

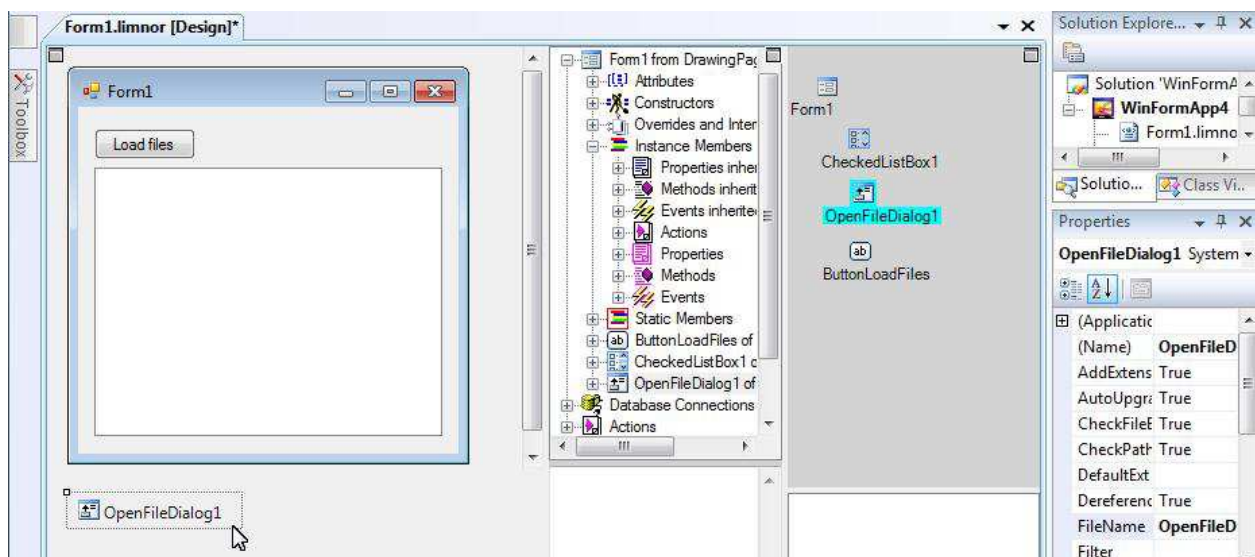
Select Multiple Files using OpenFileDialog

Contents

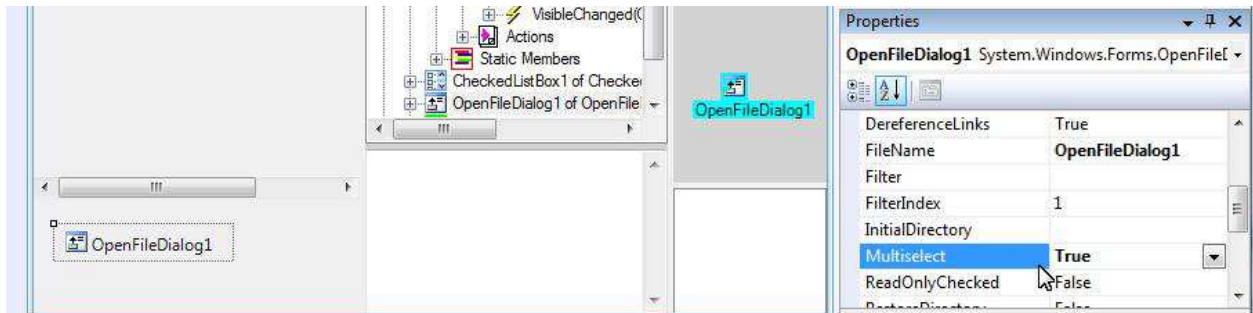
Sample Project	1
Create a Method	2
Execute the method	10

Sample Project

To show how to select multiple files using OpenFileDialog, we create a Windows Application project. An OpenFileDialog is used for selecting files. A checked list box is used to record selected files. A button is used for launching OpenFileDialog.



Make sure Multiselect is True:



Create a Method

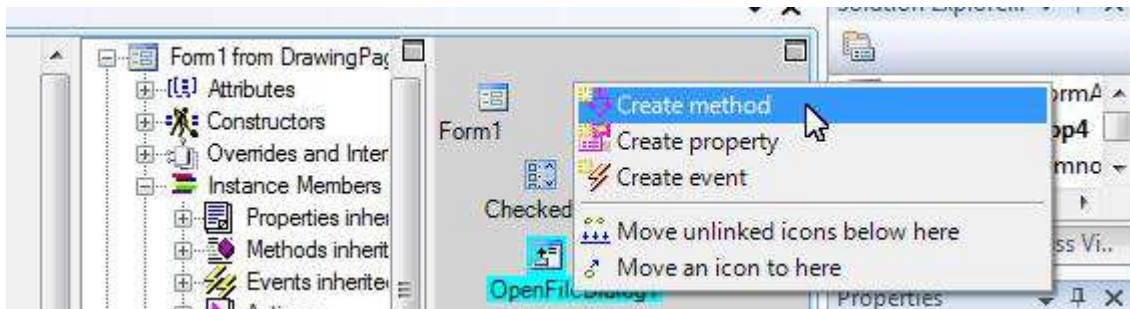
We create a method to do following actions:

Launch file-selection dialogue;

If the user made file selections with the dialogue box then execute the following actions;

Add every selected file name to the checked list box.

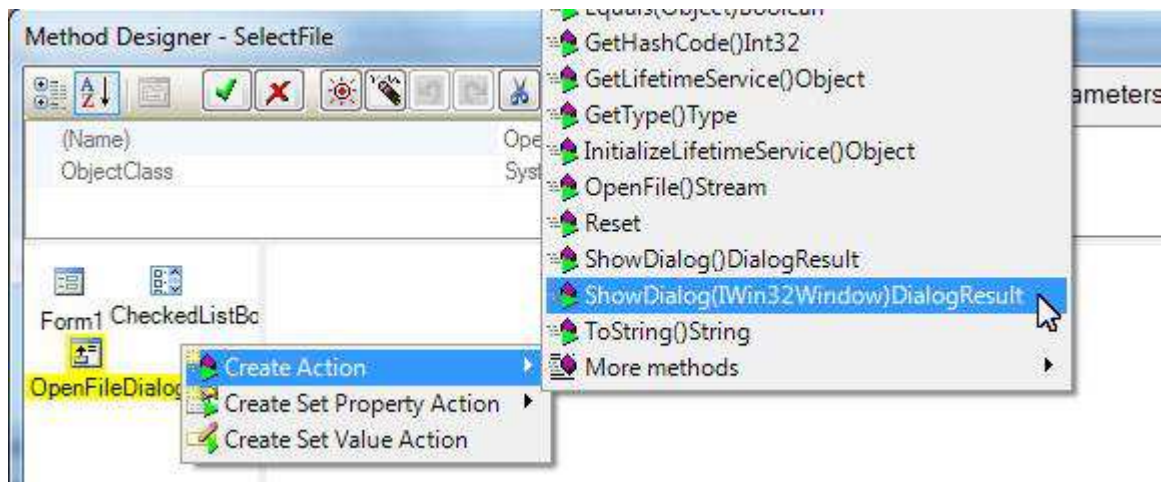
Create a method:



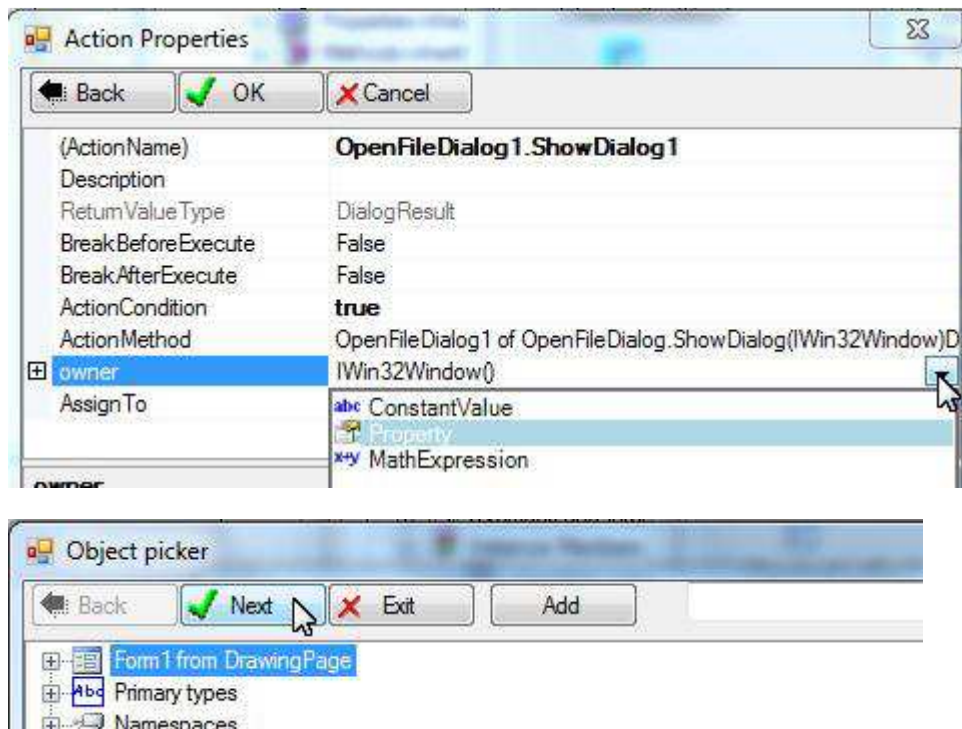
Name it "SelectFiles":

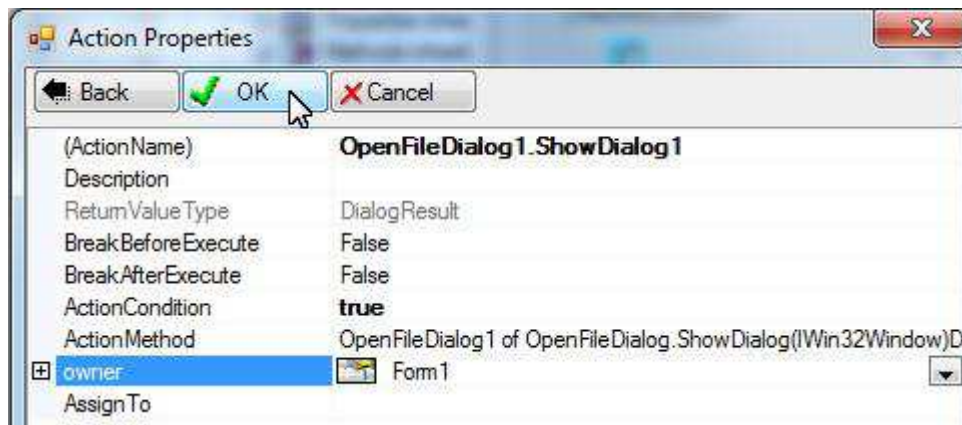


Right-click OpenFileDialog1; choose "Create Action"; choose "ShowDialog":

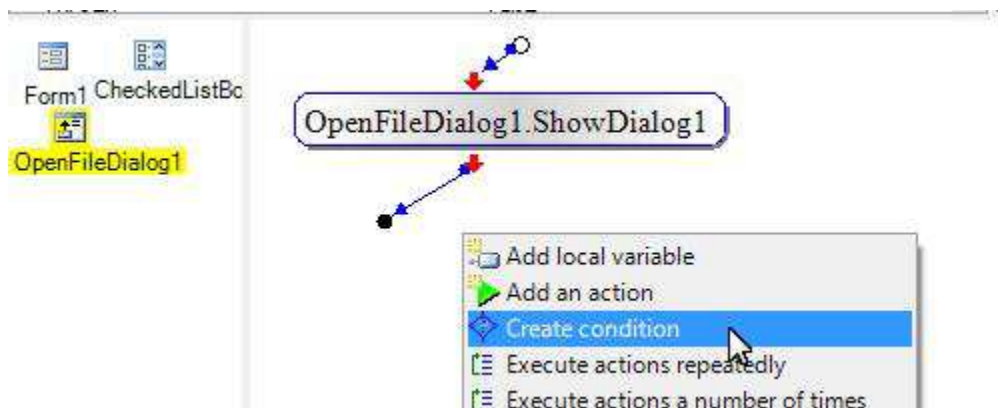


Use the current form as the owner of the dialogue box:

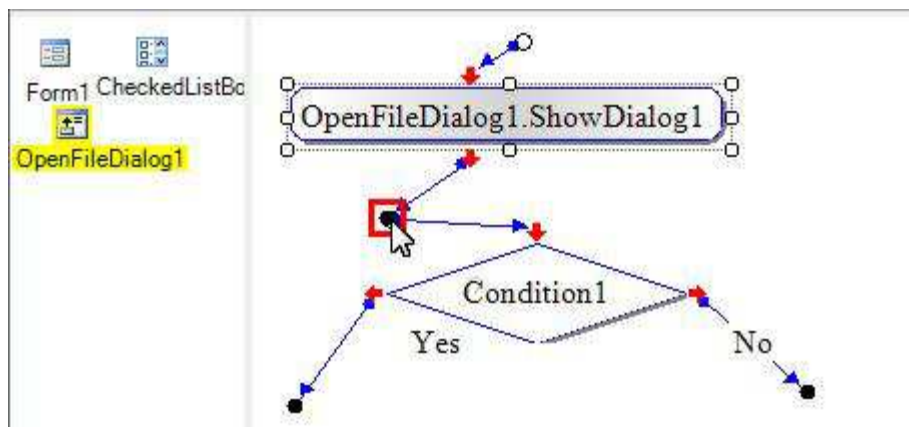




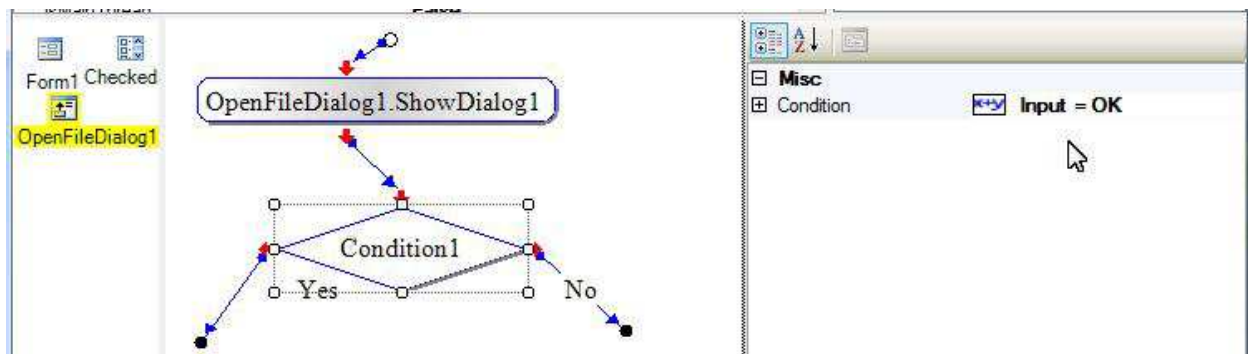
The action appears in the Method Editor. Add a condition action to control the flow according to the dialogue box result:



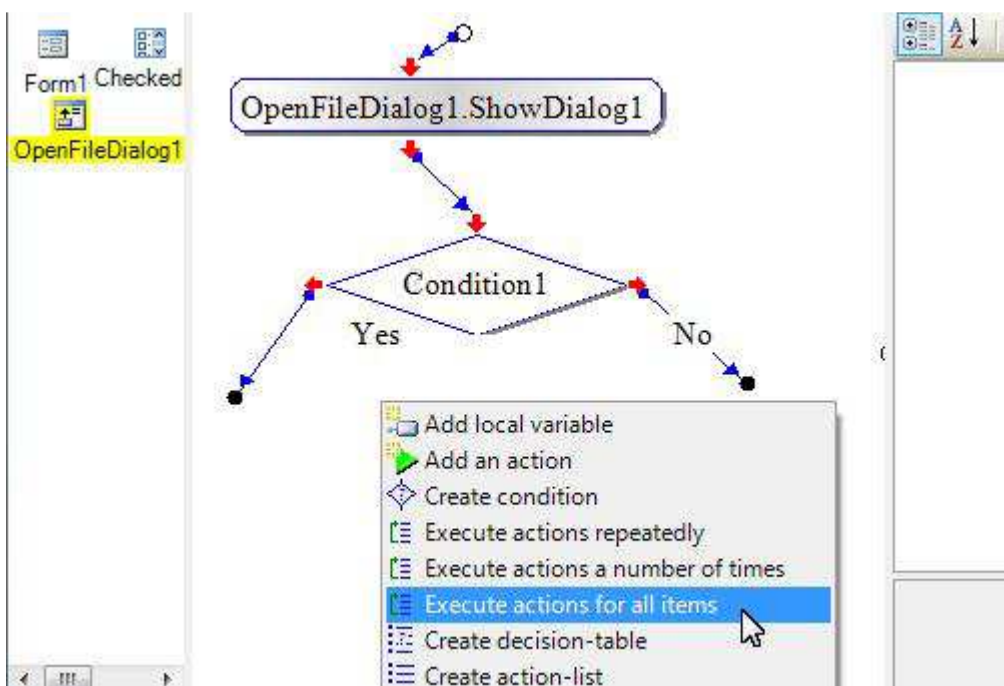
Link the condition action to the dialogue box action:



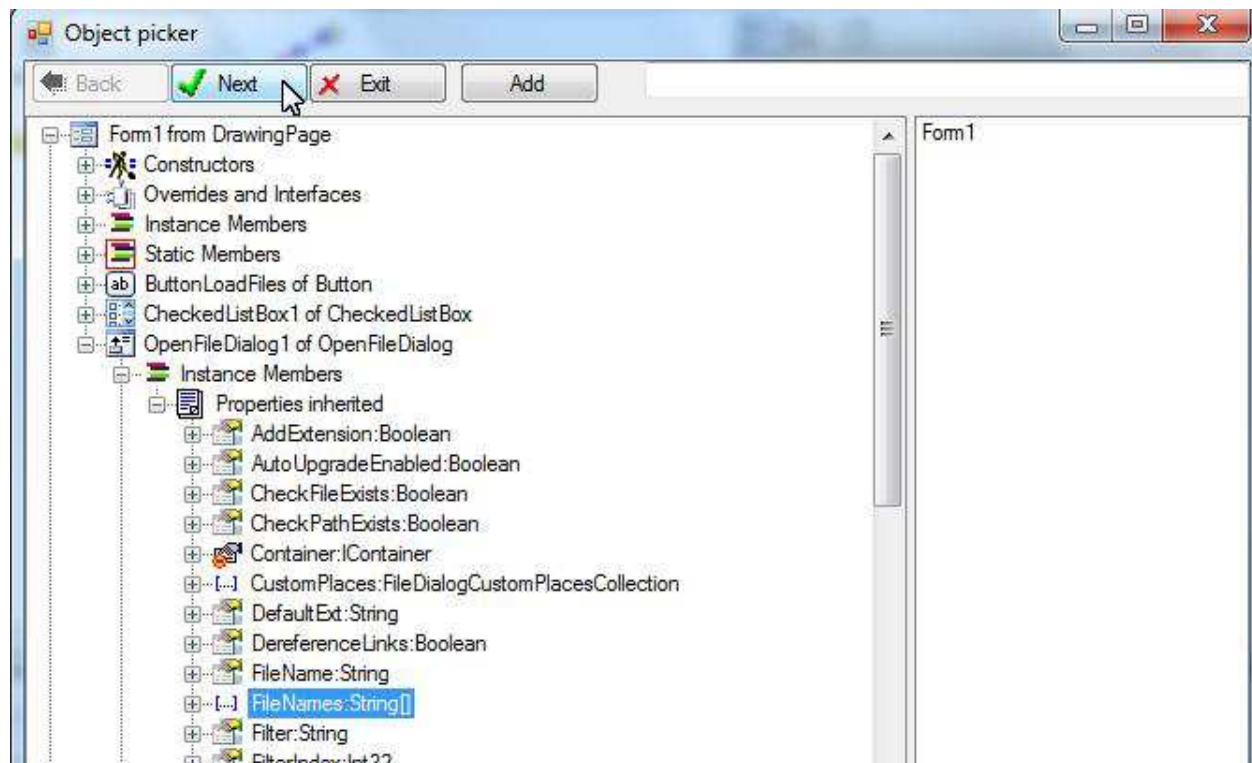
Note that by linking the condition action to the dialogue box action, the condition of the condition action is automatically set to that the dialogue box being closed by confirmation:




