



# 孤独の巡礼

博客园 首页

新随笔 联系 订阅 管理

随笔 - 191 文章 - 2 评论 - 28 阅读 - 327

昵称：太乙\_真人

园龄：12年2个月

粉丝：28

关注：0

+加关注

2025年6月								
<	日	一	二	三	四	五	六	>
	1	2	3	4	5	6	7	
	8	9	10	11	12	13	14	
	15	16	17	18	19	20	21	
	22	23	24	25	26	27	28	
	29	30	1	2	3	4	5	
	6	7	8	9	10	11	12	

## 搜索

找找看

## 常用链接

我的随笔  
我的评论  
我的参与  
最新评论  
我的标签

## 随笔档案

2025年3月(1)  
2024年6月(1)  
2024年4月(1)  
2023年11月(1)  
2023年10月(1)  
2023年8月(3)  
2023年7月(2)  
2023年5月(1)  
2023年4月(3)  
2023年3月(5)  
2023年2月(4)  
2023年1月(21)  
2022年11月(1)  
2022年9月(1)  
2022年6月(1)  
更多

## 阅读排行榜

1. Unity3d之将terrain转化成mesh(154)
- 16)
2. c#之如何转换文本文件编码格式为utf-8(12123)
3. Unity如何退出游戏(11153)
4. 四元数小总结(9945)
5. Unity shader之ColorMask(8331)

## 评论排行榜

1. Unity 之事件系统(6)
2. Unity3d之将terrain转化成mesh(4)
3. Unity性能优化之特效合并(3)
4. Unity之如何从fbx提取Animation clip文件(3)
5. unity3d 多人寻路堵塞堆叠问题(3)

## 推荐排行榜

## Unity Shader之自定义GUI

shader可以自己写一个shader gui, 即写一个类继承自**ShaderGUI**, 参考文档:

<https://docs.unity3d.com/Manual/SL-CustomEditor.html>

这是完全没有问题的, 但有时想简单点, 不想这么麻烦。

那么可以使用unity给我们提供好的一些GUI, 即**MaterialPropertyDrawer**,

参考文档: <https://docs.unity3d.com/ScriptReference/MaterialPropertyDrawer.html>

总体来说, MaterialPropertyDrawers内置了4种最常见的面板: ToggleDrawer, EnumDrawer, KeywordEnumDrawer, PowerSliderDrawer。

类别	描述
ToggleDrawer	把一个类型为float的属性显示为一个开关, 它的值要么是0要么是1。当选中它时, Unity还会设置一个名为大写属性名_ON (可以自定义名字) 的shader feature (需要在shader里使用"#pragma shader_feature"或者"#pragma multi_compile_"来定义它), 我们可以在shader里用过#if、#ifdef或者#ifndef关键词来判断它当前是否被开启。
EnumDrawer	把一个类型为float的属性显示为一个下拉列表。可以使用UnityEngine.Rendering命名空间下的各种状态来设置对应的渲染状态, 例如ZWrite、ZTest、Blend等。据实验推测, 这个值不可以随便定义, 老老实实用UnityEngine.Rendering。
KeywordEnum	和EnumDrawer类似, 也会把一个类型为float的属性显示为一个下拉列表, 但不同的是它会定义一组shader keyword (需要在shader里使用"#pragma multi_compile"来定义它们)。这些keyword的名字是大写属性名_枚举名。同样, 我们可以在shader里用过#if、#ifdef或者#ifndef配合#elif来判断当前选择是哪个keyword。最多同时支持9个keywords。
PowerSliderDrawer	显示一个非线性响应的滑动条, 其中PowerSliderDrawer中的参数指定了底数, 然后我们再根据Range()来指定范围。其实简单说来就是滑动条上的数值不再是均匀变化了, 而是xslider进行变化, 但我们在shader里还是可以直接访问到一个float数值。

下面的代码没有任何实际意义, 只是为了演示四种面板:



```

1 Shader "Custom/Material Property Drawer Example"
2 {
3     Properties
4     {
5         // Header creates a header text before the shader property.
6         [Header(Material Property Drawer Example)]
7         // Space creates vertical space before the shader property.
8         [Space]
9
10        _MainTex ("Main Tex", 2D) = "white" {}
11        _SecondTex ("Second Tex", 2D) = "white" {}
12
13        // Large amount of space
14        [Space(50)]
15
16        // Toggle displays a **float** as a toggle.
17        // The property value will be 0 or 1, depending on the toggle state.

