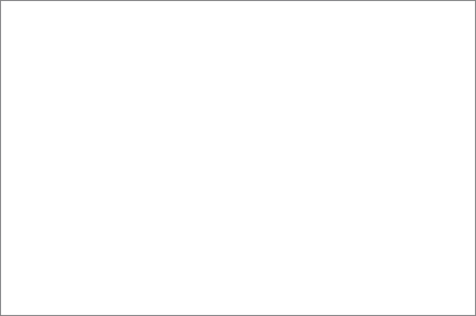
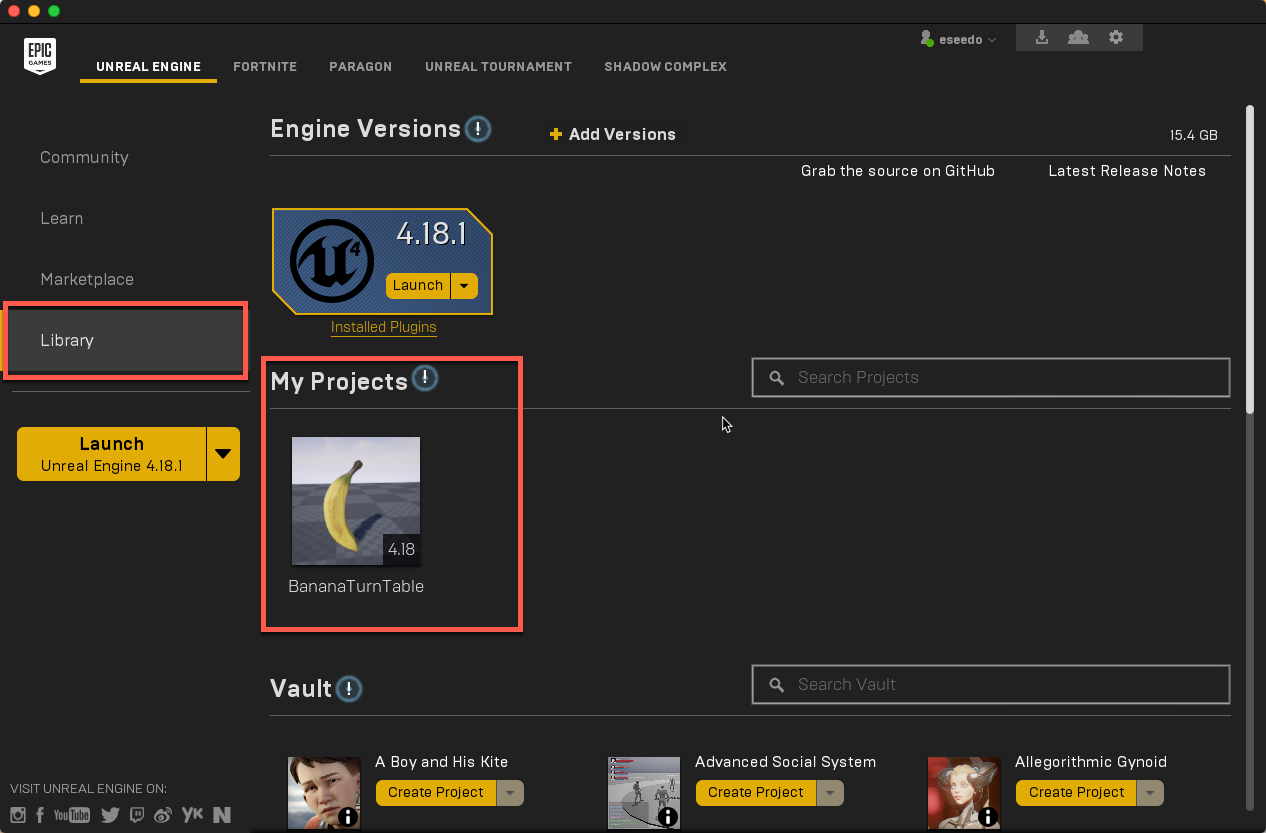
欢迎回到我们的课程，在这一课的内容中，我们将学习如何让turntable旋转。