Create an action to use the file names selected by right-clicking the Action Pane and choosing “Execute actions for all items”:

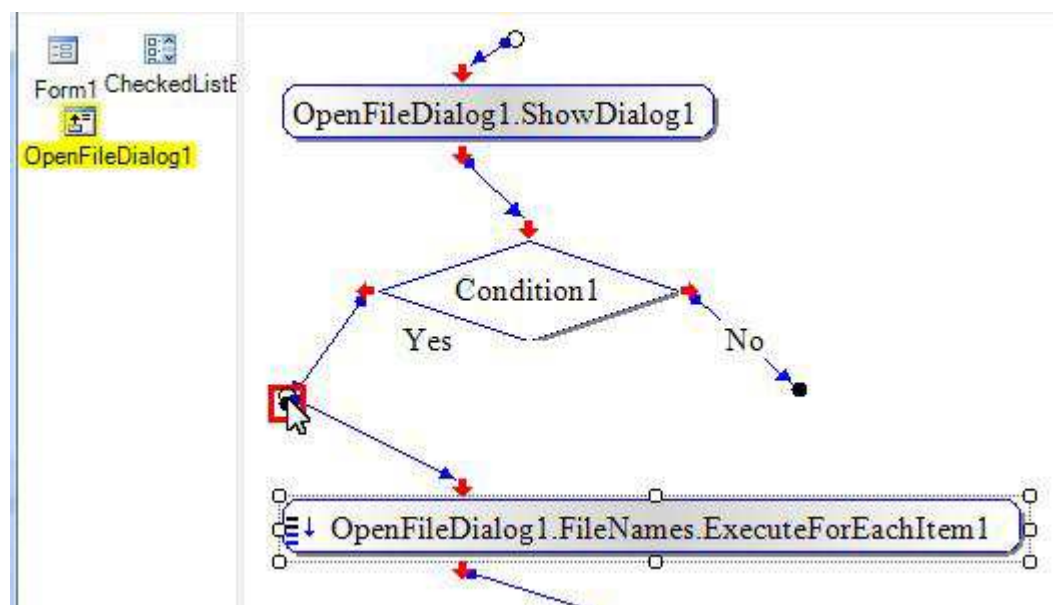


Select Filenames property of the dialogue box:



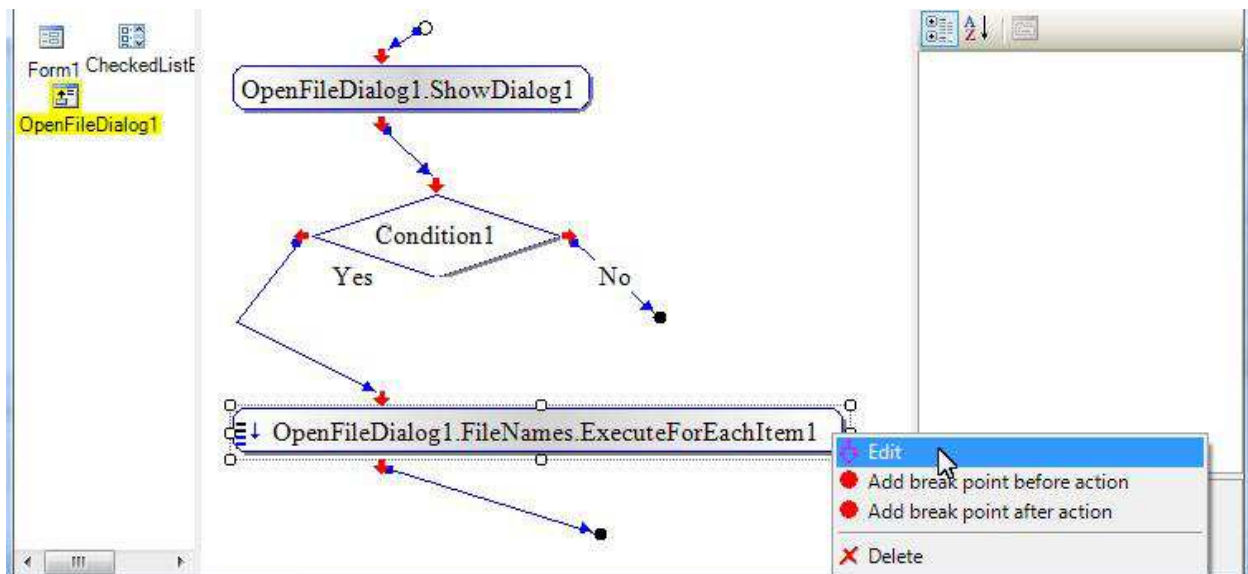
Note that icon  indicates that the property is a list, an array, or a collection of data. Only such a property can be selected for “Execute actions for all items”.

