

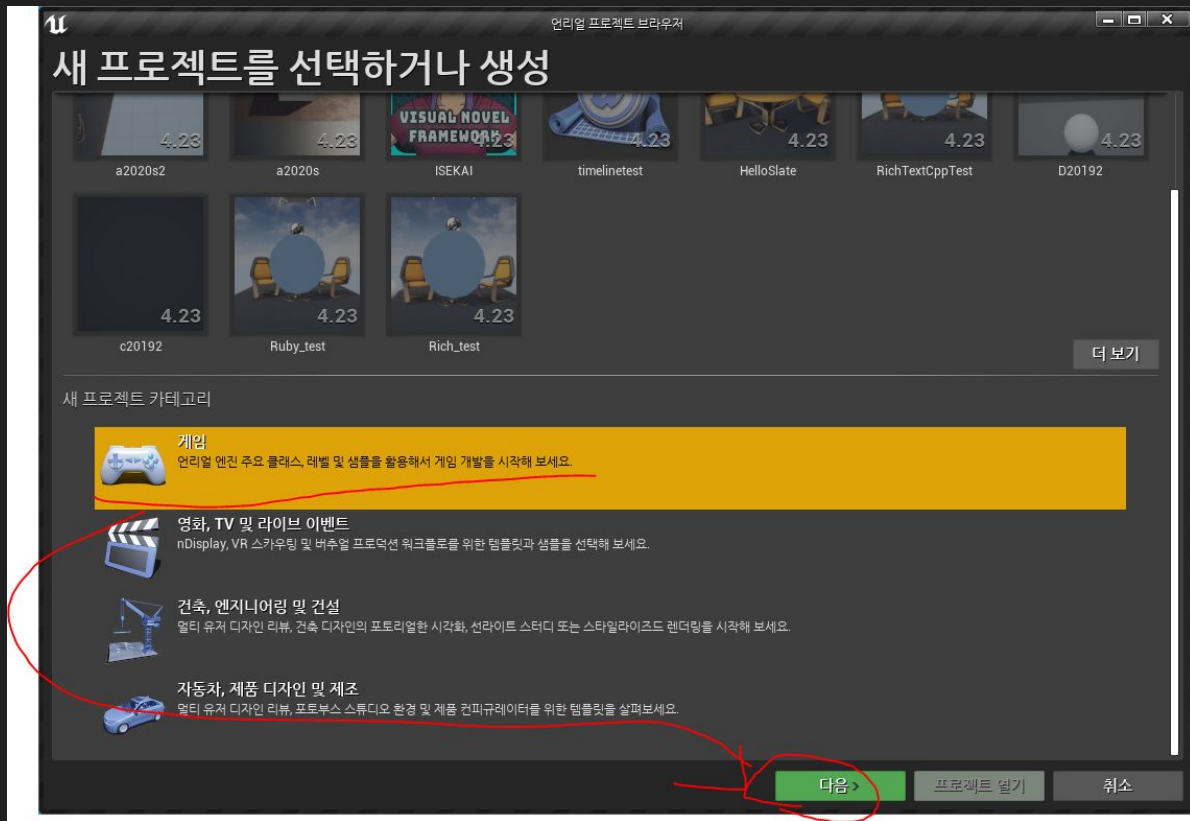
언리얼 CPP

2강

할 거

- character 클래스 활용
- c++ 와 블루프린트의 유기적 활용

프로젝트 만들기



템플릿 선택



기본



일인칭



비행



퍼즐



굴리기



삼인칭



내려보기



트윈 스틱 슈팅



휴대형 증강 현실



휠스크롤



2D 휠스크롤



비행



가상 현실



비행 고급



삼인칭

삼인칭 템플릿은 카메라가 머리위 약간 뒤쪽에 위치한 플레이가능 캐릭터를 특징으로 합니다. 마우스, 키보드, 터치 디바이스의 가상 조이스틱으로 캐릭터가 이동함에 따라 카메라는 캐릭터를 따라다니며 어깨위 시점으로 봅니다. 주인공 캐릭터를 강조하는 시점으로, 액션이나 어드벤처 게임에 많이 쓰입니다. 레벨에는 플레이어가 점프할 수 있는 오브젝트가 여럿 있습니다.

애셋 유형 레퍼런스:

Animation Sequence, Animation Blueprint, Skeleton, Blend Space 1D, Skeletal Mesh

클래스 유형 레퍼런스:

GameMode, Character, SpringArmComponent, CameraComponent, InputComponent

◀ 뒤로

다음 ▶

프로젝트 생성

취소

프로젝트 세팅



블루프린트

블루프린트 또는 C++ 프로젝트 생성 여부를 선택합니다.



최대 품질

프로젝트의 성능 특성을 선택합니다.



레이 트레이싱 비활성화됨

실시간 레이 트레이싱을 새 프로젝트에서 활성화해야 하는지 선택합니다.



데스크톱 / 콘솔

타겟 플랫폼과 가장 가까운 플랫폼을 선택합니다. 걱정하지 마세요.
프로젝트 세팅의 타겟 하드웨어 섹션에서 나중에 변경할 수 있습니다.

시작용 콘텐츠 포함

켜면 기본적인 머티리얼 및 텍스처와 함께 간단히 배치가능한 메시가 들어있는 부가 콘텐츠 팩을 포함시킵니다.
포함시키는 옵션을 끄면 선택된 프로젝트 템플릿에 꼭 필요한 것들만 된 프로젝트를 만들 수 있습니다.

프로젝트를 저장할 위치를 선택하세요.

