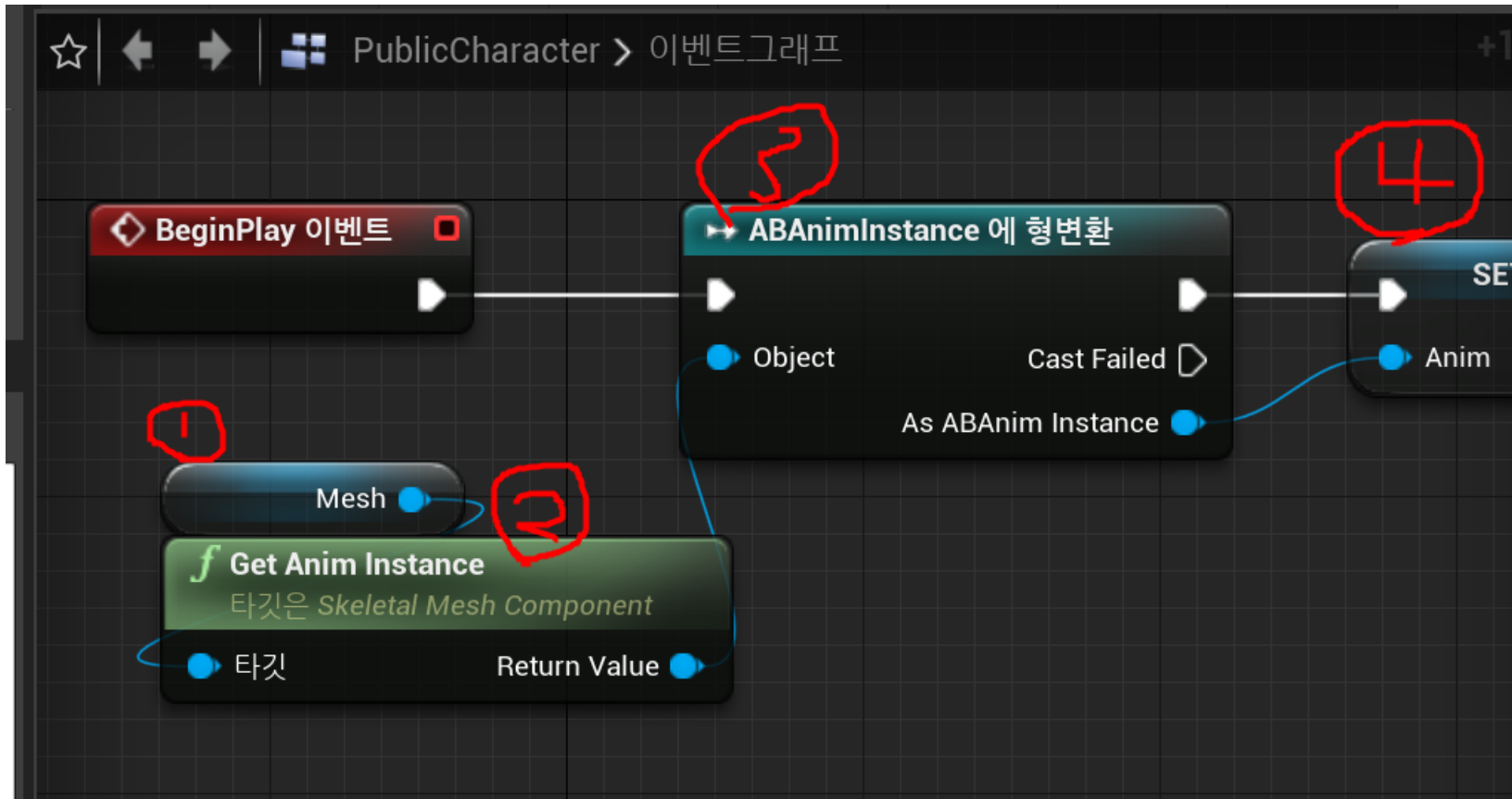


목차

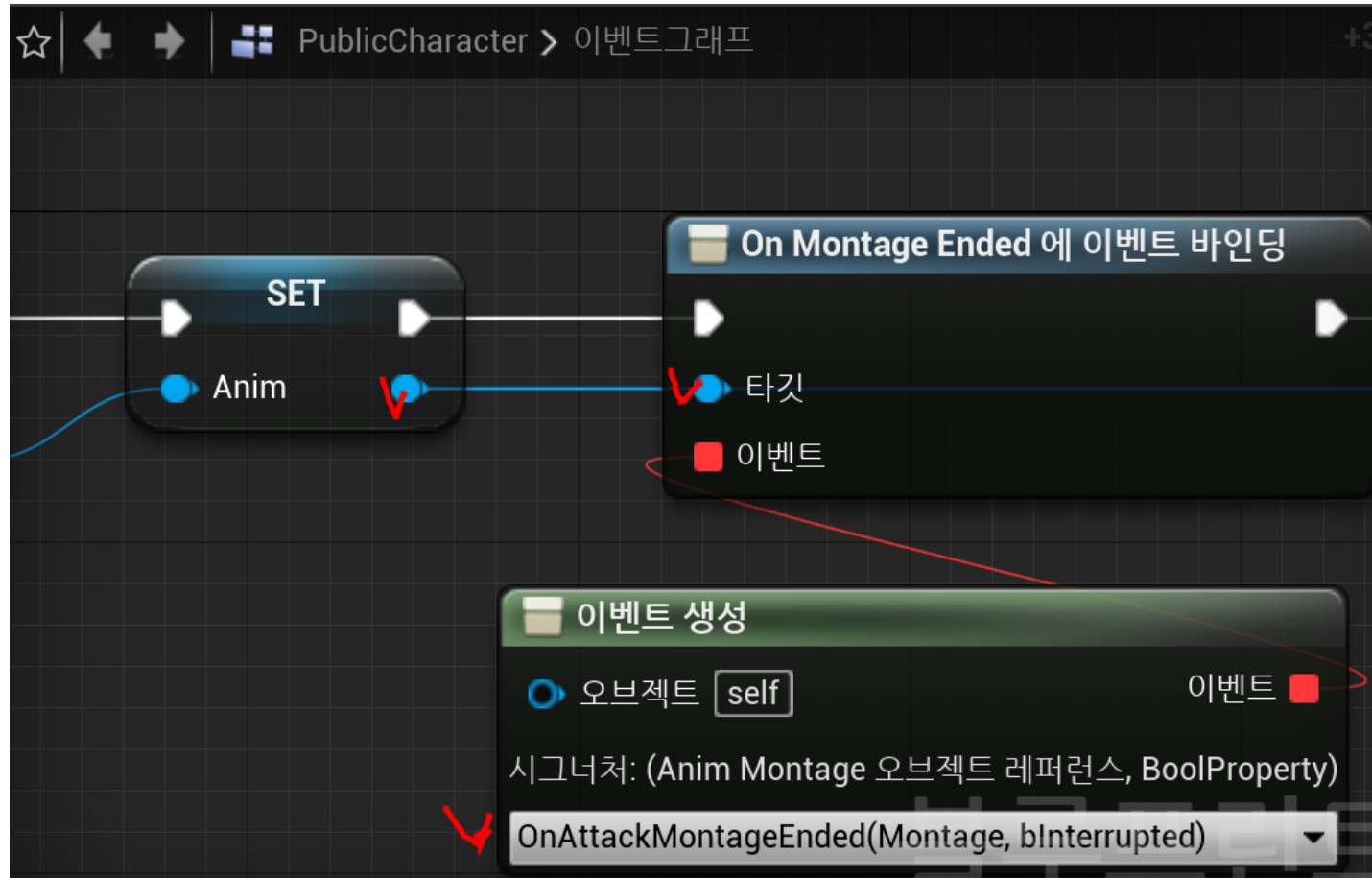
1. 블루프린트 Character를 상속받은 PublicCharacter를 생성한다
2. 프로젝트 세팅의 콜리전에 들어가서 ObjectChannels에 들어가서 오브젝트 채널을 생성한다
3. PublicCharacter를 상속받은 MariaPlayer_BP를 생성한다.
4. 카메라를 검색해서 카메라를 레벨에 배치한다.



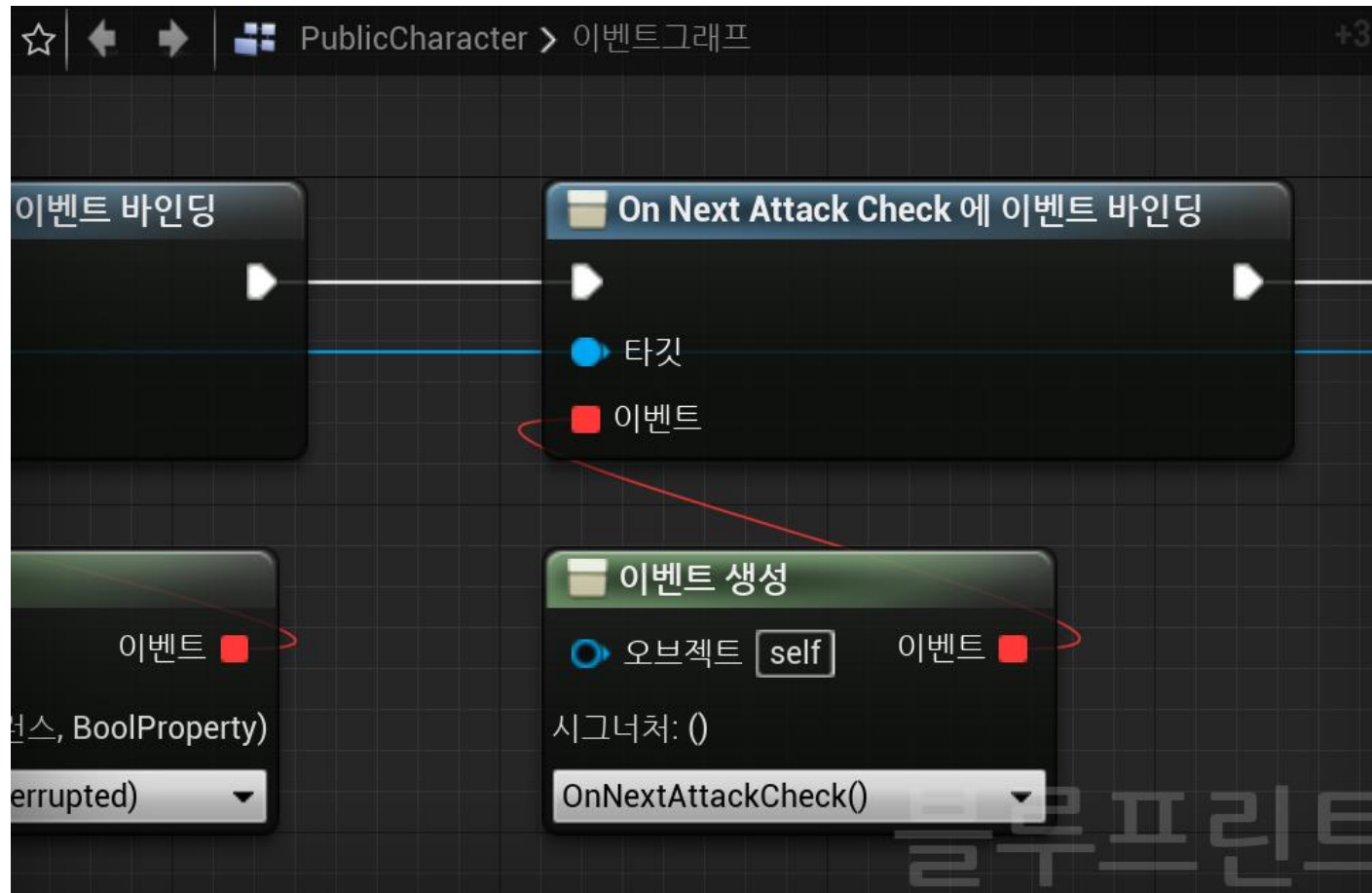
1. 블루프린트 Character를 상속받은 PublicCharacter를 생성한다



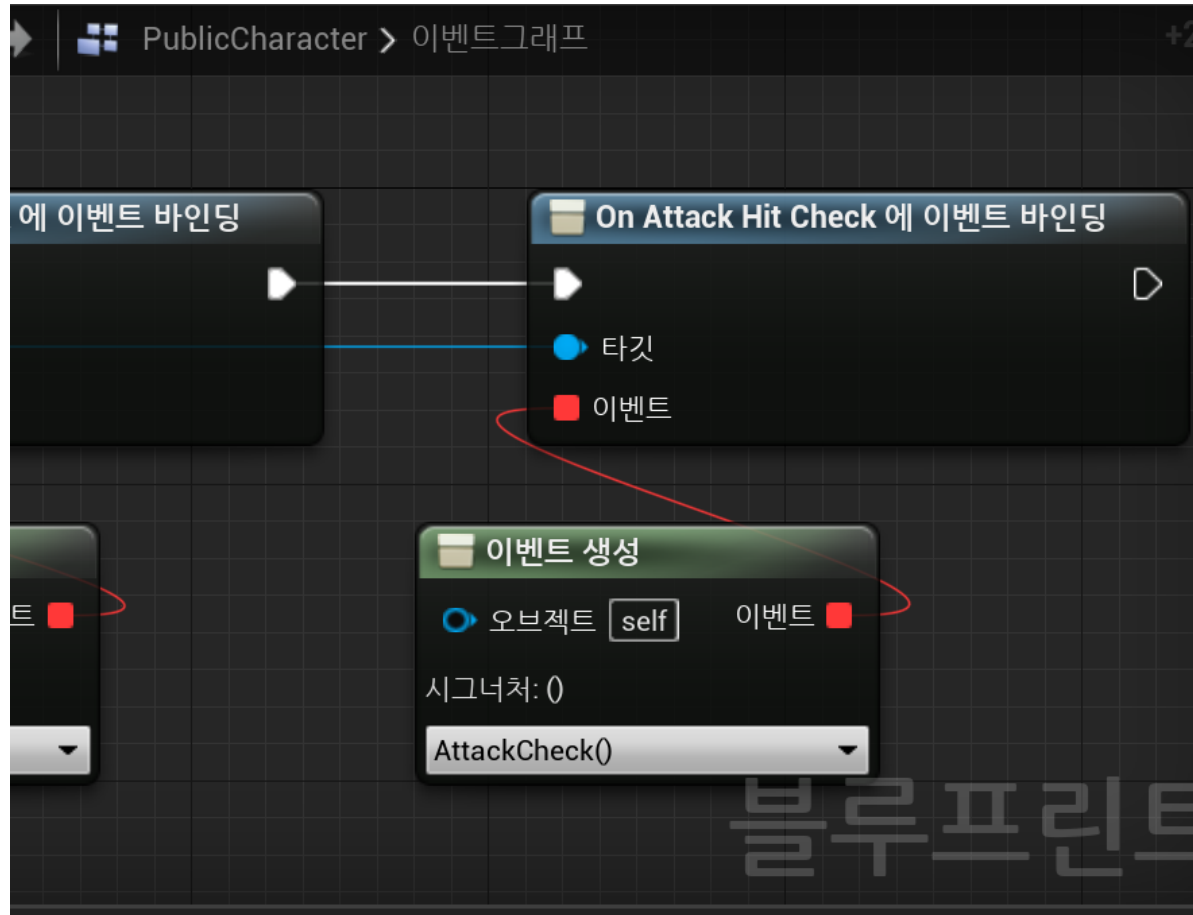
1.1 PublicCharacter에 들어가서 메시안에 있는 애님인스턴스를 ABAnimInstace로 형변환해서 변수로 승격해서 Anim을 생성한다.



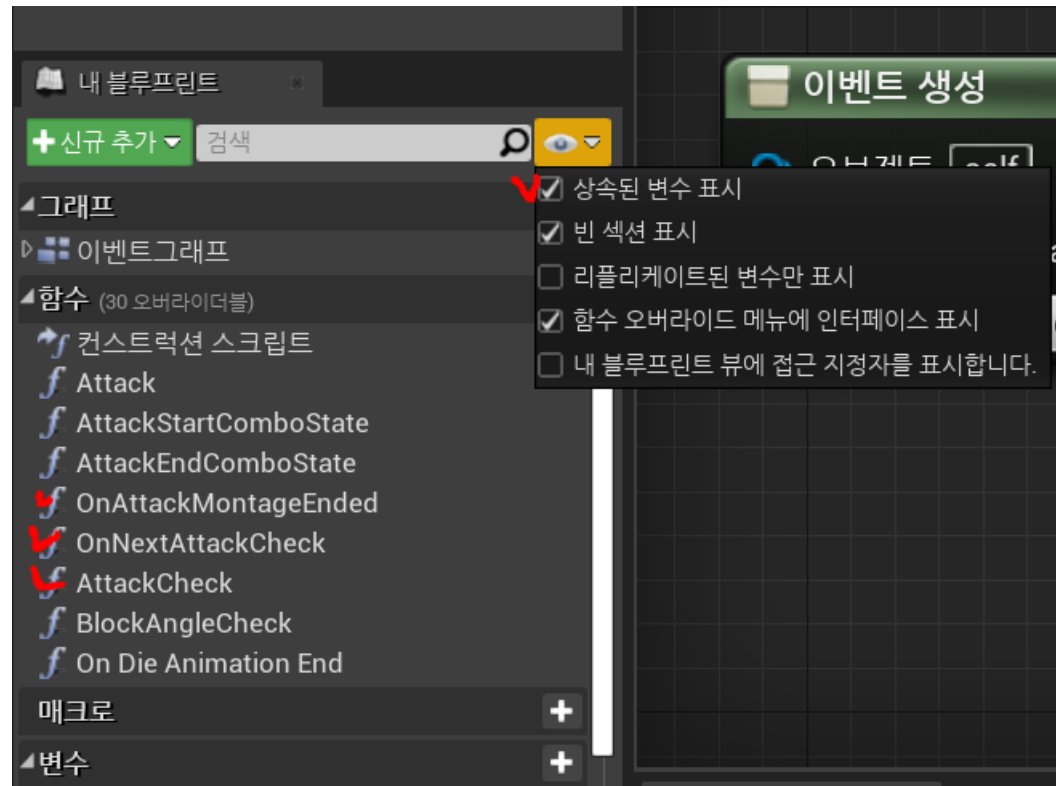
1.2 Anim에 OnMontageEnded 델리케이트를 바인딩하고 연결할 이벤트를 생성한다.



