



唐老狮系列教程

设置深度写入和渲染队列



唐老狮系列教程-设置深度写入和渲染队列

| 主要讲解内容



唐老狮系列教程-设置深度写入和渲染队列

主要讲解内容

1. Unity Shader中设置深度写入
2. Unity Shader中设置渲染队列



唐老狮系列教程-设置深度写入和渲染队列

| Unity Shader中设置深度写入



唐老狮系列教程-设置深度写入和渲染队列

Unity Shader中设置深度写入

```
Pass
{
    Tags { "LightMode"="ForwardBase" }
    //关闭深度写入我们一般放在Pass渲染通道中
    //如果存在多个Pass渲染通道，它仅仅会影响该Pass渲染通道
    ZWrite Off
```

```
SubShader
{
    //若写在SubShader语句块中
    //其中的所有Pass渲染通道都将关闭深度写入
    ZWrite Off
```

深度写入默认是开启的

我们需要通过 渲染状态中的 **ZWrite off** 指令主动关闭深度写入

当我们把它写在Pass渲染通道中时，它只会影响该Pass

若我们把它写在SubShader语句块中，它将影响其中的所有Pass



唐老狮系列教程-设置深度写入和渲染队列

| Unity Shader中设置渲染队列



唐老狮系列教程-设置深度写入和渲染队列

Unity Shader中设置渲染队列

通过上节课的学习我们知道，渲染队列非常的重要，它会影响最终的渲染效果。

在Unity Shader中我们可以通过渲染标签主动的设置物体的渲染顺序

Tags{ "Queue" = "标签值" }

- | | |
|------------------------------|--|
| 1.Background(背景)(队列号:1000) | 最早被渲染的物体的队列，一般用来渲染天空盒或者背景 |
| 2.Geometry(几何)(队列号:2000) | 不透明的几何体通常使用该队列，不设置时的默认队列 |
| 3.AlphaTest(透明测试)(队列号:2450) | 需要透明度测试的物体使用的队列 |
| 4.Transparent(透明的)(队列号:3000) | 半透明物体的渲染队列，该队列中几何体按照由远到近的顺序进行绘制，所有进行透明混合的几何体都应该使用该队列 |
| 5.Overlay(覆盖)(队列号:4000) | 用是放在最后渲染的队列，于叠加渲染的效果 |
| 6.自定义队列 | 基于Unity预先定义好的这些渲染队列标签来进行加减运算来定义自己的渲染队列 如: Tags{ "Queue" = "Geometry+1" } 代表的队列号就是 2001 |



唐老狮系列教程-设置深度写入和渲染队列

Unity Shader中设置渲染队列

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "AlphaTest" "IgnoreProjector" = "True" "RenderType" = "TransparentCutout" }
```

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "Transparent" "IgnoreProjector" = "True" "RenderType" = "Transparent" }
```

渲染队列一般都定义在SubShader语句块中，影响之后的所有Pass渲染通道

使用该Shader(着色器)的物体，就会根据你设置的渲染队列在特定的时间进行渲染

在使用 渲染队列Queue 时，一般会搭配 忽视投影器IgnoreProjector 和 渲染类型RenderType 一起使用



唐老狮系列教程-设置深度写入和渲染队列

Unity Shader中设置渲染队列

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "AlphaTest" "IgnoreProjector" = "True" "RenderType" = "TransparentCutout" }
```

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "Transparent" "IgnoreProjector" = "True" "RenderType" = "Transparent" }
```

忽视投影器 IgnoreProjector 标签:

投影器 (Projector) 是Unity中的一种特殊的光源, 它用于在场景中投射纹理 (Texture) 或简单的几何形状 (如圆形或方形) 来模拟光照、阴影或其他视觉效果。

渲染类型 RenderType 标签:

对着色器进行分类, 用于着色器替换功能, 摄像机上有对应API, 可以指定渲染类型替换成别的着色器

Opaque(不透明的)、 Transparent(透明的)、 TransparentCutout(透明切割)、 Background(背景)、 Overlay(覆盖)

```
//主要作用:
//物体是否受到Projector (投影机) 的投射
//Projector是Unity中的一个功能 (以后讲解)

//忽略Projector (一般半透明Shader需要开启该标签)
//Tags{ "IgnoreProjector" = "True" }

//不忽略Projector (默认值)
//Tags{ "IgnoreProjector" = "False" }
```

```
//1.Opaque(不透明的)
// 用于普通Shader, 比如: 不透明、自发光、反射等
//2.Transparent(透明的)
// 用于半透明Shader, 比如: 透明、粒子
//3.TransparentCutout(透明切割)
// 用于透明测试Shader, 比如: 植物叶子
//4.Background(背景)
// 用于天空盒Shader
//5.Overlay(覆盖)
// 用于GUI纹理、Halo (光环)、Flare (光晕)
```



唐老狮系列教程-设置深度写入和渲染队列

| 总结



唐老狮系列教程-设置深度写入和渲染队列

主要讲解内容

1. Unity Shader中设置深度写入

```
Pass
{
    Tags { "LightMode"="ForwardBase" }
    //关闭深度写入我们一般放在Pass渲染通道中
    //如果存在多个Pass渲染通道，它仅仅会影响该Pass渲染通道
    ZWrite Off
```

2. Unity Shader中设置渲染队列

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "AlphaTest" "IgnoreProjector" = "True" "RenderType" = "TransparentCutout"}
```

```
SubShader
{
    //渲染队列的标签一般定义在SubShader语句块中
    //影响下面的所有Pass渲染通道
    Tags{ "Queue" = "Transparent" "IgnoreProjector" = "True" "RenderType" = "Transparent"}
```




唐老狮系列教程

Thank
感谢您的聆听