C:\Users\LJH\Documents\Unreal Projects

폴더

b2020s

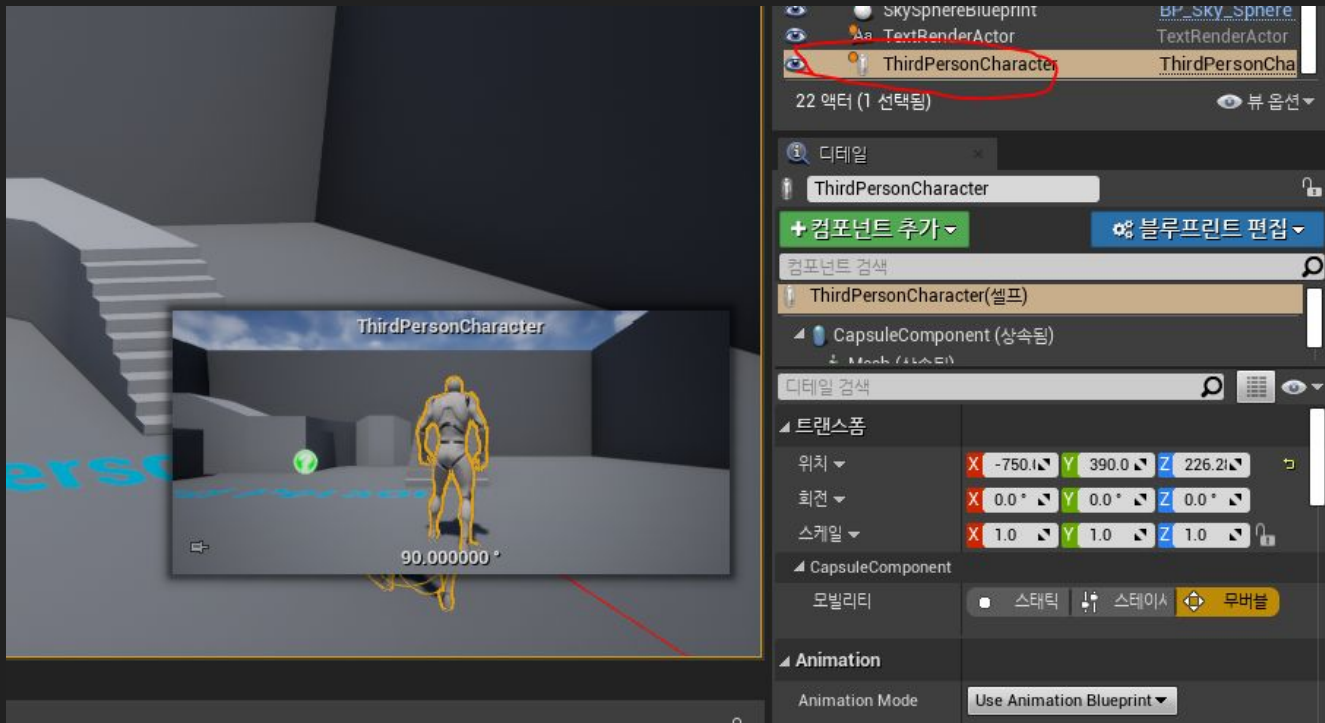
이름

< 뒤로

프로젝트 생성

취소

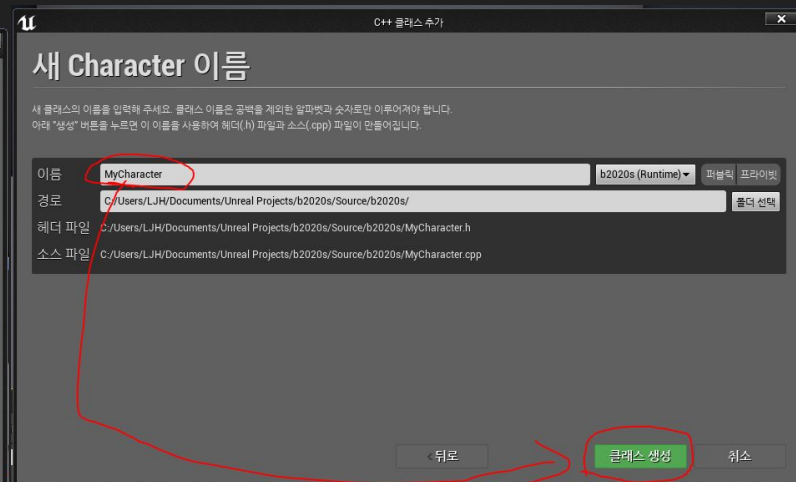
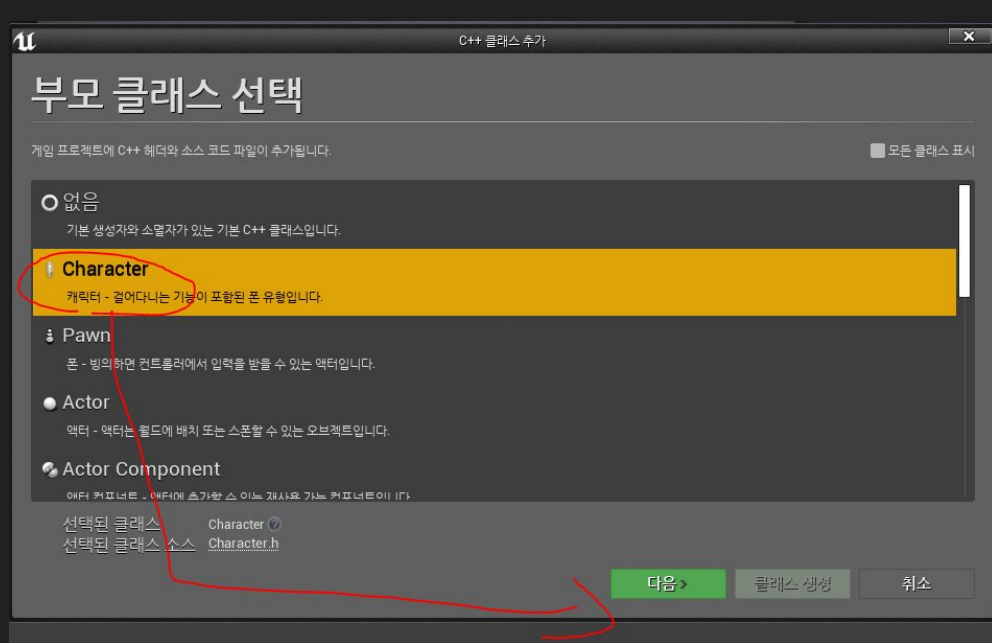
이미 있는 이 친구를 cpp 로 다시 만들거임.



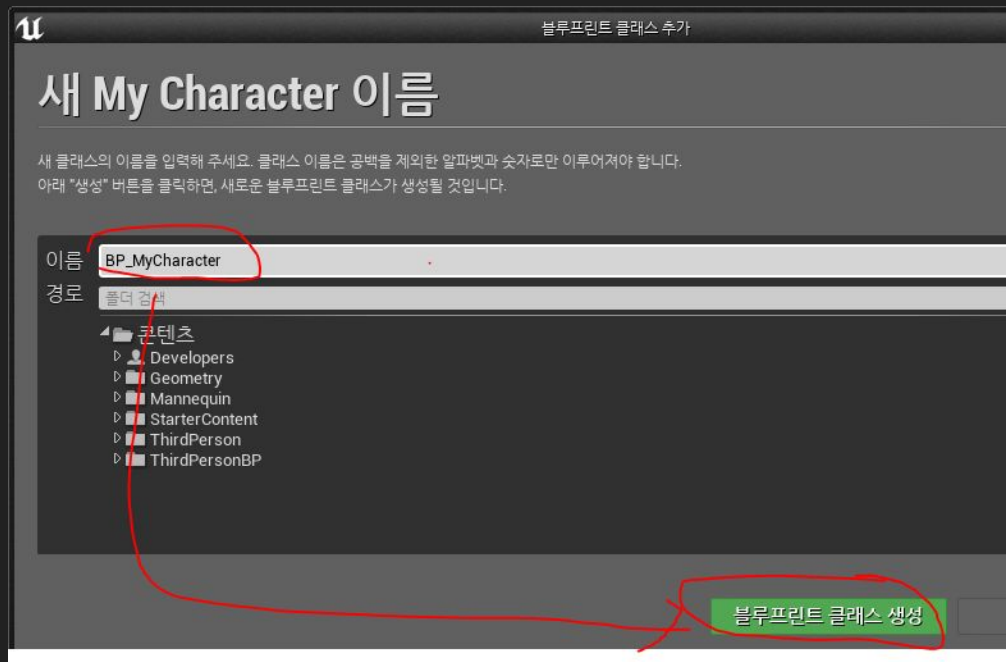
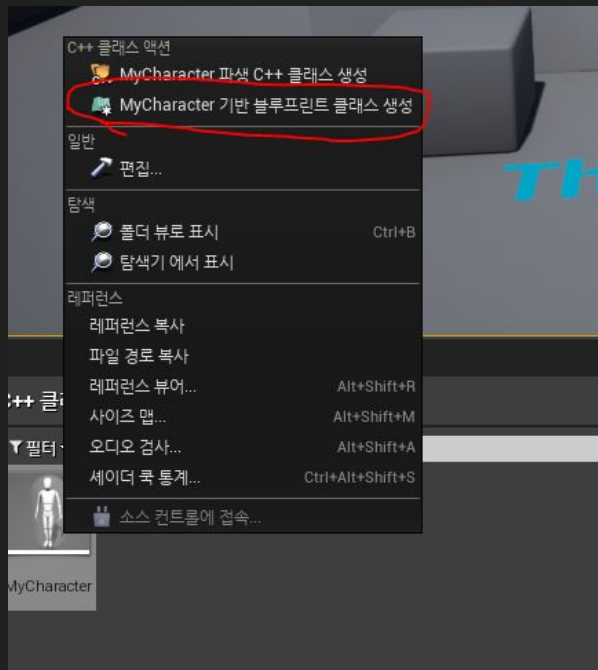
cpp 클래스 만들기