```

1. c#如何判断字符串是否含中文(2)
2. Unity3d之将terrain转化成mesh(2)
3. Jenkins+Unity自动化打包之如何跑通第一个shell脚本打印Hello world(1)
4. Unity shader学习之简单的水效果(1)
5. Unity 之事件系统(1)

## 最新评论

1. Re:Unity3d 开发之 ulua 坑的总结  
【Unity3D】Unity3D技术栈  
  
--little\_fat\_sheep
2. Re:Unity3d之将terrain转化成mesh  
@马马呼呼 var t = terrainObj.transform.parent.Find(transName); 可能是Terrain没有parent导致的，给Terrain一个父节点试试...  
  
--孤独の巡礼
3. Re:Unity3d之将terrain转化成mesh  
为啥代码复制进去之后 报错了 堆了个这个 NullReferenceException: Object reference not set to an instance of an object Te...  
  
--马马呼呼
4. Re:Unity之如何从fbx提取Animation clip文件  
大佬， AnimationClip直接在Fbx里压缩和去除Scale不是也可以吗，为啥要把AnimationClip给复制出一份呢?  
  
--Li\_sj
5. Re:Unity 之 Addressable Asset System 之用工具创建group  
你好。你可以给我一个下载这个项目download link 或 GitHub link 吗?  
  
--MikakuTosu

## Unity Shader之自定义GUI - 太乙\_真人 - 博客园

```

18 // When it is on, a shader keyword with the uppercase property name +"_ON" will be
19 // or an explicitly specified shader keyword.
20 [Toggle] _Invert ("Invert color?", Float) = 0
21
22 // Will set "ENABLE_FANCY" shader keyword when set
23 [Toggle(ENABLE_FANCY)] _Fancy ("Fancy?", Float) = 0
24
25 // Enum displays a popup menu for a **float** property.
26 // You can supply either an enum type name
27 // (preferably fully qualified with namespaces, in case there are multiple types)
28 // or explicit name/value pairs to display.
29 // Up to **7** name/value pairs can be specified
30 [Enum(UnityEngine.Rendering.BlendMode)] _SrcBlend ("Src Blend Mode", Float) = 1
31 [Enum(UnityEngine.Rendering.BlendMode)] _DstBlend ("Dst Blend Mode", Float) = 1
32 [Enum(Off, 0, On, 1)] _ZWrite ("ZWrite", Float) = 0
33 [Enum(UnityEngine.Rendering.CompareFunction)] _ZTest ("ZTest", Float) = 0
34 [Enum(UnityEngine.Rendering.CullMode)] _Cull ("Cull Mode", Float) = 1
35
36 // KeywordEnum displays a popup menu for a **float** property, and enables corresponding
37 // This is used with "#pragma multi_compile" in shaders, to enable or disable parts
38 // Each name will enable "property name" + underscore + "enum name", uppercased,
39 // Up to **9** names can be provided.
40 [KeywordEnum(None, Add, Multiply)] _Overlay ("Overlay mode", Float) = 0
41
42 // PowerSlider displays a slider with a non-linear response for a Range shader property
43 // A slider with 3.0 response curve
44 [PowerSlider(3.0)] _Shininess ("Shininess", Range (0.01, 1)) = 0.08
45 }
46 SubShader
47 {
48     Tags { "Queue"="Transparent" "RenderType"="Transparent" }
49     Blend [_SrcBlend] [_DstBlend]
50     ZWrite [_ZWrite]
51     ZTest [_ZTest]
52     Cull [_Cull]
53
54     Pass
55     {
56         CGPROGRAM
57         // Need to define _INVERT_ON shader keyword
58         #pragma shader_feature _INVERT_ON
59         // Need to define _INVERT_ON shader keyword
60         #pragma shader_feature ENABLE_FANCY
61         // No comma between features
62         #pragma multi_compile _OVERLAY_NONE _OVERLAY_ADD _OVERLAY_MULTIPLY
63
64         #pragma vertex vert
65         #pragma fragment frag
66
67         #include "UnityCG.cginc"
68
69         sampler2D _MainTex;
70         float4 _MainTex_ST;
71         sampler2D _SecondTex;
72         float4 _SecondTex_ST;
73         float _Shininess;
74
75         struct appdata
76         {
77             float4 vertex : POSITION;
78             float2 uv : TEXCOORD0;
79         };
80
81         struct v2f

