通过Assets / Create / Shader / Standard Surface Shader创建一个表面着色器。

着色器文件包含了定义表面着色器的代码。

着色器文件语法:

以Shader关键字开始一个代码块代表一个着色器，后面跟着在Shader菜单中的名字

例如Shader "Graph/Point Surface" {}

一个着色器中可以有多个子着色器，必须要有加上“回退到标准的漫反射器” （如果所有子着色器都无法在该硬件上运行，则使用标准的漫反射器）

Shader "Graph/Point Surface" {

SubShader {}

FallBack "Diffuse"

}

表面子着色器需要混合CG和HLSL两种着色器语言编写的代码部分。因此子着色器中需要包括CGPROGRAM和ENDCG关键字，在两个关键字间编写代码。在编写代码时，首先需要加入编译器指令，写作#pragma，在该关键字后需要跟一个指令

CGPROGRAM

#pragma surface ConfigureSurface Standard fullforwardshadows//生成一个标准照明和完全支持阴影的表面着色器

#pragma target 3.0//设置着色器和材质最低版本

struct Input {

float3 worldPos;

};

ENDCG

使用渲染管线之前要为他创建一个Asse，点击 Assets / Create / Rendering / Universal Render Pipeline / Pipeline Asset (Forward Renderer) ，在Edit/Project Setting中选择Graphics，给Scriptable Renderer Pipeline settings字段赋值

需要为这个渲染管线从创建一个着色器图资，Assets / Create / Shader / Universal Render Pipeline / Lit Shader Graph 创建一个新的着色器图，设置它的面板。

然后创建一个Material使用该Shader即可以正常显示材质。