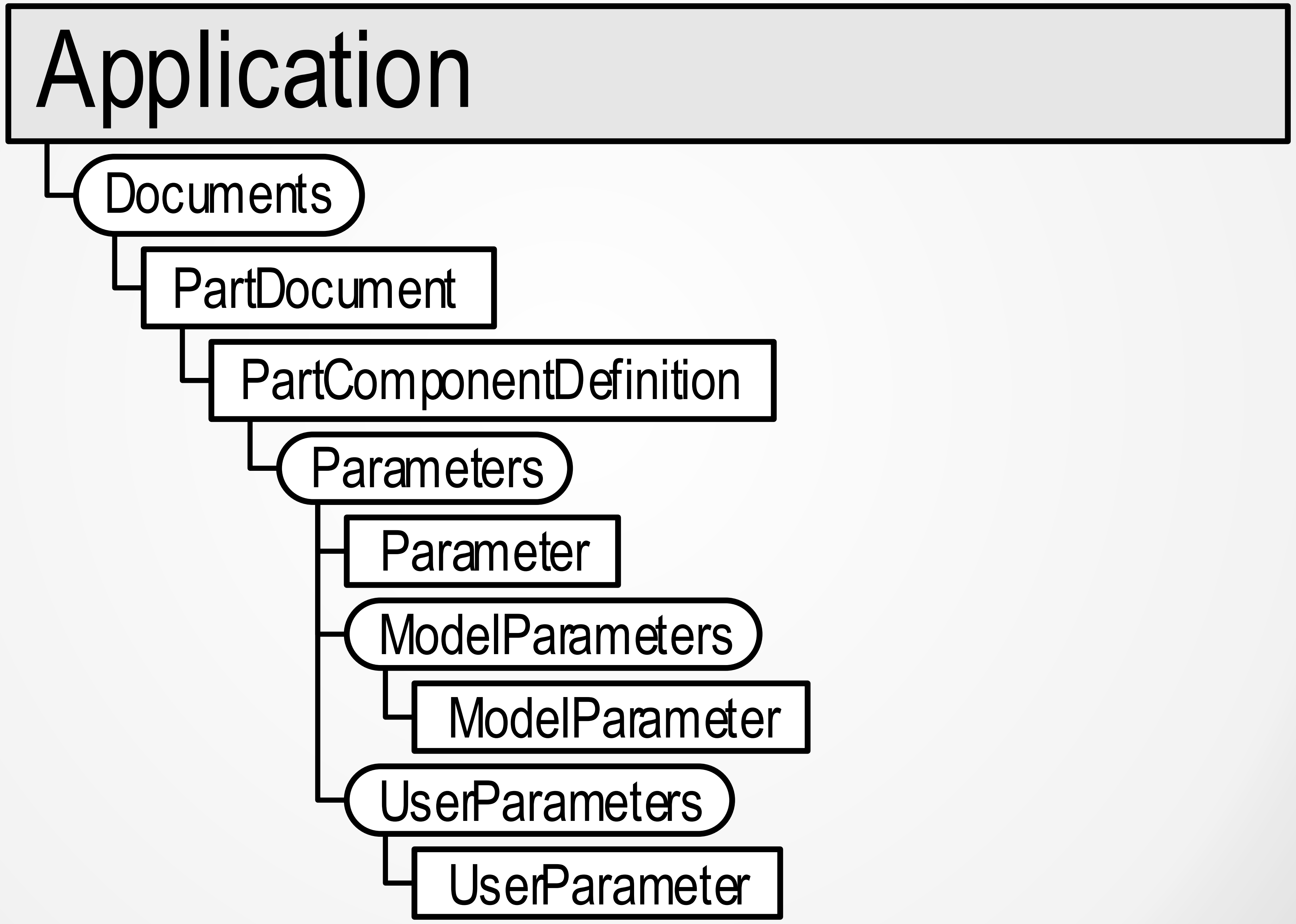
- Inventor Types
- Other less commonly used types: Date, Byte, Currency

Inventor API Object Model

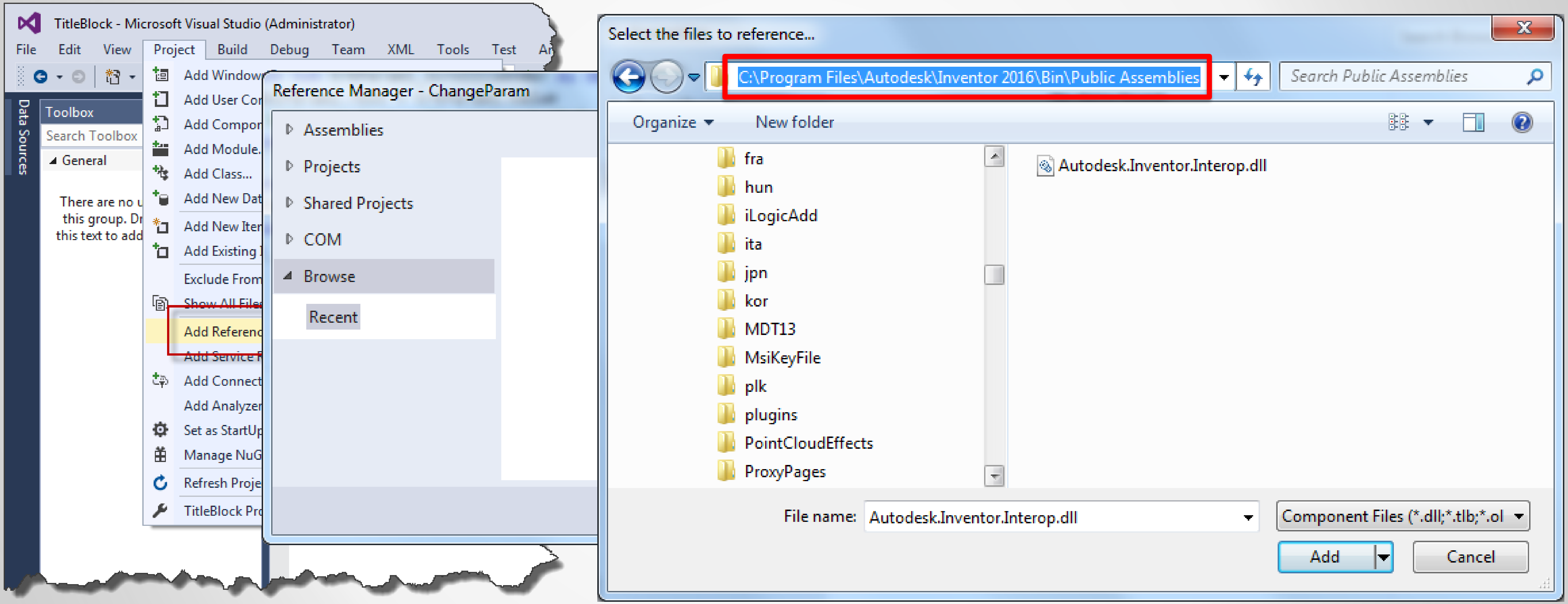
- API objects represent things in Inventor
- Hierarchy indicates ownership
- Traverse hierarchy to access specific objects