character 를 상속

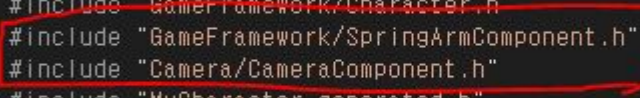


만들어진 클래스를 상속하는 블루프린트 만들기



컴포넌트를 추가 - MyCharacter.h 에 추가

```
#include "CoreMinimal.h"  
#include "GameFramework/Character.h"  
#include "GameFramework/SpringArmComponent.h"  
#include "Camera/CameraComponent.h"  
#include "MyCharacter.generated.h"
```



```
UCLASS( )
```

```
UPROPERTY(EditAnywhere)  
USpringArmComponent* SpringArm;
```

```
UPROPERTY(EditAnywhere)  
UCameraComponent* Camera;
```

.cpp 에 생성자에 추가

```
#include "MyCharacter.h"

// Sets default values
AMyCharacter::AMyCharacter()
{
    // Set this character to call Tick() every frame. You can turn this off to improve performance if you don't need it.
    PrimaryActorTick.bCanEverTick = true;

    SpringArm = CreateDefaultSubobject<USpringArmComponent>(TEXT("SpringArm"));
    Camera = CreateDefaultSubobject<UCameraComponent>(TEXT("Camera"));

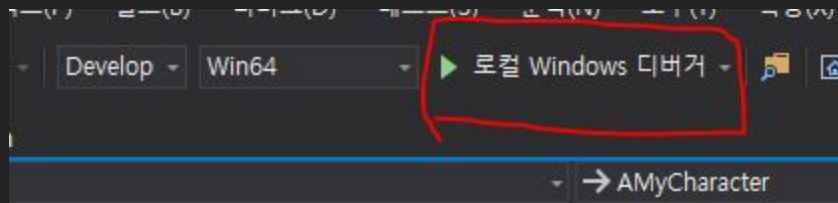
    SpringArm->SetupAttachment(RootComponent);
    Camera->SetupAttachment(SpringArm);
}
```

지금부터 매우 그지같음. 중요.

엔진을
저장하고
끈다.

비주얼 스튜디오에서 컴파일

F5 눌러도 된다.



엔진이 다시 커짐

b2020s
EDITOR

언리얼 에디터 4.25.1 - b2020s

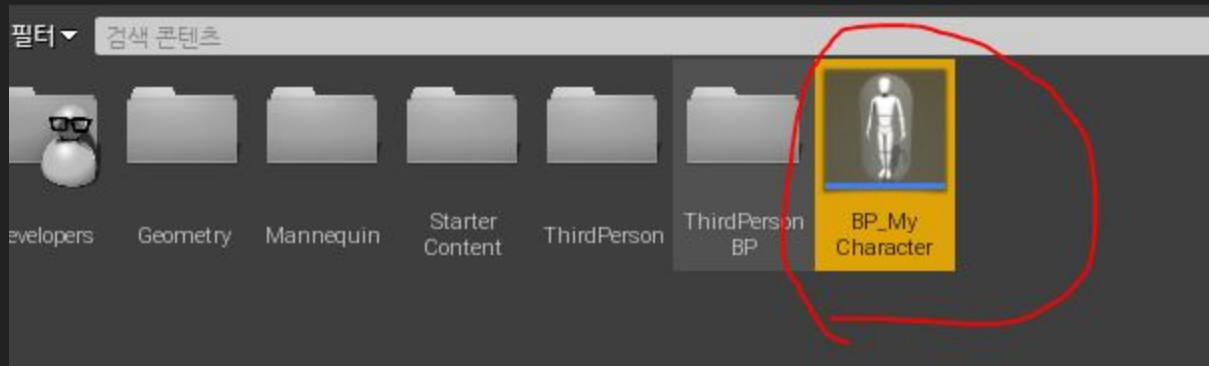
Copyright © Epic Games, Inc. All rights reserved.



초기화.. 45%

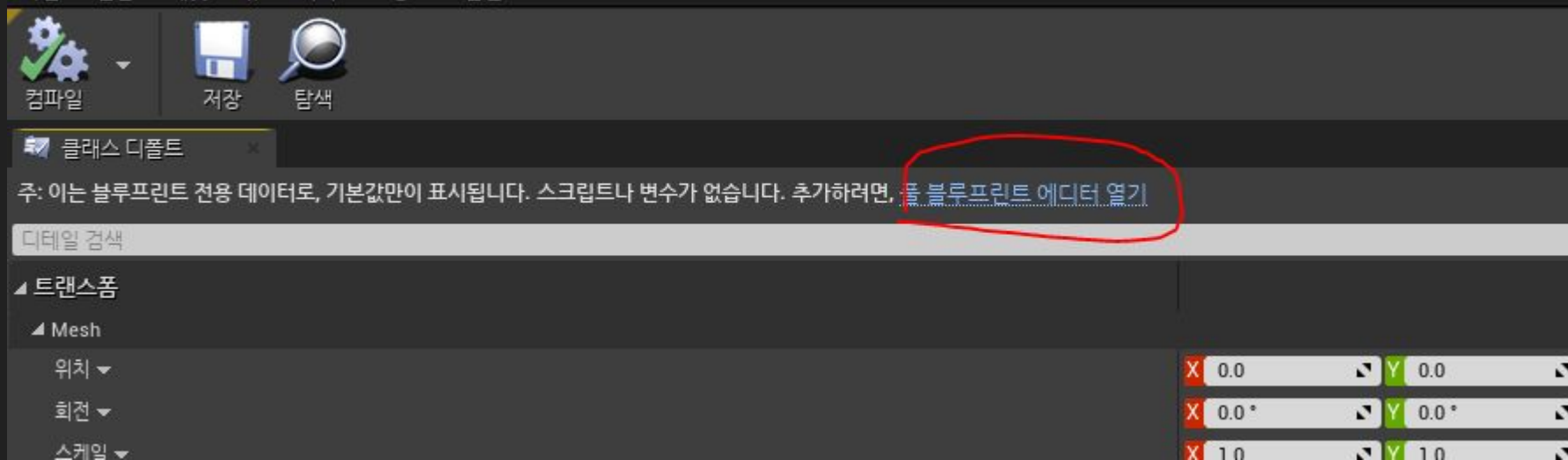
Component)

우리가 만들었던 애를 눌러본다.