```

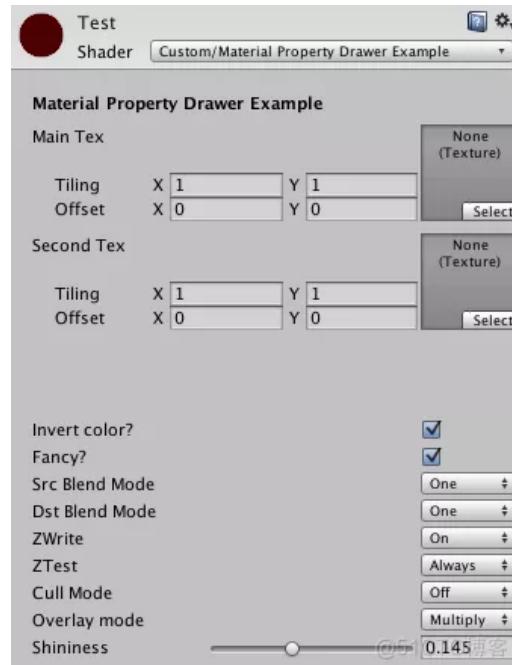
```

82
83         float4 uv : TEXCOORD0;
84         float4 vertex : SV_POSITION;
85     };
86
87     v2f vert (appdata v)
88     {
89         v2f o;
90         o.vertex = mul(UNITY_MATRIX_MVP, v.vertex);
91         o.uv.xy = TRANSFORM_TEX(v.uv, _MainTex);
92         o.uv.zw = TRANSFORM_TEX(v.uv, _SecondTex);
93         return o;
94     }
95
96     fixed4 frag (v2f i) : SV_Target
97     {
98         // sample the texture
99         fixed4 col = tex2D(_MainTex, i.uv.xy);
100
101        // Use #if, #ifdef or #if defined
102        #if _INVERT_ON
103        col = 1 - col;
104        #endif
105
106        // Use #if, #ifdef or #if defined
107        #if ENABLE_FANCY
108        col.r = 0.5;
109        #endif
110
111        fixed4 secCol = tex2D(_SecondTex, i.uv.zw);
112
113        #if _OVERLAY_ADD
114        col += secCol;
115        #elif _OVERLAY_MULTIPLY
116        col *= secCol;
117        #endif
118
119        col *= _Shininess;
120
121        return col;
122    }
123    ENDCG
124}
125}
126

```



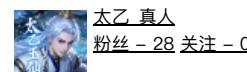
效果如下：



可参考：

[Unity 自定义Shader GUI](#)

转载请注明出处：<https://www.cnblogs.com/jietian331/p/17296261.html>



0 0

[+加关注](#)

[升级成为会员](#)

« 上一篇： [Unity Shader之如何实现Low版钻石效果](#)

» 下一篇： [Unity URP之通过Feature实现毛皮效果](#)

posted @ 2023-04-07 15:10 太乙\_真人 阅读(399) 评论(0) 收藏 举报

[刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论，立即 [登录](#) 或者 [逛逛](#) 博客园首页

【推荐】100%开源！大型工业跨平台软件C++源码提供，建模，组态！

【推荐】国内首个AI IDE，深度理解中文开发场景，立即下载体验Trae

【推荐】Flutter适配HarmonyOS 5知识地图，实战解析+高频避坑指南

【推荐】凌霞软件回馈社区，携手博客园推出1Panel与Halo联合终身会员

【推荐】轻量又高性能的SSH工具IShell：AI加持，快人一步



相关博文：

- [Unity UGUI之UI圆角shader](#)

- Unity Shader之皮肤渲染之用颜色实现sss效果
- Shader内容释义
- Unity Shader从入门到\*\*
- Uniyt Shader基础

**阅读排行:**

- 我的开源十年
- .net入行第4年
- C#/.NET/.NET Core优秀项目和框架2025年5月简报
- 杂七杂八系列----C#代码如何影响CPU缓存速度?
- (包含5w字提示词开源) 手把手教你搭建开源提示词优化平台

**历史上的今天:**

- 2013-04-07 CSS选择器总结
- 2013-04-07 HTML总结
- 2013-04-07 C#网络编程socket使用总结
- 2013-04-07 多线程总结
- 2013-04-07 ADO.NET技术总结
- 2013-04-07 C#面向对象三大特性总结

博客园 © 2004-2025



浙公网安备 33010602011771号 浙ICP备2021040463号-3