Inventor API Object Model



Telling Visual Basic about Inventor

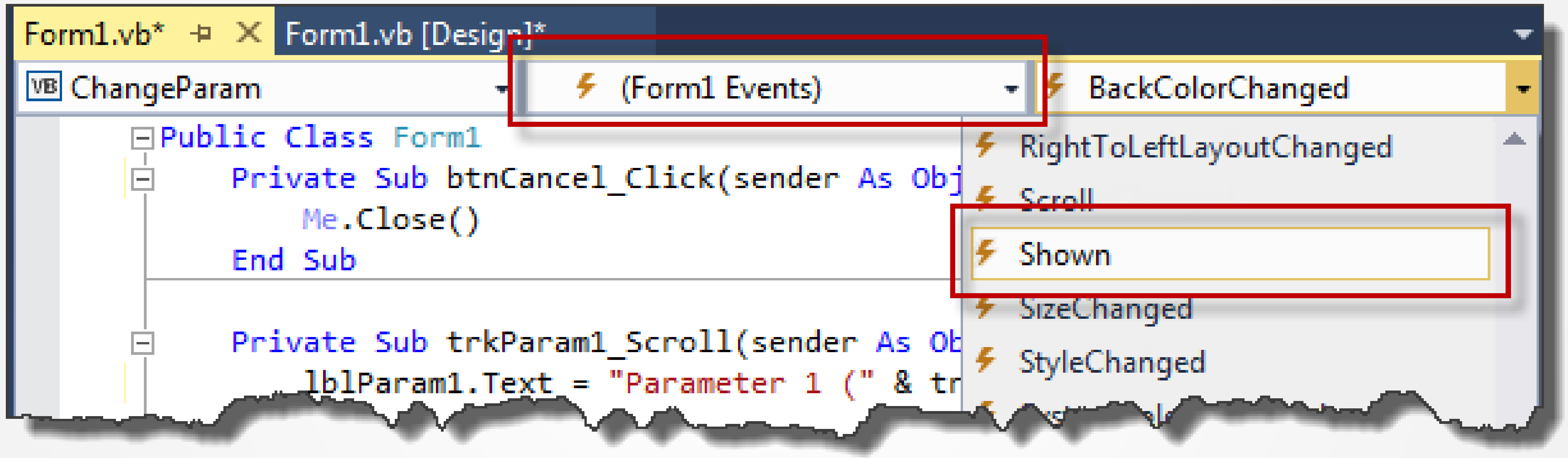


C:\Program Files\Autodesk\Inventor 2016\Bin\Public Assemblies

File name: Autodesk.Inventor.Interop.dll

Component Files (*.dll;*.tlb;*.ol

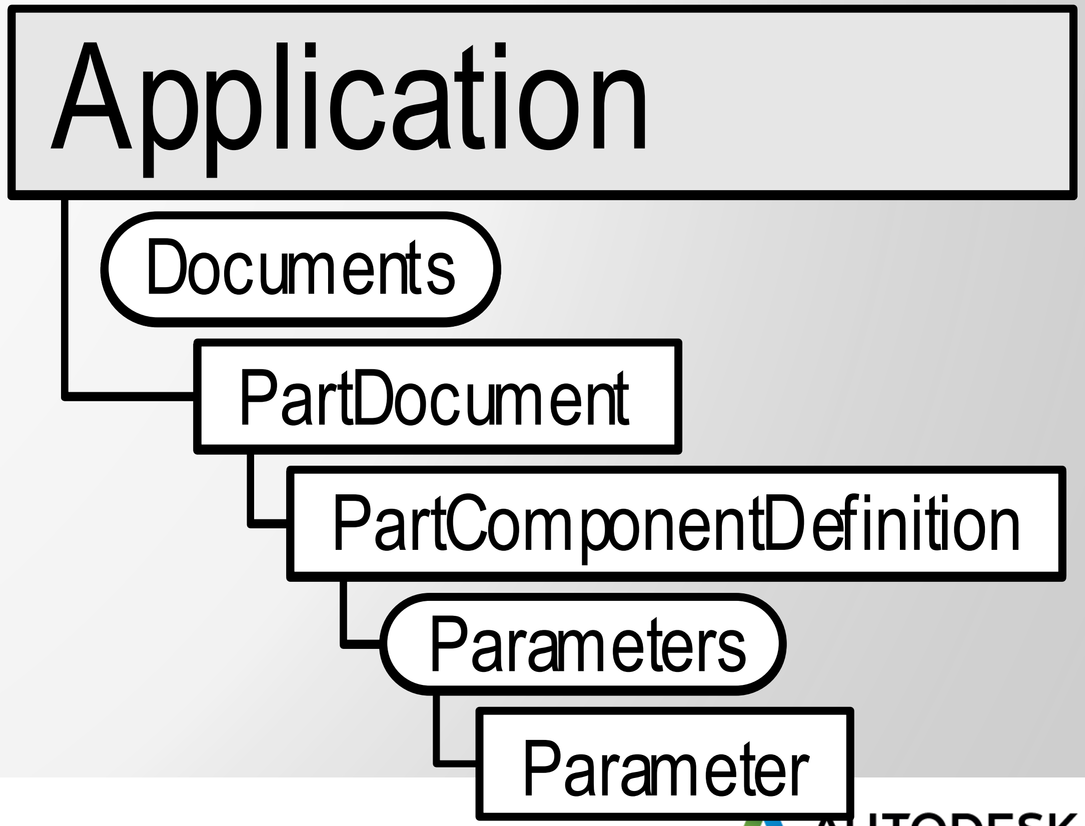
Add Event Handler for Form Shown



```
Public Class Form1
    Private Sub btnCancel_Click(sender As Obj
        Me.Close()
    End Sub

    Private Sub trkParam1_Scroll(sender As Ob
        lblParam1.Text = "Parameter 1 (" & tr
```

Connect to Inventor

```
Private Sub Form1_Shown(sender As Object, e As EventArgs) Handles Me.Shown
    ' Get the Inventor Application object.
    Dim invApp As Inventor.Application
    invApp = GetObject(, "Inventor.Application")

    ' Get the active document.  This assumes it's a part document.
    Dim partDoc As Inventor.PartDocument
    partDoc = invApp.ActiveDocument

    ' Get the Parameters collection.
    Dim params As Inventor.Parameters
    params = partDoc.ComponentDefinition.Parameters

    ' Get the Parameters using its name.
    Dim param as Inventor.Parameter
    param = params.Item("Length")
End Sub
```

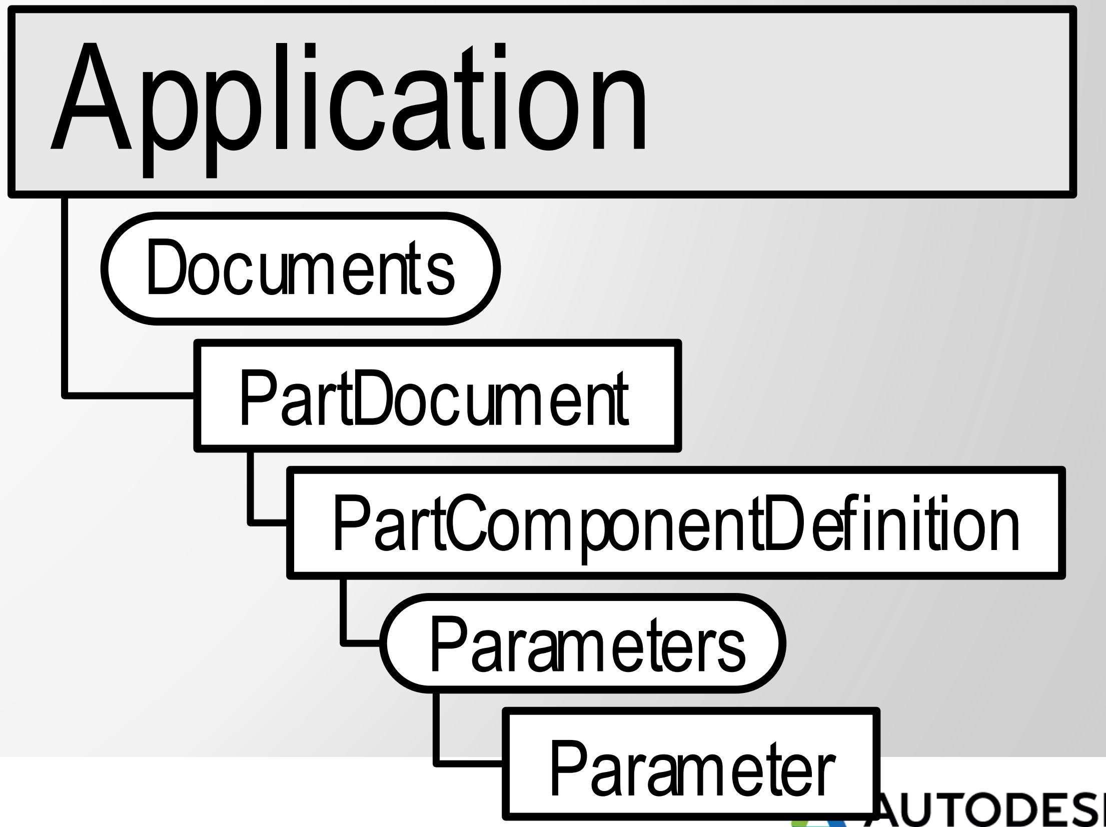


Variable Scope

```
Public Class Form1
    Dim name As String = "Brian"

    Private Sub Form1_Shown(sender As Object, e As EventArgs) Handles Me.Shown
        Dim invApp As Inventor.Application
        invApp = GetObject(, "Inventor.Application")

        For i As Integer = 1 To 10
            Dim j As Integer
            Debug.Print(invApp.Caption & name & i & j)
        Next

        ~~j = 0~~
    End Sub
End Class
```

Fix Variable Scope

```
Public Class Form1
    Dim partDoc as Inventor.PartDocument
    Dim param As Inventor.Parameter

    Private Sub Form1_Shown(sender As Object, e As EventArgs) Handles Me.Shown
        ' Get the Inventor Application object.
        Dim invApp As Inventor.Application
        invApp = GetObject(, "Inventor.Application")

        ' Get the active document.  This assumes it's a part document.
        partDoc = invApp.ActiveDocument

        ' Get the Parameters collection.
        Dim params As Inventor.Parameters
        params = partDoc.ComponentDefinition.Parameters

        ' Get the Parameters using its name.
        param = params.Item("Length")
    End Sub
End Class
```

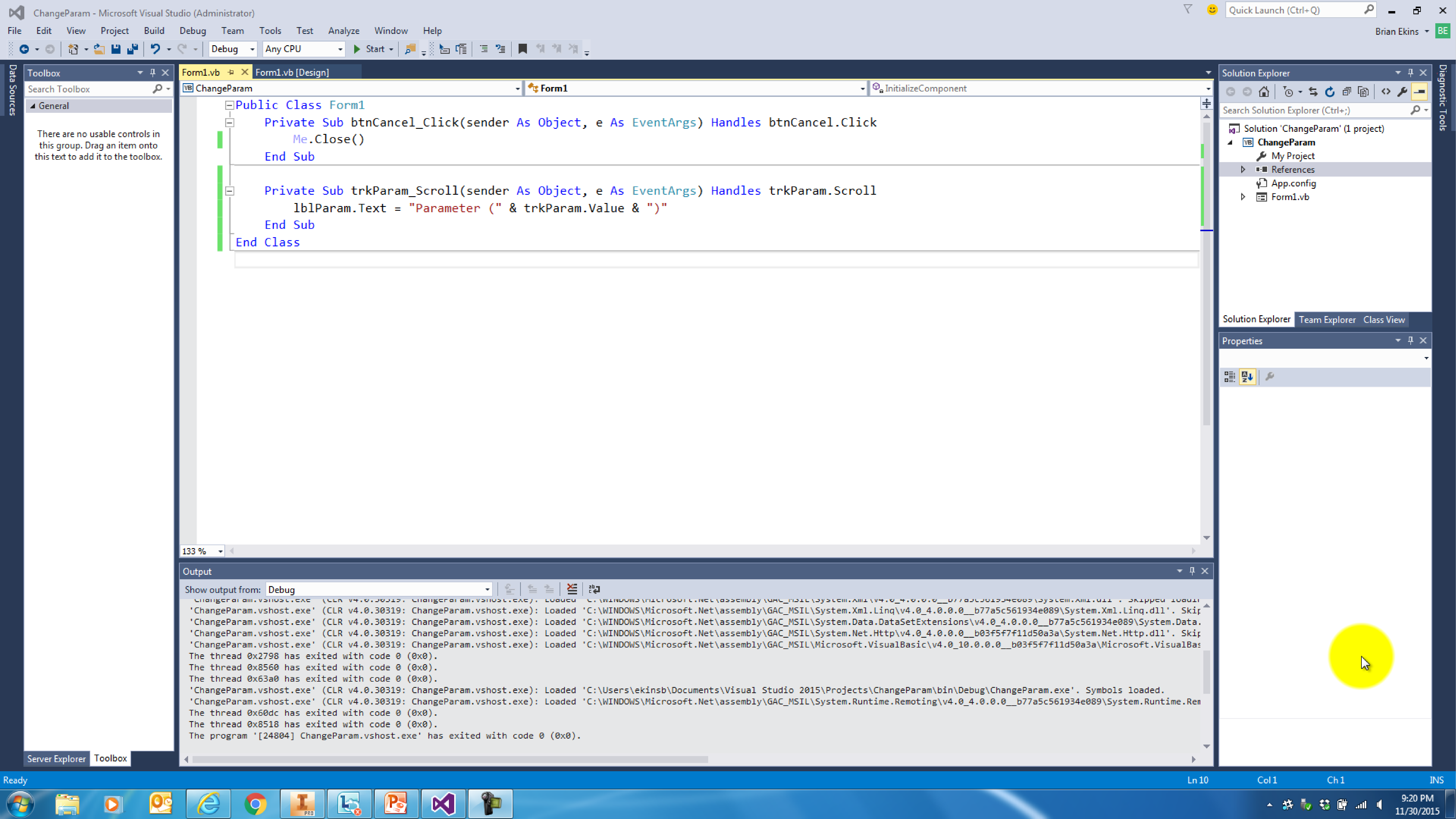


Enhancing the Track Bar

```
Private Sub trkParam1_Scroll(…) Handles trkParam1.Scroll
    lblParam1.Text = "Parameter 1 (" & trkParam1.Value & ")"
End Sub
```