1.3 Anim에 OnNextAttackCheck 델리케이트를 바인딩하고 연결할 이벤트를 생성한다.



1.4 Anim에 OnAttackHitCheck 델리케이트를 바인딩하고 연결할 이벤트를 생성한다.



1.5 상속된 변수 표시를 체크하고, 바인딩으로 생성된 함수들

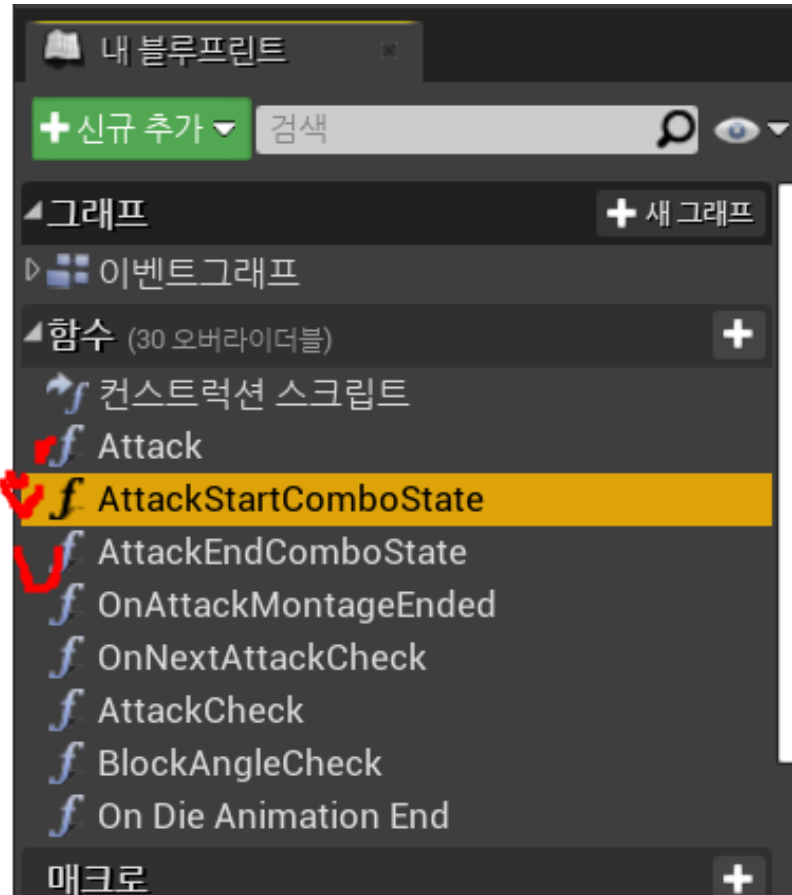


1.6 공격과 콤보에 사용될 변수를 추가하고, 카테고리 Attack에 넣는다.

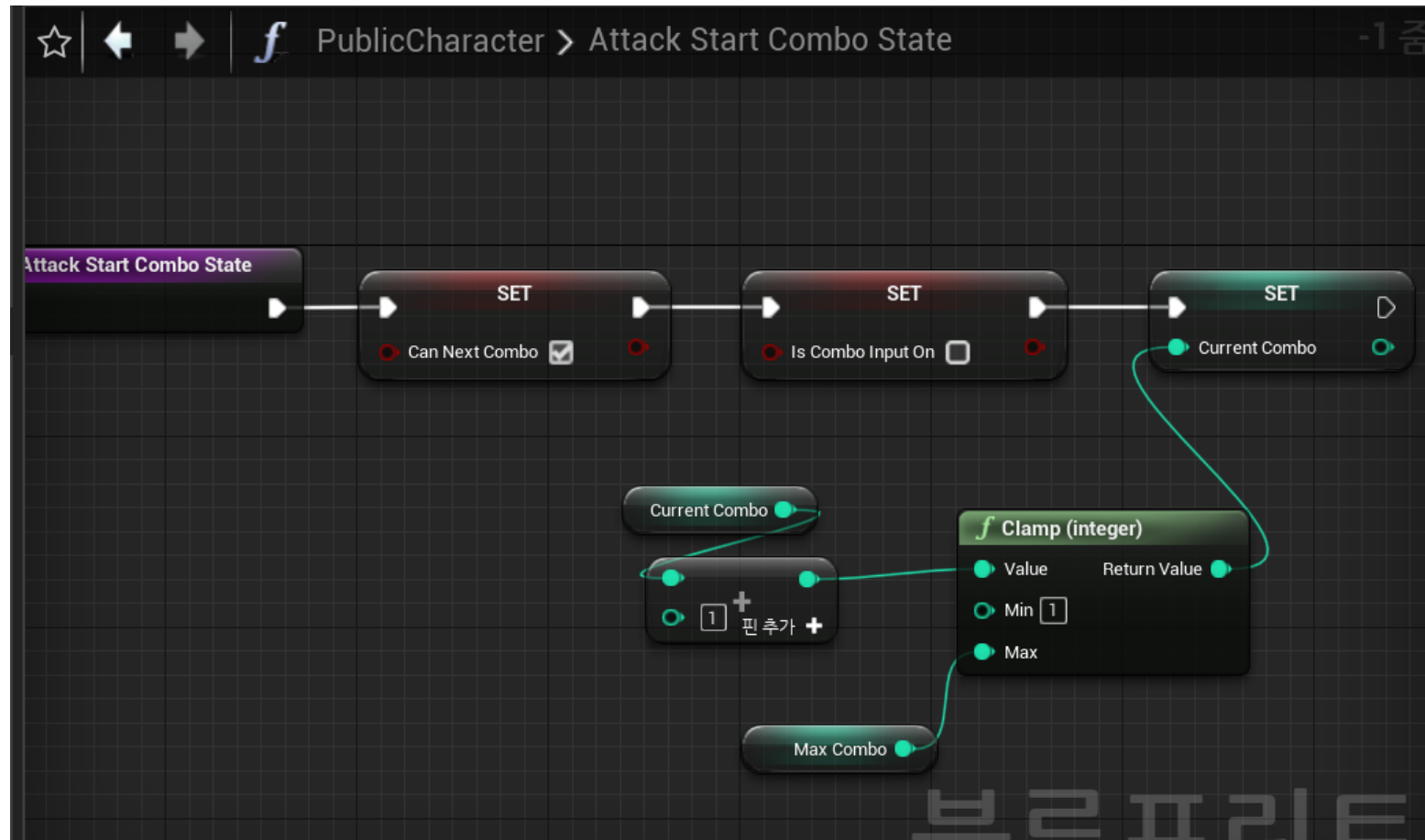
Variable	
변수 이름	MaxCombo
변수 유형	인티저
인스턴스 편집가능	☐
블루프린트 읽기 전용	☐
툴팁	
스폰시 노출	☐
프라이빗	☐
시네마틱에 노출	☐
카테고리	Attack
슬라이더 범위	
값 범위	
비트마스크	☐
비트마스크 열거형	없음
리플리케이션	None
리플리케이션 조건	None

기본값	
Max Combo	4

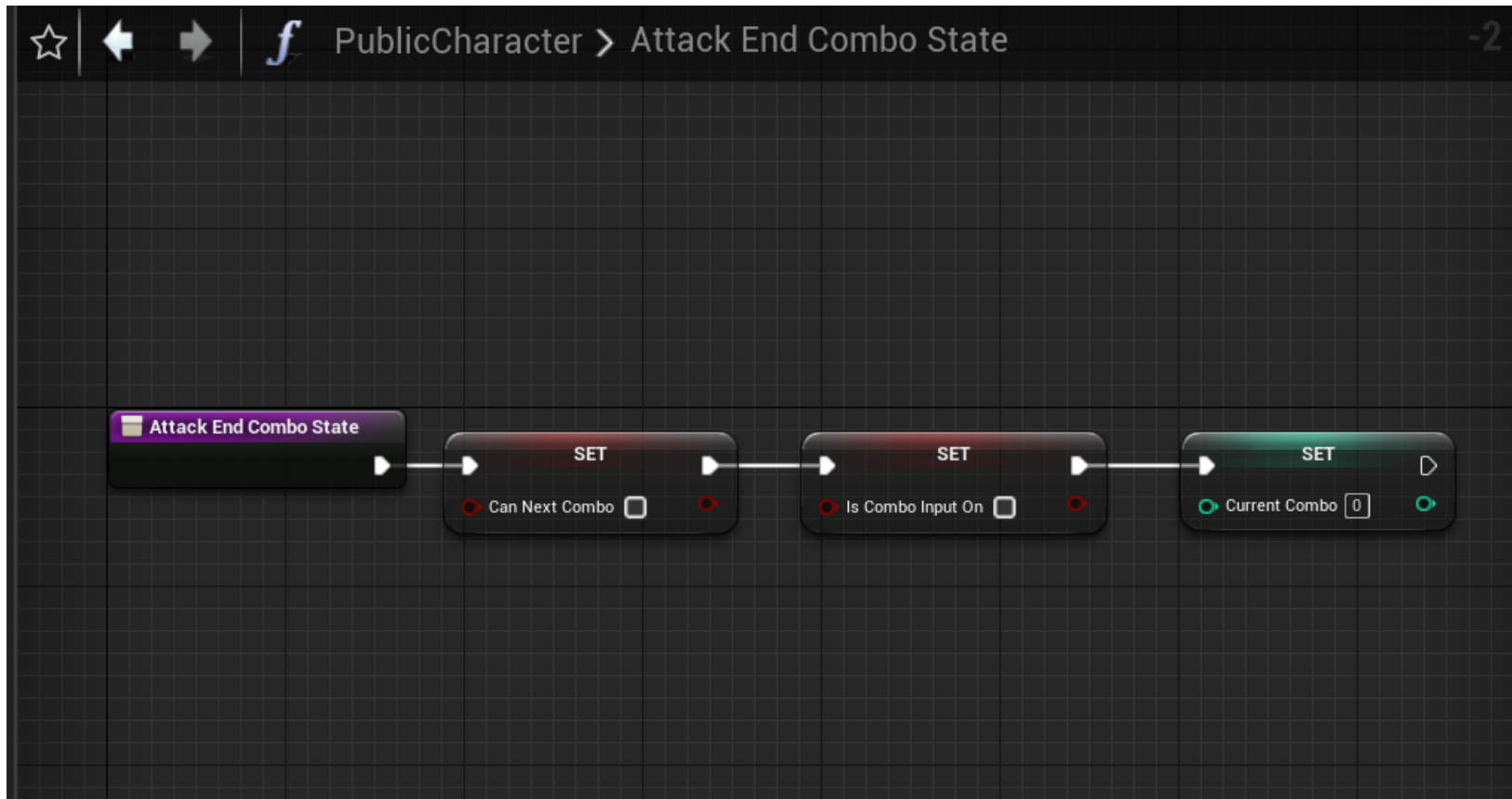
1.7 MaxCombo에는 4를 넣는다.



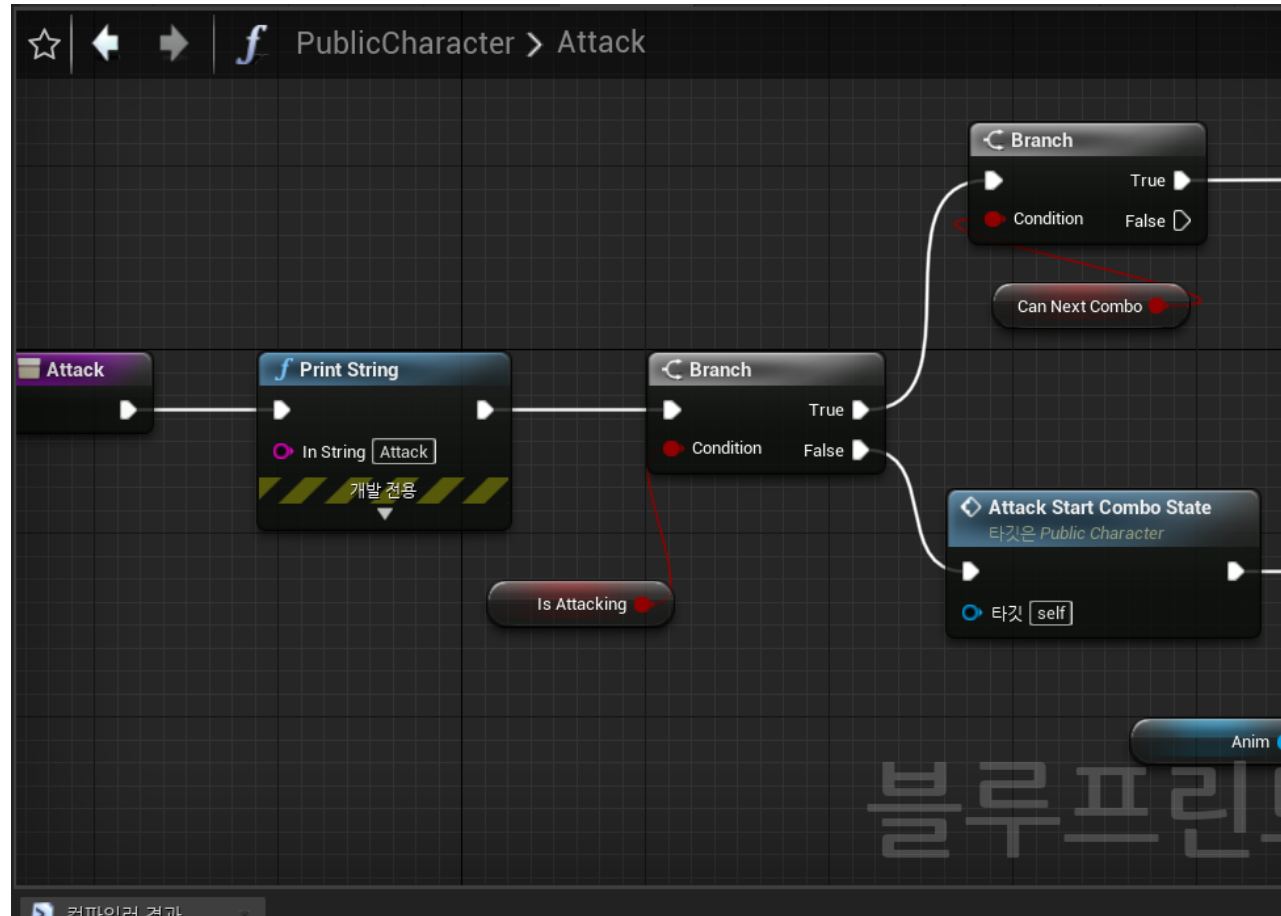
1.8 ComboState에 관한 함수를 생성한다.



