'For more information, see Custom Timer Services in Grooper x Change: https://xchange.grooper.com/discussion/779/custom-timer-service#latest

Imports Grooper

Imports Grooper.Services

Imports Grooper.Core

''' <summary>Used to delete production batches in the Grooper demo environment.</summary>

Public Class CustomTimerService : Inherits TimerService

 Private \_lastRun As DateTime = DateTime.Now

 <DataContract>

 Private Class Properties : Inherits TimerServiceProperties

 <DataMember> Public RunOnlyBefore8AM As Boolean = True

 End Class

 Private Props As Properties

#Region "Persistant Properties"

 ''' <summary>Determines if the service should only execute between midnight at 8:00 AM each day.</summary>

 <DisplayName("Run Before 8:00 AM"), Category("Delete Production Batches"), Viewable, UI("Grooper.UI.PgFolderPathEditor")>

 Public Property RunOnlyBefore8AM As Boolean

 Get

 Return Props.RunOnlyBefore8AM

 End Get

 Set(value As Boolean)

 Props.RunOnlyBefore8AM = value

 End Set

 End Property

#End Region

 Protected Overrides ReadOnly Property BaseProps As TimerService.TimerServiceProperties

 Get

 Return Props

 End Get

 End Property

 Protected Overrides Property PropertiesJson As String

 Get

 Return ToJson(Props)

 End Get

 Set(value As String)

 If (value = "") Then Props = New Properties : Return

 Props = FromJson(Of Properties)(value)

 End Set

 End Property

 Public Overrides ReadOnly Property ServiceDescription As String

 Get

 Return "Delete Production Batches is a service that deletes all Production [Batches] in Grooper."

 End Get

 End Property

 Private gdb As GrooperDb

 Public Sub New(gdb As GrooperDb)

 MyBase.New(gdb)

 Me.gdb = gdb

 Props = New Properties

 End Sub

 Protected Overrides Sub Tick()

 If (RunOnlyBefore8AM) Then

 ' Only allow the script to run from midnight up to 8 AM. Do not allow to run after 8 AM.

 If (DateTime.Now.Hour >= 8) Then Return

 End If

 If (\_lastRun.Date < DateTime.Now.Date) Then \_lastRun = DateTime.Now.Date : DeleteBatches()

 End Sub

#Region "Private Methods"

 Private Sub DeleteBatches()

 Dim gdbTemp As New GrooperDb(gdb.DataSource)

 If (gdbTemp.Connect() = False) Then Throw New GrooperException("Error connecting to database: {0}", gdbTemp.GetLastError())

 Dim BatchList As List(Of Guid) = New List(Of Guid)

 Dim ProdFolder As Folder = gdbTemp.Root.GetSpecialFolder(GrooperRoot.SpecialFolder.Production)

 Dim BatchName As String = ""

 For Each bch As Batch In ProdFolder.AllChildrenOfType(GetType(Batch))

 ' We cannot delete nodes from here because it will modify the collection, so instead we add the Guid Ids to be deleted to

 ' a list and delete those during a second step.

 BatchList.Add(bch.Id)

 Next

 For Each bchId As Guid In BatchList

 Dim bch As Batch = gdbTemp.Root.GetNode(bchId)

 BatchName = bch.DisplayName

 If (bch.ParentNode.CcDeleteChild(bchId) = False) Then gdbTemp.WriteToLog(GrooperDb.EventType.Err, "Unable to delete batch: {0}", bch.DisplayName)

 Next

 gdbTemp.Disconnect()

 End Sub

#End Region

End Class