Enhancing the Track Bar

```
Private Sub trkParam1_Scroll(…) Handles trkParam1.Scroll
    Dim newValue As Double = trkParam.Value / 100

    param.Expression = newValue
    partDoc.Update()

    lblParam1.Expression = "Parameter 1 (" & newValue & ")"
End Sub
```

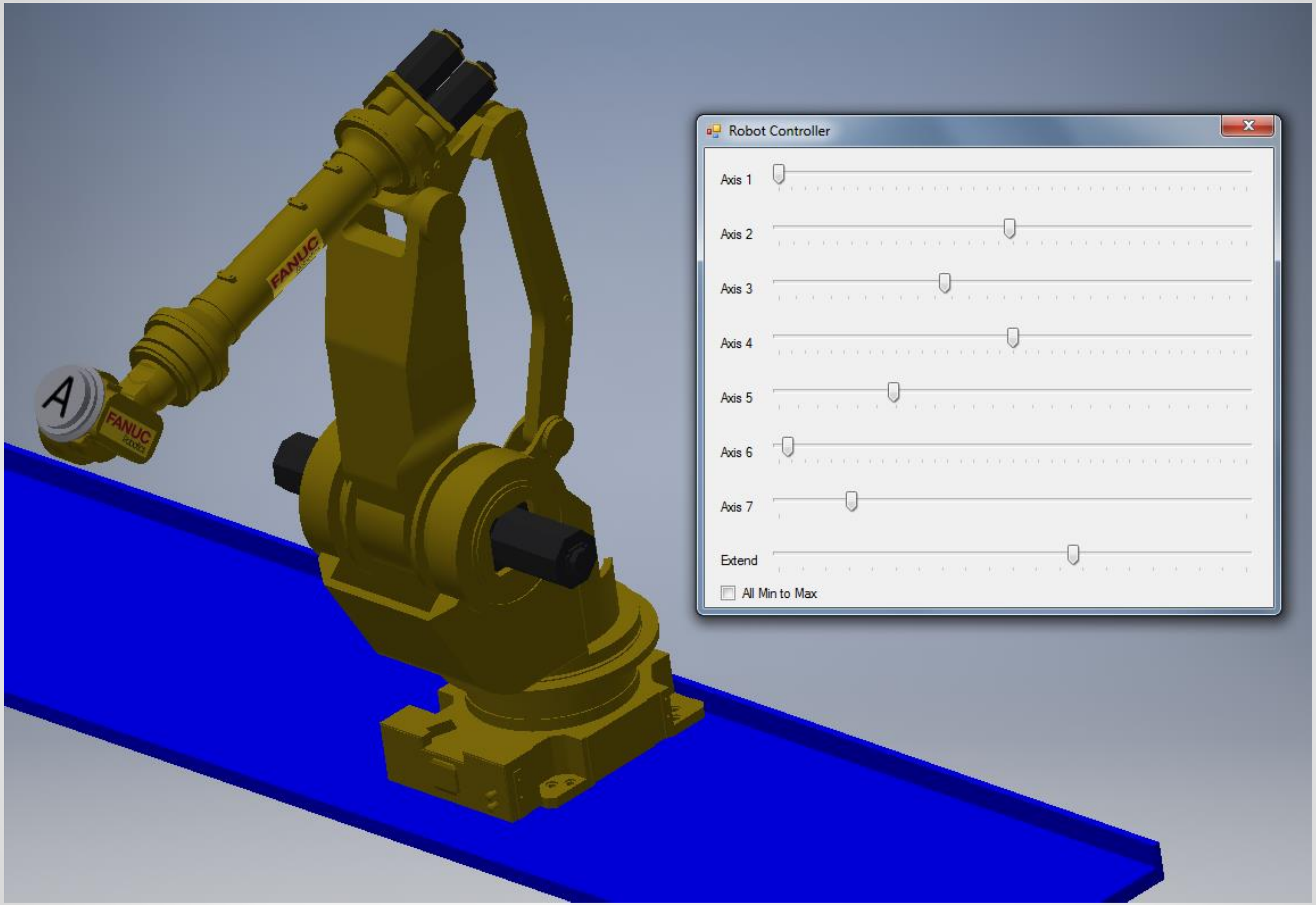


```
Public Class Form1
    Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
        Me.Close()
    End Sub

    Private Sub trkParam_Scroll(sender As Object, e As EventArgs) Handles trkParam.Scroll
        lblParam.Text = "Parameter (" & trkParam.Value & ")"
    End Sub
End Class
```

```
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll'. Skipped loadi
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml.Linq\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.Linq.dll'. Skip
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Data.DataSetExtensions\v4.0_4.0.0.0__b77a5c561934e089\System.Data.
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Net.Http\v4.0_4.0.0.0__b03f5f7f11d50a3a\System.Net.Http.dll'. Skip
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\Microsoft.VisualBasic\v4.0_10.0.0.0__b03f5f7f11d50a3a\Microsoft.VisualBas
The thread 0x2798 has exited with code 0 (0x0).
The thread 0x8560 has exited with code 0 (0x0).
The thread 0x63a0 has exited with code 0 (0x0).
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\Users\ekinsb\Documents\Visual Studio 2015\Projects\ChangeParam\bin\Debug\ChangeParam.exe'. Symbols loaded.
'ChangeParam.vshost.exe' (CLR v4.0.30319: ChangeParam.vshost.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Runtime.Remoting\v4.0_4.0.0.0__b77a5c561934e089\System.Runtime.Rem
The thread 0x60dc has exited with code 0 (0x0).
The thread 0x8518 has exited with code 0 (0x0).
The program '[24804] ChangeParam.vshost.exe' has exited with code 0 (0x0).
```

Making it Better

- What if Inventor isn’t running?
- What if a part document isn’t active?
- What if you want it to work in an assembly?
- The track bar doesn’t show the current value.
- What if the parameter doesn’t exist?



