**阴影**

**What:**

**Why:**

**How:**

**为什么会产生阴影？**

阴影区域的产生是因为光线无法到达这些区域。

**Unity使用的阴影技术及其原理**

Shadow Map，首先把摄像机的位置放在与光源重合的位置上，那么场景中的阴影区域就是那些摄像机看不到的地方。

Shadow Map阴影映射纹理本质上也是一张深度图，它记录了从该光源的位置出发，能看到的场景中距离它最近的表面位置。

**Unity中物体如何向其他物体投射阴影？**

Unity使用一个额外的Pass来专门更新光源的阴影映射纹理，这个Pass就是LightMode标签被设置为ShadowCaster的Pass。

因此，当开启了光源的阴影效果后，底层渲染引擎首先会在当前渲染物体的Unity Shader中找到LightMode为ShadowCaster的Pass，如果没有，它就会在Fallback指定的Unity Shader中继续寻找，如果仍然没有找到，该物体就无法向其他物体投射阴影(但它仍然可以接收来自其他物体的阴影)。当找到一个LightMode为ShadowCaster的Pass后，Unity会使用该Pass来更新光源的阴影映射纹理。

**屏幕空间的阴影映射技术**

在传统的阴影映射纹理的实现中，我们会在正常渲染的Pass中把顶点位置变换到光源空间下，以得到它在光源空间中的三维位置信息。然后，我们使用xy分量对阴影映射纹理进行采样，得到阴影映射纹理中该位置的深度信息。如果该深度值小于该顶点的深度值，那么说明该点位于阴影中。

但在Unity中，使用了屏幕空间的阴影映射技术。Unity首先会通过调用LightMode为ShadowCaster的Pass来得到可投射阴影的光源的阴影映射纹理以及摄像机的深度纹理。然后，根据光源的阴影映射纹理和摄像机的深度纹理来得到屏幕空间的阴影图。如果摄像机的深度图中记录的表面深度大于转换到阴影映射纹理中的深度值，就说明该表面虽然是可见的，但却处于该光源的阴影中。通过这样的方式，阴影图就包含了屏幕空间中所有有阴影的区域。如果我们想要一个物体接收来自其他物体的阴影，只需要在Shader中对阴影图进行采样。由于阴影图是屏幕空间下的，因此，我们首先需要把表面坐标从模型空间变换到屏幕空间中，然后使用这个坐标对阴影图进行采样即可。

**一个物体接收来自其他物体的阴影，以及它向其他物体投射阴影过程的总结。**

1. 如果我们想要一个物体接收来自其他物体的阴影，就必须在Shader中对阴影映射纹理进行采样，把采样结果和最后的光照结果相乘来产生阴影效果。
2. 如果我们想要一个物体向其他物体投射阴影，就必须把该物体加入到光源的阴影映射纹理的计算中，从而让其他物体在对阴影映射纹理采样时可以得到该物体的相关信息。在Unity中，这个过程是通过为该物体执行LightMode为ShadowCaster的Pass来实现的。如果使用了屏幕空间的投影映射技术，Unity还会使用这个Pass产生一张摄像机的深度纹理。

**透明物体的阴影**

透明度测试的阴影：把Fallback设置为Transparent/Cutout/VertexLit，把Cast Shadows属性设置为Two Sided。

透明度混合的阴影：Unity默认是不处理透明度混合的阴影，或者当成不透明物体处理。