# 前置知识

## 1 Camera.TargetTexture

相机上可以挂载一张RenderTexture，如果挂载了，那么就不再向屏幕渲染

而是渲染到这张RenderTexture上

## 2 Camera.RenderWithShader

Camera.RenderWithShader(depShader,"RenderType");

此方法的作用是可以使用指定的shader来代替当前物体的shader渲染一帧。当replacementTag为空时会替换视口中所有物体的shader。

SetReplacementShader方法与此方法功能相近，不同之处是，SetReplacementShader方法使用指定的shader来替换物体当前的shader，被替换后每一帧都会用替换的shader来渲染物体，而不是只渲染一帧

重置shader即调用方法ResetReplacementShader()

# 阴影映射原理

1 在灯光投影空间，渲染深度贴图

2 在地表空间，把地表上的点转换到光源投影空间，比较当前深度和深度贴图中深度，渲染影子

具体方法

ShadowMapping.cs脚本挂接在MainCamera上，其上有如下引用

RenderTexture类型的\_LightVewZDepth用于存储深度贴图

DepShader.shader 是用来渲染深度贴图的shader ，即使ShadowMapping.shader

ShadowLight 就是场景内生成影子的灯光

ShadowMapping启动后，会在灯光上绑定一个新的相机，并渲染深度贴图

ShadowMappingReceiver.shader挂接在地表，根据深度贴图来渲染影子

# 阴影映射遗留的问题

生成深度贴图时在ShadowMapping.shader中

float d=i.depth.x/i.depth.y-4/i.depth.y;用这句来算深度，后面的-4/i.depth.y是深度偏移，这个看不懂，后面找baidu google求证正解