Robot Controller

Axis 1

Axis 2

Axis 3

Axis 4

Axis 5

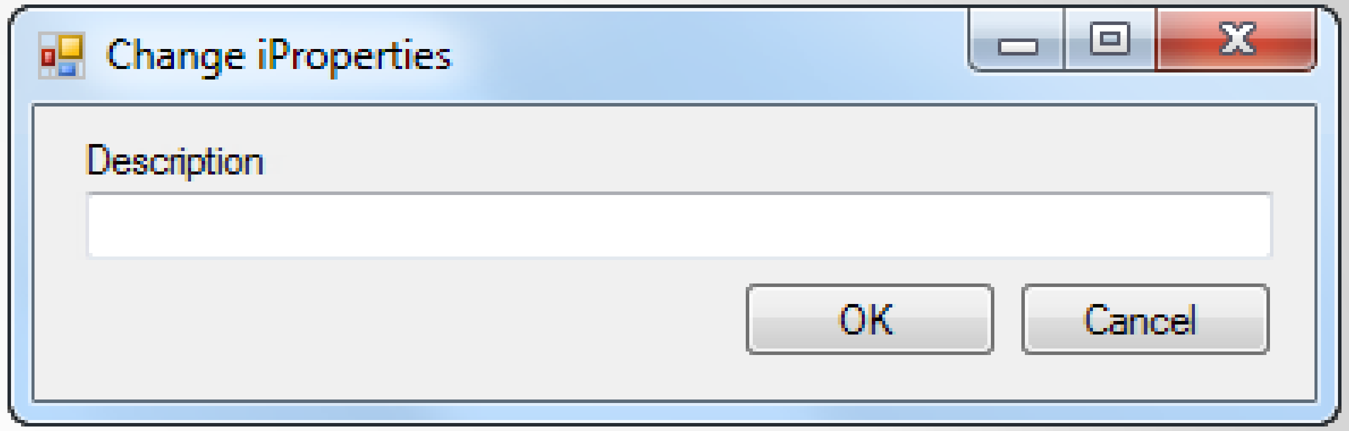
Axis 6

Axis 7

Extend

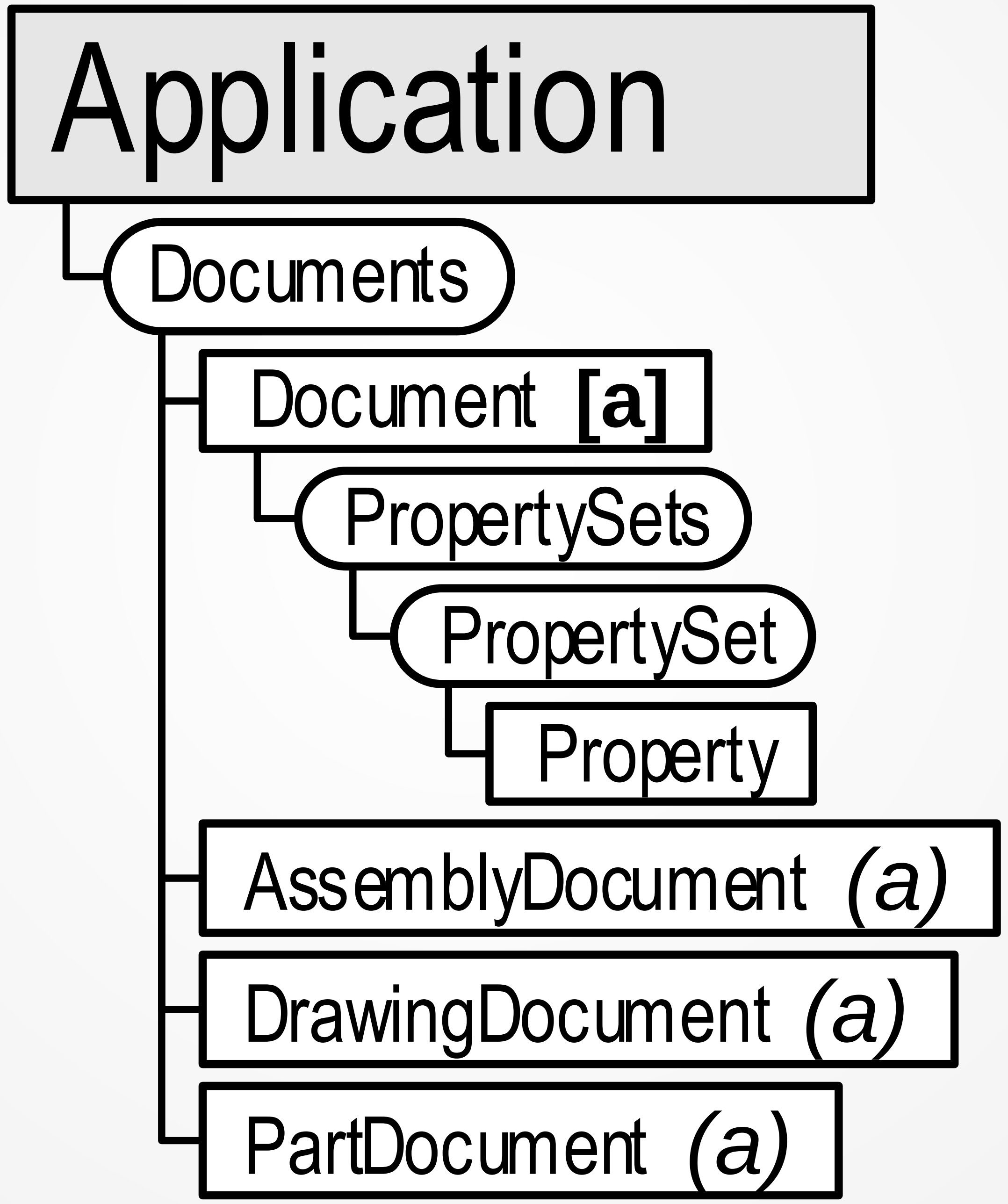
☐ All Min to Max

Exercise Two



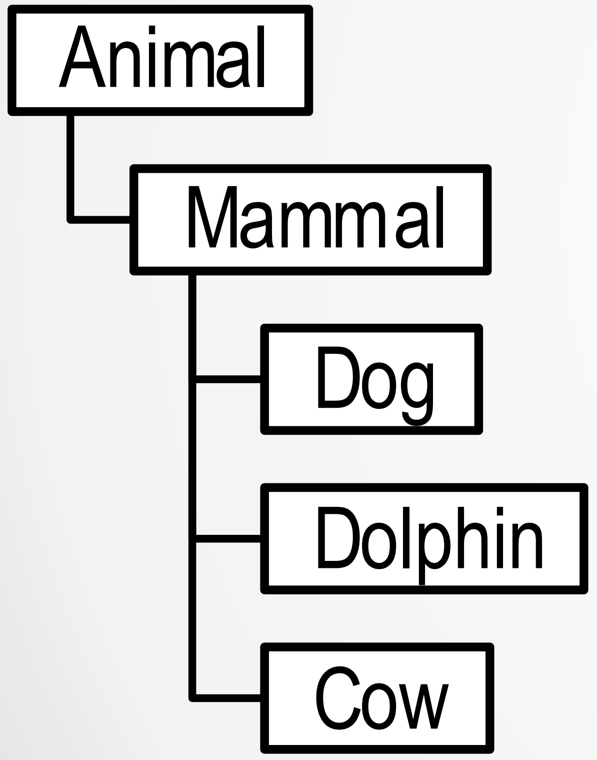
1. Connect to Inventor.
2. Get the active document.
3. Get the Description iProperty.
4. When OK is clicked:
   A. Assign the value of the text box to the iProperty.
   B. Dismiss the form.
5. When Cancel is clicked, dismiss the form.

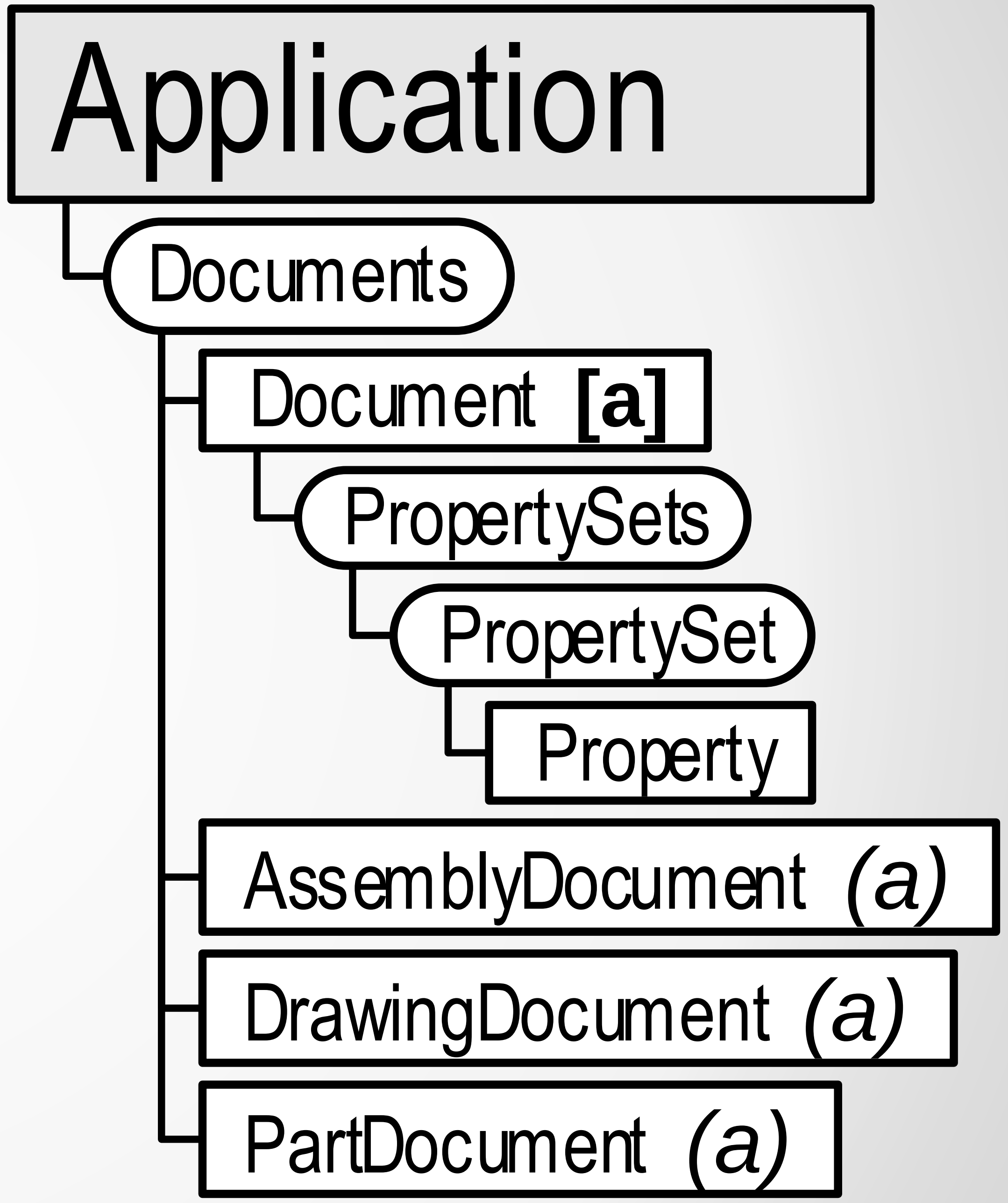
iProperties Object Model



Derivation

- Class Diagram



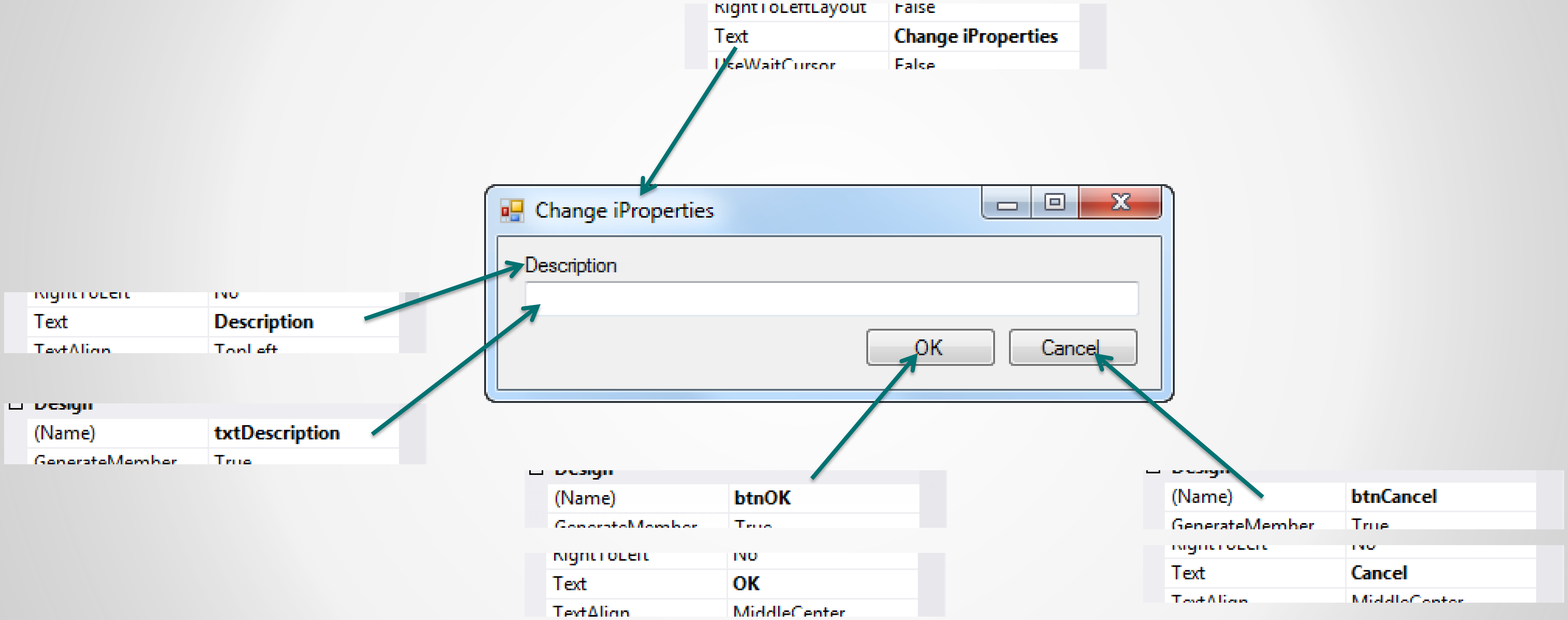


Open the Existing Project

Open existing project “ChangeiProperties.sln” from:

C:\Datasets\You Can Do It-How to write Programs for Inventor\ChangeiProperties - Start

Design Your Form



RightToLeftLayout	False
Text	Change iProperties
UseWaitCursor	False

Change iProperties

Description

OK Cancel

RightToLeft	No
Text	Description
TextAlign	TopLeft

Design

(Name)	txtDescription
GenerateMember	True

Design

(Name)	btnOK
GenerateMember	True

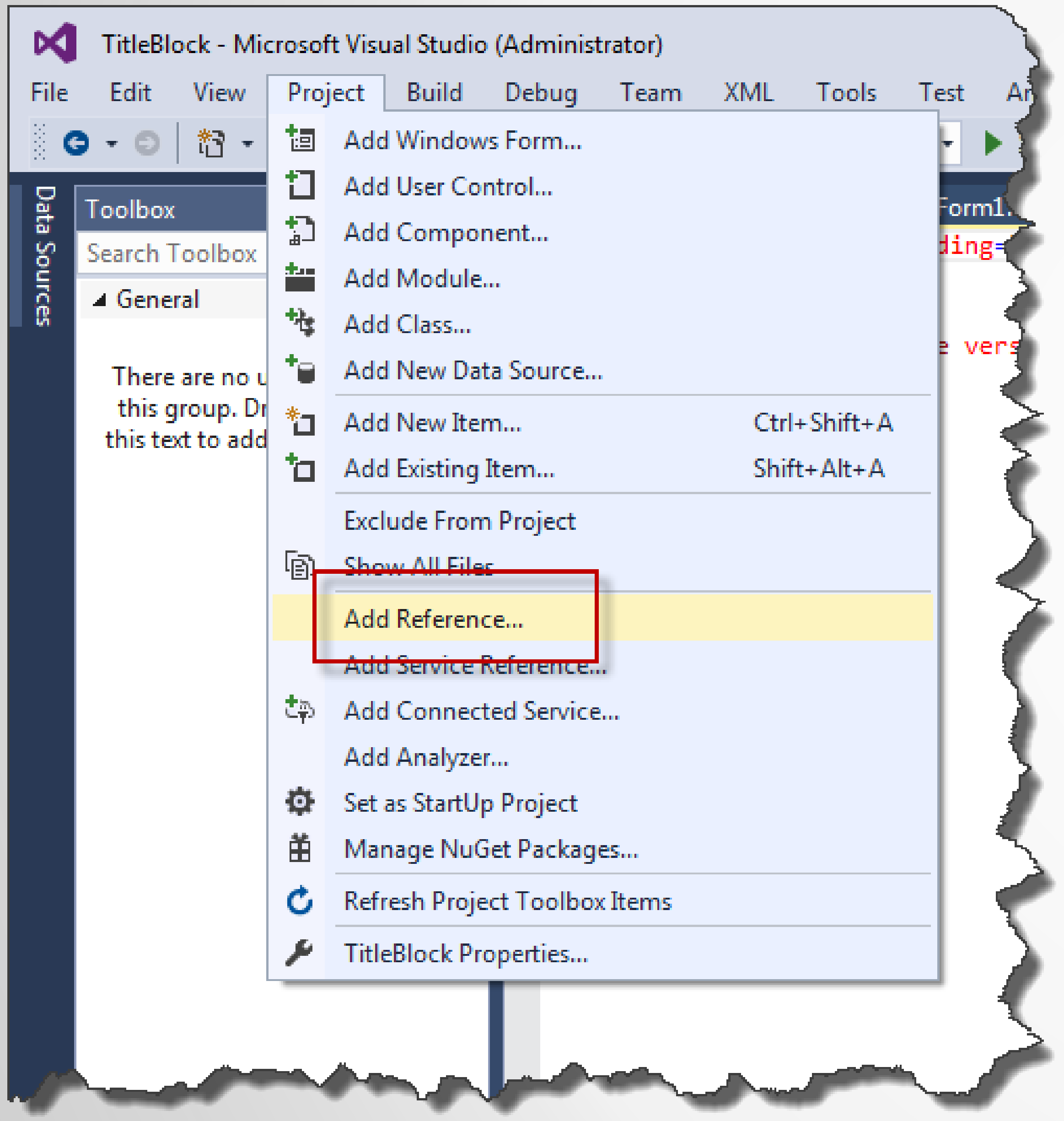
RightToLeft	No
Text	OK
TextAlign	MiddleCenter