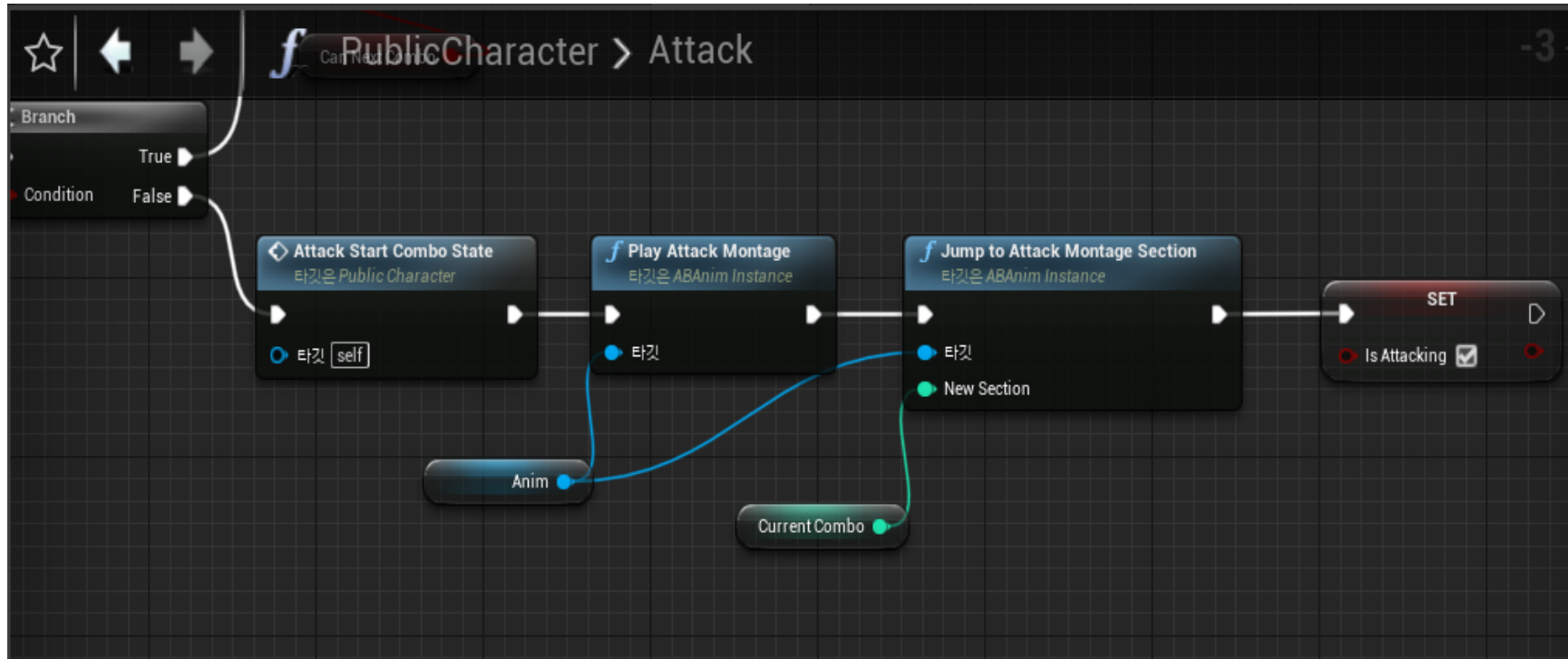
1.9 AttackStartComboState함수 내용



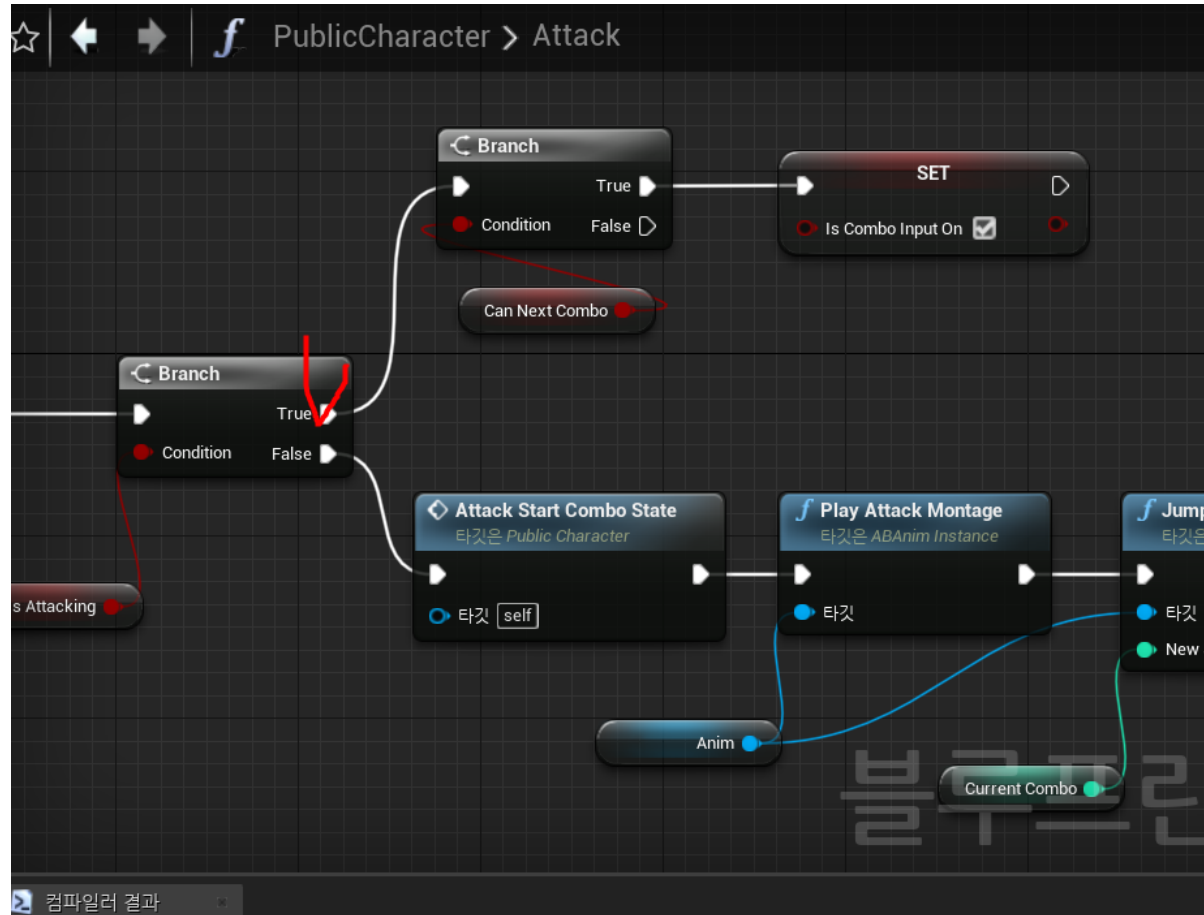
1.10 AttackEndComboState함수 내용



1.11 Attack 함수에 들어가서 IsAttacking의 값이 True인지 False인지 검사를 한다.



1.12 IsAttacking의 값이 False 이면 AttackStartComboState를 호출, Anim의 함수를 호출, IsAttacking 값을 True로 바꾼다.



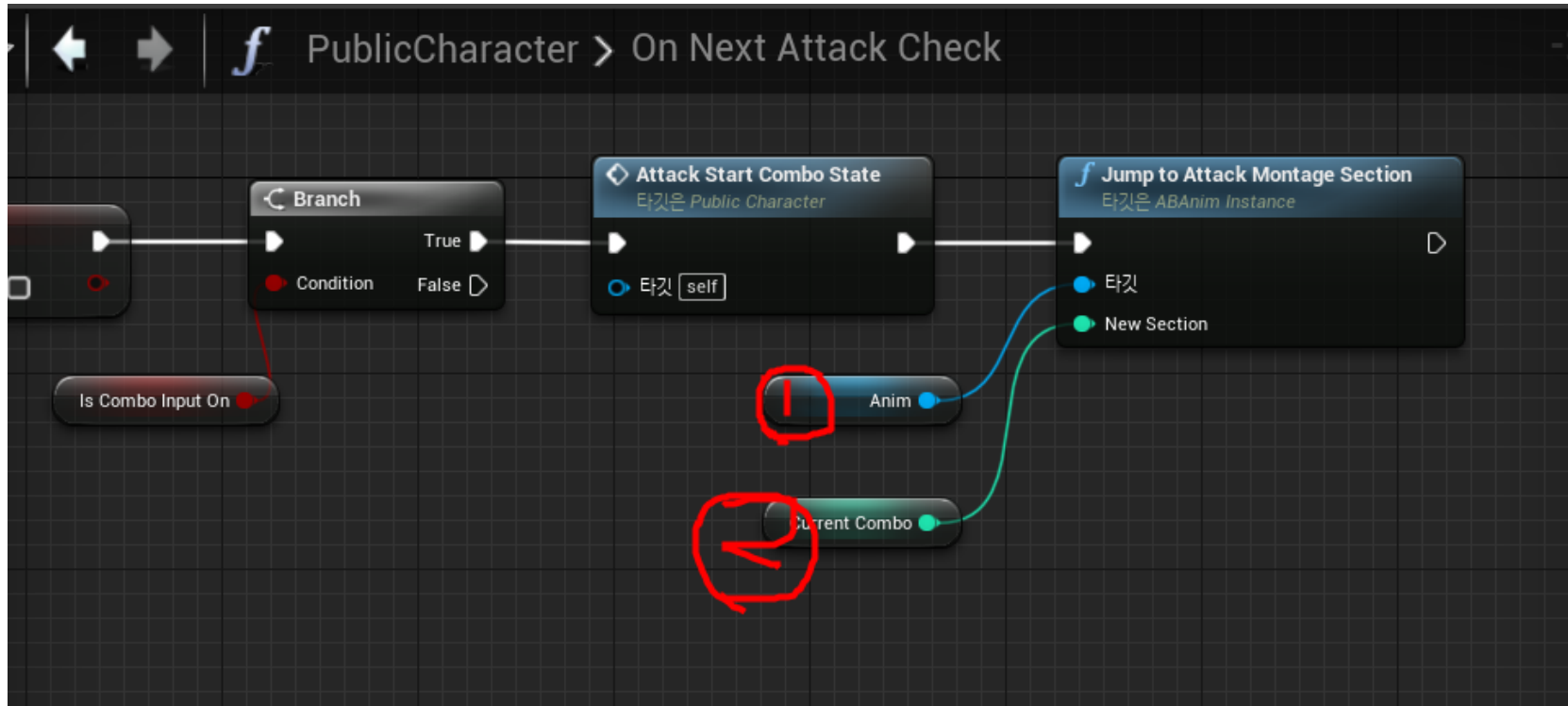
1.13 IsAttacking의 값이 True이면 CanNextCombo의 값을 검사하고 True이면 isComboInputOn을 True로 SET한다.



1.14 OnAttackMontageEnded에 들어가서 IsAttacking을 False로 SET, AttackEndComboState를 호출한다.



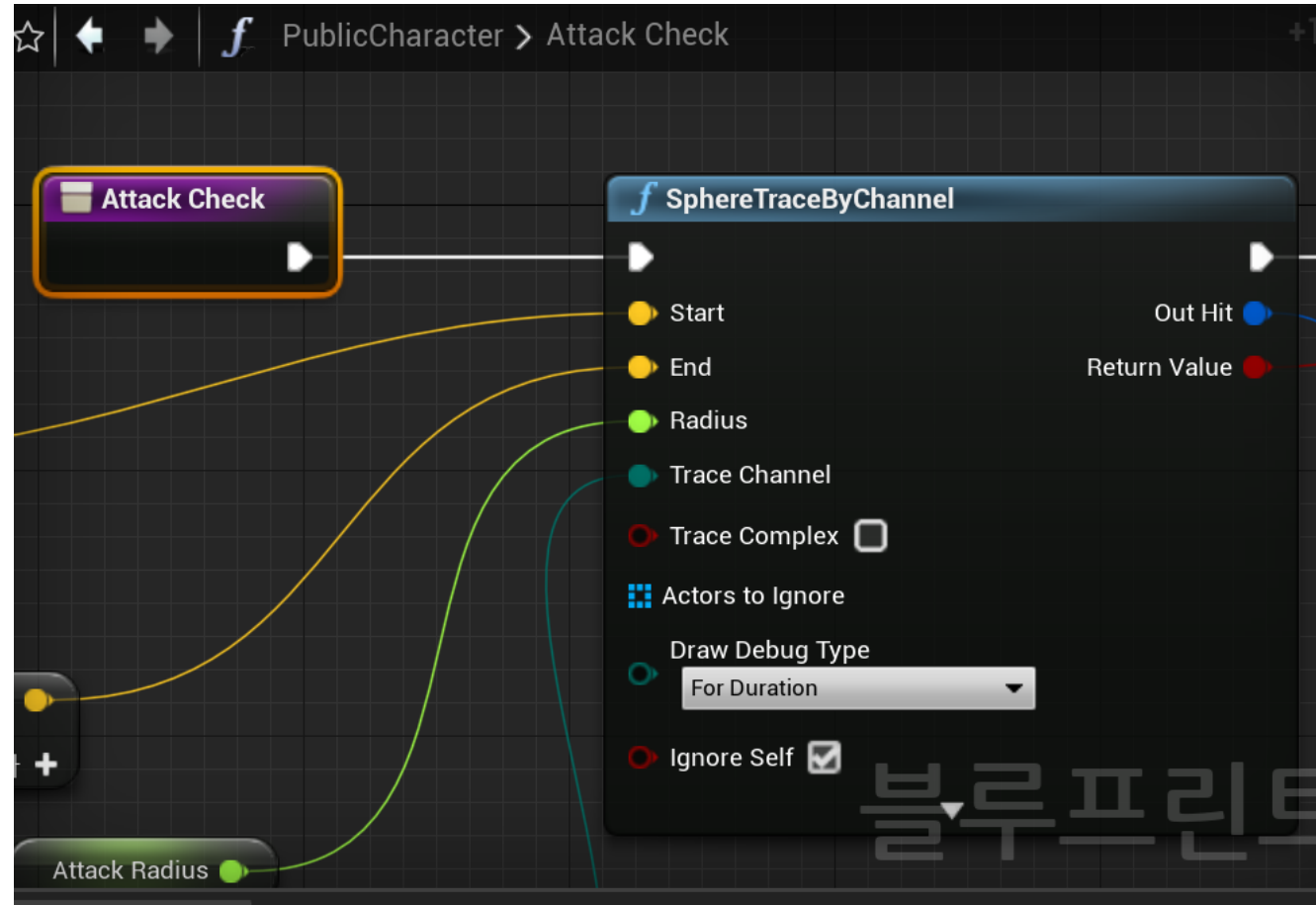
1.15 OnNextAttackCheck 함수에 들어가서 CanNextCombo를 False로 SET, IsComboInputOn을 검사한다.



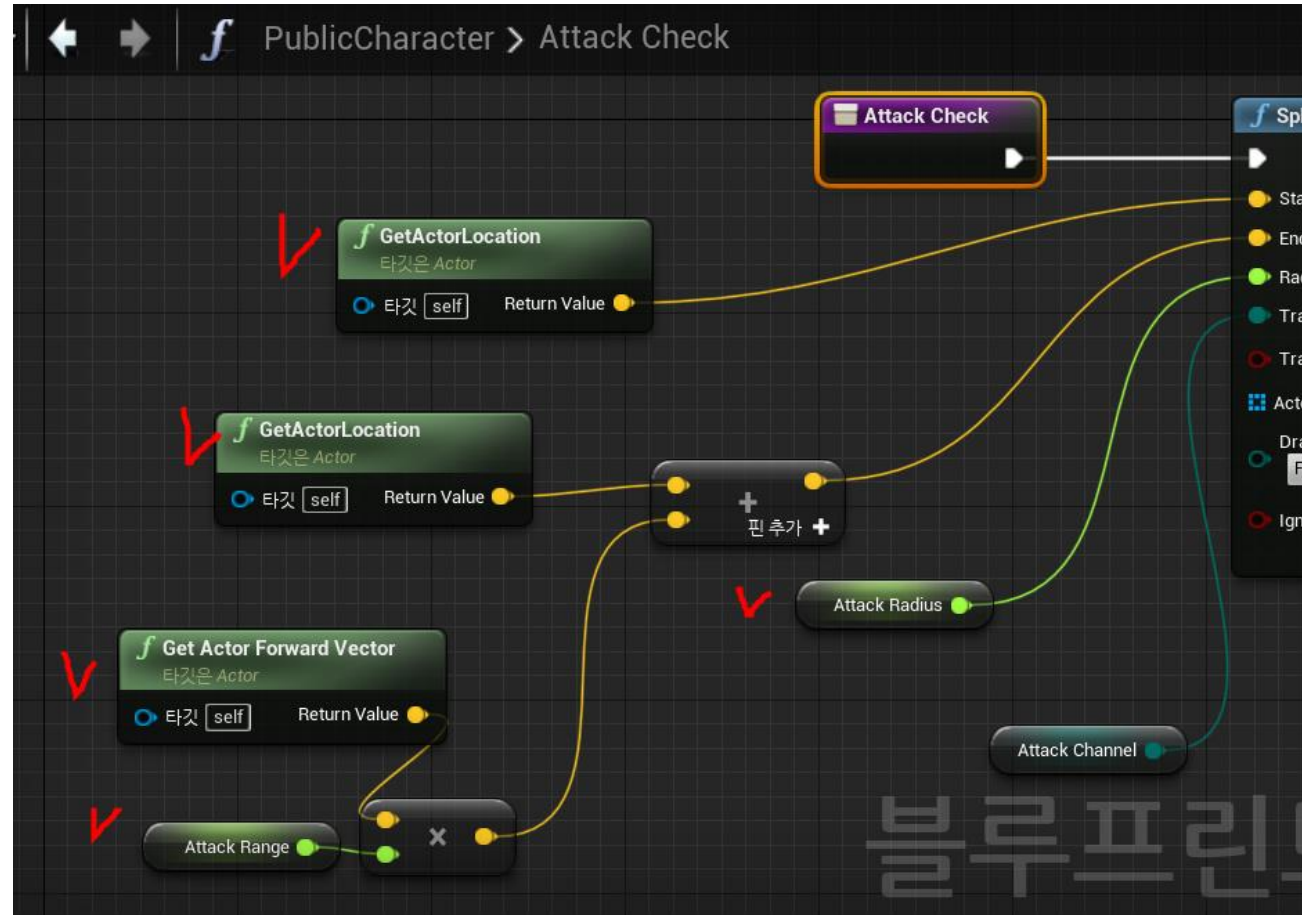
1.16 IsComboInputOn이 True이면 AttackStartComboState를 호출하고, Anim의 JumpToAttackMontageSection 함수를 호출한다.



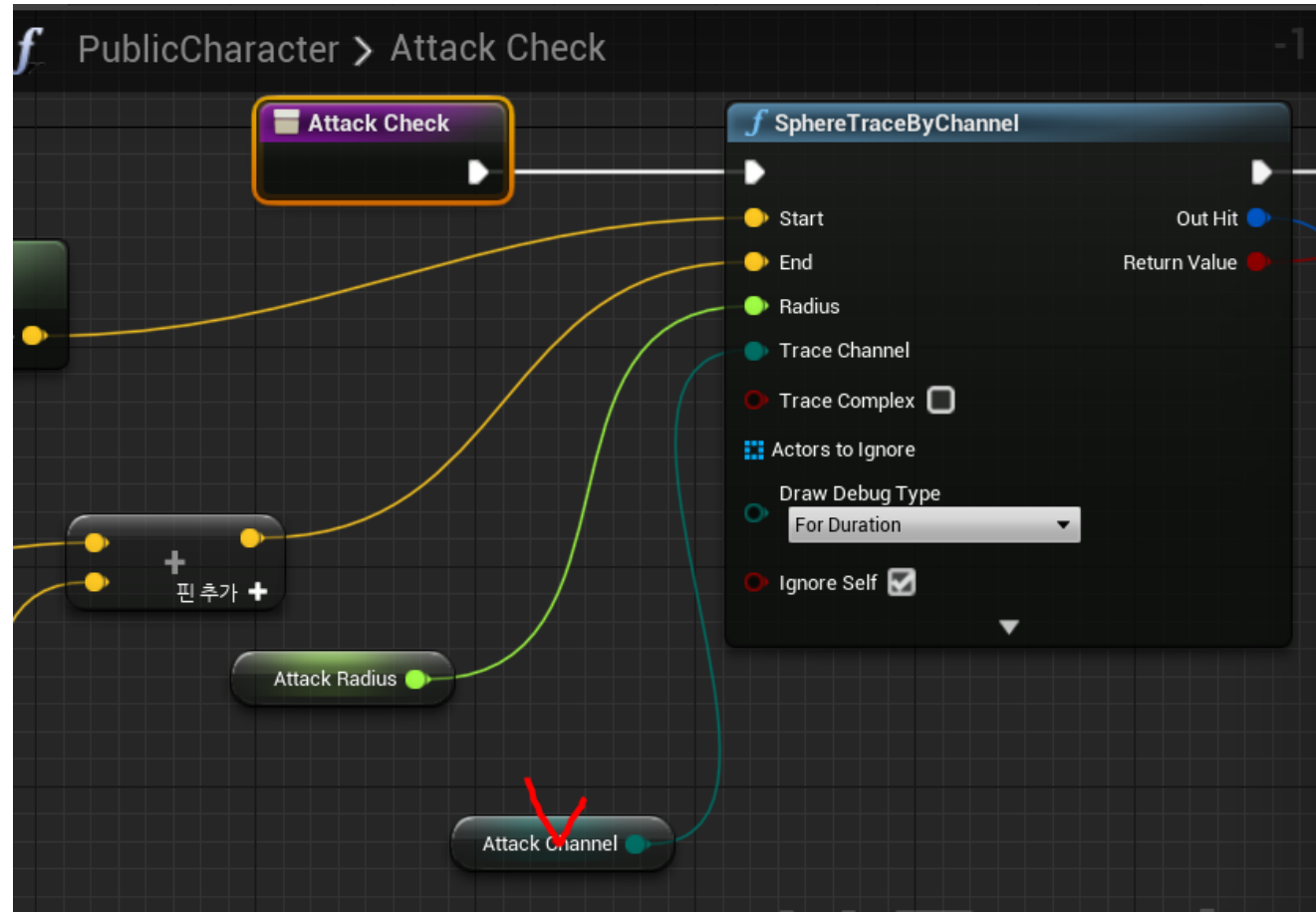
1.17 변수 AttackRange, AttackRadius를 생성한다. 각각 공격 범위와 공격의 반지름을 구하는 데 사용



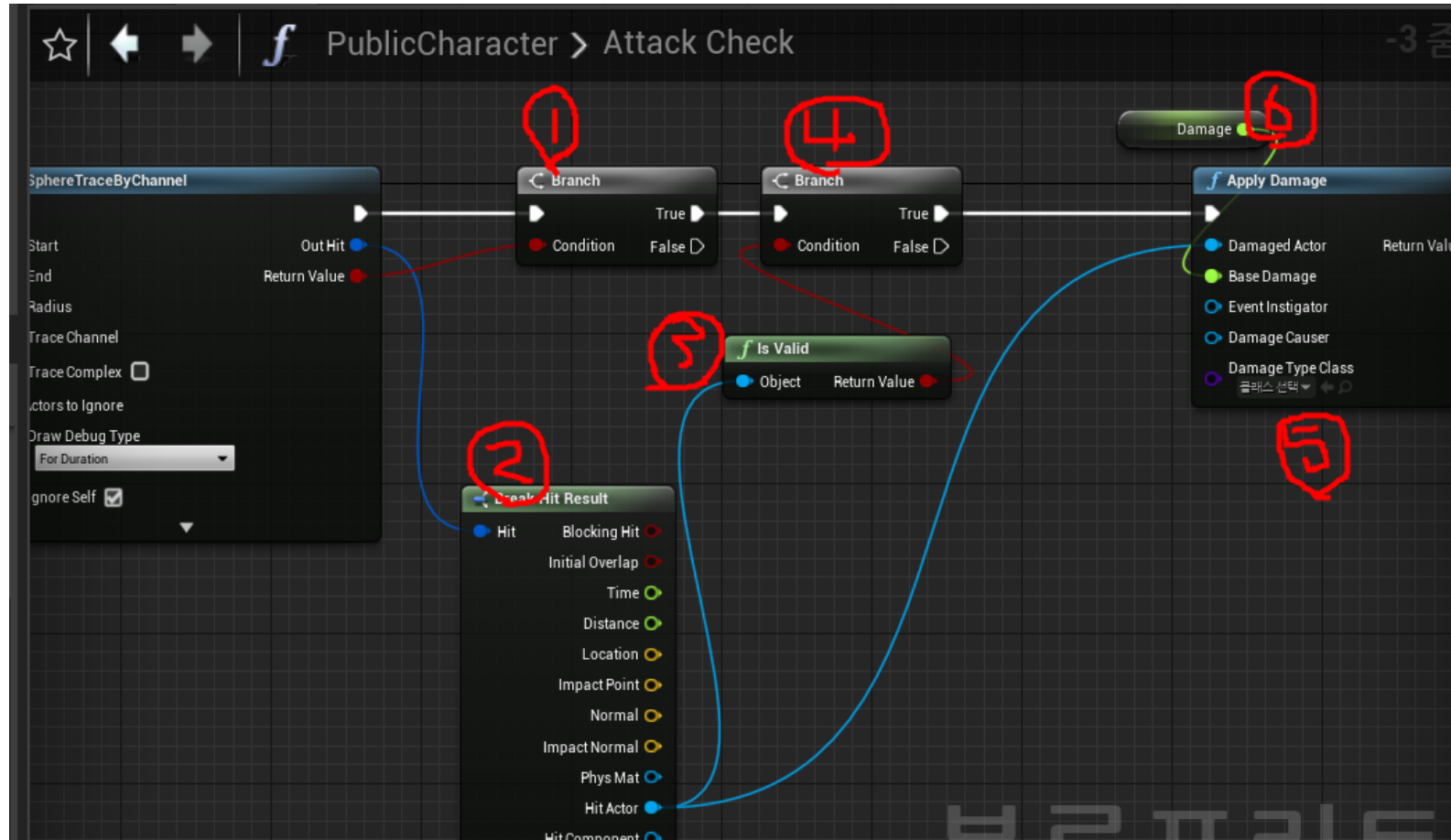
1.18 AttackCheck 함수에 들어가서 SphereTraceByCheannel을 호출한다.



