SubShader的标签主要包括：Queue、RenderType、ForceNoShadowCasting、IgnoreProjector

1）Queue

表示该shader渲染的物体是属于哪一个队列，unity默认提供了五种队列（队列值越小越先渲染）：



我们也可以自定义队列

Tags { "Queue" = "Geometry+20" } **注意：这里的Geometry+20中间不能有空格**

2）RenderType

在Shader的Tags中经常可以看到RenderType这个参数，通常可以使用的值如下：

Transparent: most semitransparent shaders (Transparent, Particle, Font, terrain additive pass shaders).  
TransparentCutout: masked transparency shaders (Transparent Cutout, two pass vegetation shaders).  
Background: Skybox shaders.  
Overlay: GUITexture, Halo, Flare shaders.  
TreeOpaque: terrain engine tree bark.  
TreeTransparentCutout: terrain engine tree leaves.  
TreeBillboard: terrain engine billboarded trees.  
Grass: terrain engine grass.  
GrassBillboard: terrain engine billboarded grass.

但是这些值并无真实意义，只是约定的一种参数，当然你也可以自己定义一组参数，比如：

Parameter1

Parameter2

…

怎么使用RenderType呢？那就是替代渲染SetReplacementShader方法（Camera的方法），即指定场景中的全部shader替换成某个shader来渲染。

1. 使用SetReplacementShader（“ShaderA”，“”）表示全部shader替换成ShaderA进行渲染
2. SetReplacementShader（“shaderA”，“RenderType”），先看场景中的shader中是否有RenderType参数，如果有在看RenderType中的数值是否与shaderA中的RenderType值相等，相等则使用shaderA渲染，否则就不渲染

这个方法当然也可以这样使用SetReplacementShader（“ShaderA”，“IgnoreProjector”）或者是其他任何参数：SetReplacementShader（“ShaderA”，“AAA”）。之后去匹配场景中的shader是否有IgnoreProjector参数，有则再匹配它的数值是否与ShaderA相同，相同则进行渲染，多有RenderType以及它的数值就是一种约定，因为它自带的shader全部使用这些数值，所有我们依照它的习惯这样使用会方便很多

3）ForceNoShadowCasting

为true表示不接受阴影

4）IgnoreProjector

为true表示不接受Projector组件的投影