**Unity3D教程：编辑器编程**

Posted on 2013年07月10日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b)/被围观 9 次

在使用Unity3D来开发大型的项目中，我们通常会自己开发一些适合自己的编辑器来便于一些非程序人员对游戏内容进行编辑。Unity3D给我们开放了这样的一些接口：

一 导航条

加入我们需要在导航条中增加一些我们自己的小工具。我们可以如下使用：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **using** UnityEditor; |
| 02 |  |
| 03 | **static** **public** **class** WayPointTool |
| 04 |  |
| 05 | { |
| 06 |  |
| 07 | **private** **static** **int** index = 0 ; |
| 08 |  |
| 09 | **private** **static** Dictionary<**string**,NvWayPoint> m\_WayPointJasonTable; |
| 10 |  |
| 11 | **private** **static** Dictionary<**int**,WayPoint> m\_WayPointTable; |
| 12 |  |
| 13 | [MenuItem(“Game Tool Chain/WayPoint Tool/Reload WayPoint”)] |
| 14 |  |
| 15 | **static** **void** ReloadWayPoint() |
| 16 |  |
| 17 | { |
| 18 |  |
| 19 | } |
| 20 |  |
| 21 | } |

因为在导航条中的工具类没有被编辑器或者使用者实例化，所以，类和对象以及函数都需要是静态的。

二 编辑器窗口

我们也可以在编辑器中添加一些我们想要的窗口，所创建的窗口会像3.5中的Navigation窗口一样。具体窗口中的内容需要定义的话，要使用Unity3D提供的类：EditorWindow。此类窗口的特点是没有对象的概念，是全局使用的窗口类。

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **using** UnityEngine; |
| 02 |  |
| 03 | **using** UnityEditor; |
| 04 |  |
| 05 | **public** **class** MyWindow : EditorWindow |
| 06 |  |
| 07 | { |
| 08 |  |
| 09 | *// Add menu named “My Window” to the Window menu* |
| 10 |  |
| 11 | [MenuItem (“Window/My Window”)] |
| 12 |  |
| 13 | **static** **void** Init () { |
| 14 |  |
| 15 | *// Get existing open window or if none, make a new one:* |
| 16 |  |
| 17 | MyWindow window = (MyWindow)EditorWindow.GetWindow (typeof (MyWindow)); |
| 18 |  |
| 19 | **void** OnGUI () |
| 20 |  |
| 21 | { |
| 22 |  |
| 23 | GUILayout.Label (“**Base** Settings”, EditorStyles.boldLabel);myString |
| 24 |  |
| 25 | EditorGUILayout.TextField (“Text Field”, myString);groupEnabled |
| 26 |  |
| 27 | EditorGUILayout.BeginToggleGroup (“Optional Settings”, groupEnabled); |
| 28 |  |
| 29 | myBool = EditorGUILayout.Toggle (“Toggle”, myBool); |
| 30 |  |
| 31 | myFloat = EditorGUILayout.Slider (“Slider”, myFloat, -3, 3); |
| 32 |  |
| 33 | EditorGUILayout.EndToggleGroup (); |
| 34 |  |
| 35 | EditorGUIUtility.LookLikeInspector (); |
| 36 |  |
| 37 | EditorGUILayout.TextField (“Text Field:”, “Hello There”); |
| 38 |  |
| 39 | EditorGUILayout.IntField(“**Int** Field:”, integer1); |
| 40 |  |
| 41 | EditorGUILayout.FloatField(“**Float** Field:”, float1); |
| 42 |  |
| 43 | EditorGUILayout.Space(); |
| 44 |  |
| 45 | EditorGUIUtility.LookLikeControls(); |
| 46 |  |
| 47 | EditorGUILayout.TextField (“Text Field”, “Hello There”); |
| 48 |  |
| 49 | EditorGUILayout.IntField(“**Int** Field:”, integer1); |
| 50 |  |
| 51 | EditorGUILayout.FloatField(“**Float** Field:”, float1); |
| 52 |  |
| 53 | } |
| 54 |  |
| 55 | } |

三 自定义对象的编辑窗口

你可能需要对自己定义的类对象暴露一些接口给使用编辑器的开发人员。比如你想让他编辑一个UI控件之类的对象，你可以使用下面的方法来实现：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **using** UnityEngine; |
| 02 |  |
| 03 | **using** UnityEditor; |
| 04 |  |
| 05 | [CustomEditor(typeof(UIWidget))] *// 对UIWidget 对象编辑窗进行的重构* |
| 06 |  |
| 07 | **public** **class** UIWidgetInspector : Editor |
| 08 |  |
| 09 | { |
| 10 |  |
| 11 | **protected** UIWidget mWidget; |
| 12 |  |
| 13 | **static** **protected** **bool** mUseShader = **false**; |
| 14 |  |
| 15 | **bool** mInitialized = **false**; |
| 16 |  |
| 17 | **protected** **bool** mAllowPreview = **true**; |
| 18 |  |
| 19 | **public** **override** **void** OnInspectorGUI () |
| 20 |  |
| 21 | { |
| 22 |  |
| 23 | } |
| 24 |  |
| 25 | } |