using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Microsoft.Crm.Sdk;

using Microsoft.Xrm.Sdk;

using Microsoft.Xrm.Sdk.Query;

using Microsoft.Xrm.Sdk.Client;

namespace PreImage

{

//Step: Creating new Related Entity Record once the Parent Entity is updated.

//Post Message Image needs to be used during Update Operation

public class Class1 : IPlugin

{

//Step: Creating a new Associate once a Candidate is updated

//Message: Update

//Step: Post-Operation

//Image: PostImage

public void Execute(IServiceProvider serviceProvider)

{

IPluginExecutionContext context = (IPluginExecutionContext)

serviceProvider.GetService(typeof(IPluginExecutionContext));

IOrganizationServiceFactory serviceFactory = (IOrganizationServiceFactory)serviceProvider.GetService(typeof(IOrganizationServiceFactory));

IOrganizationService service = serviceFactory.CreateOrganizationService(context.UserId);

if (context.MessageName.Equals("Update"))

{

if (context.PostEntityImages.Contains("Target"))

{

Entity postMessageImage = (Entity)context.PostEntityImages["Target"];

if (postMessageImage.Attributes.Contains("weavings\_name"))

{

Entity Associates = new Entity("weavings\_associates");

Associates["weavings\_lastname"] = postMessageImage.Attributes["weavings\_name"].ToString();

Associates["weavings\_dateofbirth"] = postMessageImage.Attributes["weavings\_dateofbirth"];

Associates["weavings\_gender"] = postMessageImage.Attributes["weavings\_gender"];

Associates["weavings\_source"] = postMessageImage.Attributes["weavings\_source"];

Associates["weavings\_position"] = postMessageImage.Attributes["weavings\_position"];

service.Create(Associates);

}

}

}

}

}

}