为了实现这一点，我们需要了解蓝图节点的作用。

首先打开Epic Games Launcher，在之前我们是先打开引擎，然后再打开项目的。其实也可以直接在Epic Games Launcher中打开项目。

在Launcher左侧的选项卡中切换到Library，然后在My Projects中可以看到BananaTurnTable项目，双击将其打开即可。

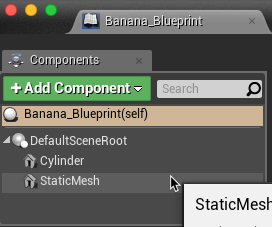


和之前学过的材质编辑器中的节点类似，蓝图节点也有特殊的接口，名为Execution pins，或者说执行接口。

和材质节点类似，左侧的接口为输入接口，而右侧接口为输出接口。所有的节点都至少有一个接口。

如果某个节点有一个输入接口，那么在其被执行前必须有一个连接到该节点上。如果某个节点没有被连接，那么所有其后的节点都不会被执行。

下面是一个简单的示例：

在上图中，节点A和B是可以执行的，因为它们的输入接口之间有一个连接。但是节点C和D无法执行，因为节点C有一个输入接口，但是没有连接。



旋转Turntable

在我们开始之前，先打开Banbaba\_Blueprint这个蓝图，然后查看一下Components面板。

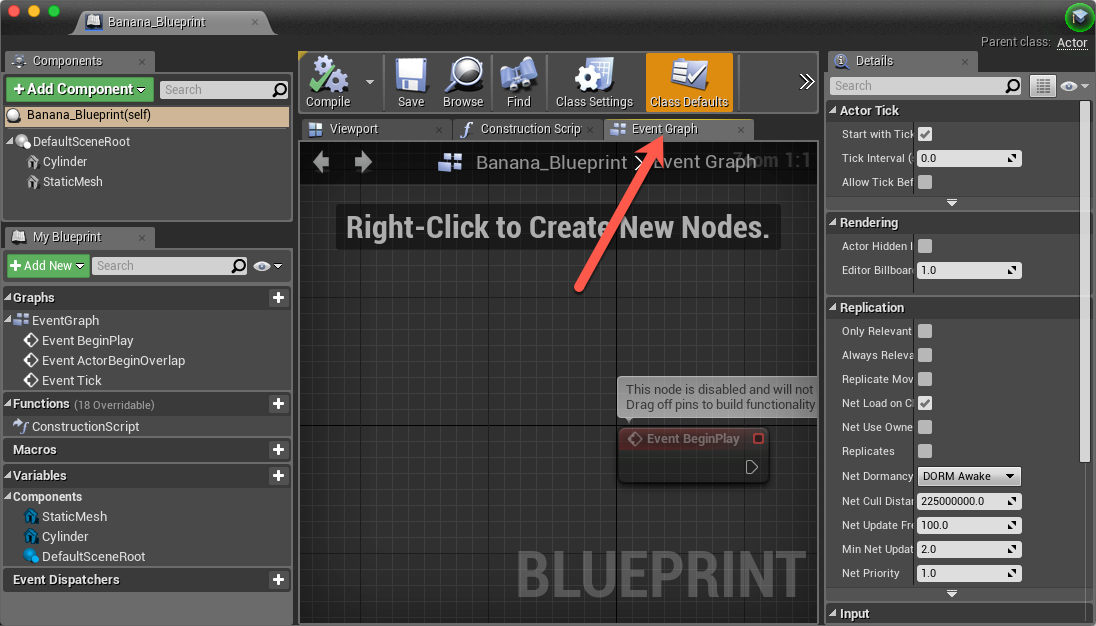
可以看到这里Cylinder和StaticMesh是缩进显示的，但DefaultSceneRoot却没有。这是因为Cylinder和StaticMesh是附属于DefaultSceneRoot上的。