1.19 Start, End, Radius 핀에 연결한다.



1.20 TraceChannel을 변수로 승격해서 AttackChannel를 생성한다.

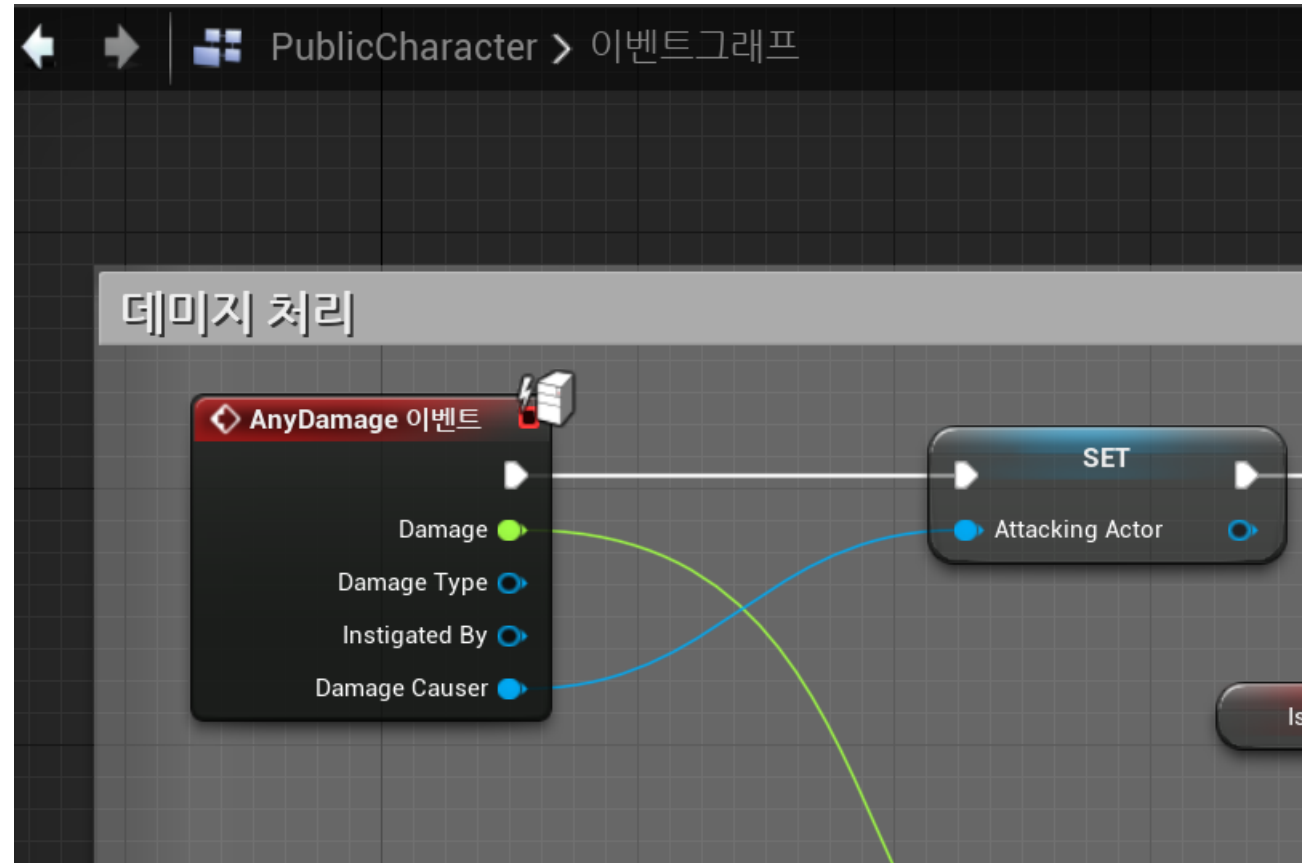


1.21 OurHit을 Break해서 HitActor를 ApplyDamge 함수에 연결하고, float 형 Damage 변수를 생성해서 연결한다.

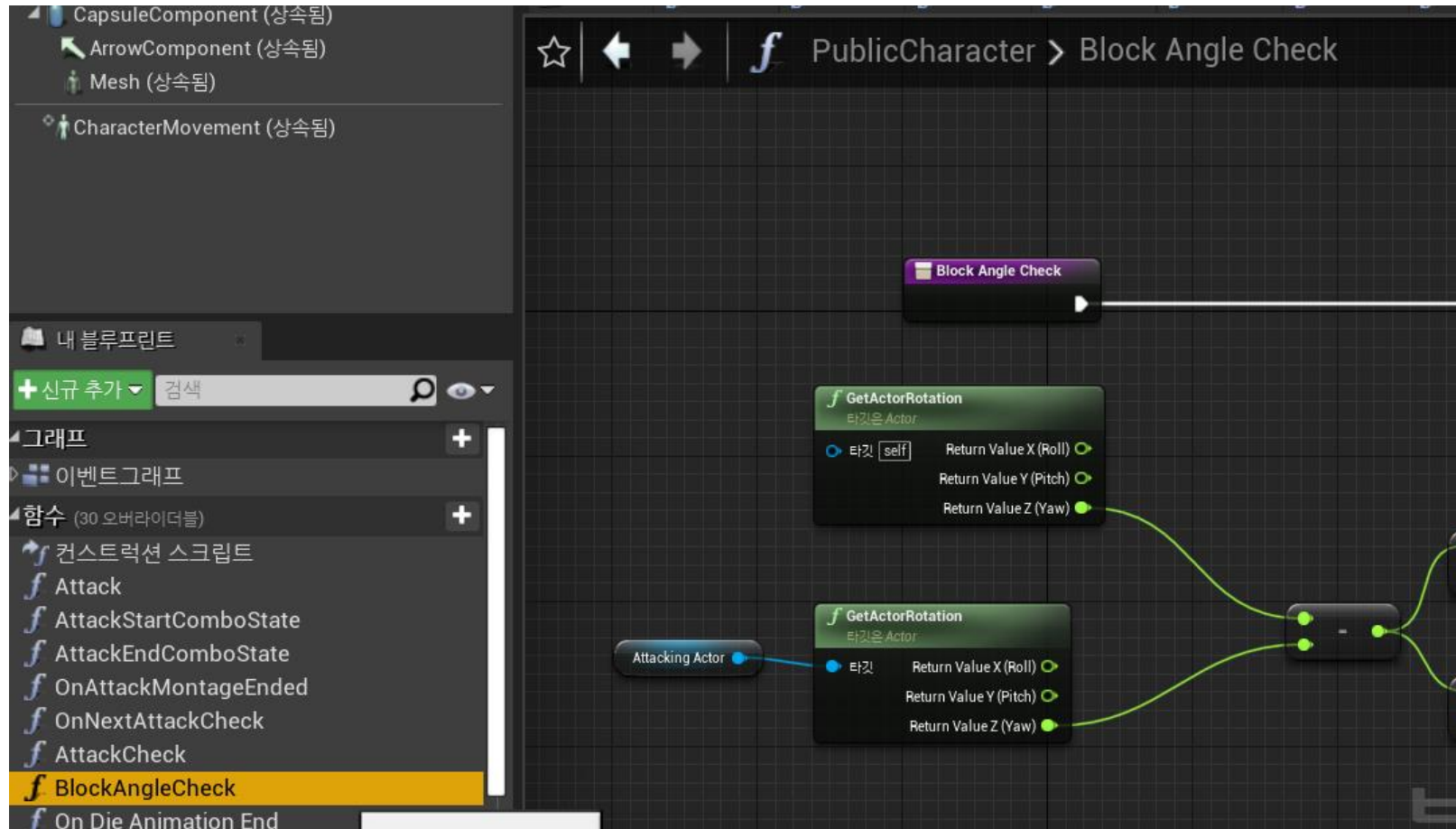
Variable	
변수 이름	HP
변수 유형	플로트
인스턴스 편집가능	☐
블루프린트 읽기 전용	☐
툴팁	
스폰시 노출	☐
프라이빗	☐
시네마틱에 노출	☐
카테고리	Default
슬라이더 범위	
값 범위	
리플리케이션	None
리플리케이션 조건	None

기본값	
HP	0.0

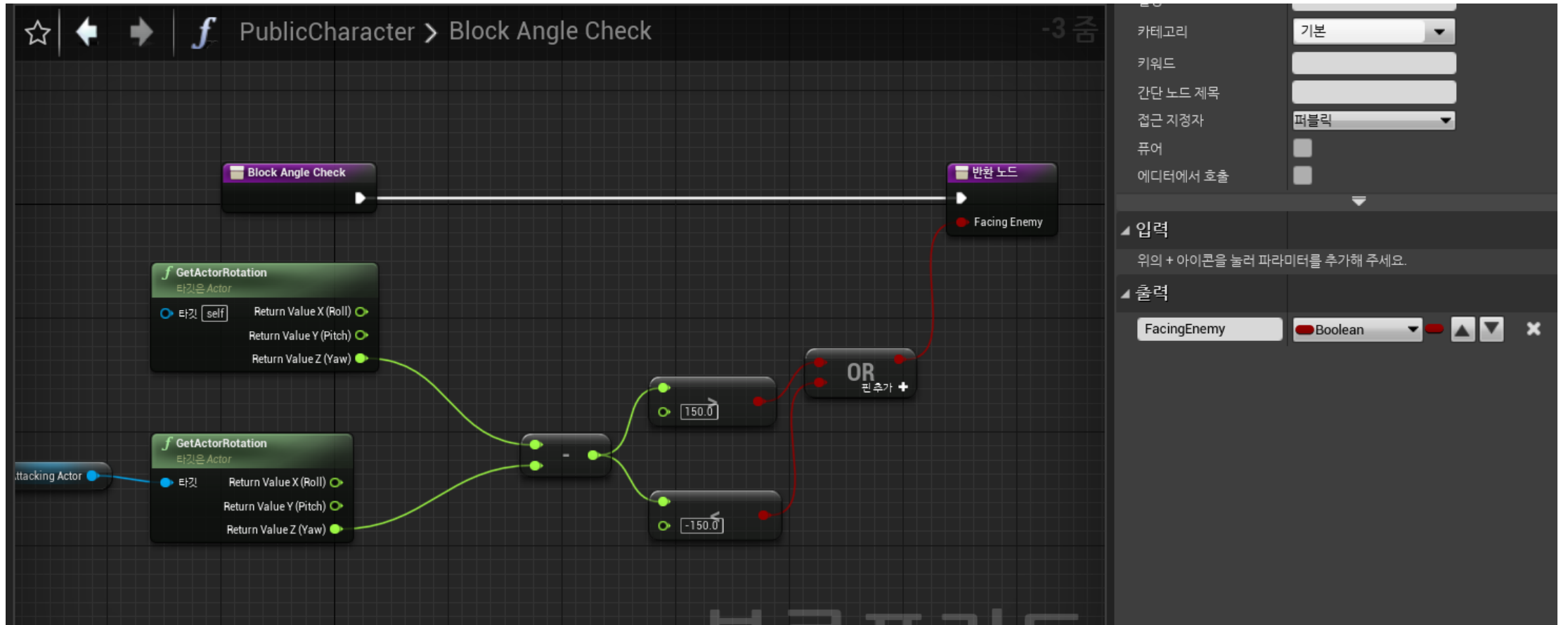
1.22 Float 형 변수 HP를 생성



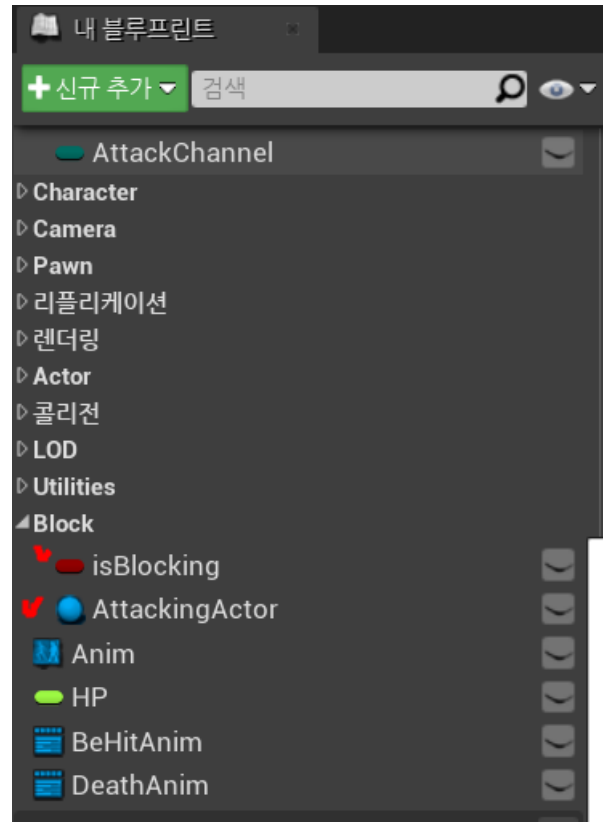
1.23 AnyDamage 이벤트의 DamageCauser를 변수로 승격해서 AttackingActor를 생성한다.



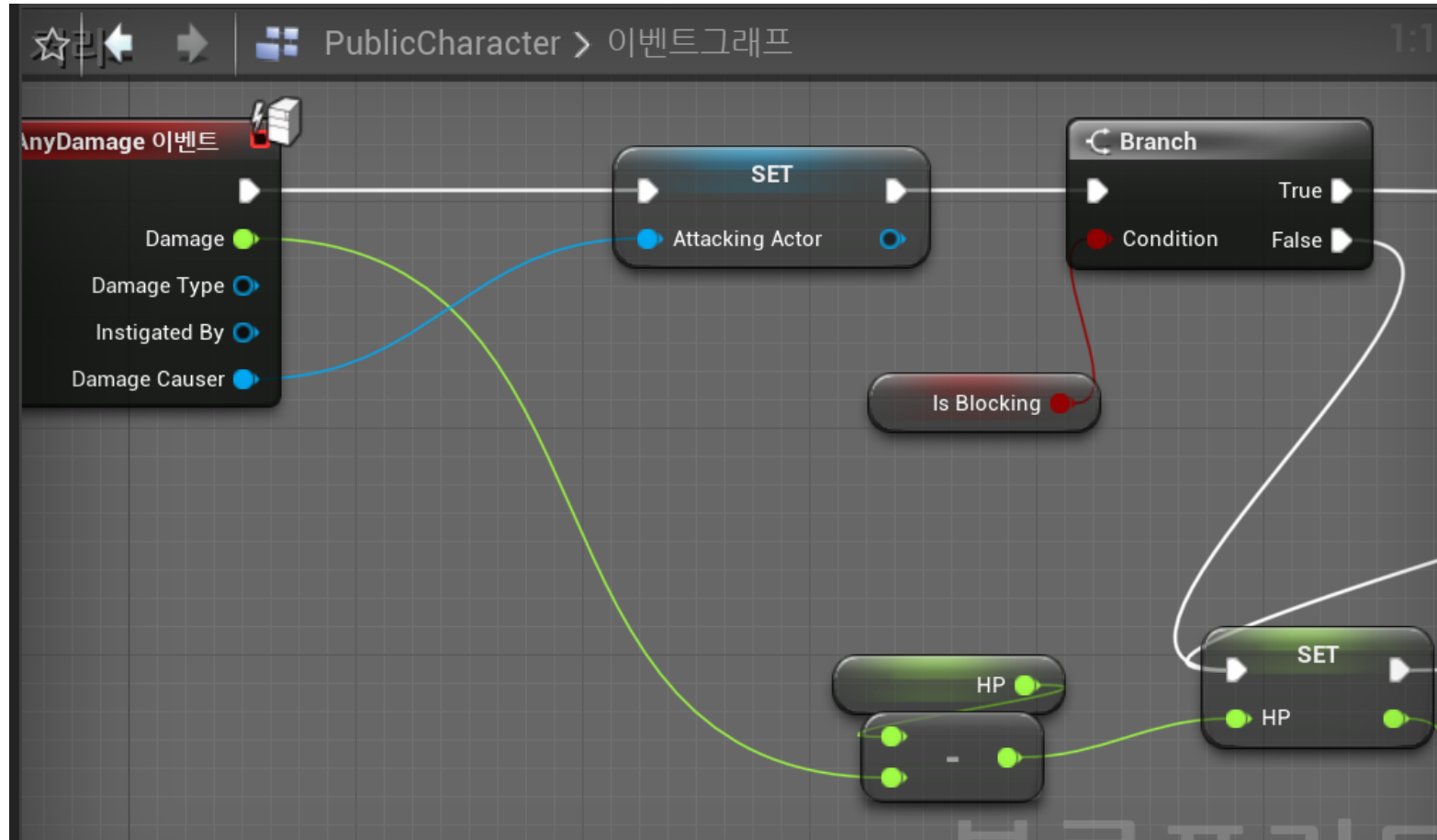
1.24 BlockAngleCheck 함수를 추가한다.



