

Invoke One-Click Action

Last Modified on 04/24/2017 7:43 am EDT

Use the API to invoke the One-Click Action feature.

```
//*****
//
// 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.
//
//*****

using System;
using PNMsoft.Sequence;
using PNMsoft.Sequence.Activities;
using PNMsoft.Sequence.Data;
using PNMsoft.Sequence.Expressions;
using PNMsoft.Sequence.Forms.Activities;
using PNMsoft.Sequence.Messaging;

/*
Assemblies:
    PNMsoft.Sequence
    PNMsoft.Sequence.Security
    PNMsoft.Sequence.Extensions
    PNMsoft.Sequence.Forms
*/
namespace SequenceEx.Samples
{
    // This sample shows how to invoke a One Click Action programmatically.
    // NOTE:
    // You need at least a ReadAndWrite permission in order to perform this operation.
    internal class InvokingOneClickActionSample
    {
        public void Invoke(TaskInstance taskInstance, string actionName)
        {
            Activity taskActivity = (Activity)taskInstance.ActivityInstance.Activity;
            TaskActivityOneClickActionExtension oca = taskActivity.Extensions.Find();
            if (oca.OneClickAction != null)
            {
                foreach (var command in oca.OneClickAction.Commands)
                {
                    if (string.Equals(command.Name, actionName, StringComparison.OrdinalIgnoreCase))
                    {
                        InvokeCommand(command, taskInstance);
                    }
                }
            }
        }

        private void InvokeCommand(OneClickActionCommand command, TaskInstance taskInstance)
        {
            TaskExecutor executor = new TaskExecutor(taskInstance);
            DataContextManager dataManager = executor.CreateDataManager();

            EntityPropertyAssignOperationContext operationContext = new EntityPropertyAssignOperationContext(
                dataManager);

            using (AssignContext assignCtx = new AssignContext(RuntimeProperties.Create(taskInstance)

```

```
        using (AssignContext assignCtx = new AssignContext(runtimeExpression.Create(taskInstance), operationContext))
        {
            command.Assignments.Invoke(assignCtx, /*throwOnError*/true);
        }

        executor.SubmitChanges(dataManager, true, true);
    }
}
```