

```

1 Imports System.Collections.Generic
2
3 Imports Grasshopper.Kernel
4 Imports Rhino.Geometry
5 Imports Interop.gsa_8_7
6
7 Public Class SaveGsaFile
8     Inherits GH_Component
9     ''' <summary>
10     ''' Initializes a new instance of the SaveGsaFile class.
11     ''' </summary>
12     Public Sub New()
13         MyBase.New("Save Gsa File", "Save", _
14             "Saves Gsa File", _
15             "GsaComTools", "I/O")
16     End Sub
17
18     ''' <summary>
19     ''' Registers all the input parameters for this component.
20     ''' </summary>
21     Protected Overrides Sub RegisterInputParams(pManager As GH_Component.GH_InputParamManager)
22         pManager.AddGenericParameter("Gsa", "Gsa", "Gsa COM Object", GH_ParamAccess.item)
23         pManager.AddTextParameter("SaveAs", "SaveAs", "File path and name (Optional)", GH_ParamAccess.item)
24
25         pManager(1).Optional = True
26     End Sub
27
28     ''' <summary>
29     ''' Registers all the output parameters for this component.
30     ''' </summary>
31     Protected Overrides Sub RegisterOutputParams(pManager As GH_Component.GH_OutputParamManager)
32         pManager.AddGenericParameter("Gsa", "Gsa", "Gsa COM Object", GH_ParamAccess.item)
33     End Sub
34
35     ''' <summary>
36     ''' This is the method that actually does the work.
37     ''' </summary>
38     ''' <param name="DA">The DA object is used to retrieve from inputs and store in outputs.</param>
39     Protected Overrides Sub SolveInstance(DA As IGH_DataAccess)
40         Dim myObjectWrapper As Types.GH_ObjectWrapper
41         Dim IntResult As Integer
42         Dim s As Short
43         Dim SaveAs As Boolean = True
44         Dim Path As String
45
46         'Import GSA File Name
47         If (Not DA.GetData(0, myObjectWrapper)) Then Return
48         If (Not DA.GetData(1, Path)) Then SaveAs = False
49
50         Dim GsaObj As ComAuto = TryCast(myObjectWrapper.Value, ComAuto)

```

```
51
52     'Save GSA File
53     If IntResult = 0 Then
54         If SaveAs = False Then
55             s = GsaObj.Save
56         Else
57             s = GsaObj.SaveAs(Path)
58         End If
59         IntResult = IntResult + s
60
61         Select Case s
62             Case 1
63                 AddRuntimeMessage(GH_RuntimeMessageLevel.Warning, "No GSA  ↗
64                                     file is open")
65             Case 2
66                 AddRuntimeMessage(GH_RuntimeMessageLevel.Warning, "No  ↗
67                                     default path is available; use SaveAs")
68             Case 3
69                 AddRuntimeMessage(GH_RuntimeMessageLevel.Warning, "Failed  ↗
70                                     to save")
71             End Select
72         End If
73
74         ' Assign the data to the output parameter.
75         DA.SetData(0, GsaObj)
76
77         'Send Usage Data
78         Dim myDataCollector As New DataCollector
79         myDataCollector.UsageStatistics("SaveGsaFile")
80
81     End Sub
82
83     ''' <summary>
84     ''' Provides an Icon for every component that will be visible in the User  ↗
85     Interface.
86     ''' Icons need to be 24x24 pixels.
87     ''' </summary>
88     Protected Overrides ReadOnly Property Icon() As System.Drawing.Bitmap
89     Get
90         Return My.Resources.IO_SaveGsaFile
91     End Get
92 End Property
93
94     ''' <summary>
95     ''' Gets the unique ID for this component. Do not change this ID after  ↗
96     release.
97     ''' </summary>
98     Public Overrides ReadOnly Property ComponentGuid() As Guid
99     Get
100         Return New Guid("{b56f2ea7-f5b0-4b54-9c87-d953a75862eb}")
101     End Get
102 End Property
103 End Class
```