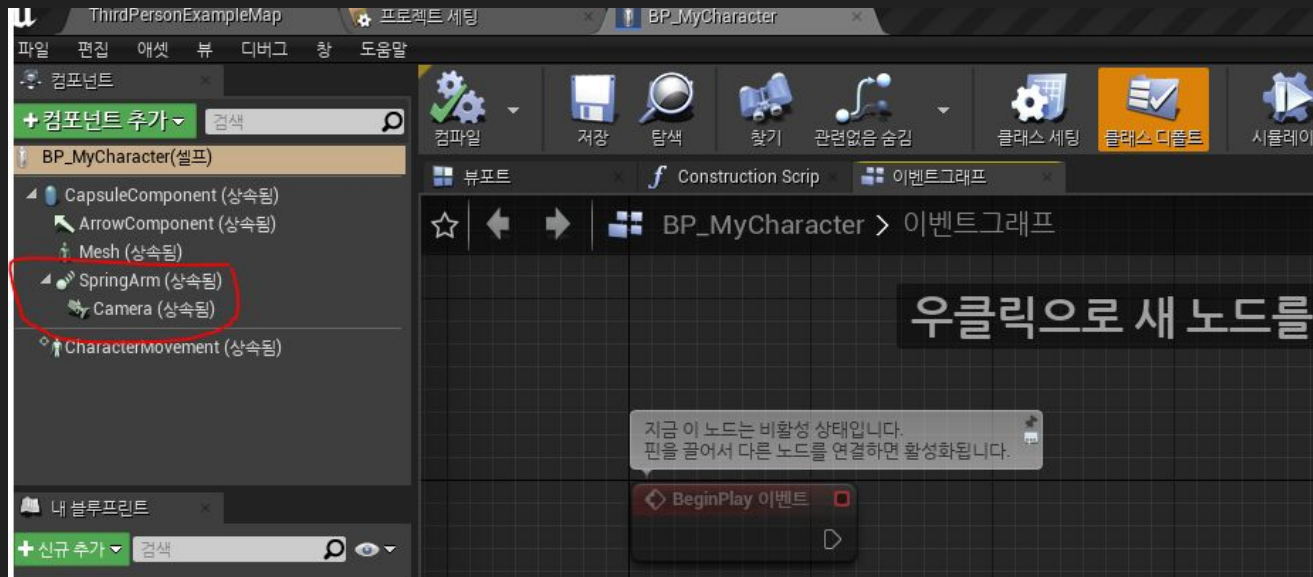


폴 플루프린트 에디터 열기

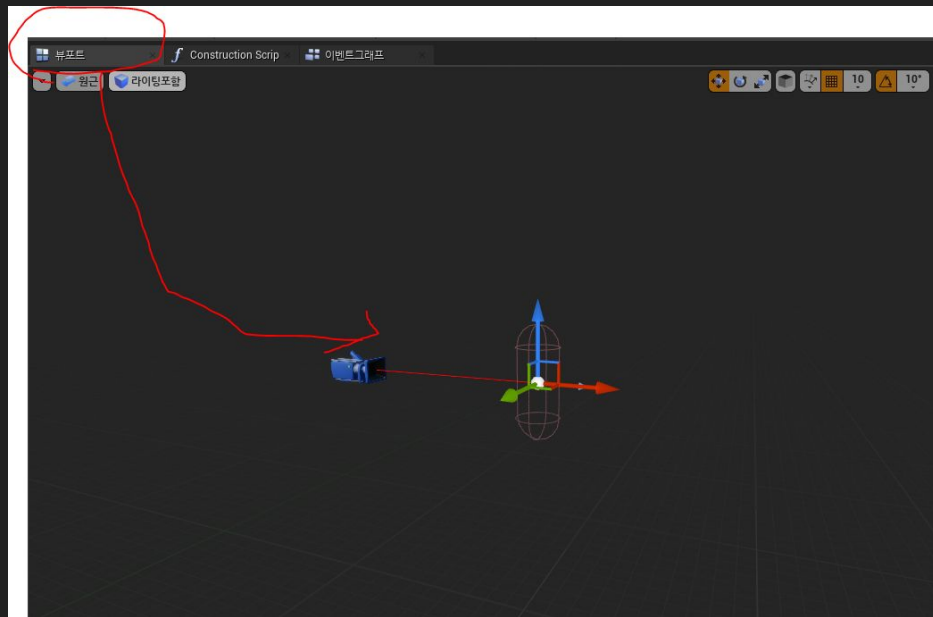
아무 블루프린트가 없어서 이렇게 뜨는건데 일단 누른다.



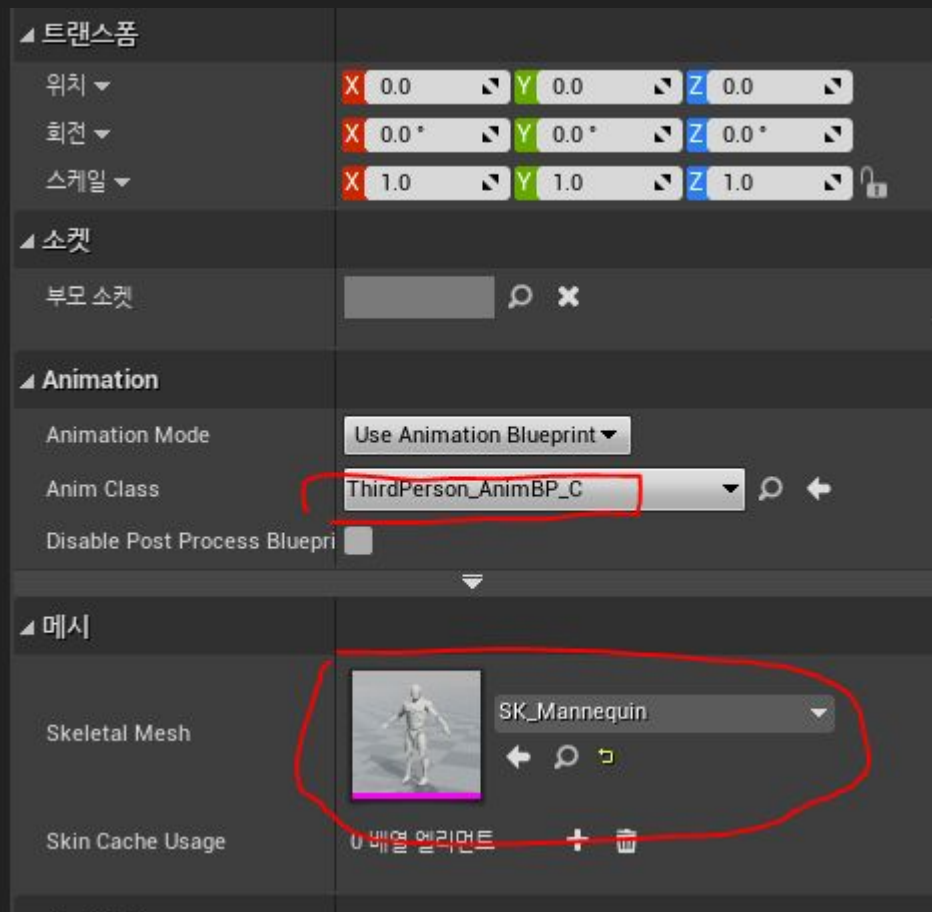
우리가 추가한 컴포넌트가 보인다.