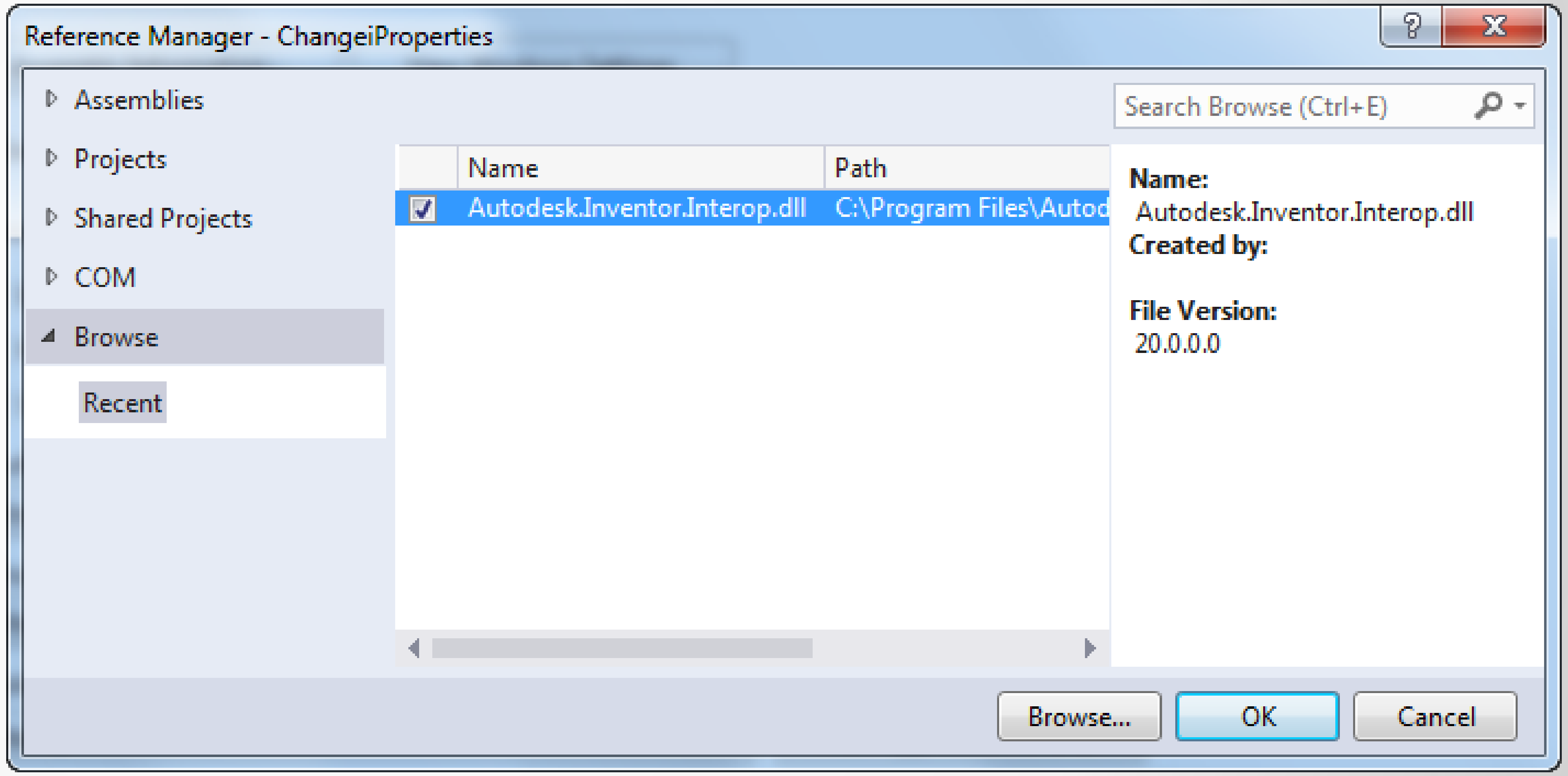
Design

(Name)	btnCancel
GenerateMember	True

RightToLeft	No
Text	Cancel
TextAlign	MiddleCenter

Reference the Inventor Library



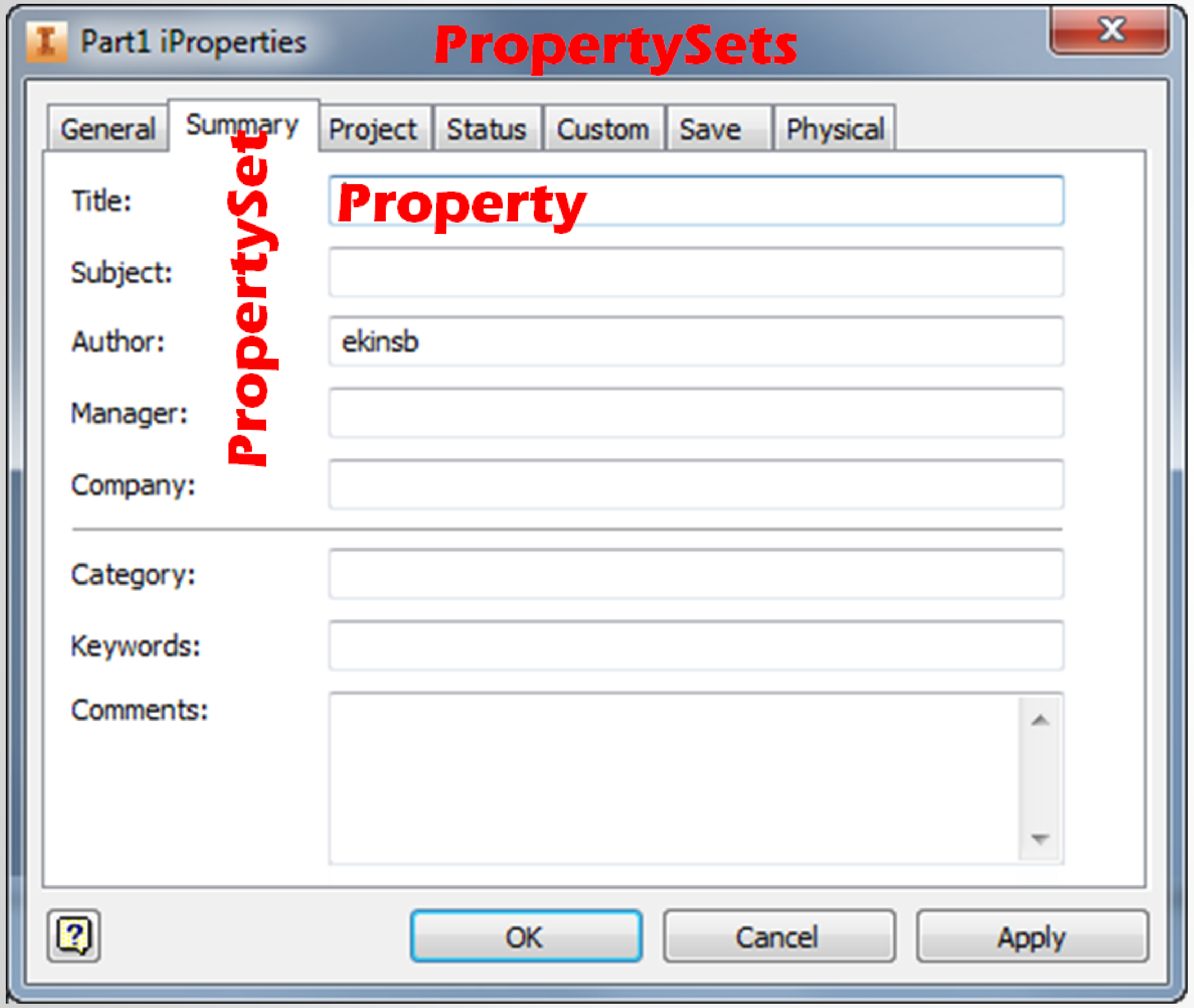


	Name	Path
✓	Autodesk.Inventor.Interop.dll	C:\Program Files\Autod

Name:
Autodesk.Inventor.Interop.dll
Created by:

File Version:
20.0.0.0

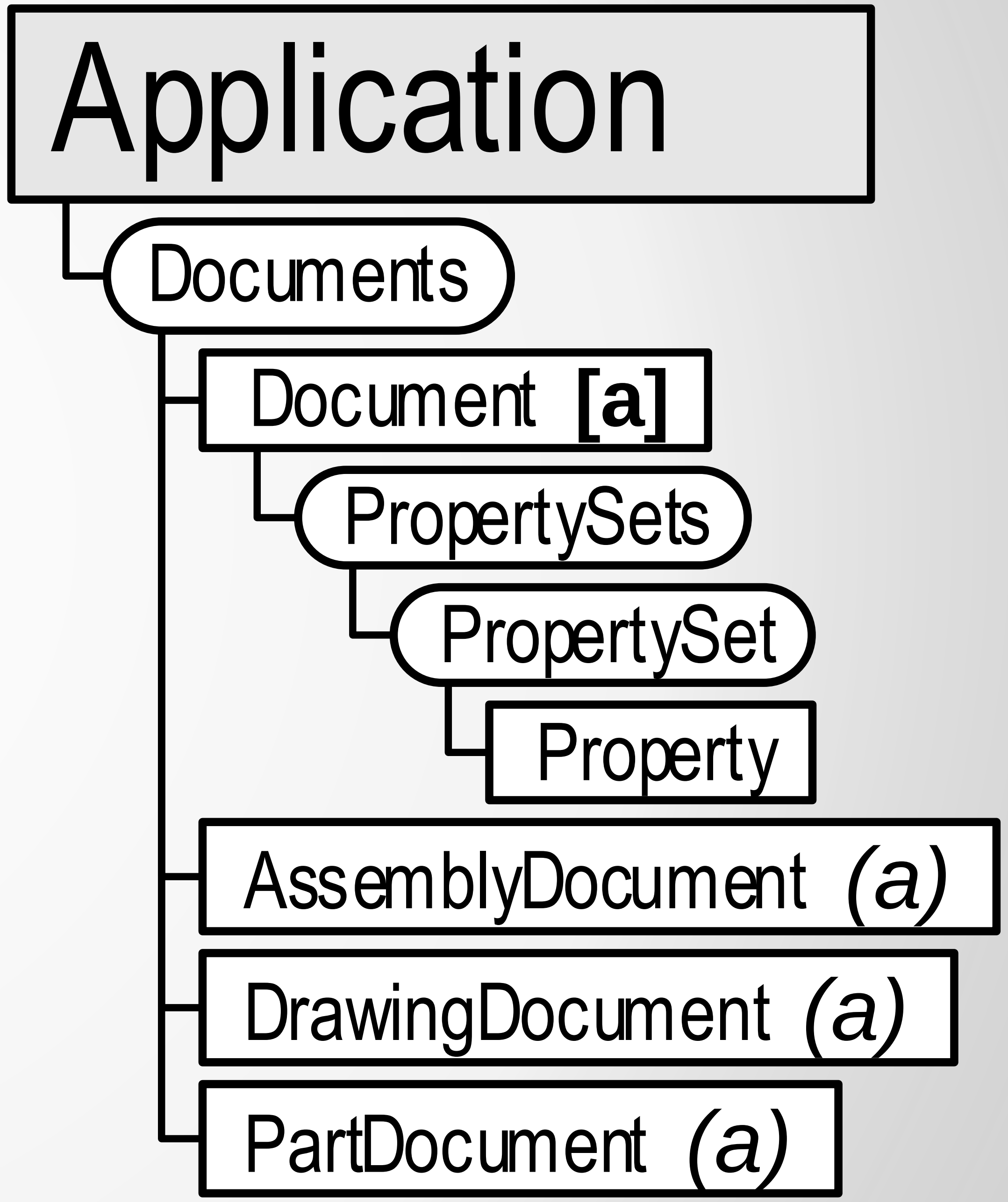
iProperties Object Model



Part1 iProperties

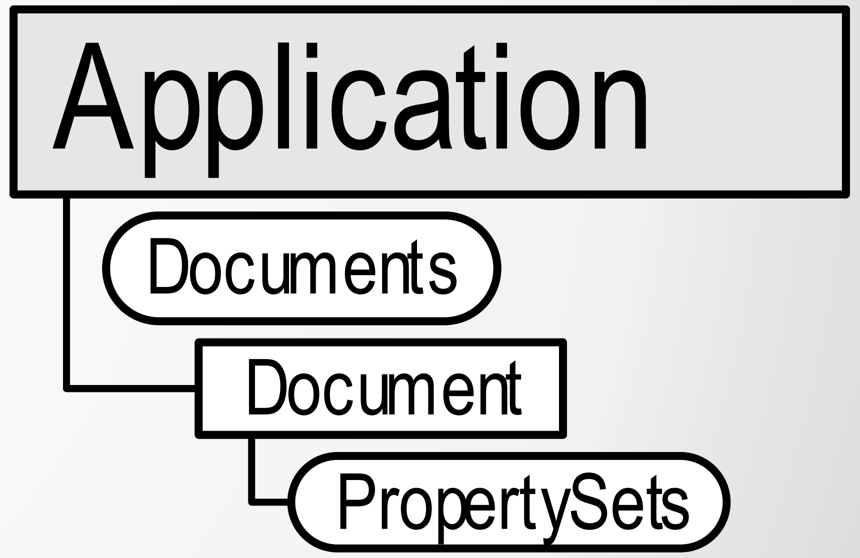
General | Summary | Project | Status | Custom | Save | Physical

Title:	
Subject:	
Author:	ekinsb
Manager:	
Company:	
Category:	
Keywords:	
Comments:	



Connect to Inventor

```
btnOK_Click(sender As Object, e As EventArgs) Handles btnOK.Click
    ' Get the Inventor Application object.
    Dim invApp As Inventor.Application
    invApp = GetObject(, "Inventor.Application")

    ' Get the active document.
    Dim doc As Inventor.Document
    doc = invApp.ActiveDocument

    ' Get the PropertySets collection.
    Dim propSets As Inventor.PropertySets
    propSets = doc.PropertySets
End Sub
```



PropertySet and Property Names

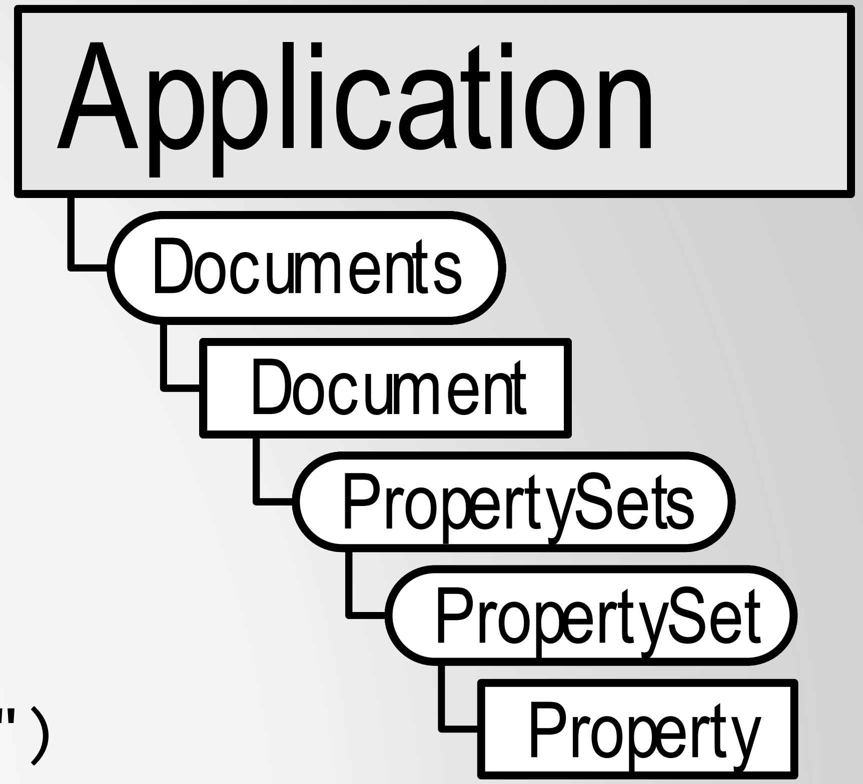
Inventor User Defined Properties	
Inventor Summary Information	
Author	String
Comments	String
Keywords	String
Last Saved By	String
Revision Number	String
Subject	String
Thumbnail	Picture
Title	String
Inventor Document Summary Information	
Category	String
Company	String
Manager	String
Design Tracking Properties	
Appearance	String
Authority	String
Catalog Web Link	String
Categories	String
Checked By	String
Cost	Currency
Cost Center	String
Creation Time	Date