Link the new action to the Yes port of the condition so that it will be executed only when the dialogue box is closed by confirmation:



Action “OpenFileDialog1.FileNames.ExecuteForEachItem1” is an indication that we want to use every one item in the FileNames collection. But we haven’t indicated what actions we want to execute for

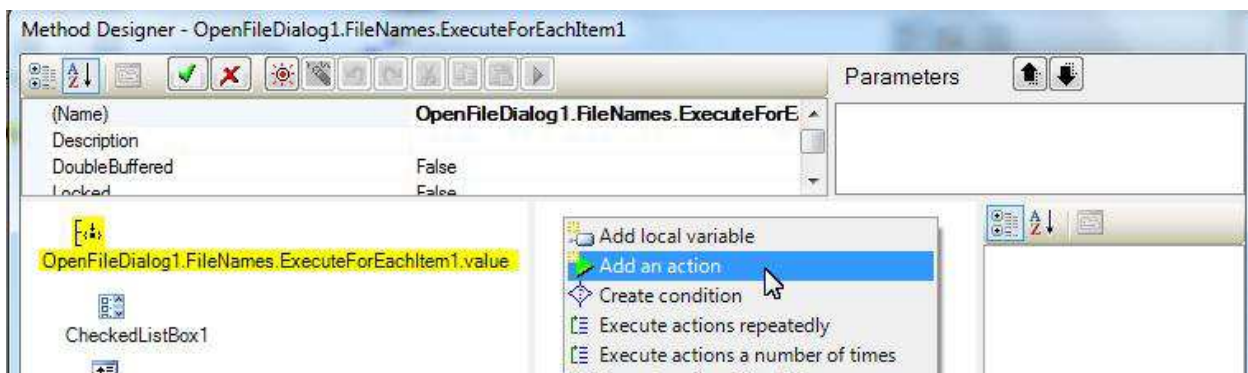
each item. Edit action “OpenFileDialog1.FileNames.ExecuteForEachItem1” to specify the actions to be executed for each file name:



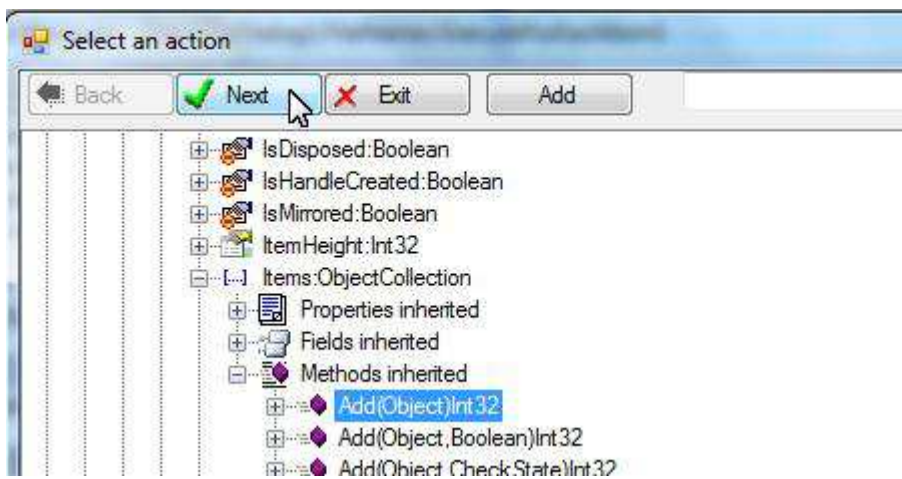
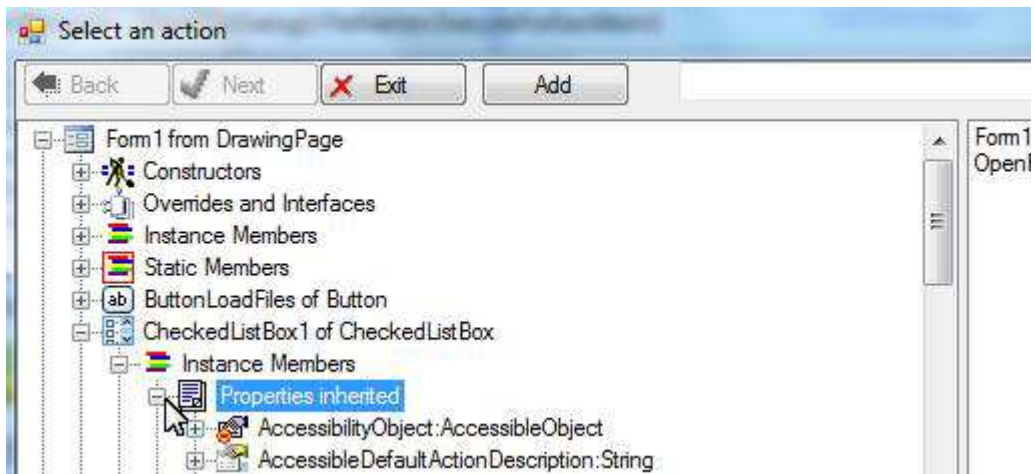
A new Method Editor appears for us to specify actions to be executed for each file name.

Note that the variable named “OpenFileDialog1.FileNames.ExecuteForEachItem1.value” represents the file name to be acted on.

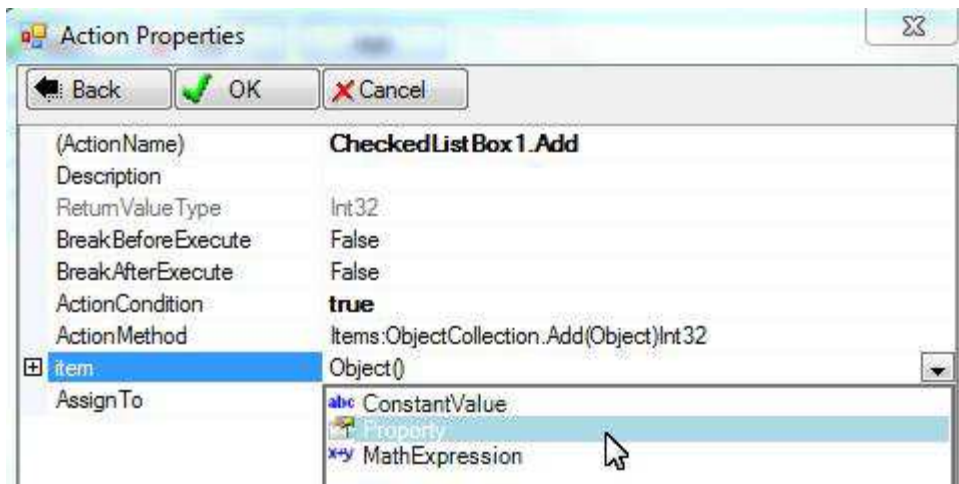
For this sample, we want to add this file name to the list box. We need to create an action to do it. Right-click the Action Pane; choose “Add an action”:



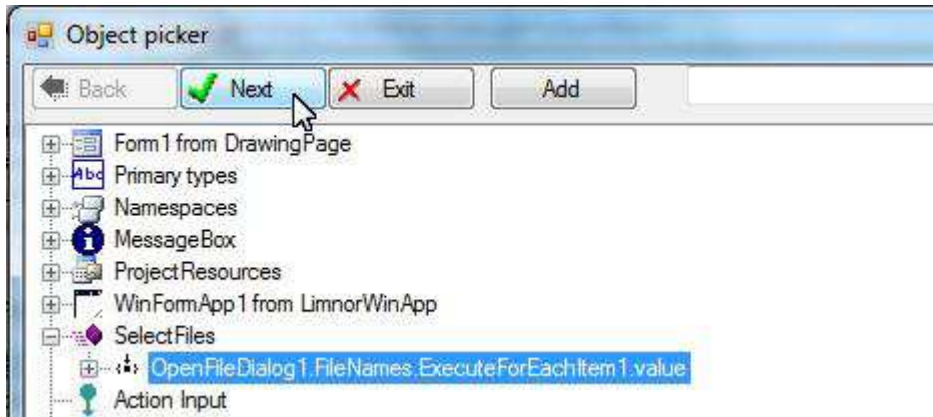
Find and select the Add method of the Items property of the list box:



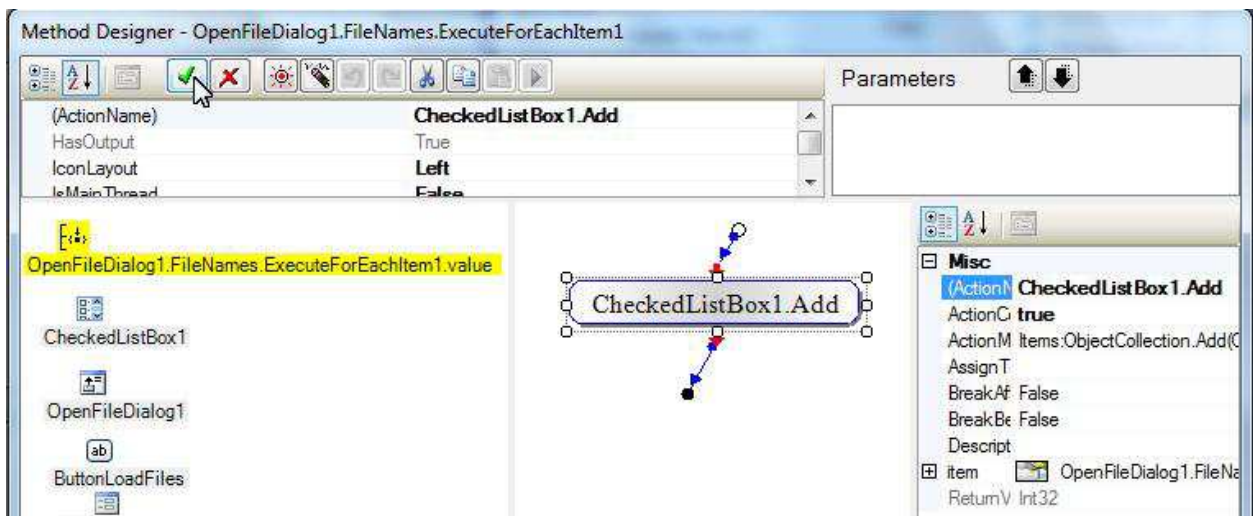
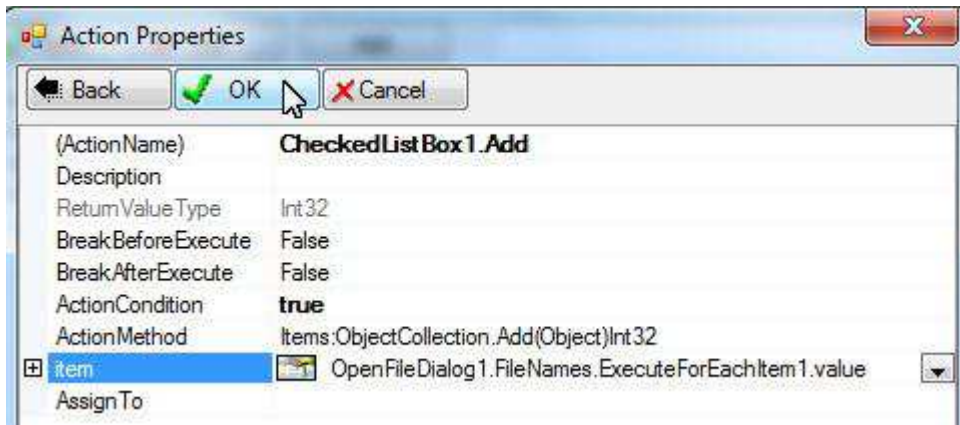
Choose "Property" for the "item" parameter of the action:



Select "OpenFileDialog1.FileName.ExecuteForEachItem1.value"