뷰포트를 누르면 전체 모양을 볼수 있다.

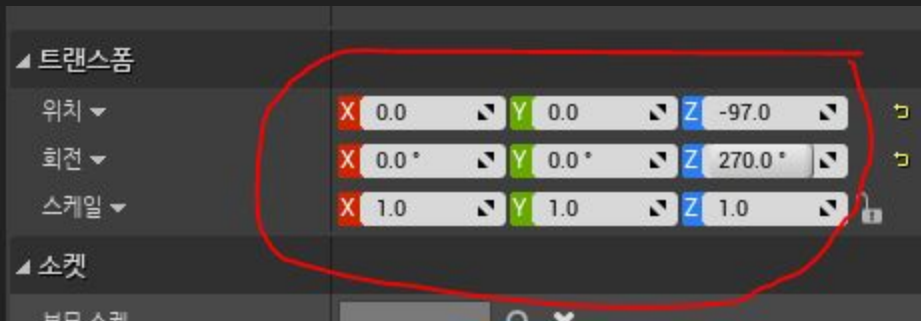


Mesh 를 누르고 2개 수정.



위치가 좀 이상하니까 위치도 수정

※좌표표가 있는 방향이 앞방향이다



그상태로 필드에 드래그 해서 올리면

잘 있다.



기본 캐릭터 바꾸기

지난주에 했던 프로젝트
세팅을 들어가서

맵&모드

프로젝트

게임플레이 태그

▶ 맵 & 모드

무비

설명

암호화

지원 플랫폼

타겟 하드웨어

패키징

게임

게임 데이터

기본 캐릭터를 우리가 만든거로.

이 세팅은 DefaultEngine.ini 에 저장되었으며, 현재 쓰기가능 입니다.

Default Modes

기본 게임모드

선택된 GameMode

Default Pawn Class

HUD Class

Player Controller Class

Game State Class

Player State Class

Spectator Class

ThirdPersonGameMo ◀ 🔍 +

BP_MyCharacter ◀ 🔍 +

HUD ◀ 🔍 +

PlayerController ◀ 🔍 +

GameStateBase ◀ 🔍 +

PlayerState ◀ 🔍 +

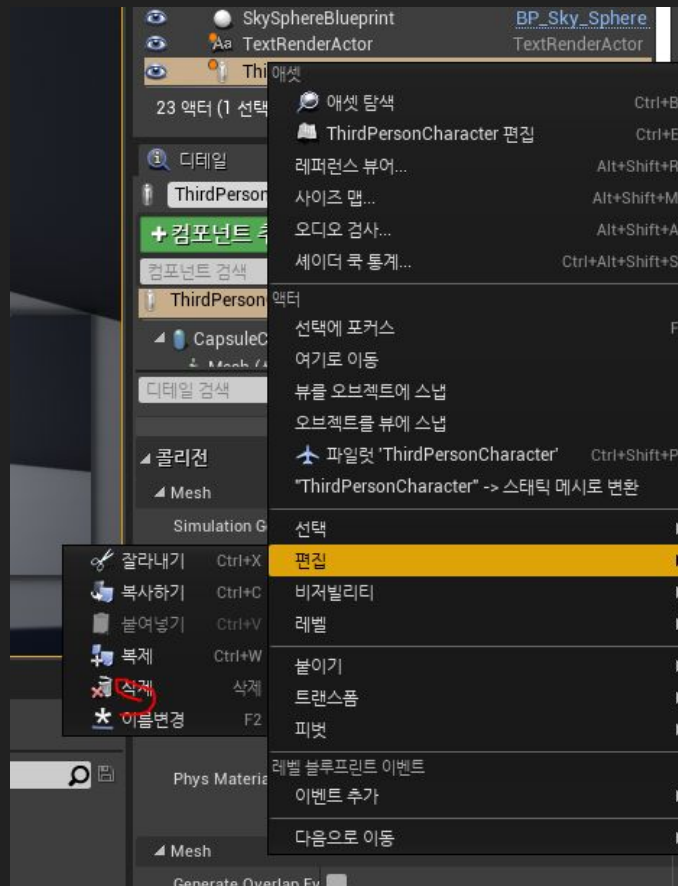
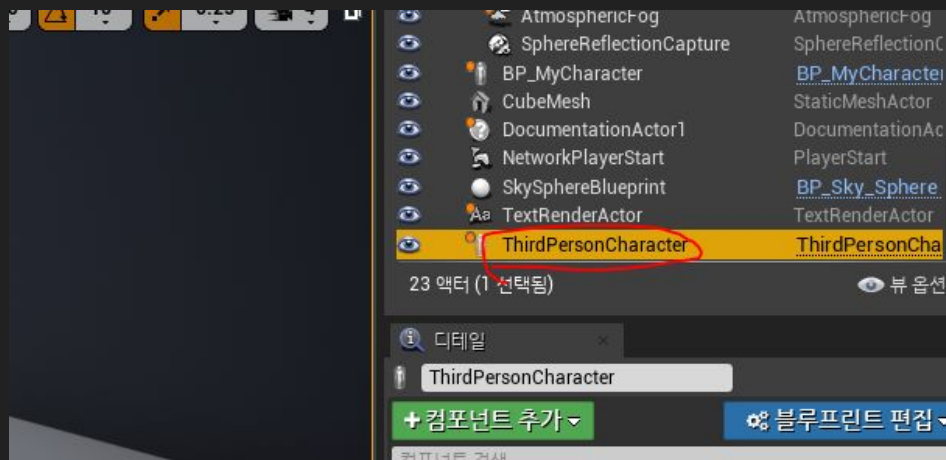
SpectatorPawn ◀ 🔍 +

