

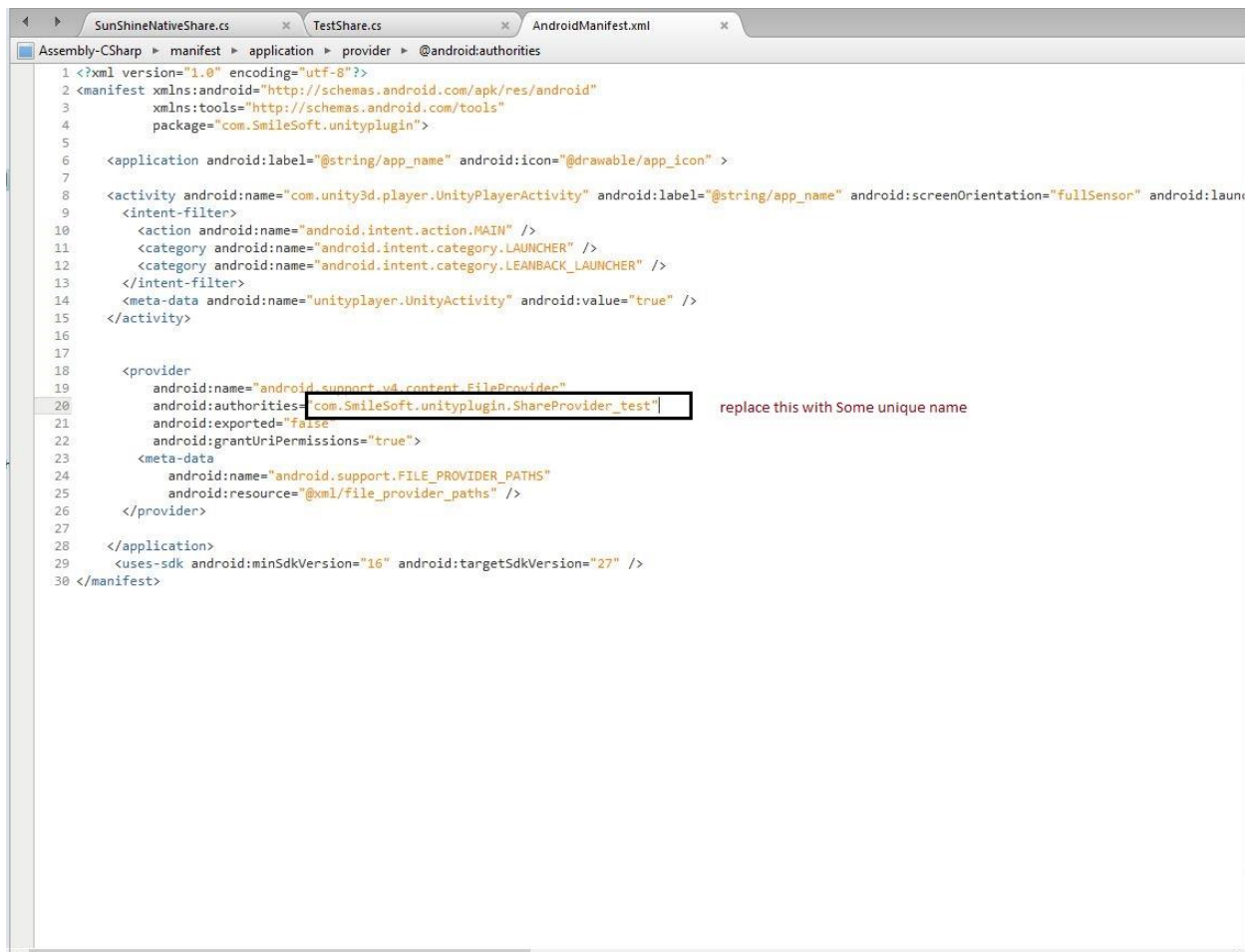
## Sunshine Android Native Share:

Just import the package in your unity project and you are ready to use the plugin. You don't need to drag any prefab. The only thing you need to do is to call some static functions.