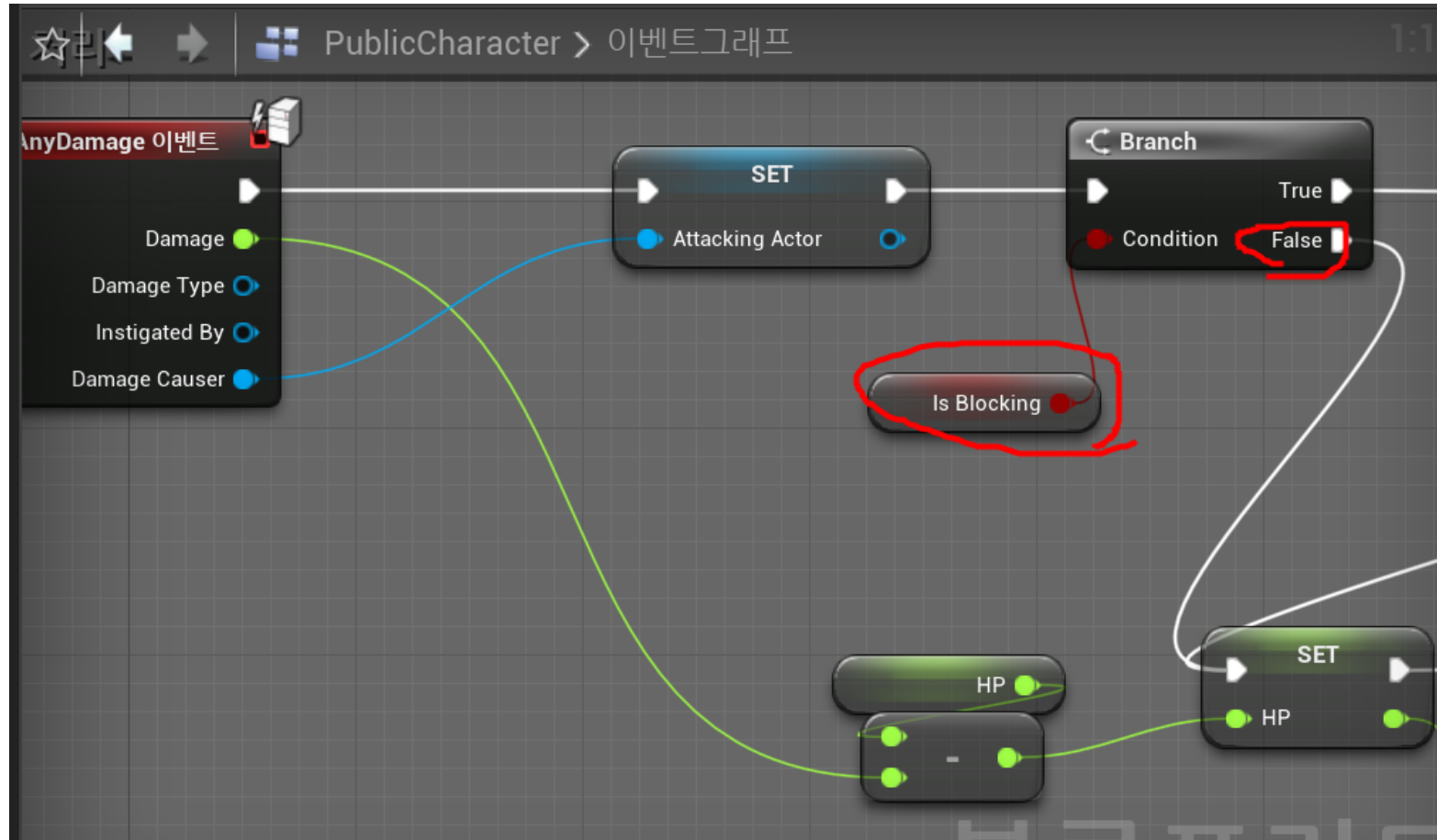
1.25 AttackingActor의 회전값과 자신의 회전값의 차이로 앞인지 뒤인지 확인을 하는 함수



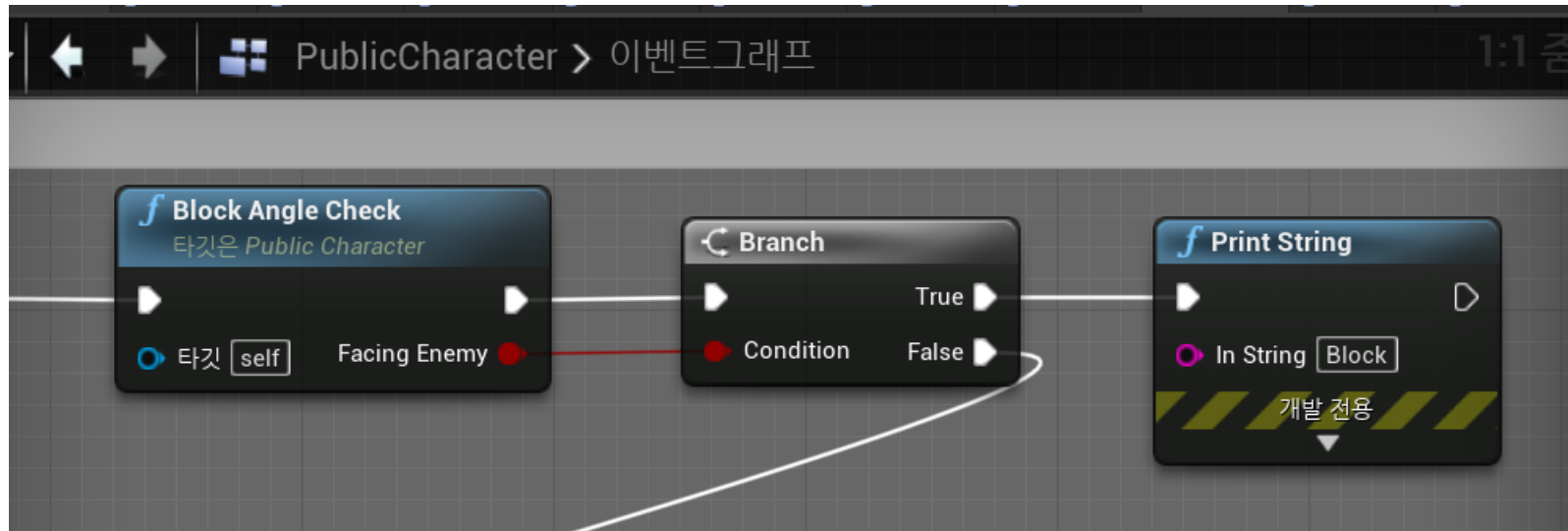
1.26 Block 카테고리를 만들고, IsBlocking boolean 변수를 추가



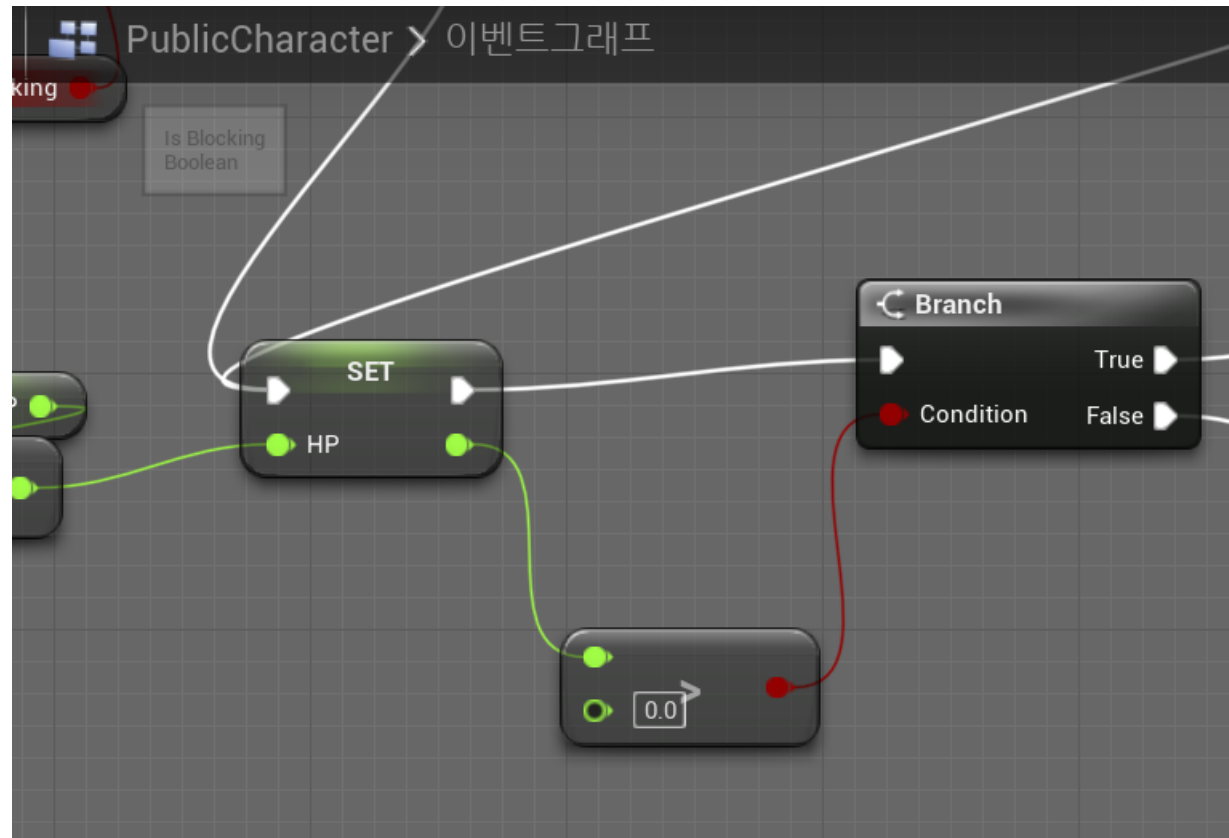
1.27 HP와 Damage 값을 빼고 그 값을 HP에 SET한다.



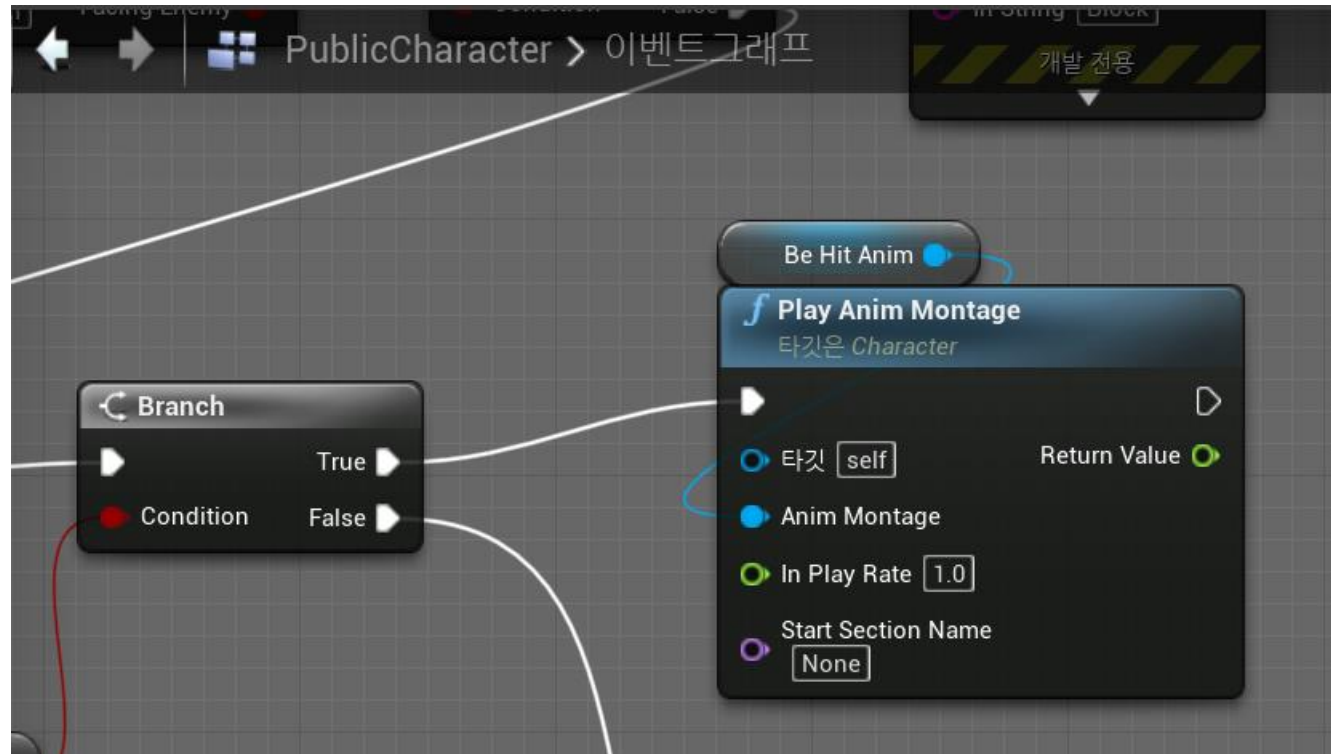
1.28 IsBlocing 변수를 검사해서 False 이면 데미지 처리를 한다.



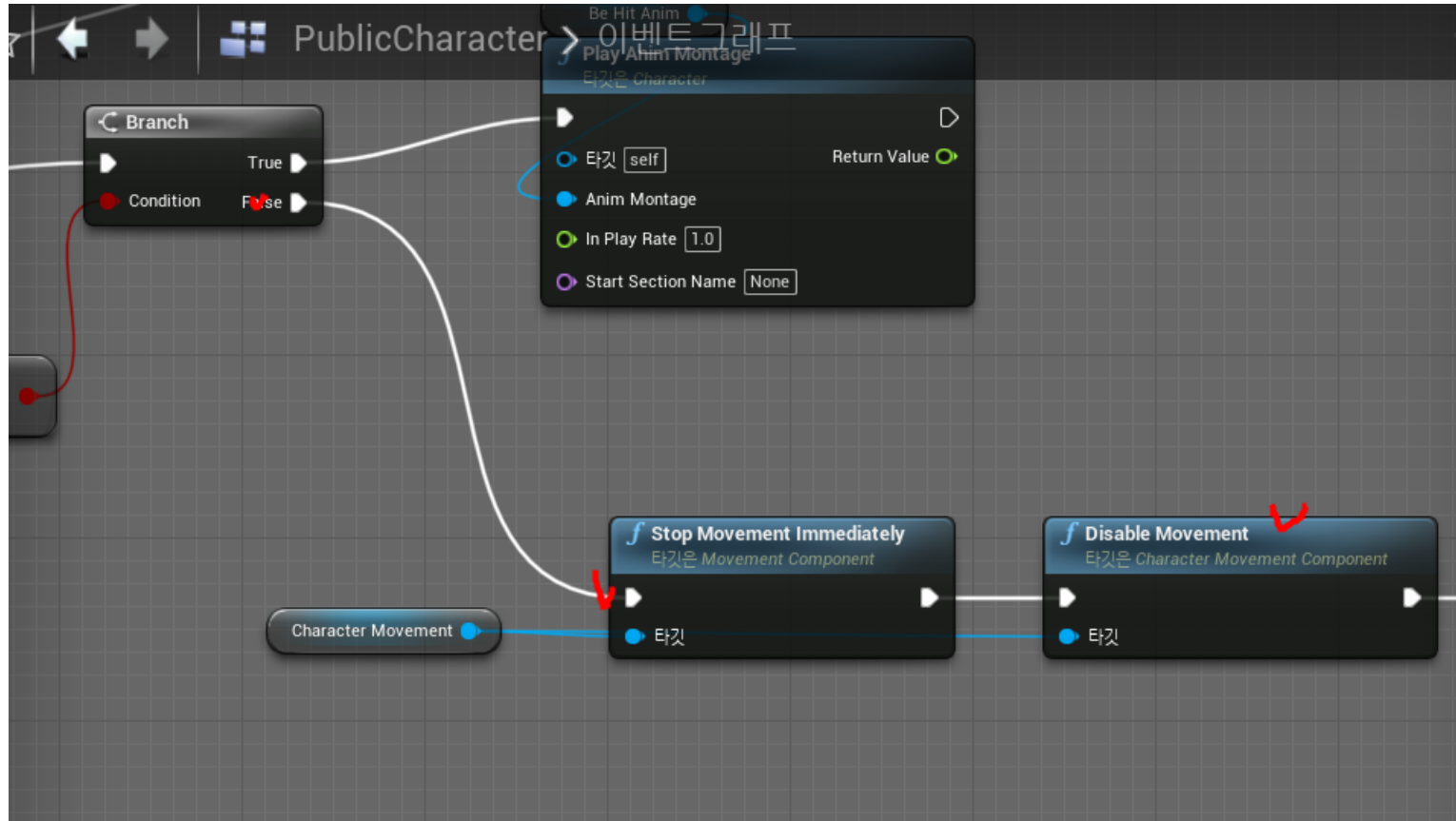
1.30 BlockAngleCheck 함수의 리턴값이 True이면 Block이라는 로그를 쓴다.