Default Maps



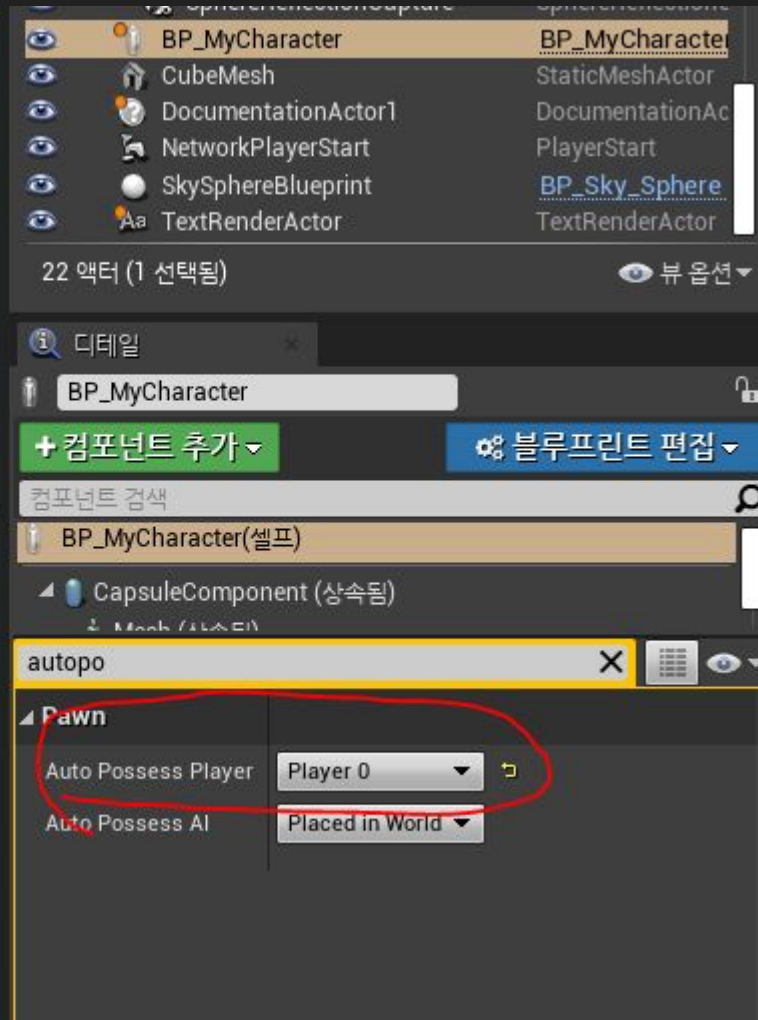
ThirdPersonExampleMan

원래 있던애를 삭제



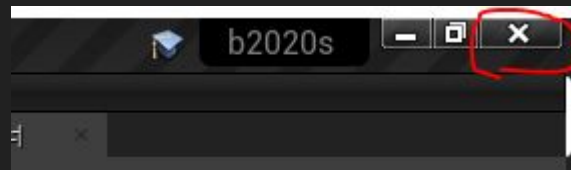
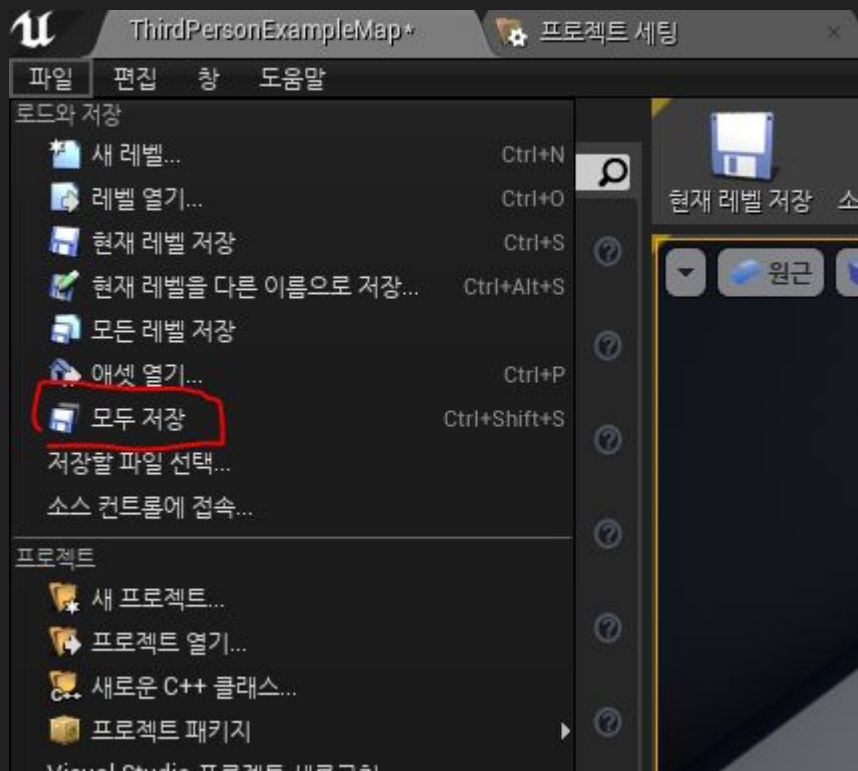
Auto Possess

우리가 필드에 올려둔 우리
캐릭터를 강제로 possess
시킴.



play 눌러보면 캐릭터가 안움직인다.

저장하고 끄기



이동 등등 구현.

이미 축 이랑 액션 이 되어있다.

엔진 - 입력

기본 입력 동작 및 축 설정이 포함된 입력 세팅입니다.

 이 세팅은 DefaultInput.ini 에 저장되었으며, 현재 쓰기가 가능합니다.

바인딩

액션 및 축 매핑은 입력 동작과 그것을 실행하는 키 사이에 중간 레이어를 삽입하는 방식으로, 입력 동작에 대한 키와 축 매핑

액션 매핑 +

▷ Jump + ×

▷ ResetVR + ×

축 매핑 +

▷ MoveForward + ×

▷ MoveRight + ×

▷ TurnRate + ×

▷ Turn + ×

▷ LookUpRate + ×

▷ LookUp + ×

스피치 매핑

0 배

이동 함수 만들기

.h 파일에 만든다.

```
__FUNCTION()  
void MoveForward(float Value);
```

```
__FUNCTION()  
void MoveRight(float Value);
```

바인딩

```
// Called to bind functionality to input
void AMyCharacter::SetupPlayerInputComponent(UInputComponent* PlayerInputComponent)
{
    Super::SetupPlayerInputComponent(PlayerInputComponent);

    PlayerInputComponent->BindAxis("MoveForward", this, &AMyCharacter::MoveForward);
    PlayerInputComponent->BindAxis("MoveRight", this, &AMyCharacter::MoveRight);
}
```

함수 구현

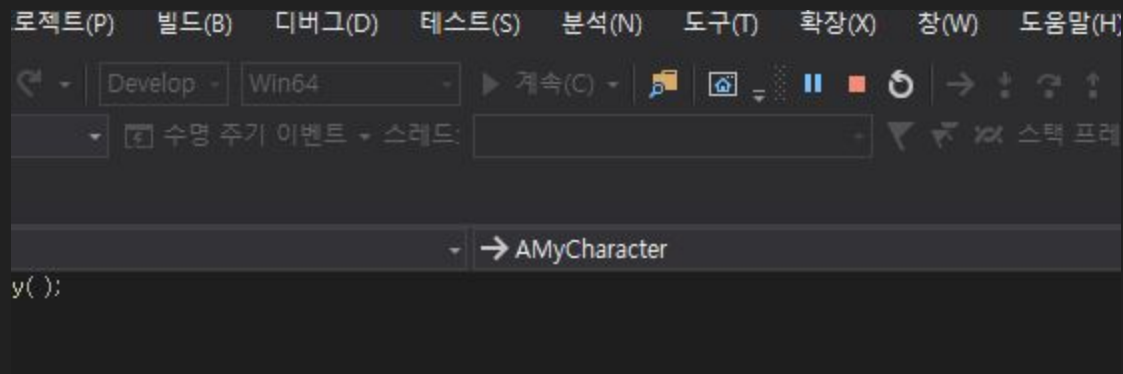
```
void AMyCharacter::MoveForward(float Value)
{
    FVector ForwardVector = FRotationMatrix(Controller->GetControlRotation()).GetScaledAxis(EAxis::X);

    AddMovementInput(ForwardVector, Value);
}
```

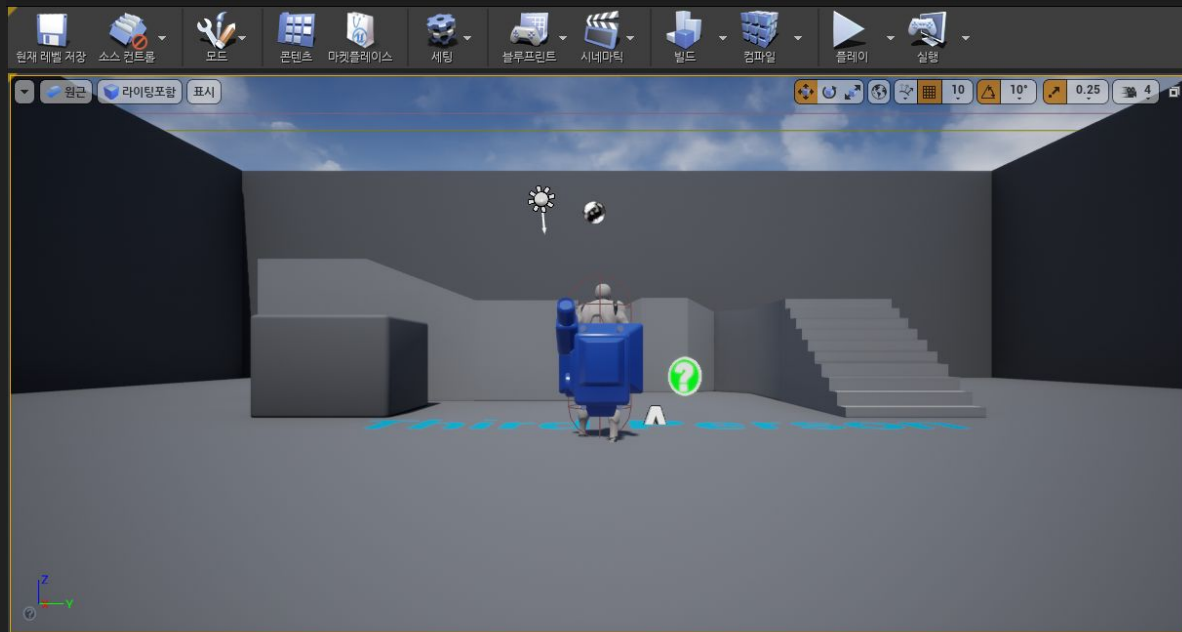
```
void AMyCharacter::MoveRight(float Value)
{
    FVector RightVector = FRotationMatrix(Controller->GetControlRotation()).GetScaledAxis(EAxis::Y);

    AddMovementInput(RightVector, Value);
}
```

다시 또 vs 에서 컴파일



뭔가 이상하지만 움직이긴 한다.



마우스 입력

이미 있는 함수랑도 연결해줄수 있다.

이미 **character** 클래스에 구현되어있다.

```
// Called to bind functionality to input
void AMyCharacter::SetupPlayerInputComponent(UInputComponent* PlayerInputComponent)
{
    Super::SetupPlayerInputComponent(PlayerInputComponent);

    PlayerInputComponent->BindAxis("MoveForward", this, &AMyCharacter::MoveForward);
    PlayerInputComponent->BindAxis("MoveRight", this, &AMyCharacter::MoveRight);

    PlayerInputComponent->BindAxis("Turn", this, &AMyCharacter::AddControllerYawInput);
    PlayerInputComponent->BindAxis("LookUp", this, &AMyCharacter::AddControllerPitchInput);
}
```

점프도 추가.

점프도 이미 구현되어있다.

```
// Called to bind functionality to input
void AMyCharacter::SetupPlayerInputComponent(UInputComponent* PlayerInputComponent)
{
    Super::SetupPlayerInputComponent(PlayerInputComponent);

    PlayerInputComponent->BindAxis("MoveForward", this, &AMyCharacter::MoveForward);
    PlayerInputComponent->BindAxis("MoveRight", this, &AMyCharacter::MoveRight);

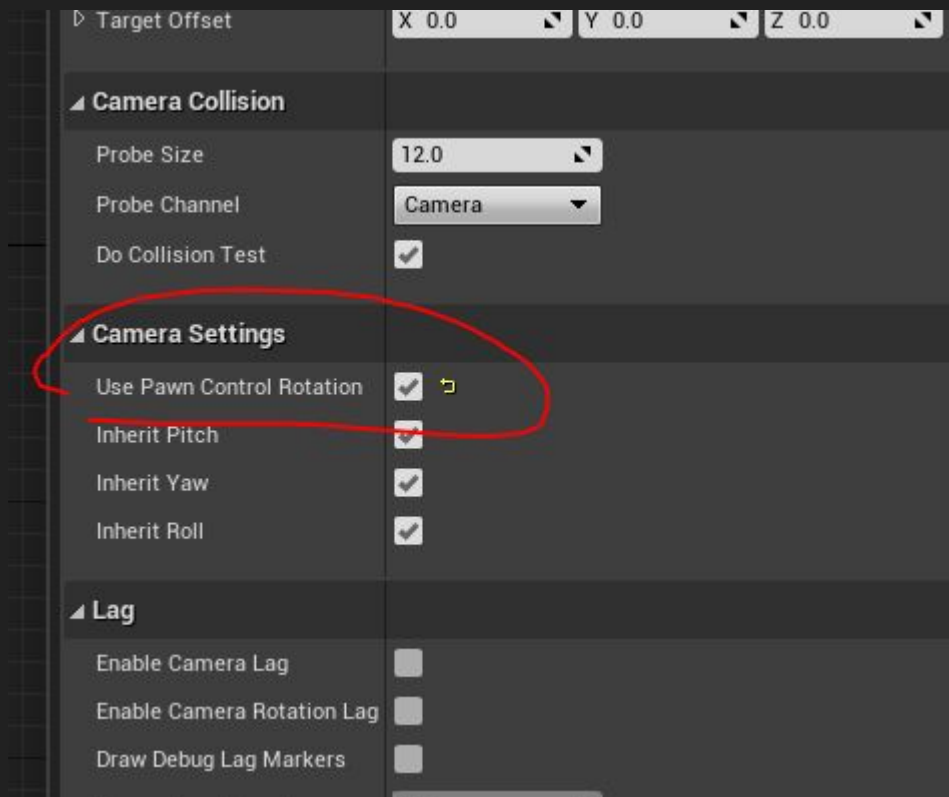
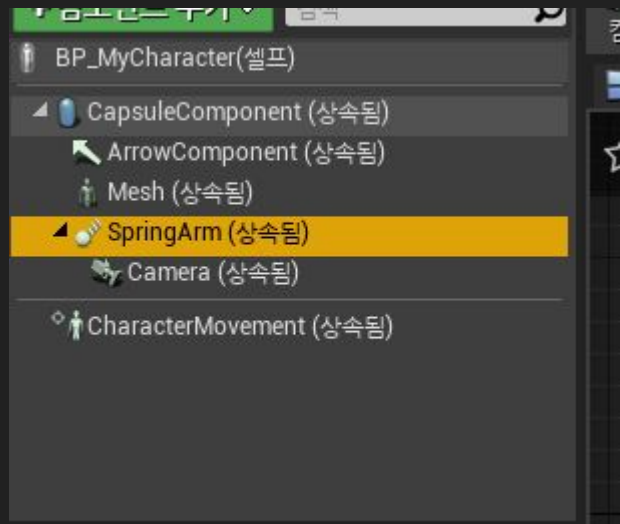
    PlayerInputComponent->BindAxis("Turn", this, &AMyCharacter::AddControllerYawInput);
    PlayerInputComponent->BindAxis("LookUp", this, &AMyCharacter::AddControllerPitchInput);

    PlayerInputComponent->BindAction("Jump", IE_Pressed, this, &AMyCharacter::Jump);
    PlayerInputComponent->BindAction("Jump", IE_Released, this, &AMyCharacter::StopJumping);
}
```

