Actor组件，对应普通组建，

举个例子，在角色类蓝图的组件中，横线下面的就是Actor组件



Actor组件我们是可以自定义的

创建蓝图类，选择Actor组件

既然是组件，我们可以选择一个比较通用的组件

比如选择健康组件

健康组件自己有一个变量，命名为健康值，float型，可编辑实例，生成时公开。

然后我们要给健康组件定义一个函数，命名为“触发伤害”。函数需要一个形式参数，来传递伤害值



然后我们需要创建一个事件分发器，对事件分发器进行调用。

Actor组件和使用该组件的Actor之间通信的方式：事件分发器

在Actor组件中调用一个事件分发器就是再说：

“嘿，我要让所有对这件事感兴趣的人直到角色正在触发伤害”。但是Actor组件本身并不决定在受到伤害时除了减少生命值之外应该做什么。（组件本身并不关心事件被调用时会发生什么，只是通知使用组件的部分：事件发生了）

然后我们可以再创建一个事件分发器，命名为死亡。用于在角色死亡时发出通知，



待会就会看到如何在玩家角色中配置事件。

然后我们需要在一个蓝图中使用这个组件

创建一个角色蓝图，在组件中添加刚创建的健康组件，然后我们就可以在细节中看到我们刚创建的变量和事件分发器。

进入事件图表，创建该组件的节点，可以在细节的事件中看到我们刚创建的两个事件分发器。同时也可以通过组件节点引出得到健康值



同样的也可以通过组件节点使用“触发伤害”函数



现在想在玩家角色的头顶上添加一个血条Widget，我们可以在Actor组件内部来完成这项工作。

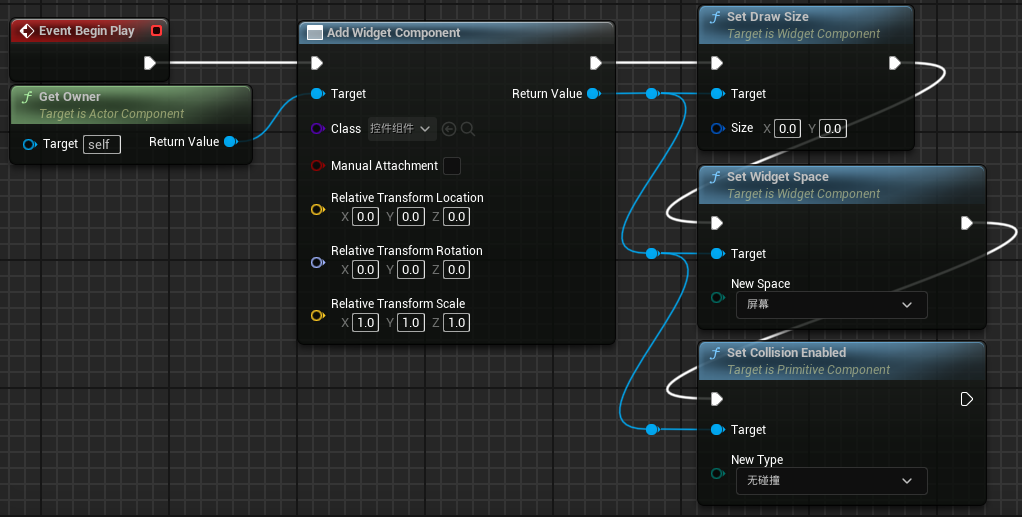
你可以使用一个Actor组件向Actor添加这个组建所依赖的其他组件。

如果你想在角色的头上添加一个血条，通常的做法是给Mesh组件添加一个Widget组件，然后把widget放到头上

现在我们用Health组件来完成这一要求。

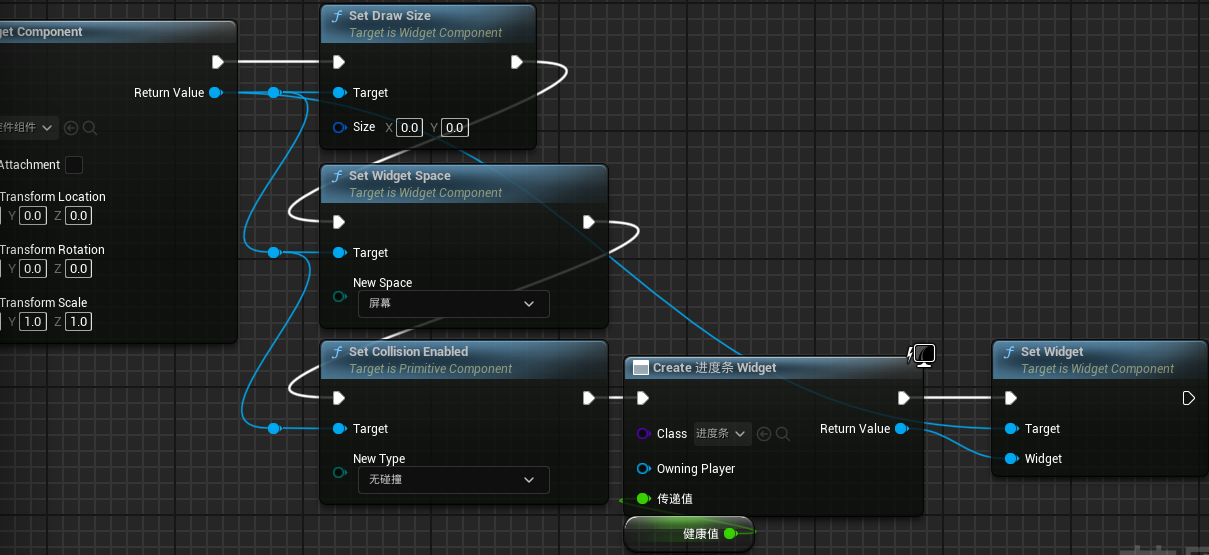
在健康组件中GetOwner，然后AddComponentByClass，选择UE自带的WidgetComponent。

然后对WidgetComponent进行SetDrawSize、SetWidgetSpace、SetCollisionEnabled



现在我们创建一个控件蓝图，蓝图中在画布面板下只有一个进度条。将进度条设置为变量后进入事件图表，创建一个float型参数，命名为“传递值”。我们要通过传递的值（当前生命值）与最大生命值（10）进行计算求出进度条的位置。

然后回到健康组件，在前面的节点完成之后CreateWidget。Create之后不用add to viewport，而是SetWidget。这个函数接受一个Widget引用作为参数，这个引用是来自于Add Widget Component节点。



这样就可以实现。

Attack组件同理

https://www.bilibili.com/video/BV1ygHeeSEXv/?spm\_id\_from=333.788.recommend\_more\_video.1&vd\_source=8dc4bce5a7680a01e3e5813461aae91c