How to Use NotifyIcon

Contents
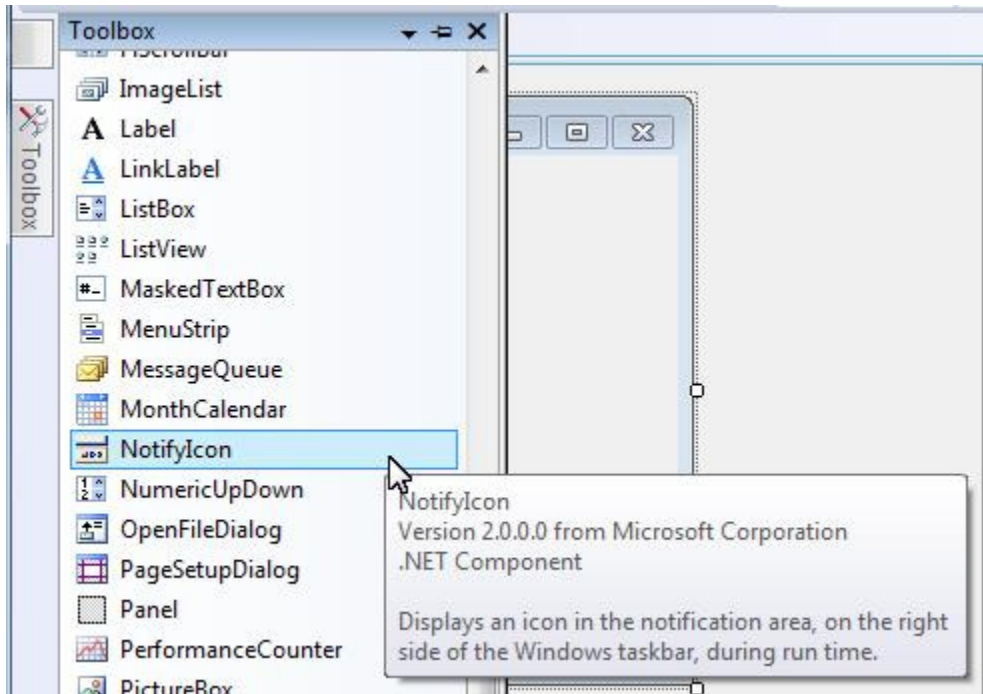
NotifyIcon Class

When a program runs and a Form is loaded, if the form contains a NotifyIcon instance as its component then the NotifyIcon will appear in the System Tray as an icon.
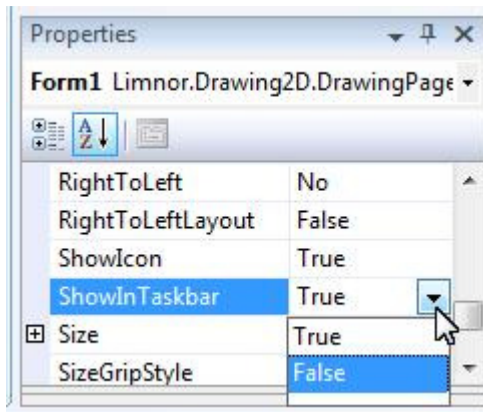
Use NotifyIcon on a form
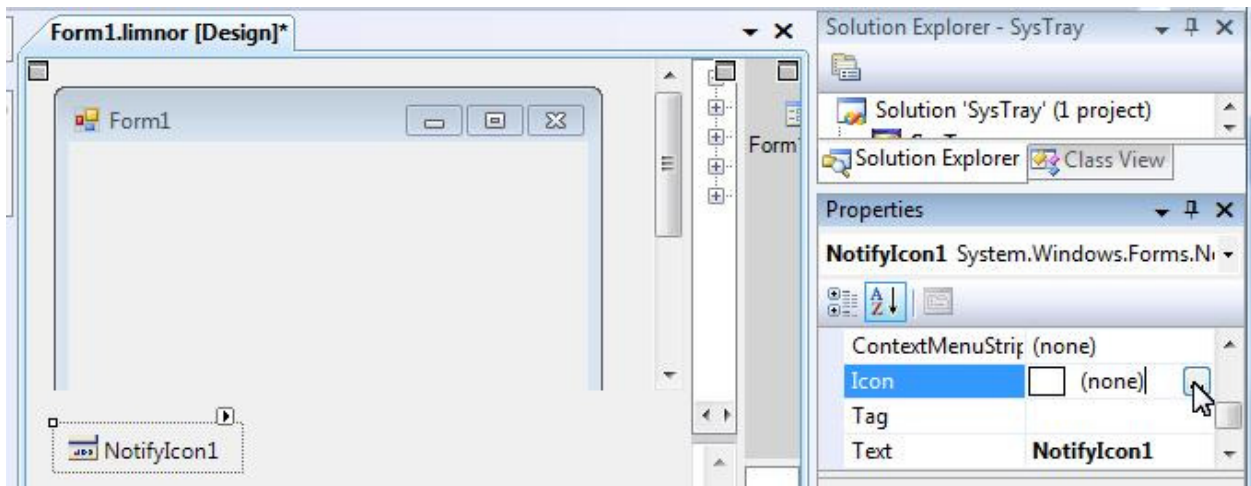
Drop NotifyIcon to a form:

Remove form from the Taskbar

Since the application will be represented by an icon in the System Tray, we may not want the form to appear in the Taskbar. Set the form's ShowInTaskbar property to False:
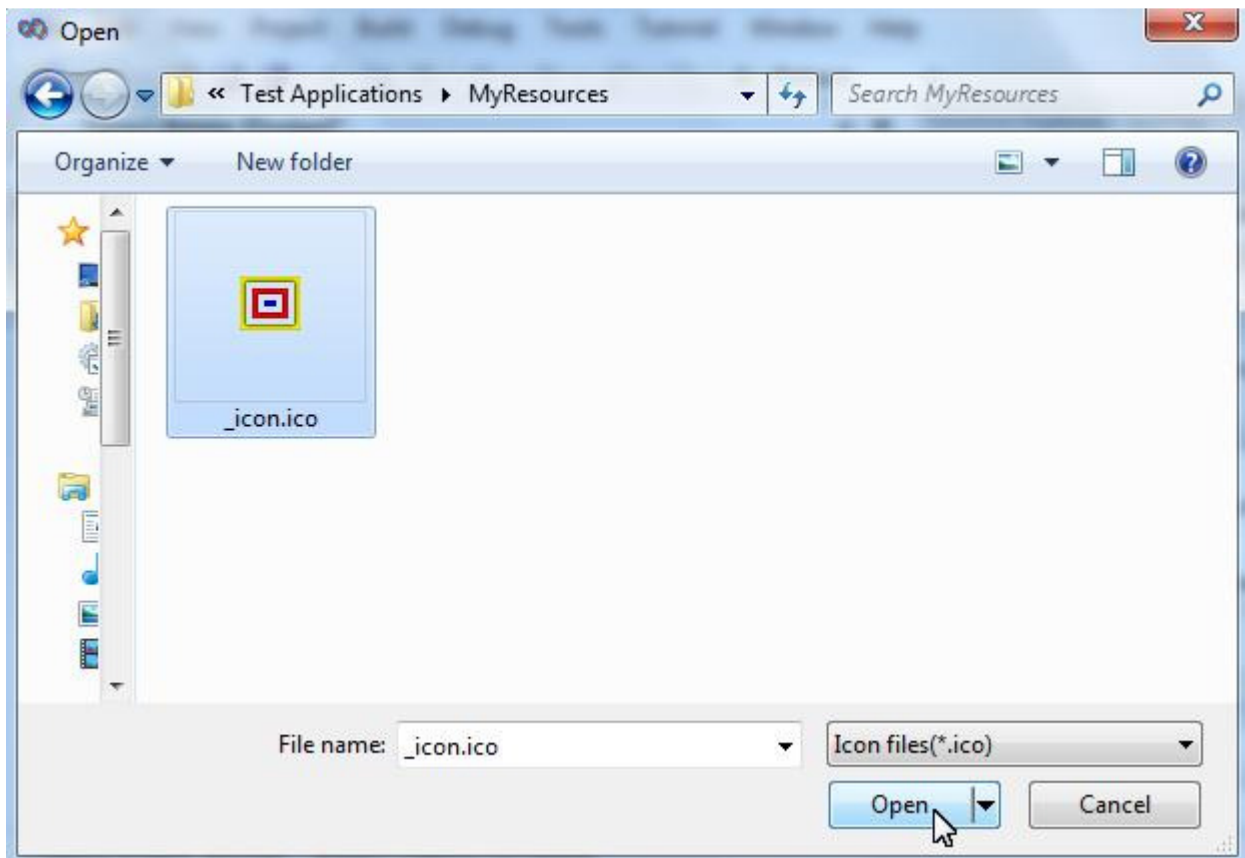

Set Icon

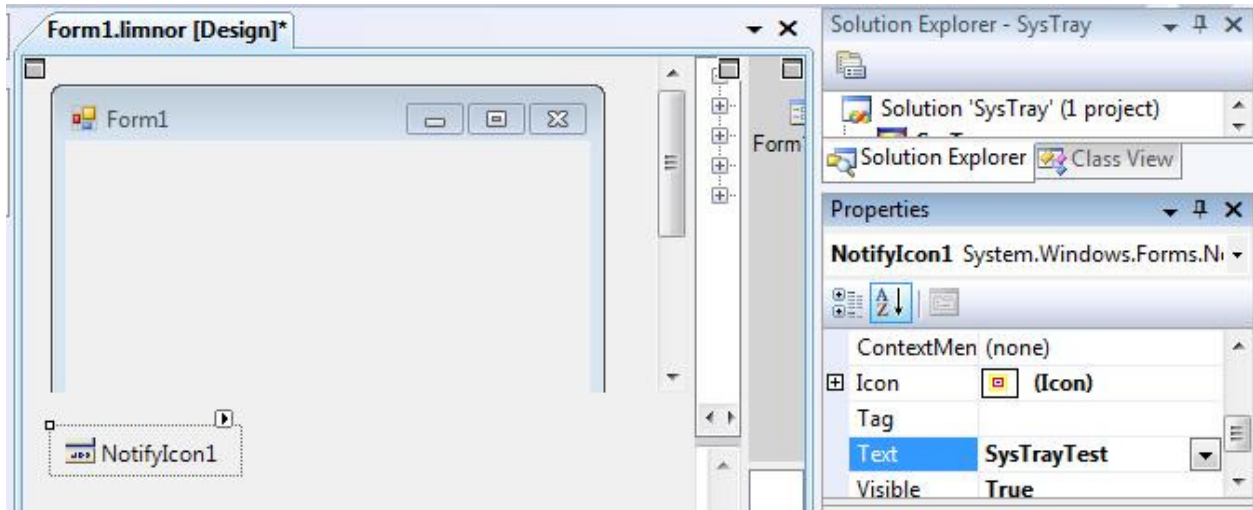We must set its Icon property to a valid icon file:

Select an icon file for NotifyIcon1:

Set title
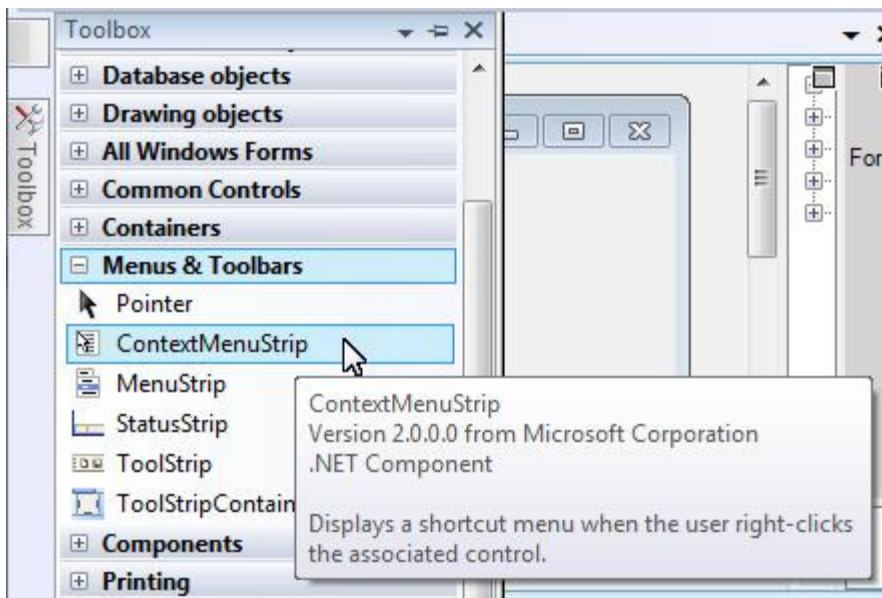
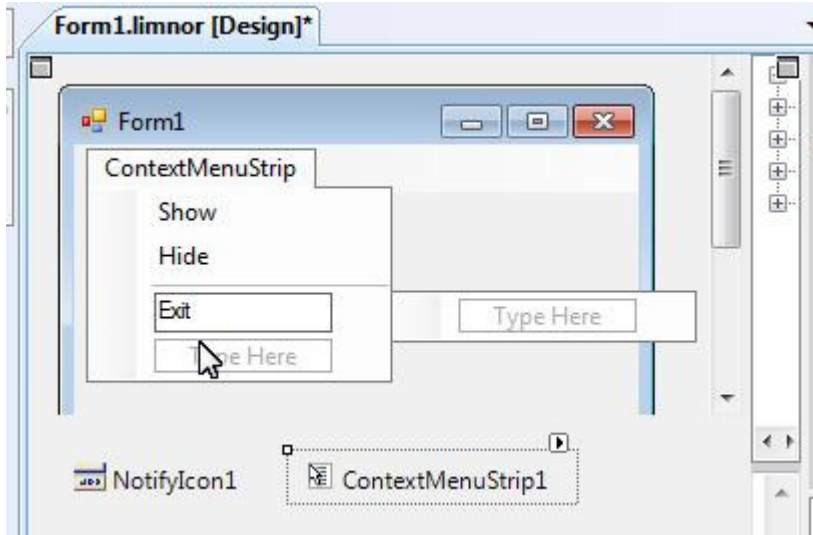Set Text property of NotifyIcon1 to a title we want to use:

Menu for the icon

For an icon in the system tray we may use menus to provide interface for application functionality.

Create context menu
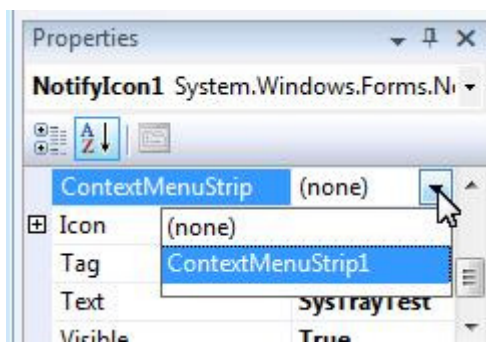
Drop ContextMenuStrip to the form:

Create some menu items. In this sample, we create menu items Show, Hide, and Exit.
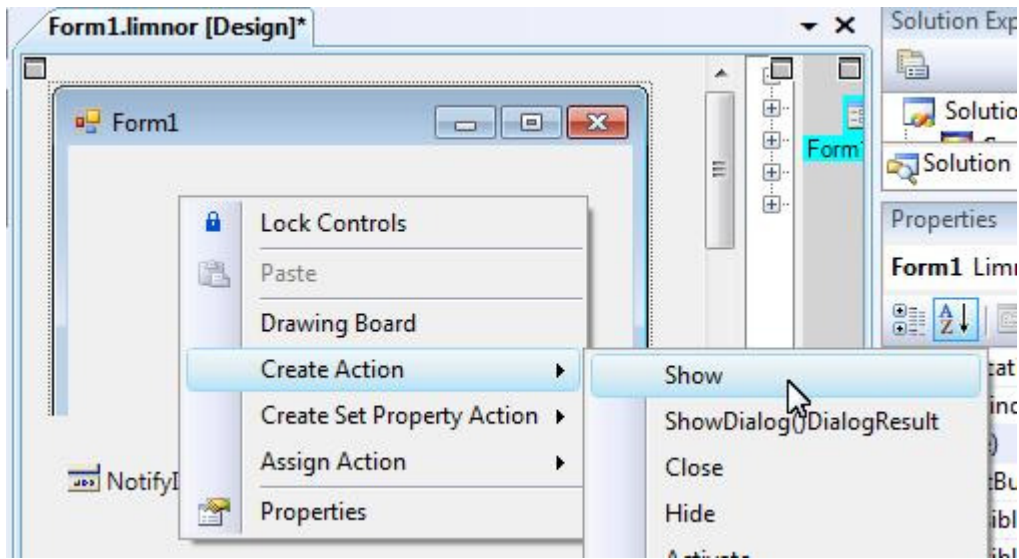
Set menu to NotifyIcon

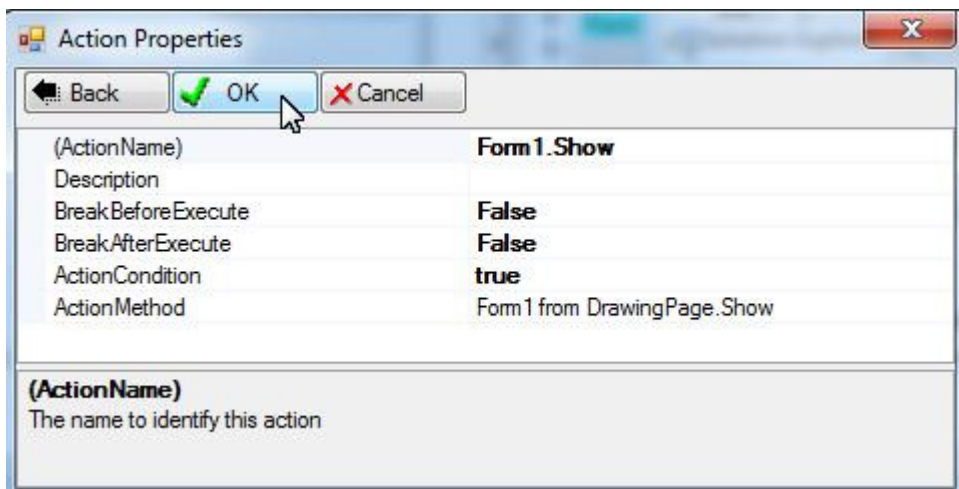Set ContextMenuStrip property of NotifyIcon1 to the ContextMenuStrip1:

Create actions

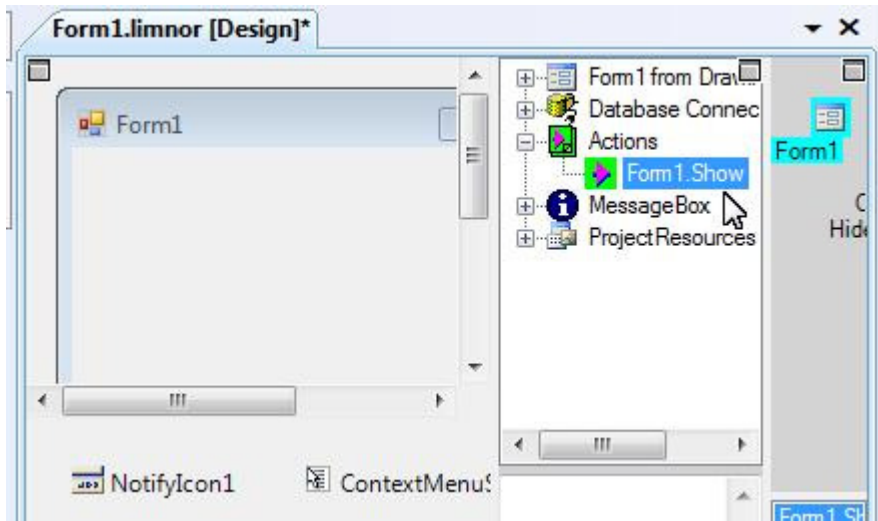Create 3 actions, using Show, Hide and Close methods, for the menu items.

Right-click the form, choose "Create action". Choose "Show" method:
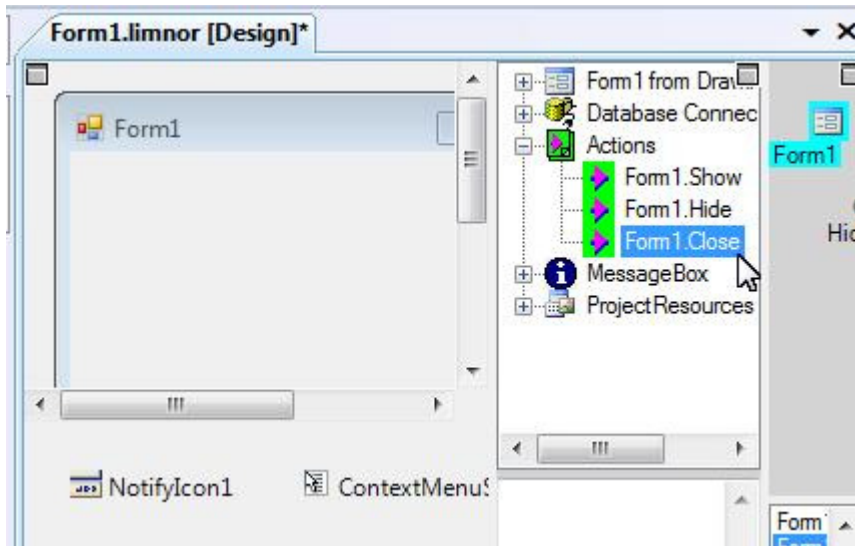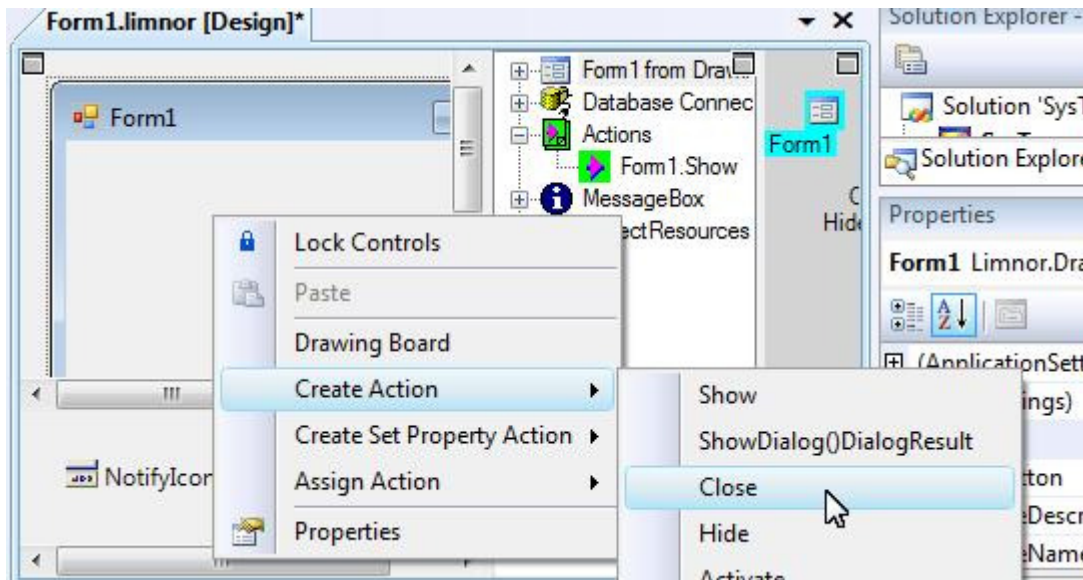
Click OK:

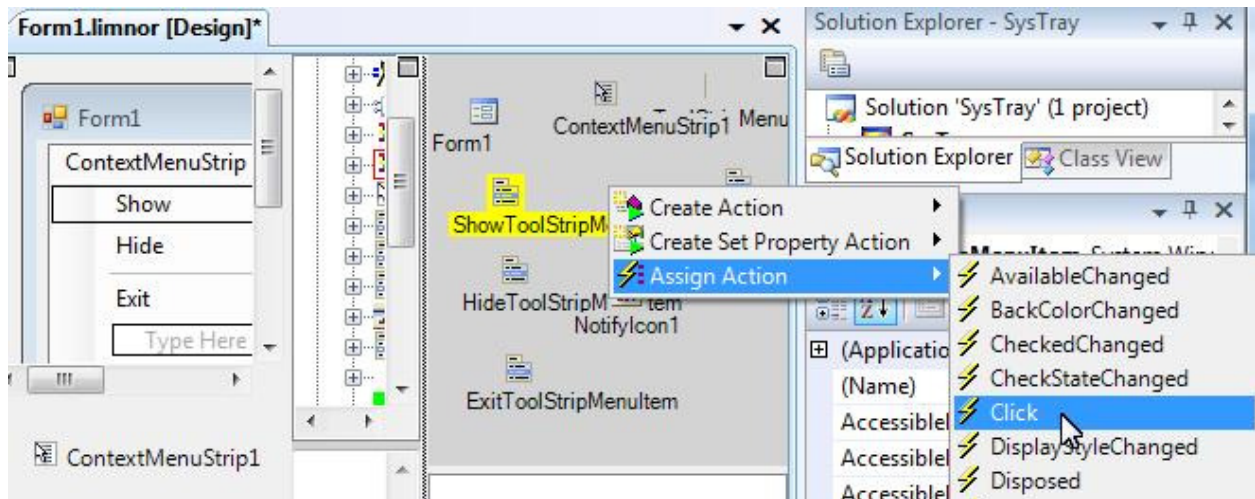A Show action, Form1.Show, is created:

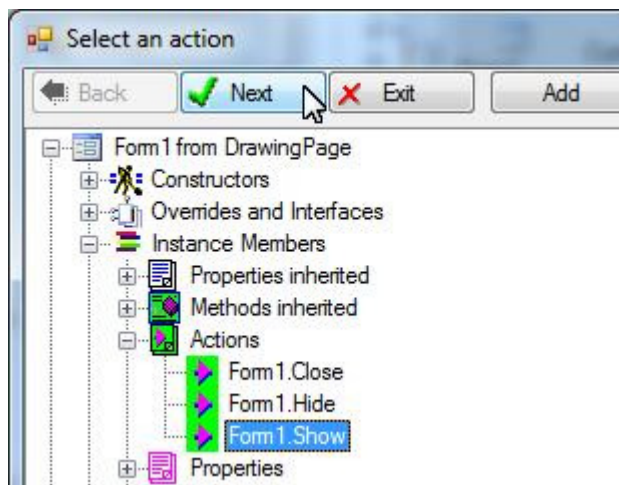Similarly, we may create the other two actions:
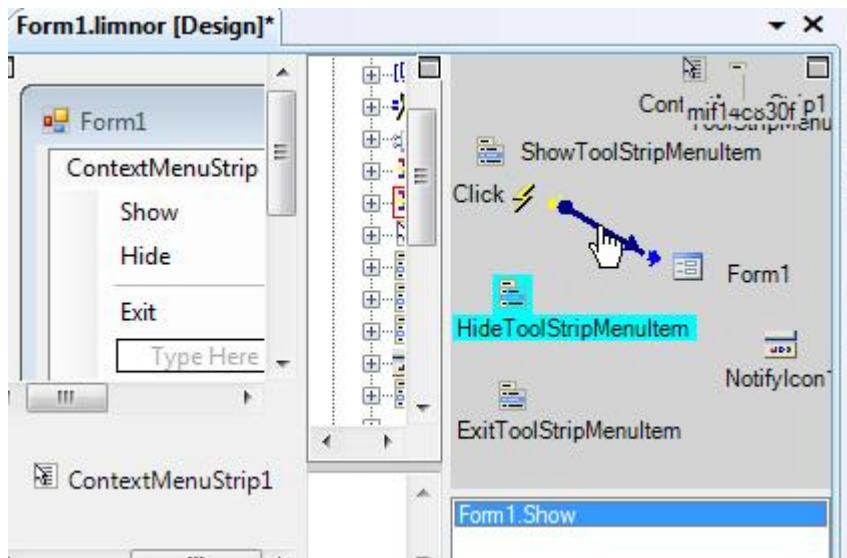
Assign actions to menu items

Right-click the menu item for Show menu; choose "Assign Action"; choose "Click" event:
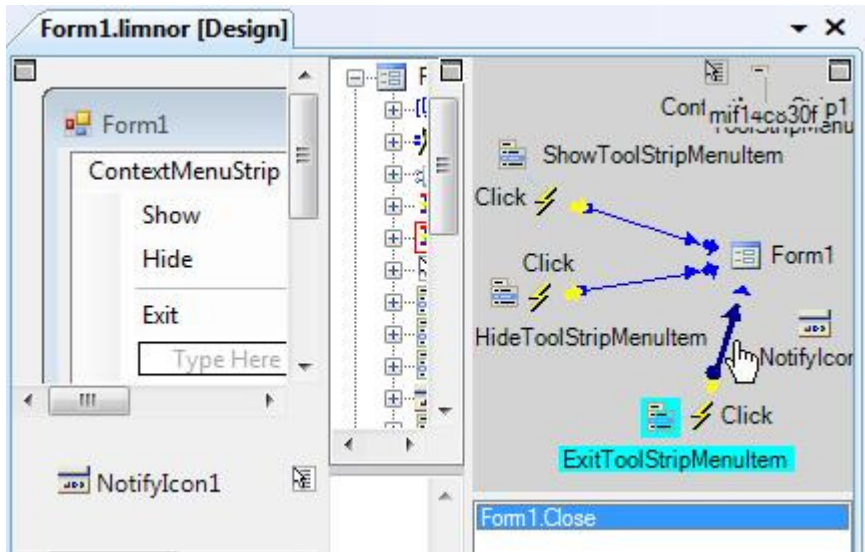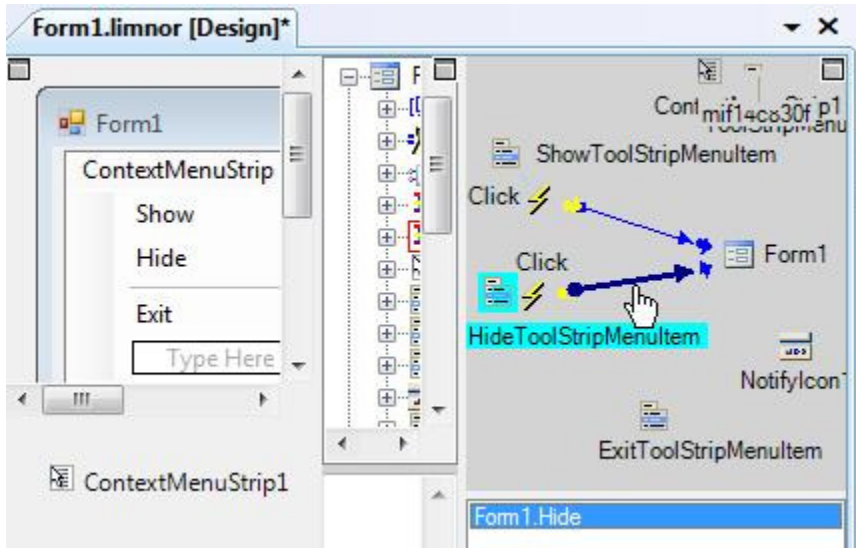
Select action Form1.Show; click Next:

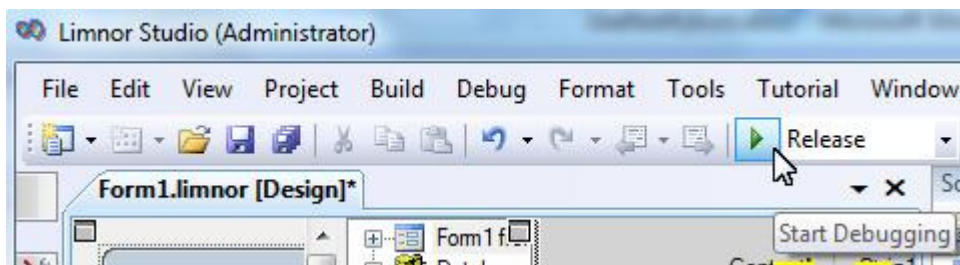The action is assigned to the menu item:

We may assign actions Form1.Hide and Form1.Close to menu items respectively:
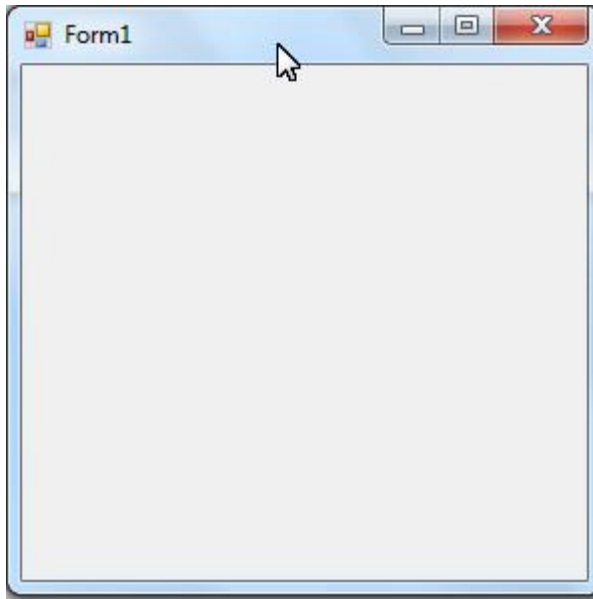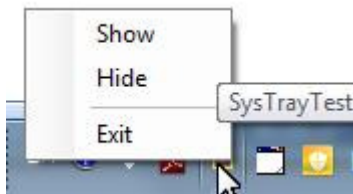

Test

We may test the application now

The Form1 appears:

NotifyIcon1 appears in the System Tray:

Right-click the icon, the menu appears:

Choose menu "Hide", the form disappears.

Choose menu "Show", the form appears.

Choose menu "Exit", the application exits.