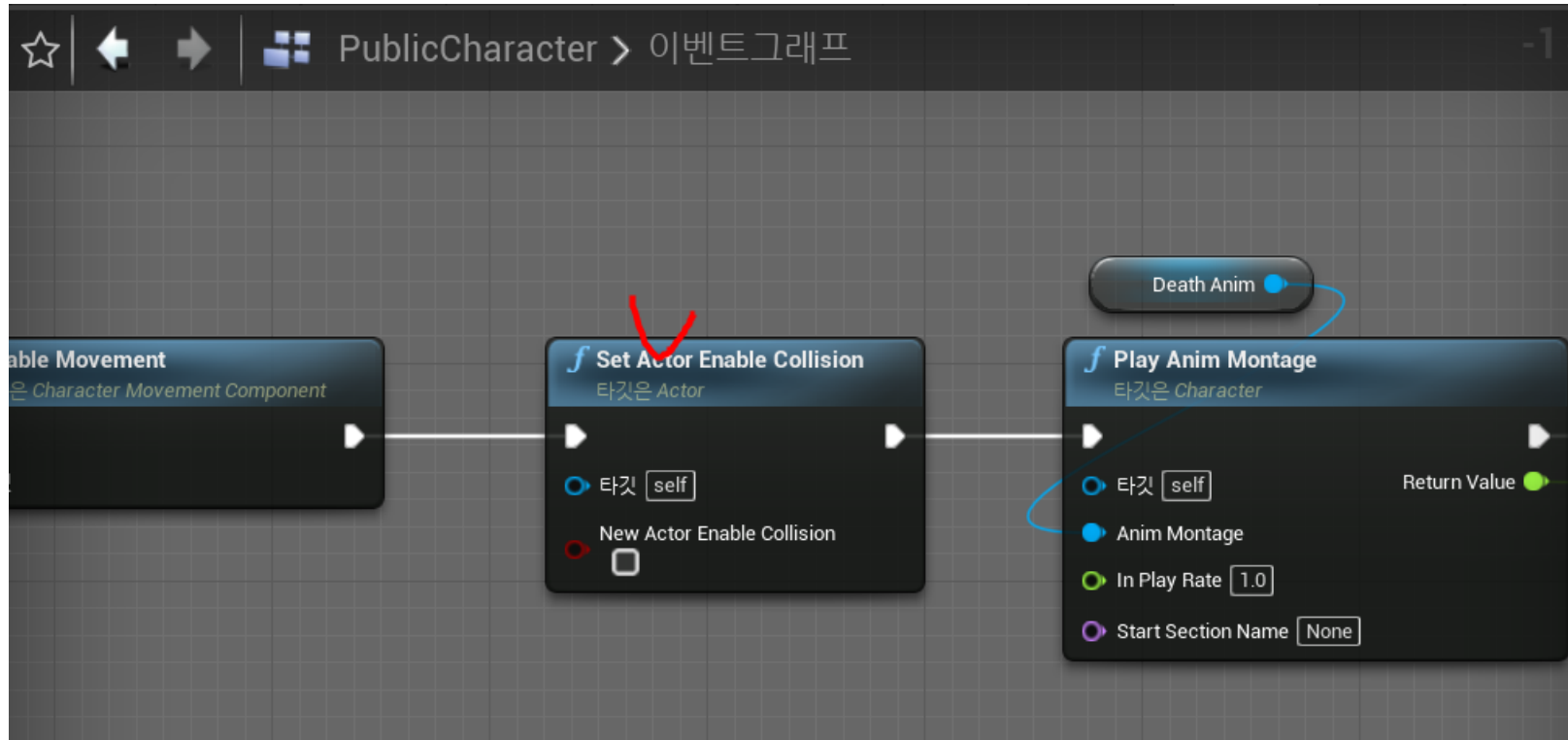
1.31 데미지를 처리해서 HP값이 0보다 큰지 검사한다.



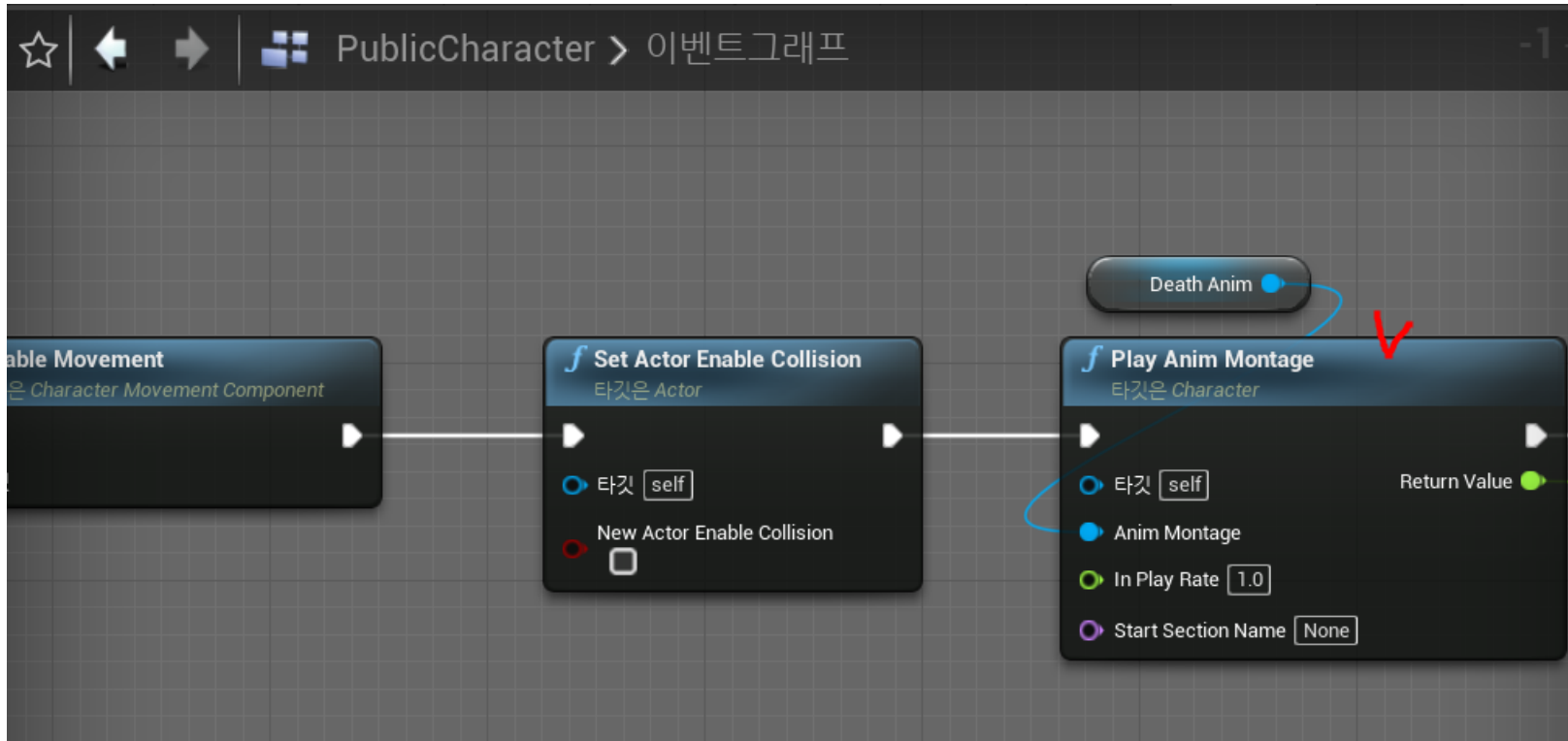
1.32 HP가 0보다 크면 PlayAnimMontage를 호출하고, AnimMontage를 변수로 승격해서 BeHitAnim을 생성한다.



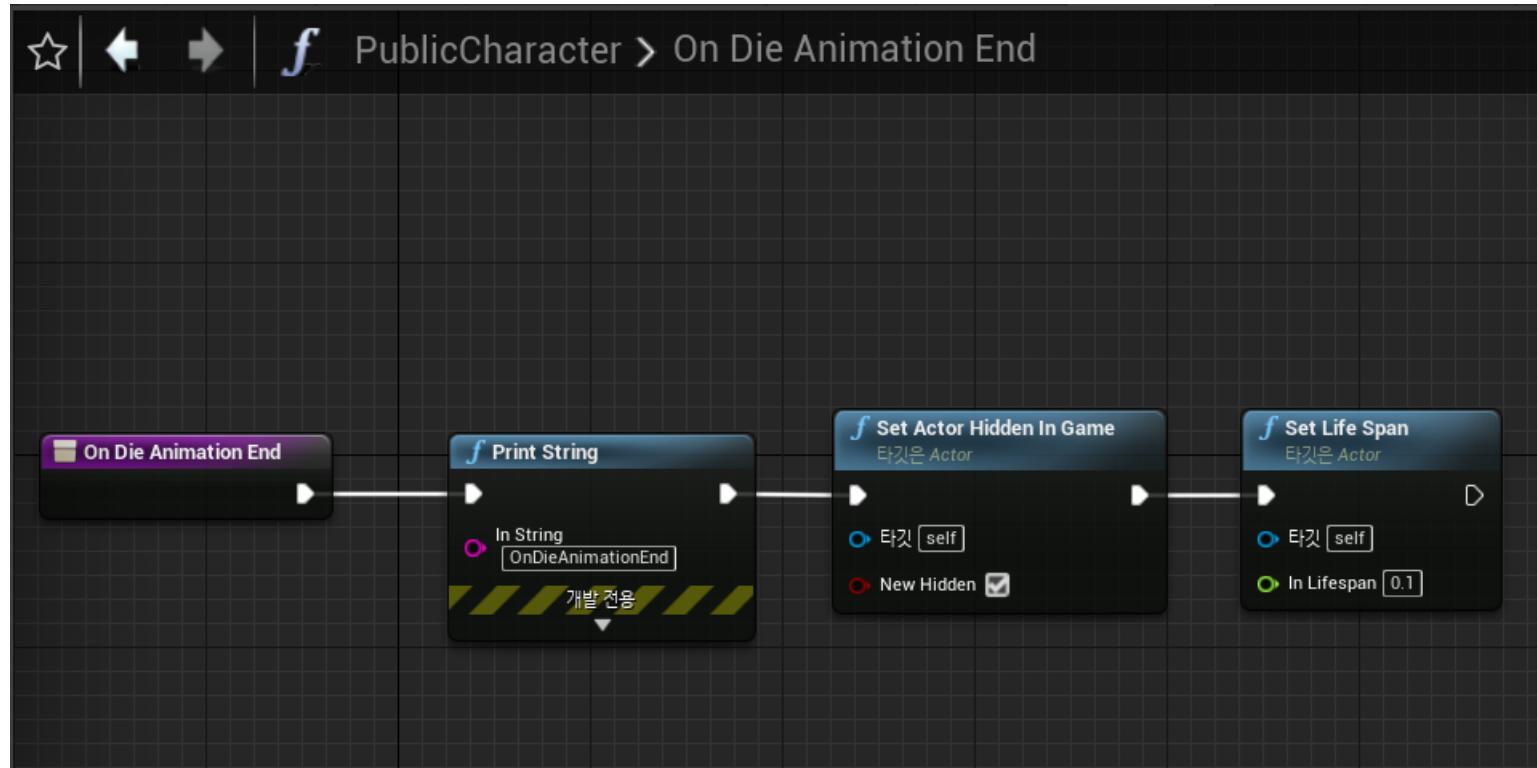
1.33 HP가 0보다 작으면 캐릭터의 움직임을 멈추게 한다.



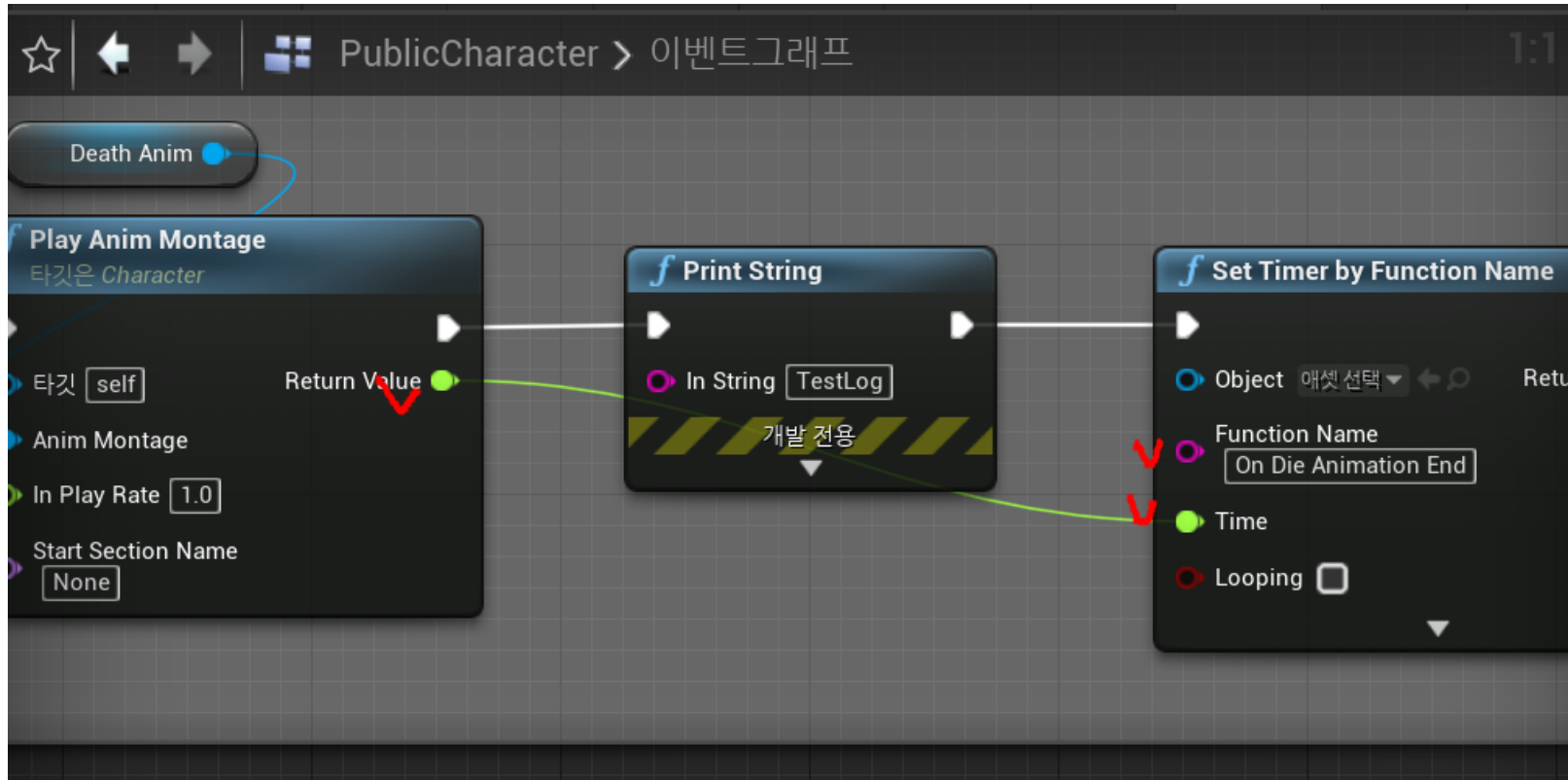
1.34 SetActorEnableCollision을 호출해서 False로 만든다.



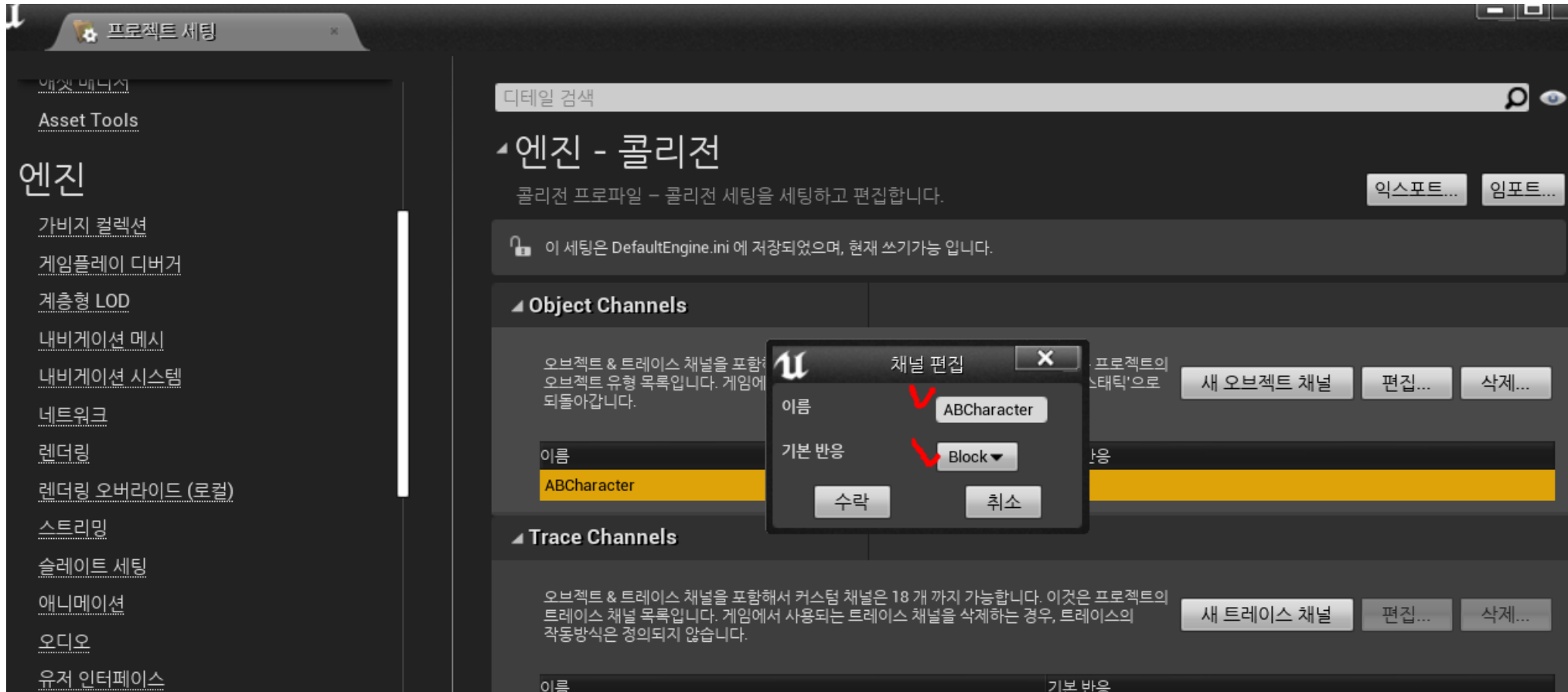
1.35 PlayAnimMontage를 호출하고 AnimMontage를 변수로 승격해서 DeathAnim을 생성한다.



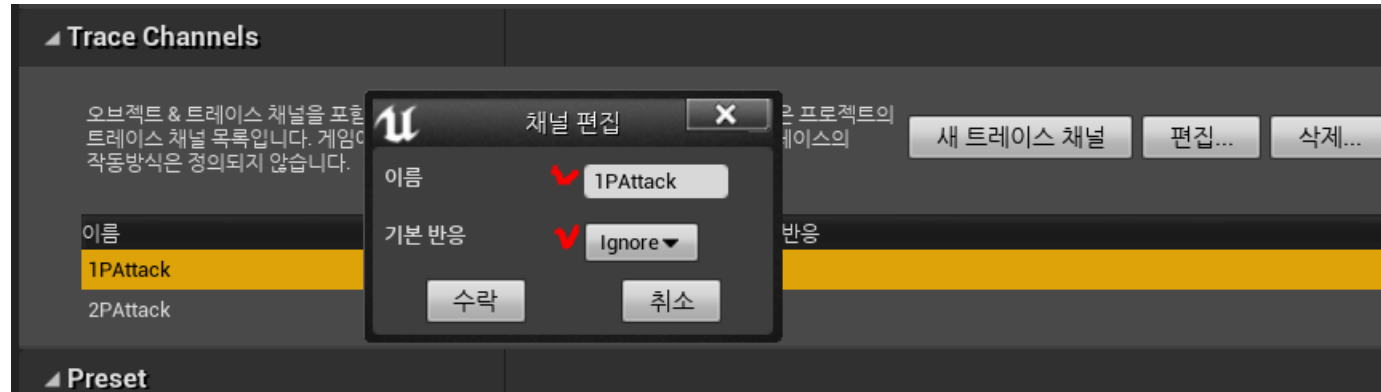
1.36 OnDieAnimationEnd 함수를 추가하고 시간이 지나면 사라지게 하는 기능을 넣는다.



