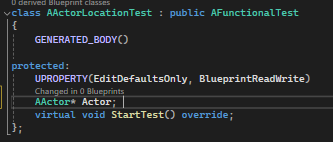
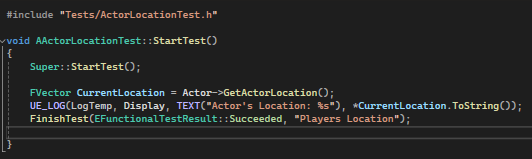
**Creating an ActorLocationTest Class**

1. Open Unreal Engine.
2. Select "C++ Classes" folder in the Content Browser.
3. Right Click and add a new C++ class.
4. Select "FunctionalTest" under "Actor" for the parent class.
5. Select "Private".
6. Input the class name "GetActorLocationTest.cpp".
7. Add "Tests/" to the end of the file path.
8. Once compiling has finished open the "ActorLocationTest.h" file in visual studio.

**Writing a ActorLocationTest C++ Base Class**

1. Inside the "ActorLocationTest.h" file add a variable for the Actor that we would like the location of. We want to add some UPROPERTYs to this variable which are: "EditDefaultsOnly" and "BlueprintReadWrite". These properties will allow us to reuse this test to find any actors location.
2. Override the StartTest function from the AFunctionalTest Parent class.
3. Next in the "GetActorLocationTest.cpp" file add the StartTest function and inside this function we will add a log, ensuring the type is set to "Display" so as to not fail the test, and the log message will say "Actors Location: %s" including the Actor.GetActorLocation() Value as a string.
4. Finally, we will call the "FinishTest" function to make the test succeed.



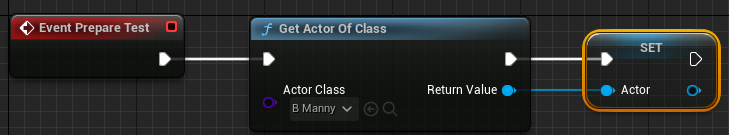


**Writing a GetPlayerLocationTest Blueprint Class**

1. Now that we have a C++ base class we can create a blueprint from this class. In Unreal Engine navigate to the C++ class we created. Right click it and select "Create blueprint class based on…" and save to a test folder with the name "GetPlayerLocation".



1. For getting the player in a map that its spawns into (not present from the start) we will need to first find the player actor and assign to the Actor variable. We will use the "Prepare Test" event for this. From there we call the blueprint function "GetActorOfClass" and select our player class. We can then set our Actor variable with the returned actor.
2. Finally, we can drag the blueprint class into the map, save It and then the test will be present in the "Session Frontend".



**Writing a ActorLocationTest Blueprint Class**

1. If we want to find the location of another actor we can create another blueprint class like above, with a relevant name, but instead of finding the player class in the "Prepare Test" event we can find the other actor.
2. Optionally if the other actor is already present in the map you can select the Actor variable on the left hand side of the screen, if it's not present ensure "show Inherited Variables" is selected in the settings. And set the default Actor value on the right hand side of the screen.