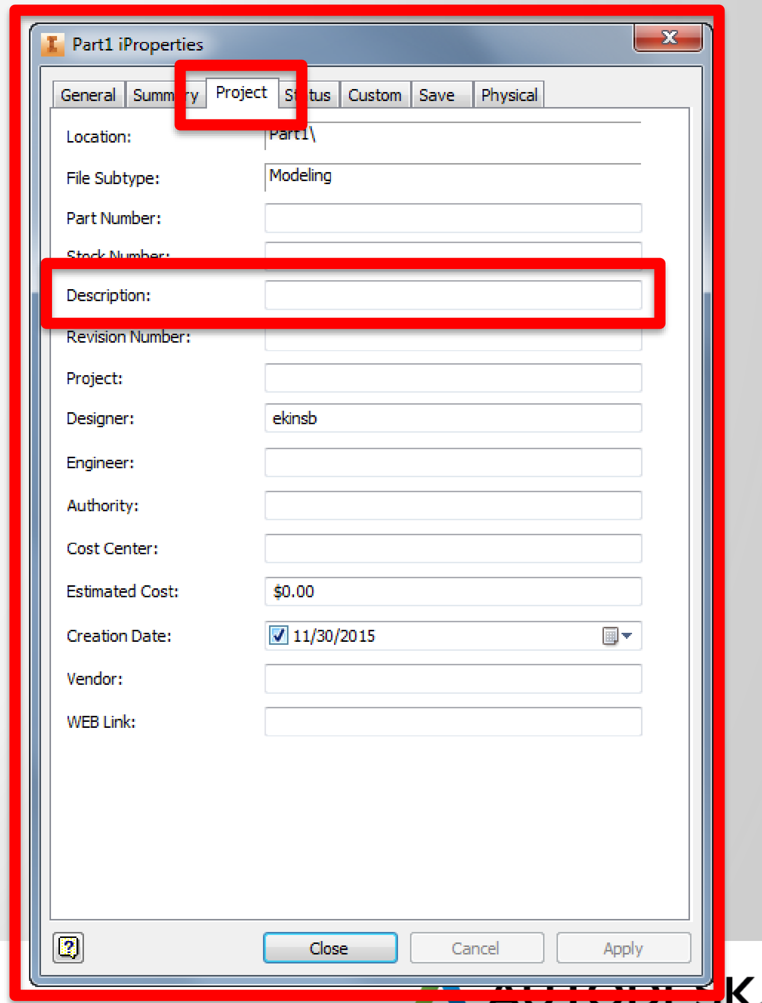
Date Checked	Date
Defer Updates	Boolean
Density	Double
Description	String
Design Status	Long
Designer	String
Document SubType	String
Document SubType Name	String
Engineer	String
Engr Approved By	String
Engr Date Approved	Date
External Property Revision Id	String
Flat Pattern Area	Double
Flat Pattern Length	Double
Flat Pattern Width	Double
Language	String
Last Updated With	String
Manufacturer	String
Mass	Double
Material	String
Material Identifier	String
Mfg Approved By	String
Mfg Date Approved	Date

Parameterized Template	Boolean
Part Icon	Picture
Part Number	String
Part Property Revision Id	String
Project	String
Proxy Refresh Date	Date
Sheet Metal Area	String
Sheet Metal Length	String
Sheet Metal Rule	String
Sheet Metal Width	String
Size Designation	String
Standard	String
Standard Revision	String
Standards Organization	String
Stock Number	String
SurfaceArea	Double
Template Row	String
User Status	String
Valid MassProps	Long
Vendor	String
Volume	Double
Weld Material	String

Get the Properties

```
    ' Get the PropertySets collection.
    Dim propSets As Inventor.PropertySets
    propSets = doc.PropertySets

    ' Get the design tracking property set.
    Dim designTrackPropSet As Inventor.PropertySet
    designTrackPropSet = propSets.Item("Design Tracking Properties")

    ' Get the description property.
    Dim descProp As Inventor.Property
    descProp = designTrackPropSet.Item("Description")

    ' Set the value of the property using the value in the text box.
    descProp.Value = txtDescription.Text

    Me.Close()
End Sub
```



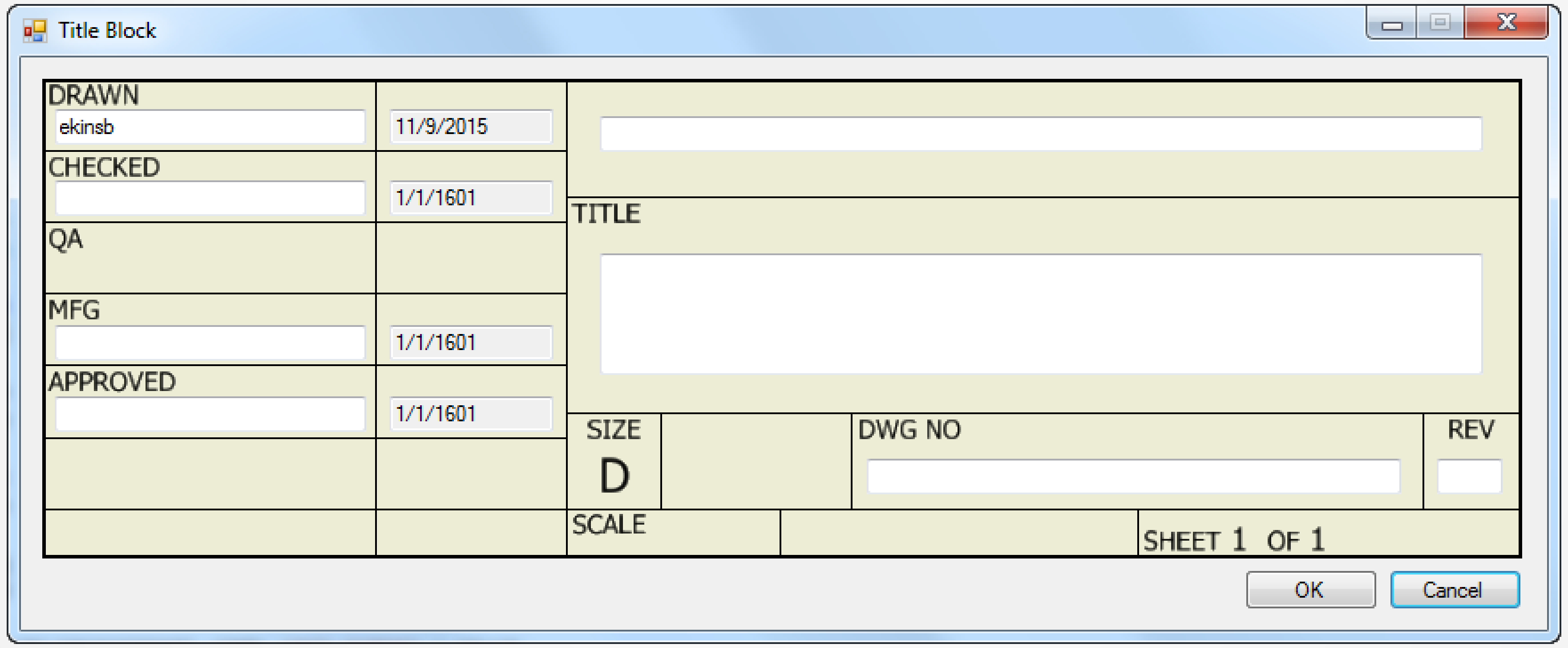


Location:	Part1\
File Subtype:	Modeling
Part Number:	
Stock Number:	
Description:	
Revision Number:	
Project:	
Designer:	ekinsb
Engineer:	
Authority:	
Cost Center:	
Estimated Cost:	$0.00
Creation Date:	✓ 11/30/2015
Vendor:	
WEB Link:	

Code for OK

```
 Private Sub btnOK_Click(sender As Object, e As EventArgs) Handles btnOK.Click
    ' Get the Inventor Application object.
    Dim invApp As Inventor.Application
    invApp = GetObject(, "Inventor.Application")

    ' Get the active document.
    Dim doc As Inventor.Document
    doc = invApp.ActiveDocument

    ' Get the design tracking property set.
    Dim designTrackPropSet As Inventor.PropertySet
    designTrackPropSet = doc.PropertySets.Item("Design Tracking Properties")

    ' Get the description property.
    Dim descProp As Inventor.Property
    descProp = designTrackPropSet.Item("Description")

    ' Set the values of the properties using the values in the text boxes.
    descProp.Value = txtDescription.Text

    ' Close the form.
    Me.Close()
End Sub
```