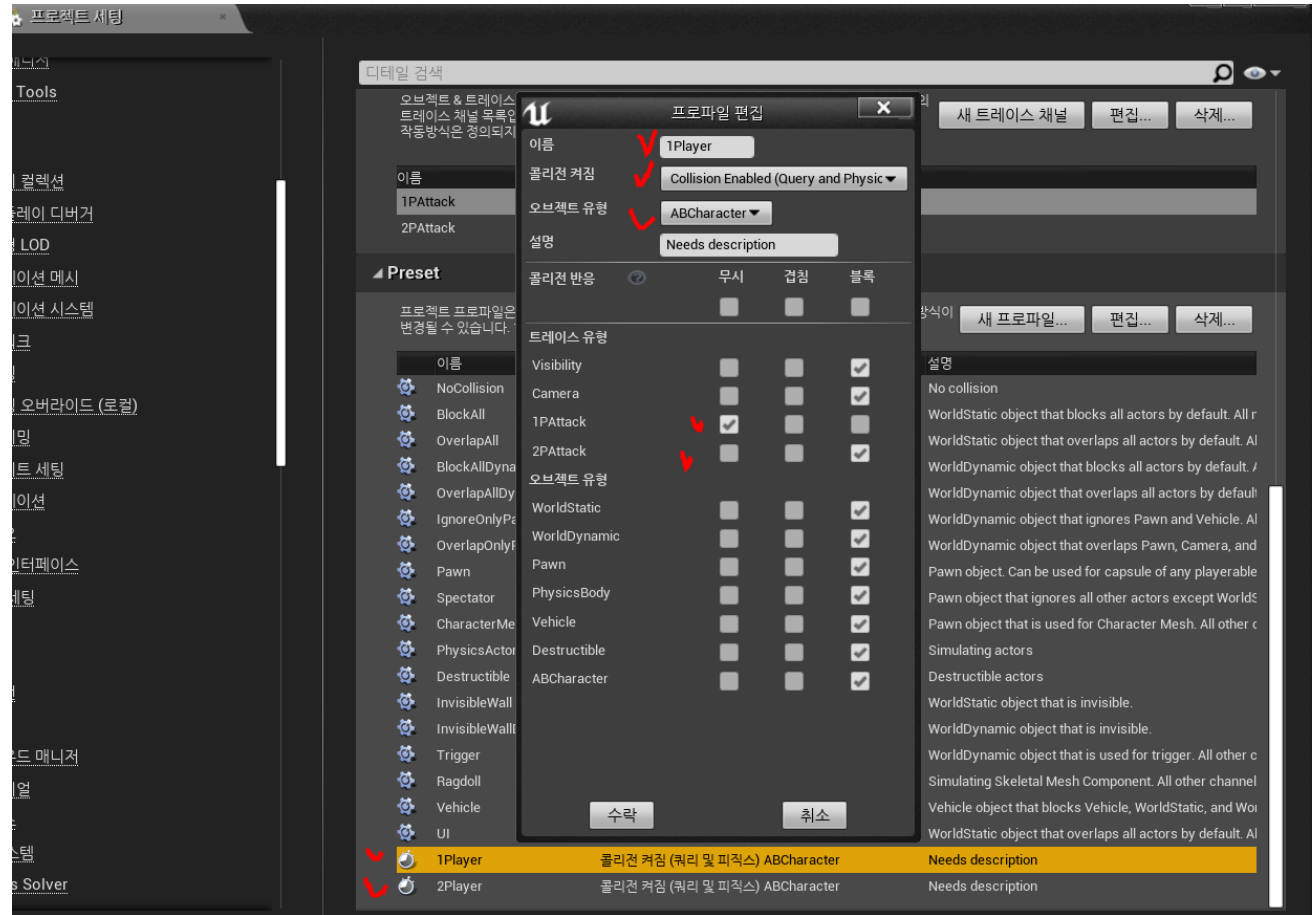
1.37 SetTimerbyFuctionName을 호출하고, FuctionName에 OnDieAnimationEnd를 넣고 Time에는 PlayAnimMontage의 결과값을 연결한다.



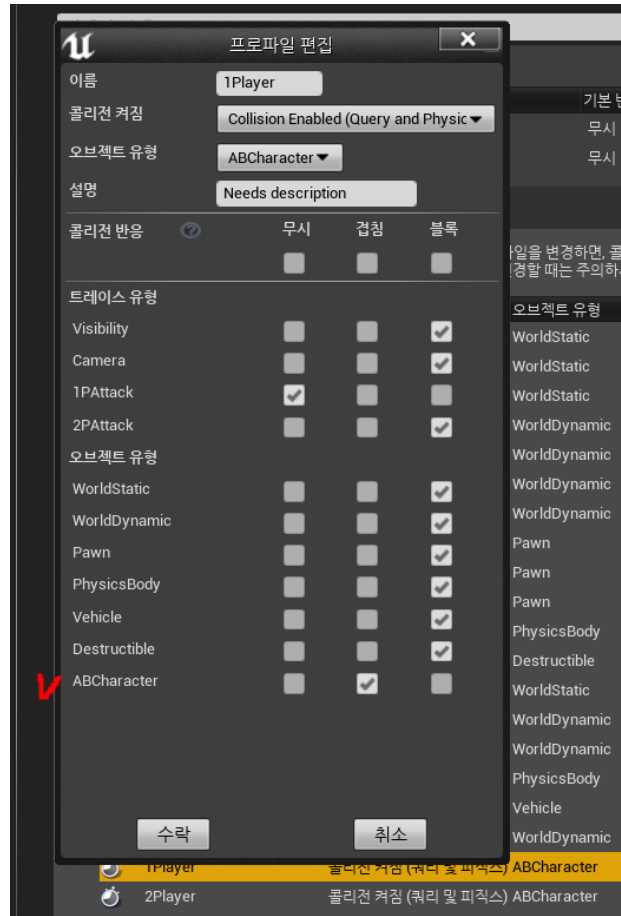
2. 프로젝트 세팅의 콜리전에 들어가서 ObjectChannels에 들어가서 오브젝트 채널을 생성한다.



2.1 1PAttack, 2PAttack 트레이스 채널을 각 각 추가한다.



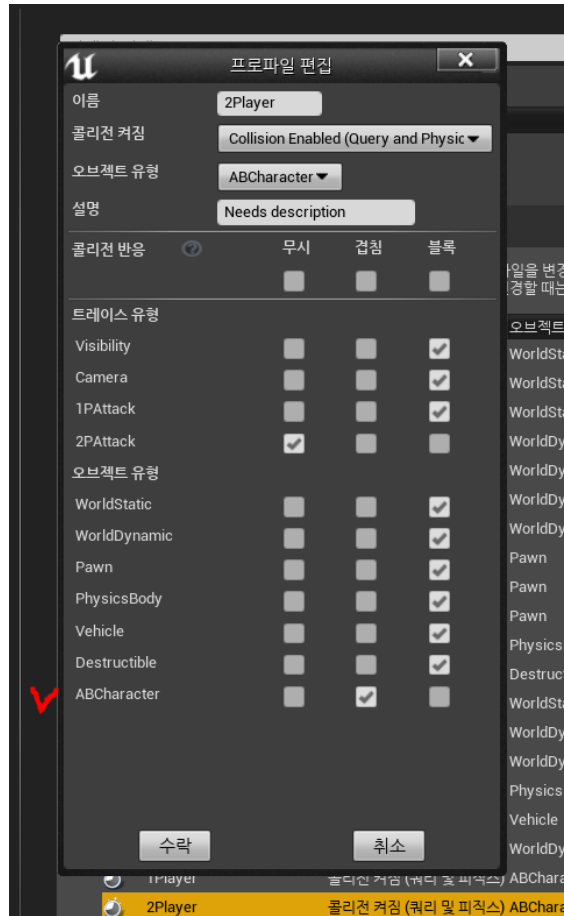
2.2 프리셋에 새 프로파일 1Player를 생성한다.



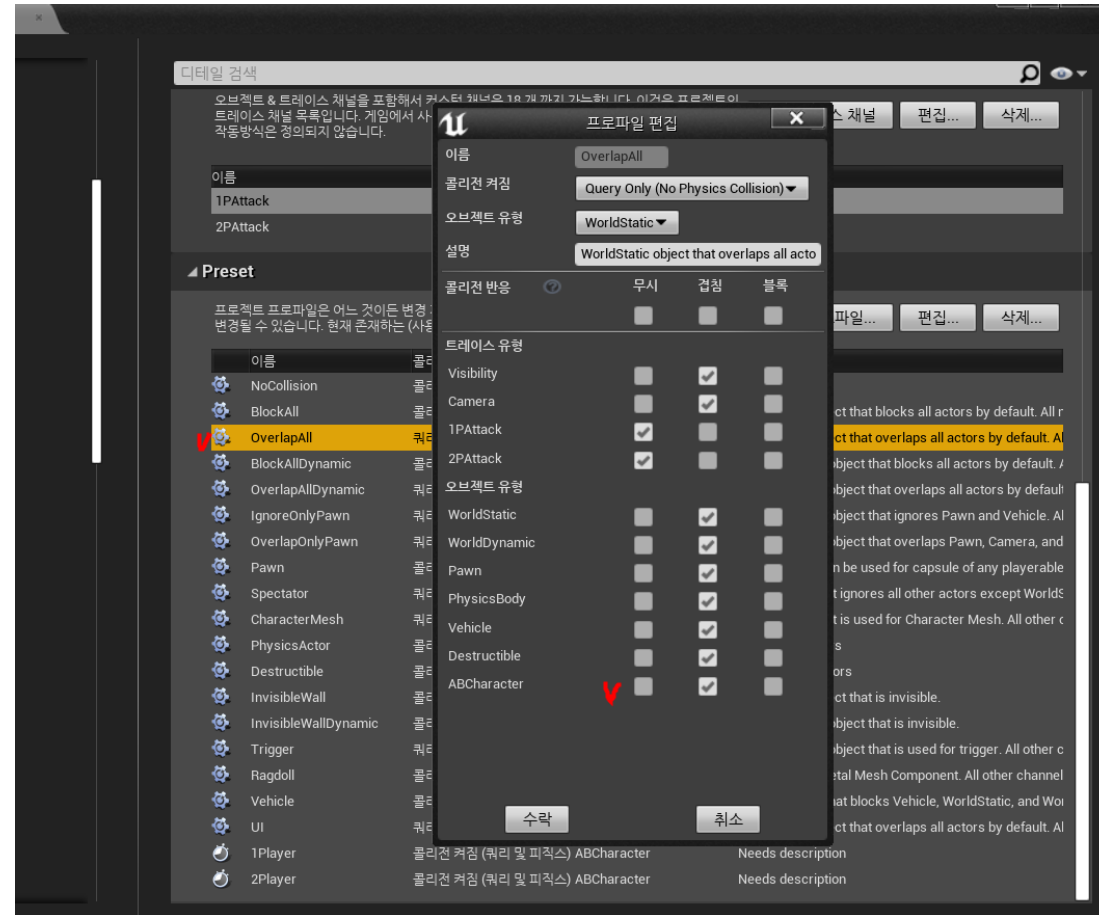
2.2.1 ABCharacter에 대해서 Overlap으로 처리한다.



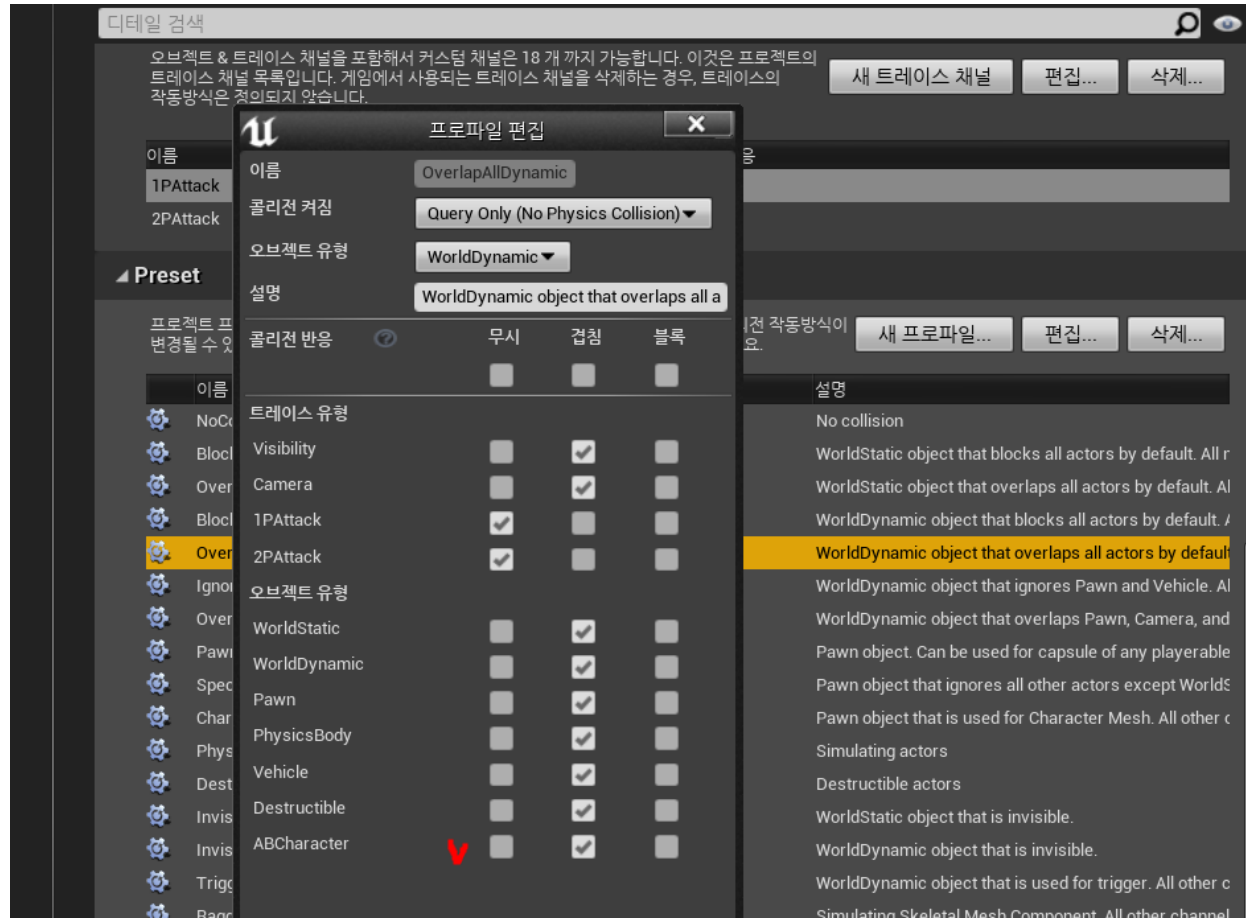
2.3 새 프로파일 2Player를 생성한다.



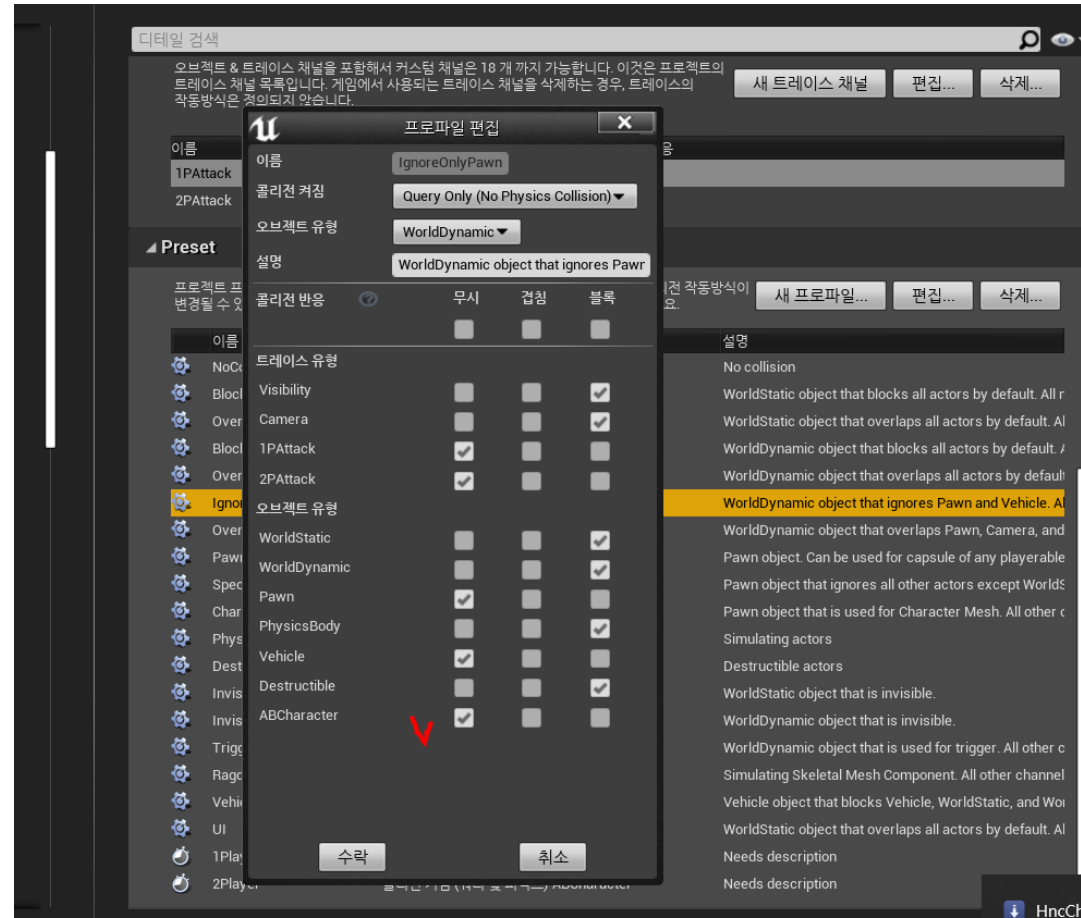
2.3.1 ABCharacter에 대해서 Overlap으로 처리한다.