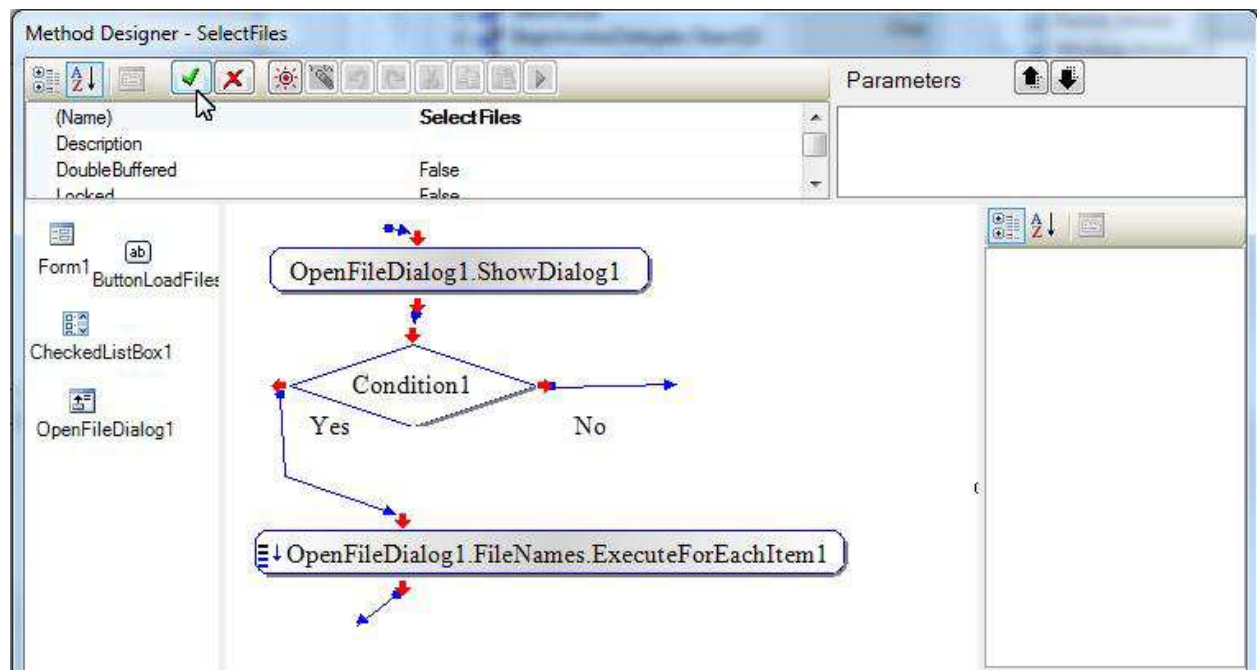


Note that “OpenFileDialog1.FileName.ExecuteForEachItem1.value” represents the file name being used, so, this action adds the file name to the list box.



Note that because all actions in this Method Editor will be executed for all file names, thus all file names will be added to the list box.

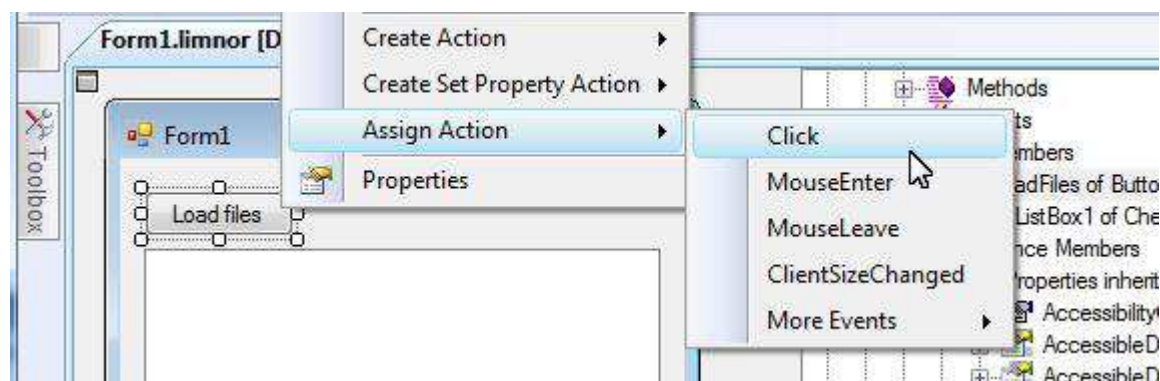
We are done creating this method:



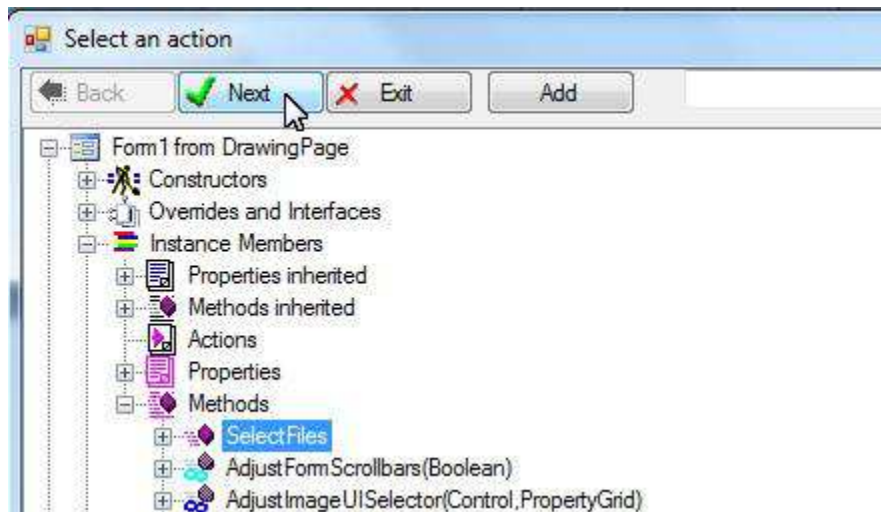
Execute the method

Suppose we want to execute the above method when the user clicks the button.

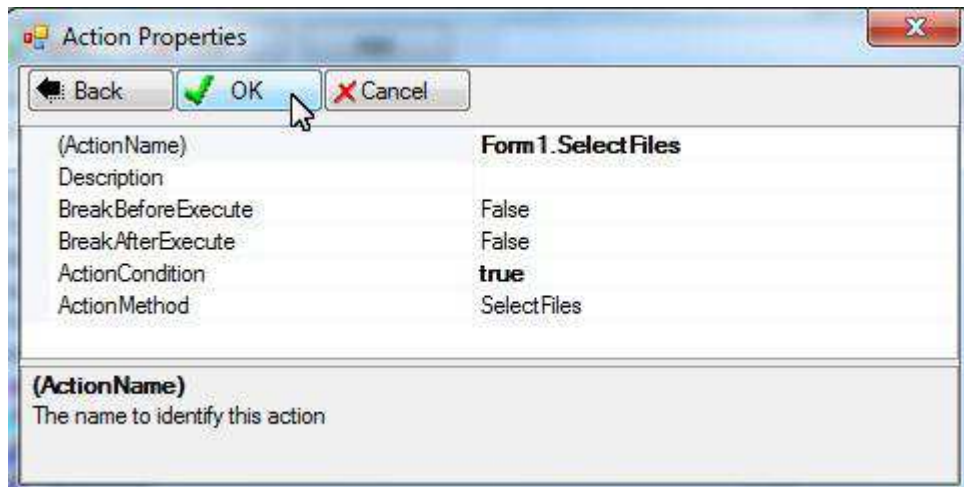
Right-click the button; choose “Assign Action”; choose “Click” event:



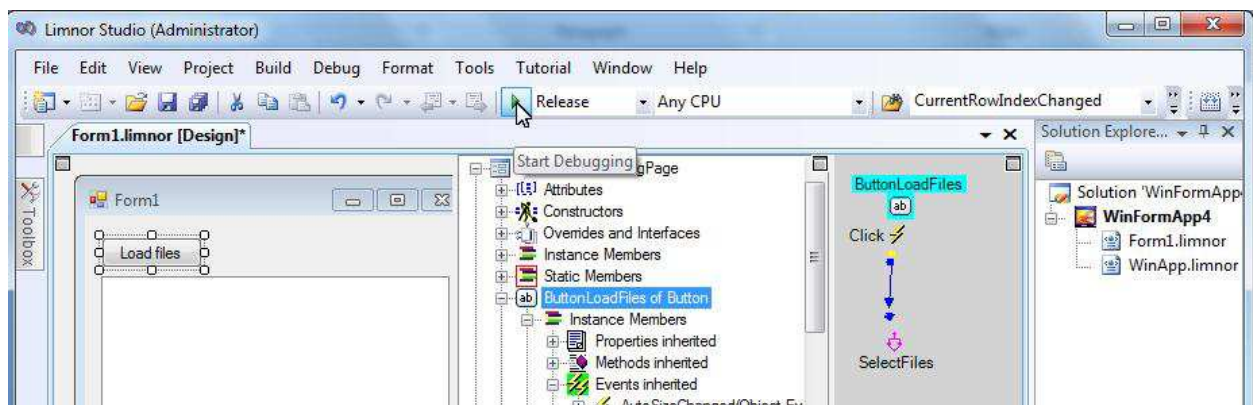
Select the method we just created:



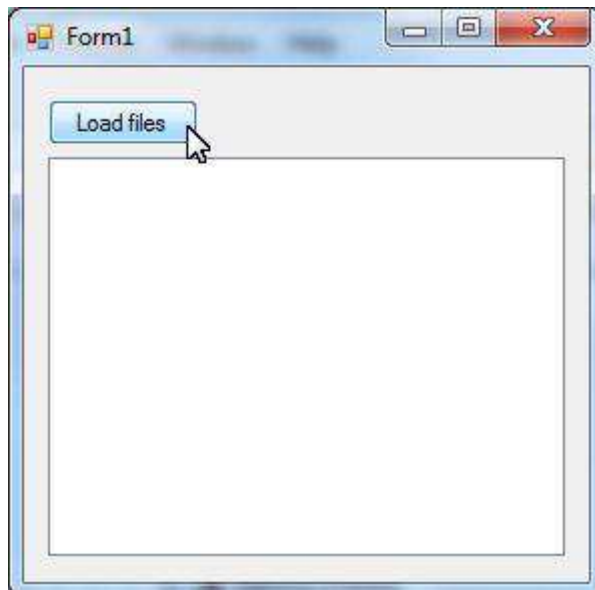
Click OK to create an action and assign the action to the button:



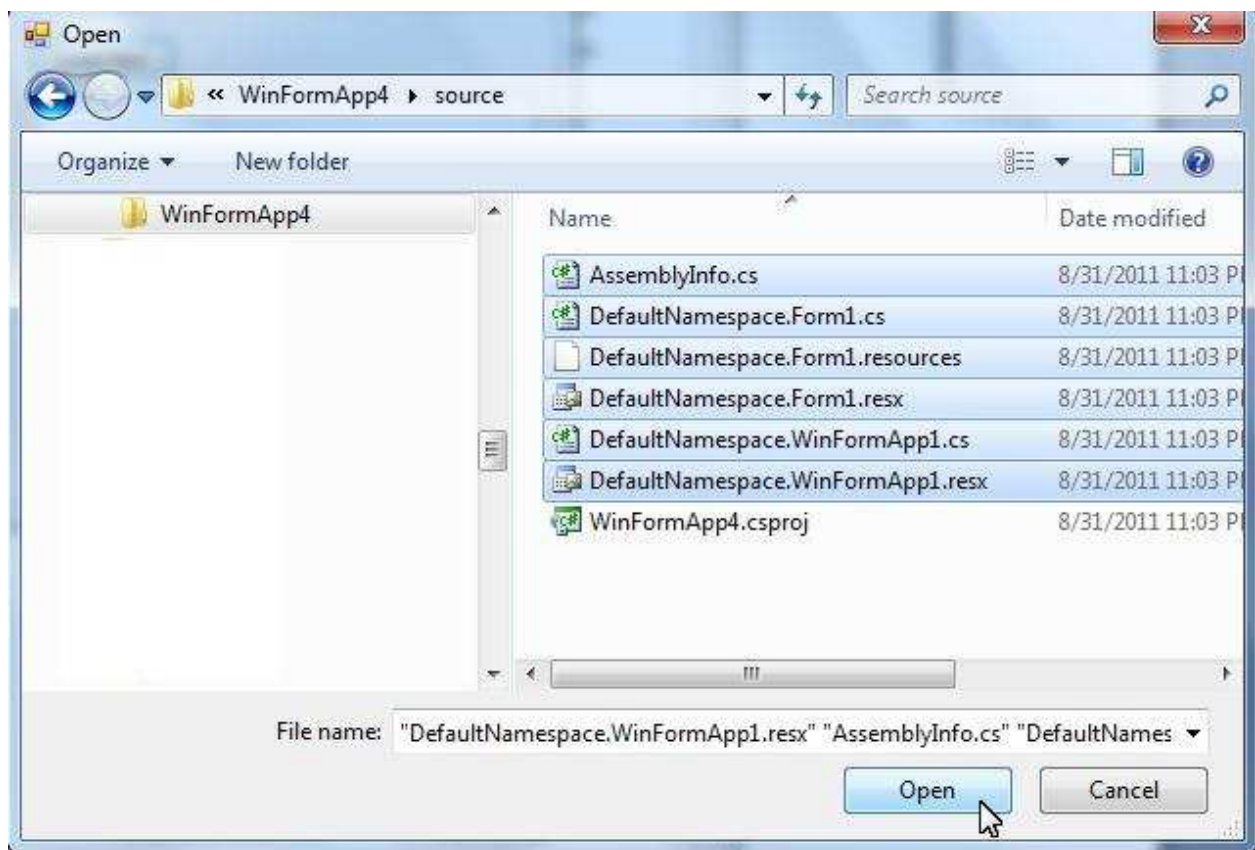
We may now test the application:



A form appears. Click the button:



A file selection dialogue box appears. Select some files and click Open:



The selected files appear in the checked list box:

