Extend Functionality When You Create an Activity

Last Modified on 04/25/2017 3:31 am EDT

Oftentimes, an activity requires additional functionality. To accomplish this, you can write an extension for the activity.

Activate the Code

1. Add a class library to your solution. Add the relevant references.
   - *PNMsoft.Sequence*
   - *PNMsoft.Sequence.Forms*
   - *PNMsoft.Sequence.Security*
   - *PNMsoft.Sequence.Extension*
2. Strongly sign the project.
3. Complete the code in the relevant places, and compile the assembly.
4. Put the assembly in the GAC.
5. Insert a new record into *tblTemplateWorkflowExtensions*.
   a. In *fldType*, type the assembly strong name.
   b. In *fldGuid*, type the GUID in the class decorator.

NOTE: If your code raises an exception, execution will continue.

```csharp
//****************************************************
//
// Copyright (C) PNMsoft. All rights reserved.
//
// THIS CODE IS PROVIDED *AS IS* WITHOUT WARRANTY OF
// ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING ANY
// IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR
// PURPOSE, MERCHANTABILITY, OR NON-INFRINGEMENT.
//
//****************************************************

// Assemblies
//
//
//
//
//


using System;
using PNMsoft.Sequence;
using PNMsoft.Sequence.Forms.Activities;

namespace SequenceEx.Samples.TaskActivityExtensions
{
 [WorkflowElementExtension("put extension guid")]
 public class ExecutionListenerTaskActivityExtension : WorkflowElementExtension
 {
 public ExecutionListenerTaskActivityExtension()
 {
 }

 protected override void OnAttached()
 {
 base.OnAttached();

 this.Owner.InstanceCreated += new EventHandler(OnActivityInstanceCreated);
 }

 protected override void OnDetached()
 {
 base.OnDetached();

 this.Owner.InstanceCreated -= new EventHandler(OnActivityInstanceCreated);
 }

 private void OnActivityInstanceCreated(object sender, ActivityInstanceEventArgs e)
 {
 }
 }
}
```