Real-World Example

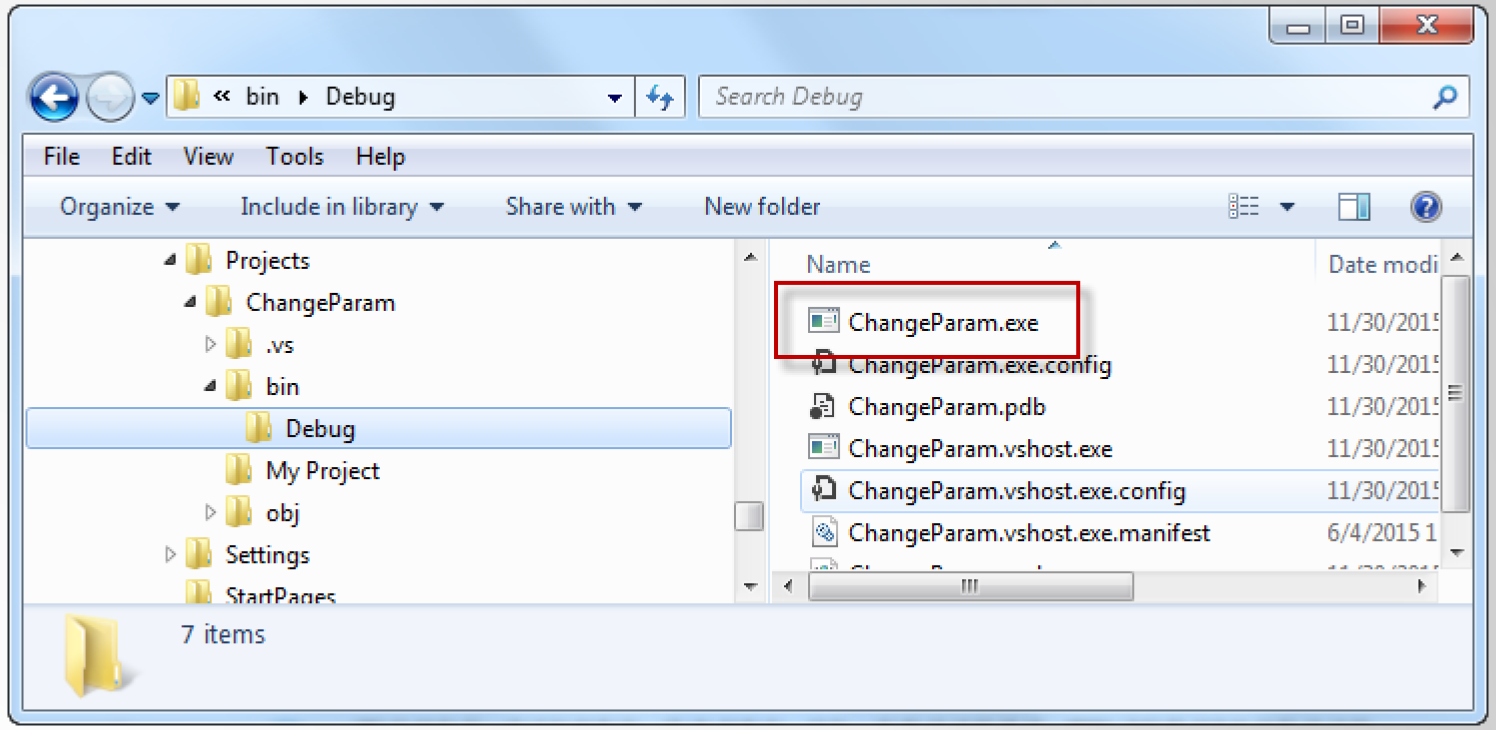


Title Block

DRAWN ekinsb	11/9/2015	
CHECKED	1/1/1601	TITLE
QA		
MFG	1/1/1601	
APPROVED	1/1/1601	SIZE D / DWG NO / REV
		SCALE / SHEET 1 OF 1

OK Cancel

Running Your Programs

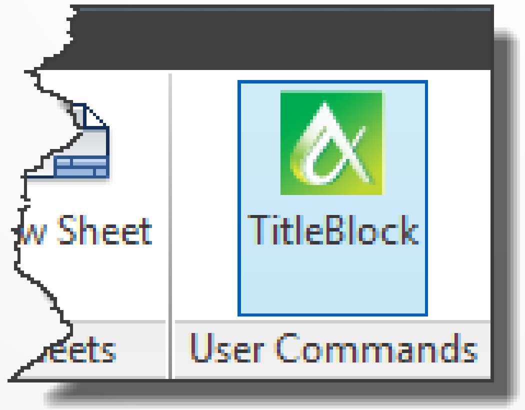

bin ▸ Debug
Search Debug
File Edit View Tools Help
Organize ▾ Include in library ▾ Share with ▾ New folder

```
Projects
  ChangeParam
    .vs
    bin
      Debug
    My Project
    obj
  Settings
  StartPages
```

Name	Date modi
ChangeParam.exe	11/30/201
ChangeParam.exe.config	11/30/201
ChangeParam.pdb	11/30/201
ChangeParam.vshost.exe	11/30/201
ChangeParam.vshost.exe.config	11/30/201
ChangeParam.vshost.exe.manifest	6/4/2015 1

7 items

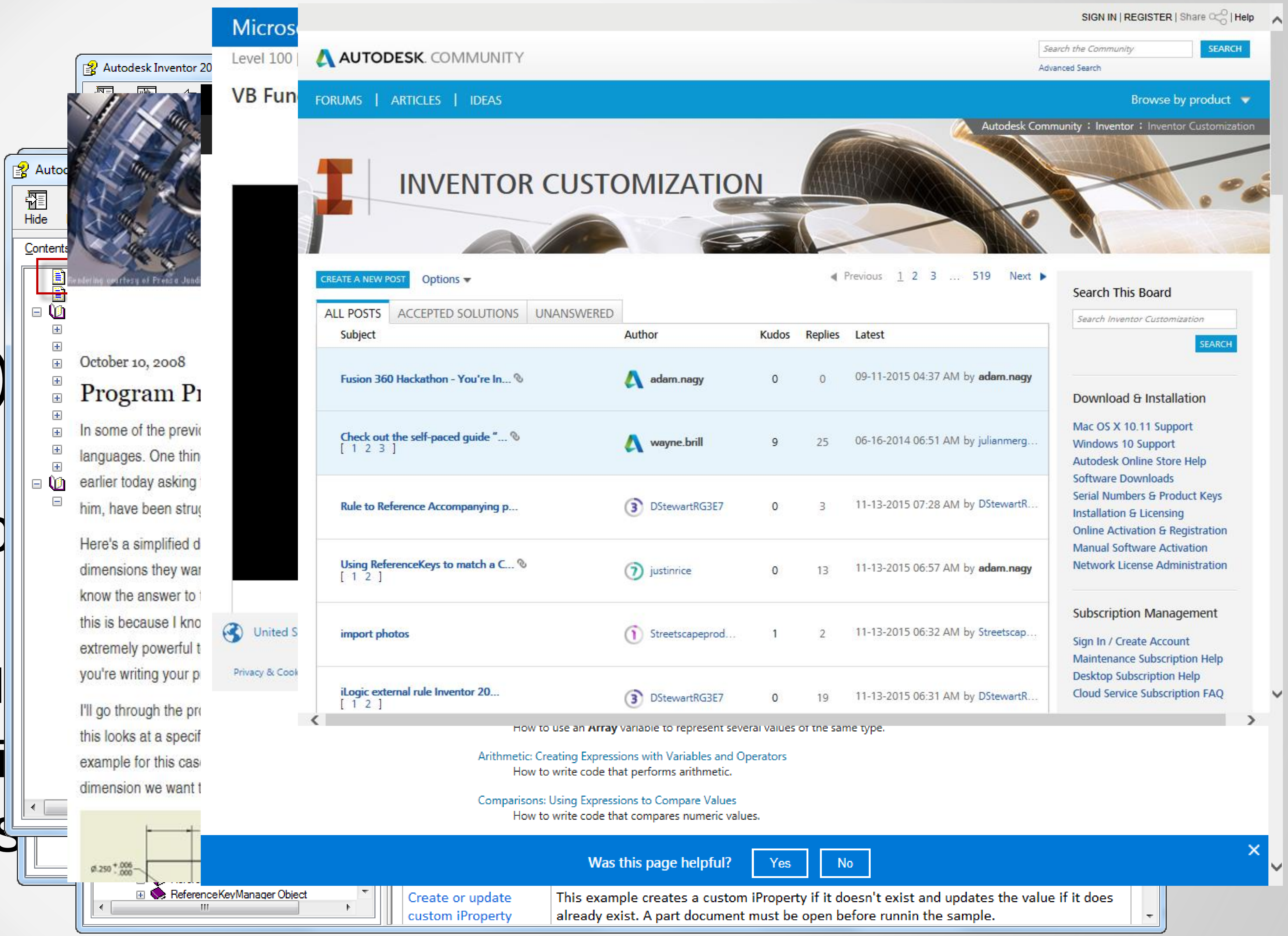
- You've created an executable.
- It can be distributed.
- You can run it from Explorer.
- You can create a one-line VBA macro to run it from inside Inventor.



- You can convert your program to an add-in.

What Now?

- API Help
- SDK
- Web (Blogs)
- Books
- Online Reso
  - Google
  - Microsoft M
  - Microsoft Vi
- Inventor Cus



SIGN IN | REGISTER | Share | Help

AUTODESK. COMMUNITY

FORUMS | ARTICLES | IDEAS — Browse by product

Autodesk Community : Inventor : Inventor Customization

INVENTOR CUSTOMIZATION

CREATE A NEW POST — Options — Previous 1 2 3 ... 519 Next

ALL POSTS | ACCEPTED SOLUTIONS | UNANSWERED

Subject	Author	Kudos	Replies	Latest
Fusion 360 Hackathon - You're In...	adam.nagy	0	0	09-11-2015 04:37 AM by adam.nagy
Check out the self-paced guide "... [1 2 3]	wayne.brill	9	25	06-16-2014 06:51 AM by julianmerg...
Rule to Reference Accompanying p...	DStewartRG3E7	0	3	11-13-2015 07:28 AM by DStewartR...
Using ReferenceKeys to match a C... [1 2]	justinrice	0	13	11-13-2015 06:57 AM by adam.nagy
import photos	Streetscapeprod...	1	2	11-13-2015 06:32 AM by Streetscap...
iLogic external rule Inventor 20... [1 2]	DStewartRG3E7	0	19	11-13-2015 06:31 AM by DStewartR...

Search This Board

Download & Installation
- Mac OS X 10.11 Support
- Windows 10 Support
- Autodesk Online Store Help
- Software Downloads
- Serial Numbers & Product Keys
- Installation & Licensing
- Online Activation & Registration
- Manual Software Activation
- Network License Administration

Subscription Management
- Sign In / Create Account
- Maintenance Subscription Help
- Desktop Subscription Help
- Cloud Service Subscription FAQ

Micros — Level 100 — VB Fun

How to use an **Array** variable to represent several values of the same type.

Arithmetic: Creating Expressions with Variables and Operators
How to write code that performs arithmetic.

Comparisons: Using Expressions to Compare Values
How to write code that compares numeric values.

Was this page helpful? Yes No

Autodesk Inventor 20 — Hide — Contents

October 10, 2008

Program Pr

In some of the previ languages. One thin earlier today asking him, have been strug

Here's a simplified d dimensions they wa know the answer to this is because I kno extremely powerful t you're writing your p

I'll go through the pr this looks at a specif example for this cas dimension we want t

ReferenceKeyManager Object

Create or update custom iProperty	This example creates a custom iProperty if it doesn't exist and updates the value if it does already exist. A part document must be open before runnin the sample.

AUTODESK®