```vb
1   Imports System.Collections.Generic
2
3   Imports Grasshopper.Kernel
4   Imports Rhino.Geometry
5   Imports Interop.gsa_8_7
6   Imports UsageDataCollection
7
8
9   Public Class BuildList
10      Inherits GH_Component
11      ''' <summary>
12      ''' Initializes a new instance of the BuildList class.
13      ''' </summary>
14      Public Sub New()
15          MyBase.New("Build List", "BuildList", _
16                  "Build List from Record Set", _
17                  "GsaComTools", "Misc")
18      End Sub
19
20      ''' <summary>
21      ''' Registers all the input parameters for this component.
22      ''' </summary>
23      Protected Overrides Sub RegisterInputParams(pManager As
          GH_Component.GH_InputParamManager)
24          pManager.AddGenericParameter("Gsa", "Gsa", "Gsa COM Object",
              GH_ParamAccess.item)
25          pManager.AddTextParameter("List Name", "Name", "Name of List",
              GH_ParamAccess.item)
26          pManager.AddIntegerParameter("List Type", "Type", "Type of List
              (0=Node, 1=Element)", GH_ParamAccess.item)
27          pManager.AddIntegerParameter("Records", "Records", "Record Numbers",
              GH_ParamAccess.list)
28      End Sub
29
30      ''' <summary>
31      ''' Registers all the output parameters for this component.
32      ''' </summary>
33      Protected Overrides Sub RegisterOutputParams(pManager As
          GH_Component.GH_OutputParamManager)
34          pManager.AddGenericParameter("Gsa", "Gsa", "Gsa COM Object",
              GH_ParamAccess.item)
35          pManager.AddTextParameter("List Number", "Num", "List Number",
              GH_ParamAccess.item)
36      End Sub
37
38      ''' <summary>
39      ''' This is the method that actually does the work.
40      ''' </summary>
41      ''' <param name="DA">The DA object is used to retrieve from inputs and
          store in outputs.</param>
42      Protected Overrides Sub SolveInstance(DA As IGH_DataAccess)
43
44          'Declare Variables
45          Dim Highest As Integer
46          Dim ListNum As Integer
47          Dim FileName As String
```

```vb
48             Dim ListName As String
49             Dim ListTypeBool As Integer
50             Dim Refs As New List(Of Integer)
51             Dim ListString As String
52             Dim ListType As String
53             Dim s As Short
54             Dim WriteString As String
55             Dim myDataCollector As New DataCollector
56             Dim myObjectWrapper As Types.GH_ObjectWrapper
57
58             'Import GSA File Name
59             If (Not DA.GetData(0, myObjectWrapper)) Then Return
60             If (Not DA.GetData(1, ListName)) Then Return
61             If (Not DA.GetData(2, ListTypeBool)) Then Return
62             If (Not DA.GetDataList(3, Refs)) Then Return
63
64             'Build List String
65             ListString = ""
66             For Each item In Refs
67                 ListString = ListString & item & " "
68             Next
69
70             'Set List Type
71             Select Case ListTypeBool
72                 Case 0
73                     ListType = "Node"
74                 Case 1
75                     ListType = "Element"
76             End Select
77
78             Dim GsaObj As ComAuto = TryCast(myObjectWrapper.Value, ComAuto)
79
80             'Get Highest List
81             Highest = CInt(GsaObj.GwaCommand("HIGHEST, LIST"))
82             ListNum = Highest + 1
83
84             'Write List
85             WriteString = "SET_AT, " & CStr(ListNum) & ", LIST, " & CStr(ListNum) ⮎
                 & ", " & ListName & ", " & ListType & ", " & ListString
86             s = GsaObj.GwaCommand(WriteString)
87
88             'Send Usage Data
89             myDataCollector.UsageStatistics("BuildList")
90
91             'Send Data
92             DA.SetData(0, GsaObj)
93             DA.SetData(1, CStr(ListNum))
94
95         End Sub
96
97     ''' <summary>
98     ''' Provides an Icon for every component that will be visible in the User ⮎
           Interface.
99     ''' Icons need to be 24x24 pixels.
100    ''' </summary>
101    Protected Overrides ReadOnly Property Icon() As System.Drawing.Bitmap
```

```vb
102            Get
103                'You can add image files to your project resources and access them ⮧
                       like this:
104                ' return Resources.IconForThisComponent;
105                Return My.Resources.Misc_BuildList
106            End Get
107        End Property
108
109        ''' <summary>
110        ''' Gets the unique ID for this component. Do not change this ID after    ⮧
               release.
111        ''' </summary>
112        Public Overrides ReadOnly Property ComponentGuid() As Guid
113            Get
114                Return New Guid("{469e526b-e777-45b4-8f31-3eb85947d832}")
115            End Get
116        End Property
117 End Class
```