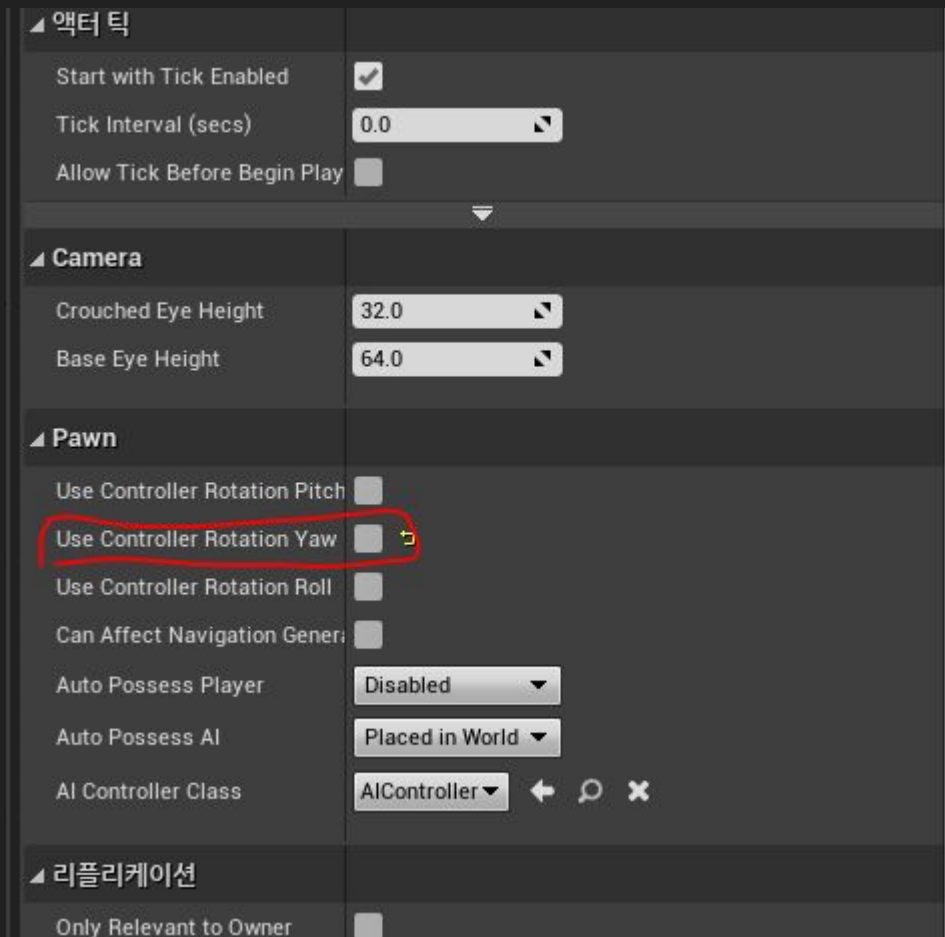
또 다시 비주얼 스튜디오에서 컴파일

이제 잘 움직이고 화면도 돌아간다.

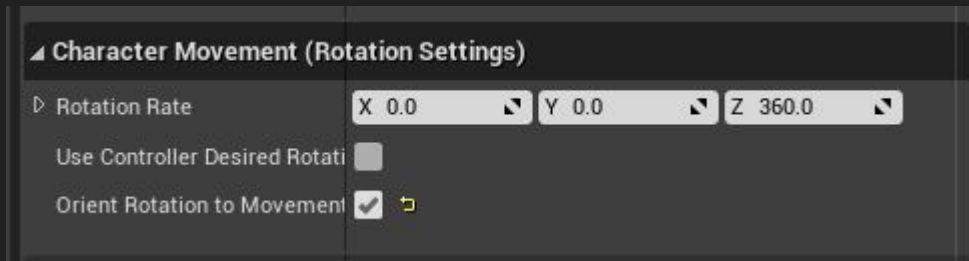
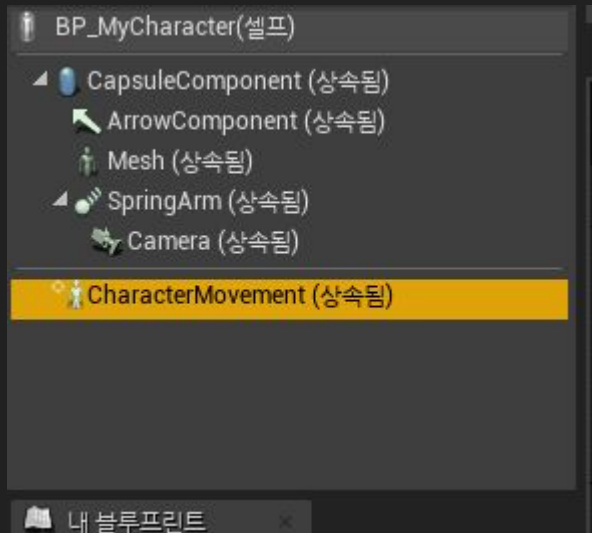
설정 세가지 해주면 덜 어색해진다. -1



2번째



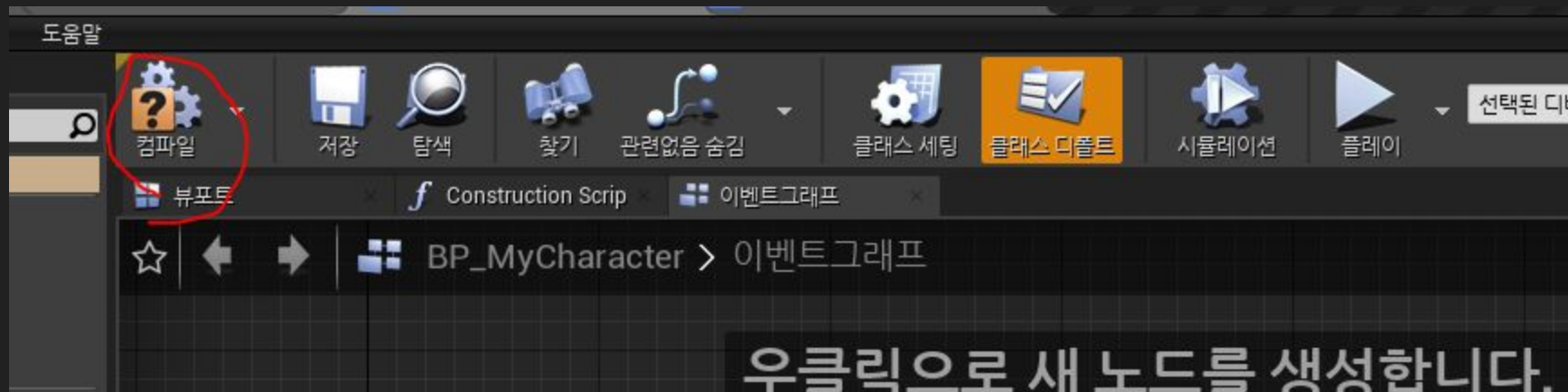
3번째



3인칭에서 자주쓰이는 설정임.

자주 쓰이는 설정인데 맨날 까먹으니까 찾아서 써야된다.

컴파일



실행하면 잘 된다.

