**编辑器扩展**

**1，创建自定义编辑器窗口：**

1，创建一个派生自 EditorWindow 的脚本。

2，使用代码触发窗口自行显示。

[MenuItem("Window/My Window")]

public static void ShowWindow(){

EditorWindow.GetWindow(typeof(MyWindow));

}

3，实现工具的OnGUI代码。

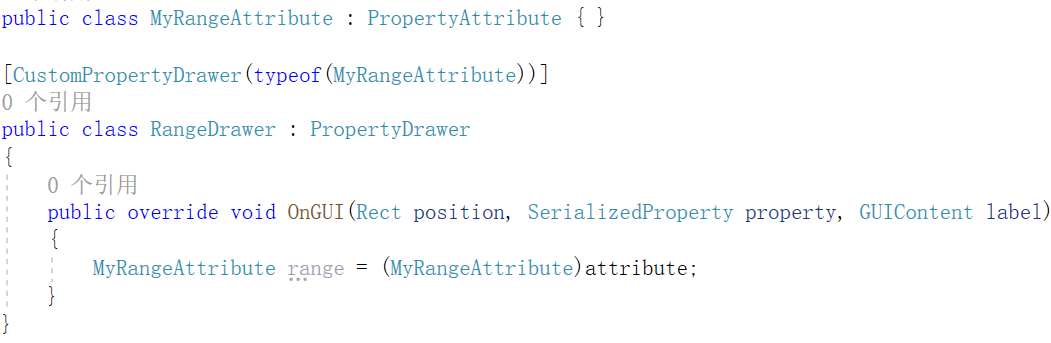
**2，[Serializable]的作用**

在unity里，自定义数据类型无法显示在inspector面板里，需要对定义数据类型的类或者结构体使用[System.Serializable]。

**3，自定义Inspector的三种方法**

1. **自定义 Serializable 类的每个实例的GUI。**

**2，通过自定义的属性特性来自定义脚本成员的GUI。**



**3，自定义属性器**



**4，TreeView重要的类和方法**

**TreeView**

TreeView 是一个IMGUI控件，用于显示可以展开和折叠的层级数据。

**TreeViewState**

TreeViewState 包含与 Editor 中的TreeView 字段交互时更改的状态信息，例如选择状态、展开状态、导航状态和滚动状态。TreeViewState 是唯一[可序列化](file:///D:\BaiduNetdiskDownload\Unity2019.1%E4%B8%AD%E6%96%87%E6%8A%80%E6%9C%AF%E6%89%8B%E5%86%8C%E7%A6%BB%E7%BA%BF%E7%89%88\UnityDocumentation_2019.1\Manual\script-Serialization.html)的状态。

**TreeViewItem**

每一行代表一个TreeViewItem。

TreeView 有一个单独的根 TreeViewItem，此项是隐藏的。

每个 TreeViewItem 必须以唯一的整数ID进行构造。如果树表示 Unity 对象，应为每个对象使用 [GetInstanceID](file:///D:\BaiduNetdiskDownload\Unity2019.1%E4%B8%AD%E6%96%87%E6%8A%80%E6%9C%AF%E6%89%8B%E5%86%8C%E7%A6%BB%E7%BA%BF%E7%89%88\UnityDocumentation_2019.1\ScriptReference\Object.GetInstanceID.html) 作为 TreeViewItem 的 ID。

所有 TreeViewItems 都有一个 depth 属性，表示视觉缩进。

**BuildRoot**

使用此方法可以创建树的根项。每次对树调用 [Reload](file:///D:\BaiduNetdiskDownload\Unity2019.1%E4%B8%AD%E6%96%87%E6%8A%80%E6%9C%AF%E6%89%8B%E5%86%8C%E7%A6%BB%E7%BA%BF%E7%89%88\UnityDocumentation_2019.1\ScriptReference\IMGUI.Controls.TreeView.Reload.html) 时都会调用此方法。对于使用小数据集的简单树，应在 BuildRoot 中的根项下创建 TreeViewItems 的整个树。对于非常大的树，应先创建根，然后重写 BuildRows 方法，仅创建当前行的项。

**BuildRows**

如果只在 BuildRoot 中创建了根，则应重写此方法来处理展开的行。每次展开或折叠 TreeViewItem 时也会调用。

**5，阅读TreeViewExamples源码**

ScriptableObject