当我们移动、旋转或缩放某个根组件时，附属于它的所有组件也会做同样的操作。通过这样，我们就可以同时旋转Cylinder和Static Mesh两个游戏对象了。

创建节点

为了编写蓝图的逻辑，让我们切换到Event Graph选项。

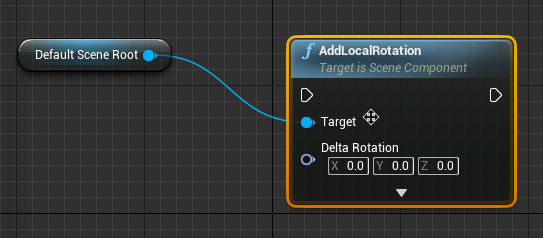
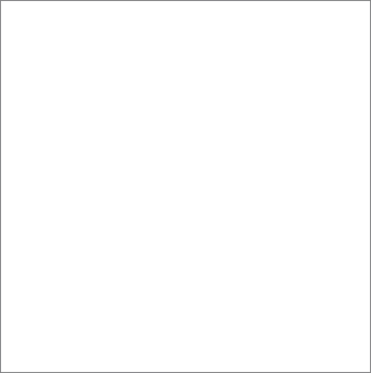
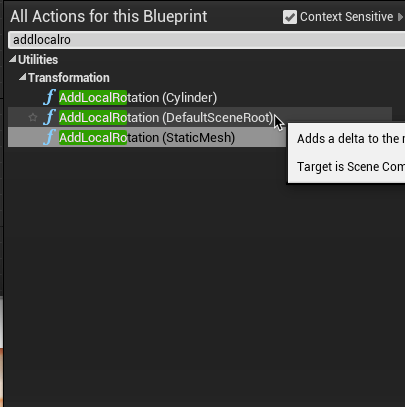
让某个游戏对象非常简单，我们只需创建一个节点就可以了。



在Event Graph的空白区域右键单击，然后可以看到一系列可用的节点列表，在里面搜索AddLocalRotation。因为我们要同时旋转底座和香蕉，所以只需要旋转根组件即可。这里我们选择AddLocalRotation(DefaultSceneRoot）。

注意，如果在搜索的时候没有发现自己需要的节点，那么请取消对Context Sensitive（大小写敏感）的勾选。

此时在节点中可以看到类似下面的画面：



可以看到，在Target输入接口有一个连线，自动连接到所选的组件。

为了设置旋转角度，可以在Delta Rotation中将Z值更改为1.0，这样蓝图就会围绕Z轴旋转。数值越大，turntable旋转的速度也就越快。

为了让turntable持续旋转，我们需要让AddLocalRotation每一帧都执行。为此，需要用到Event Tick节点。

默认情况下在Event Graph中已经存在该节点了，如果没有，我们可以用刚才的方法来添加一个。



从Event Tick节点的输出接口连一条线到AddLocalRotation节点的输入接口，如下图。

需要注意的是，turntable的旋转速度跟帧速有关。换句话说，如果你的电脑配置比较低，运行速度比较慢，那么turntable的旋转速度也会比较慢。对本教程来说，这样就可以了。在后续的教程中，我们将学习如何解决这个小问题。

最后，在工具栏上点击Compile来更新蓝图，然后关闭蓝图编辑器。



在场景中添加蓝图

在向场景中添加蓝图之前，首先返回虚幻4主编辑器，然后从Viewport中删除之前所添加的香蕉模型，

选中该物体（最保险的方法是在World Outliner中选择），然后使用Edit-Delete将其删除，或者直接使用Delete键来删除。

接下来的事情就很简单了，在Content Browser中找到Banana\_Blueprint这个蓝图文件，然后使用鼠标左键选中，把它拖到场景中就可以了。

从编辑器的工具栏中点击Play按钮，就可以预览游戏的效果了~

注意：如果你没有删除之前的🍌模型，那么就会收到一个警告，大概意思是光照需要rebuild。当我们删除此前的🍌模型后，一切就会正常了。

好了，本系列教程的第一个小环节就到此结束了。

在接下来的教程中，我们将进一步了解虚幻4的蓝图系统。

微信公众号：vrlife