**光照探针**

**What:**

**Why:**

**How:**

**光照探针的原理是什么？**

在光照探针的所在位置点上对光照信息进行采样，然后从该光照探针相邻的其它光照探针的位置上对光照信息进行采样，把这些光照信息进行插值运算。

**什么是光照探针组？**

光照探针组件不能直接挂到游戏对象上面，通常需要依赖光照探针组（Light Probe Group）组件挂接。光照探针组默认情况下在一个立方体空间中包含六个探针，场景中用黄色小球表示光照探针，它可以通过组件上的Edit Light Probes来编辑探针，可以移动、复制或删除单个探针，场景中光照探针要达到一定数量才能被正确烘焙。

**光照探针的布局**

最简单的布局方式是将光照探针排列成一个规则的3D网格样式，这样的设置方式简单高效，但会消耗大量内存，因为每一个光照探针本质上是一个球形的且记录了当前采样点周围环境的纹理图像。如果一片区域的照明信息都差不多，那么就没必要使用大量光照探针了。光照探针通常用于照明效果突然改变的场合。

**采样光照探针**

1. 光照探针的插值数据需要逐对象的传递给GPU。

perObjectData = PerObjectData.Lightmaps | PerObjectData.LightProbe;

2. 在UnityInput.hlsl文件的UnityPerDraw缓冲区中定义7个float4类型变量来接收CPU传递来的光探针数据，它们是代表红色、绿色和蓝光的多项式组件。

3. 在GI.hlsl文件中创建SampleLightProbe方法对光照探针进行采样。首先判断，若该对象正在使用光照贴图就直接返回0，否则返回0和使用SampleSH9()方法得到的光照数据之间最大值。

4. 在GetGI中将光照探针的采样结果和光照贴图的采样结果相加得到漫反射照明。