Video tutorial for setup is here <https://youtu.be/GuCw5plwxtI>

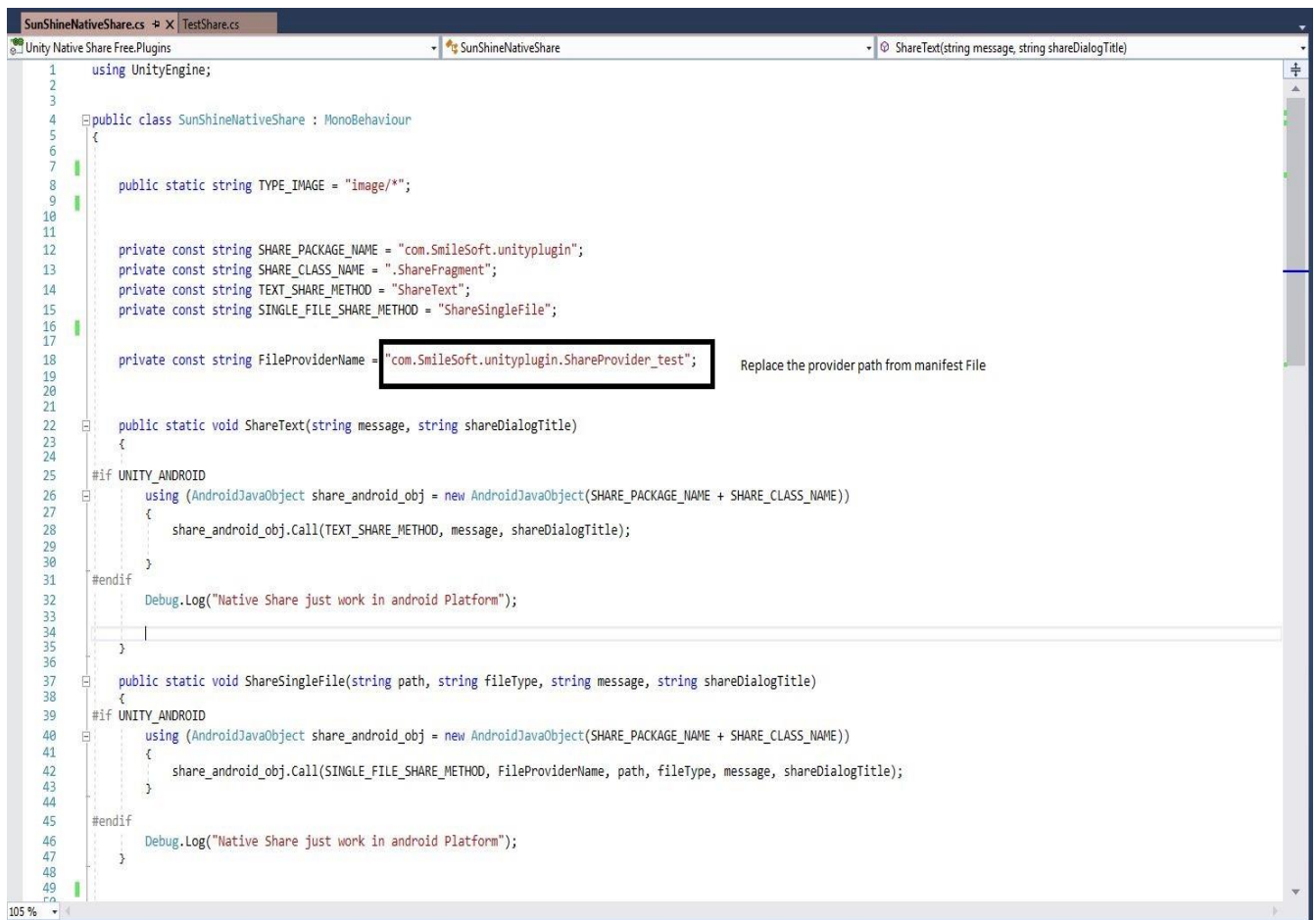
Setup:

Open androidManifest.xml file from “Plugins / Android / androidManifest.xml “. replace the `android:authorities` name from provider block with some unique name. This will be your file provider path.



Then Again open SunShineNativeShare.cs script from “Plugins / SunShine Android Native Share / SunShineNativeShare.cs “. Copy the provider path from androidmanifest file and paste it in “FileProviderName “variable.

**\*\* Strongly recommended that you should use your package name as your provider name \*\***



```
1 using UnityEngine;
2
3
4 public class SunShineNativeShare : MonoBehaviour
5 {
6
7     public static string TYPE_IMAGE = "image/*";
8
9
10
11     private const string SHARE_PACKAGE_NAME = "com.SmileSoft.unityplugin";
12     private const string SHARE_CLASS_NAME = ".ShareFragment";
13     private const string TEXT_SHARE_METHOD = "ShareText";
14     private const string SINGLE_FILE_SHARE_METHOD = "ShareSingleFile";
15
16
17     private const string FileProviderName = "com.SmileSoft.unityplugin.ShareProvider_test"; // Replace the provider path from manifest File
18
19
20
21
22     public static void ShareText(string message, string shareDialogTitle)
23     {
24
25     #if UNITY_ANDROID
26         using (AndroidJavaObject share_android_obj = new AndroidJavaObject(SHARE_PACKAGE_NAME + SHARE_CLASS_NAME))
27         {
28             share_android_obj.Call(TEXT_SHARE_METHOD, message, shareDialogTitle);
29         }
30     #endif
31     Debug.Log("Native Share just work in android Platform");
32
33
34     }
35
36
37     public static void ShareSingleFile(string path, string fileType, string message, string shareDialogTitle)
38     {
39     #if UNITY_ANDROID
40         using (AndroidJavaObject share_android_obj = new AndroidJavaObject(SHARE_PACKAGE_NAME + SHARE_CLASS_NAME))
41         {
42             share_android_obj.Call(SINGLE_FILE_SHARE_METHOD, FileProviderName, path, fileType, message, shareDialogTitle);
43         }
44     #endif
45     Debug.Log("Native Share just work in android Platform");
46
47     }
48
49 }
```

## 1. Share Text :

`SunShineNativeShare.ShareText` (string message, string share\_dialog)

2. **Share Single File:** ( free version only support image type file . Pro version supports all most all type of files )

`SunShineNativeShare.ShareSingleFile`( string path, string fileType, string message, string shareDialogTitle )

Ex: `SunShineNativeShare.ShareSingleFile`( "path" ,  
`SunShineNativeShare.TYPE_IMAGE`,"Share Message","Share By sunshine")

3. **Share Multiple File of Same Type :** ( Pro Version Only)

`SunShineNativeShare.ShareMultipleFileOfSameType` ( string[] path , string fileType, string message, string shareDialogTitle )

4. **Share Multiple File of Multiple Type :** ( Pro version Only )

`SunShineNativeShare.ShareMultipleFileOfMultipleType ( string[] path ,  
string message, string shareDialogTitle )`

**\*\* If you previously import the free version of this plugin then please delete the “SunShine Android Native Share” folder first for unexpected conflict of scripts before using the pro version. \*\***