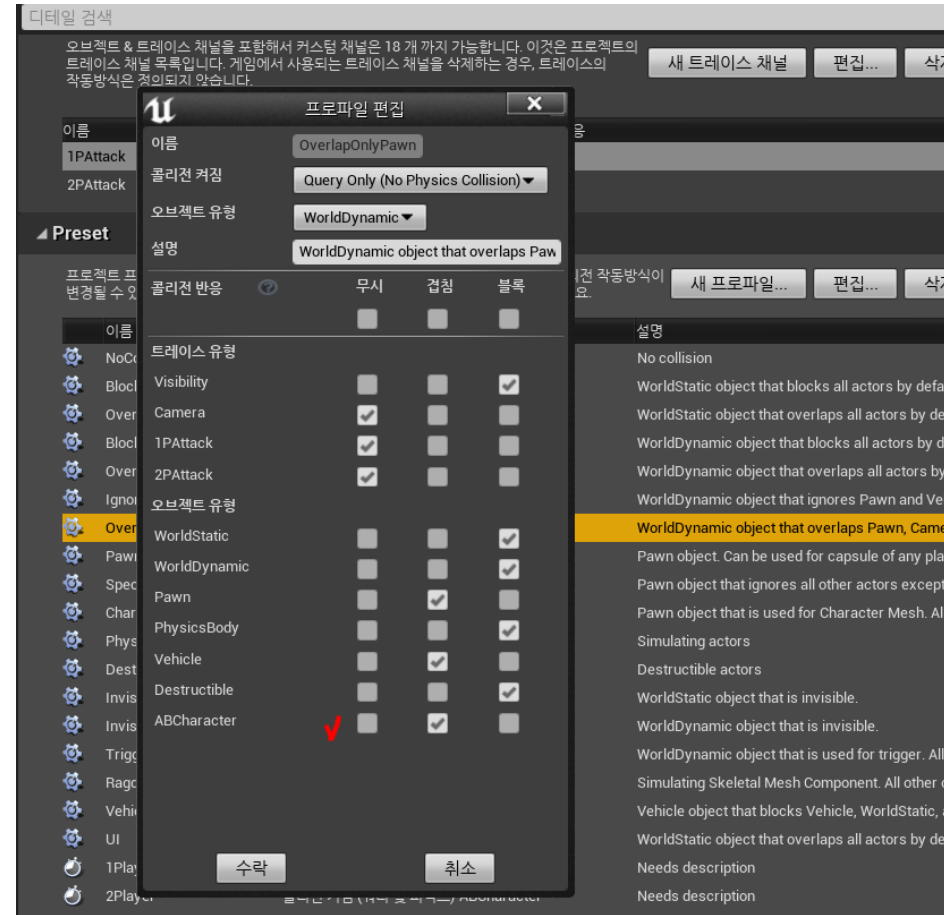
2.4 OverlapAll 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



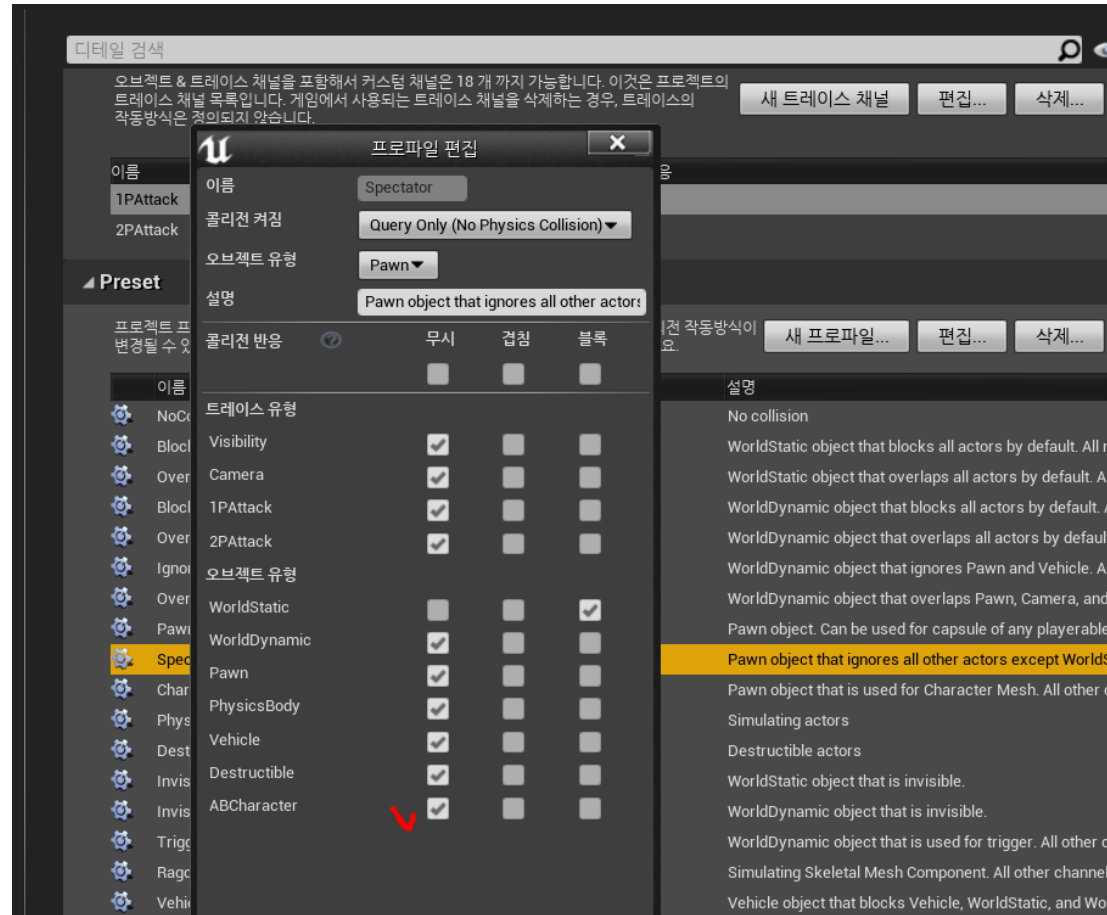
2.5 OverlapAllDynamic 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



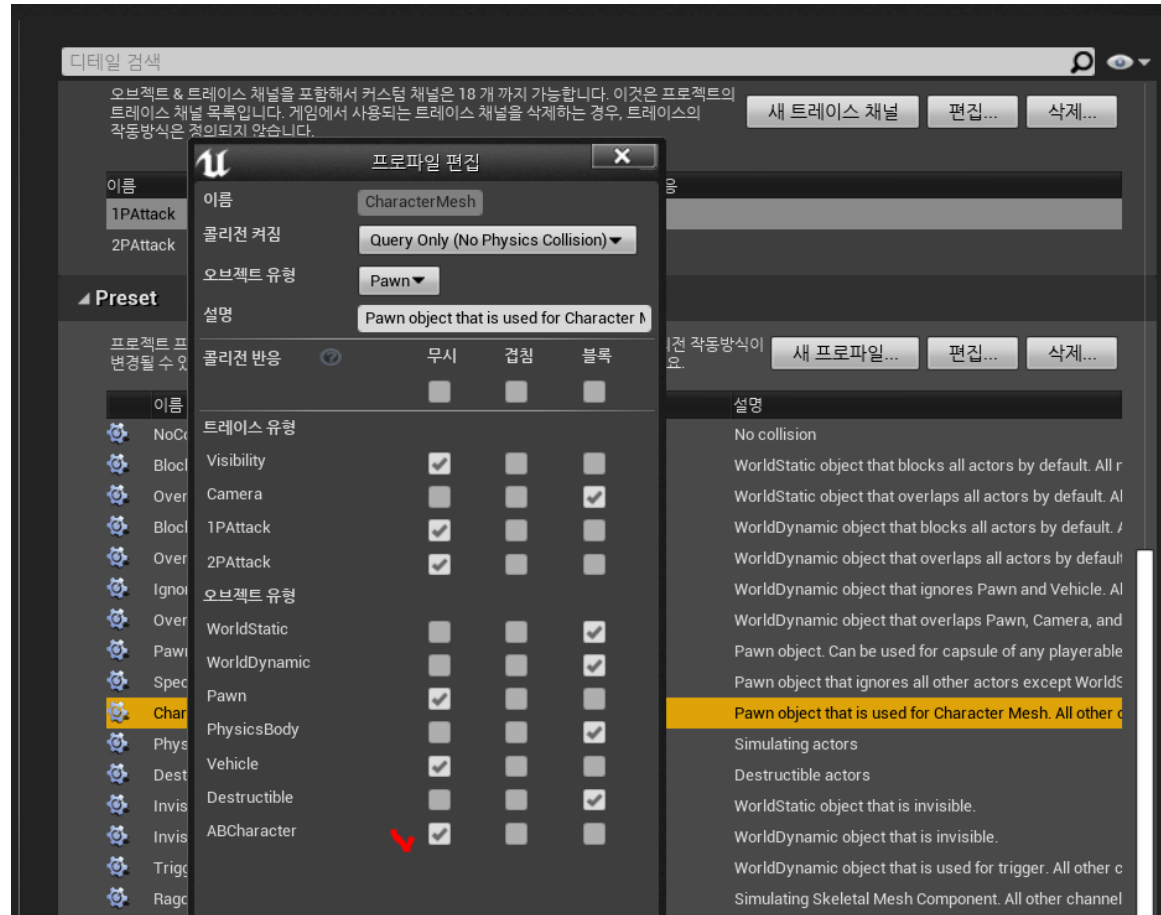
2.6 IgnoreOnlyPawn 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



2.7 OverlapOnlyPawn 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



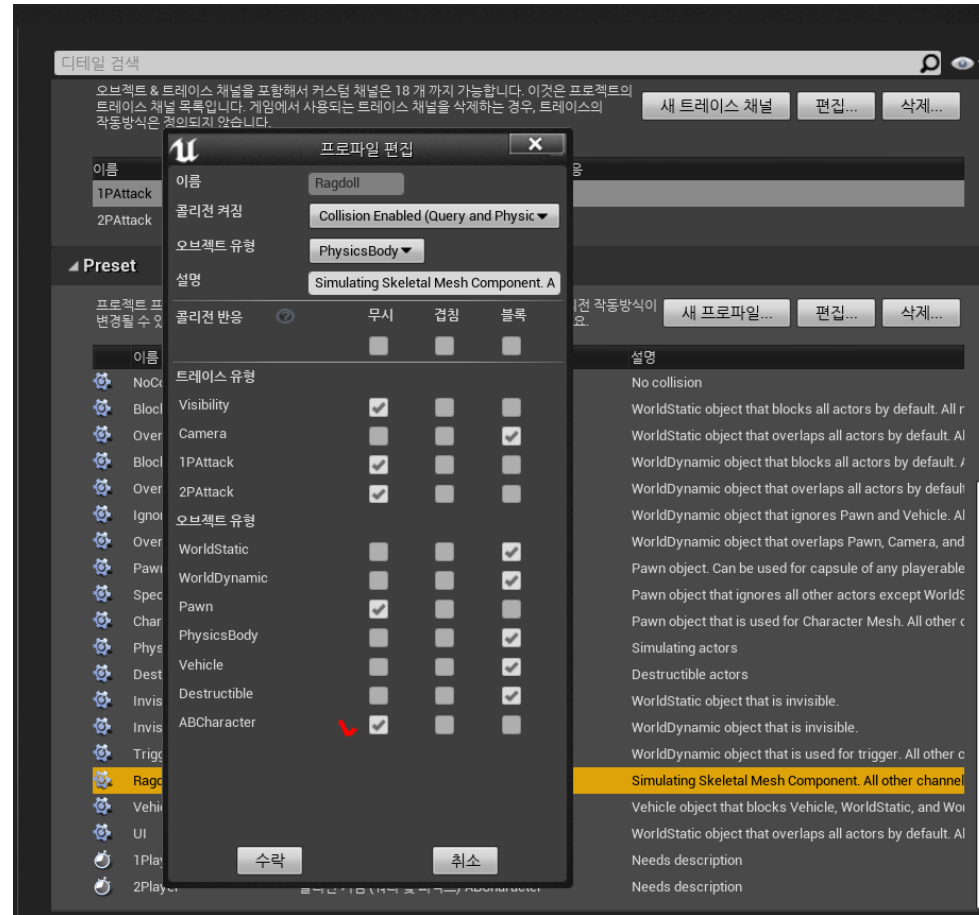
2.8 Spectator 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



2.9 CharacterMesh 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



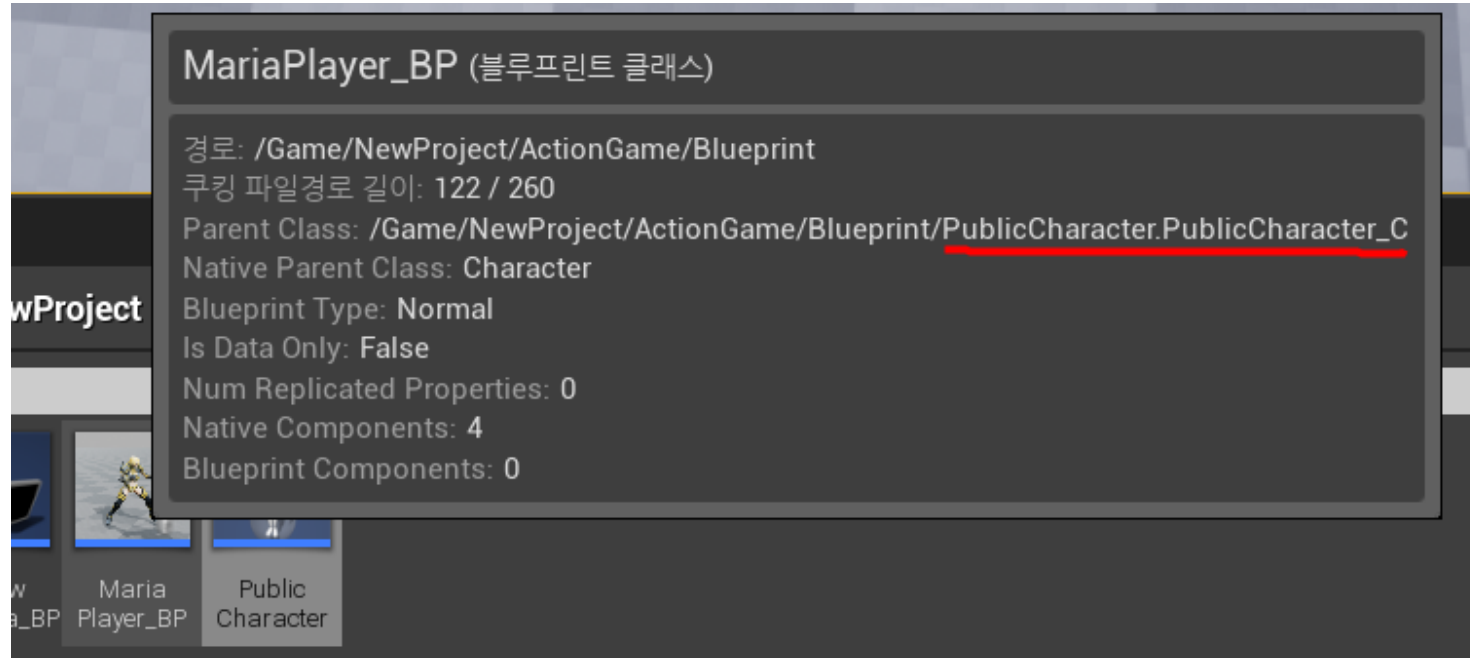
2.10 Trigger 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



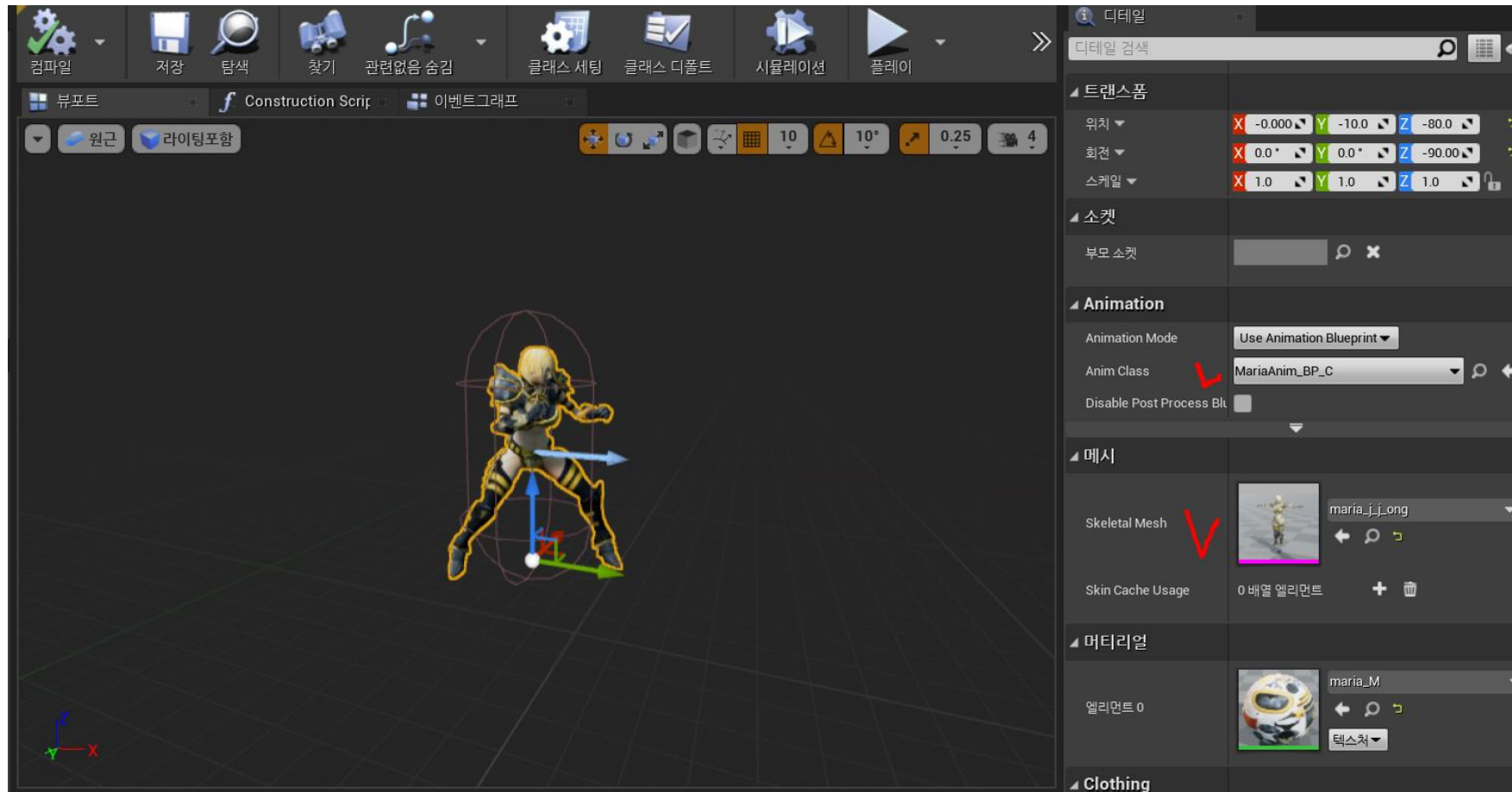
2.11 Ragdoll 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



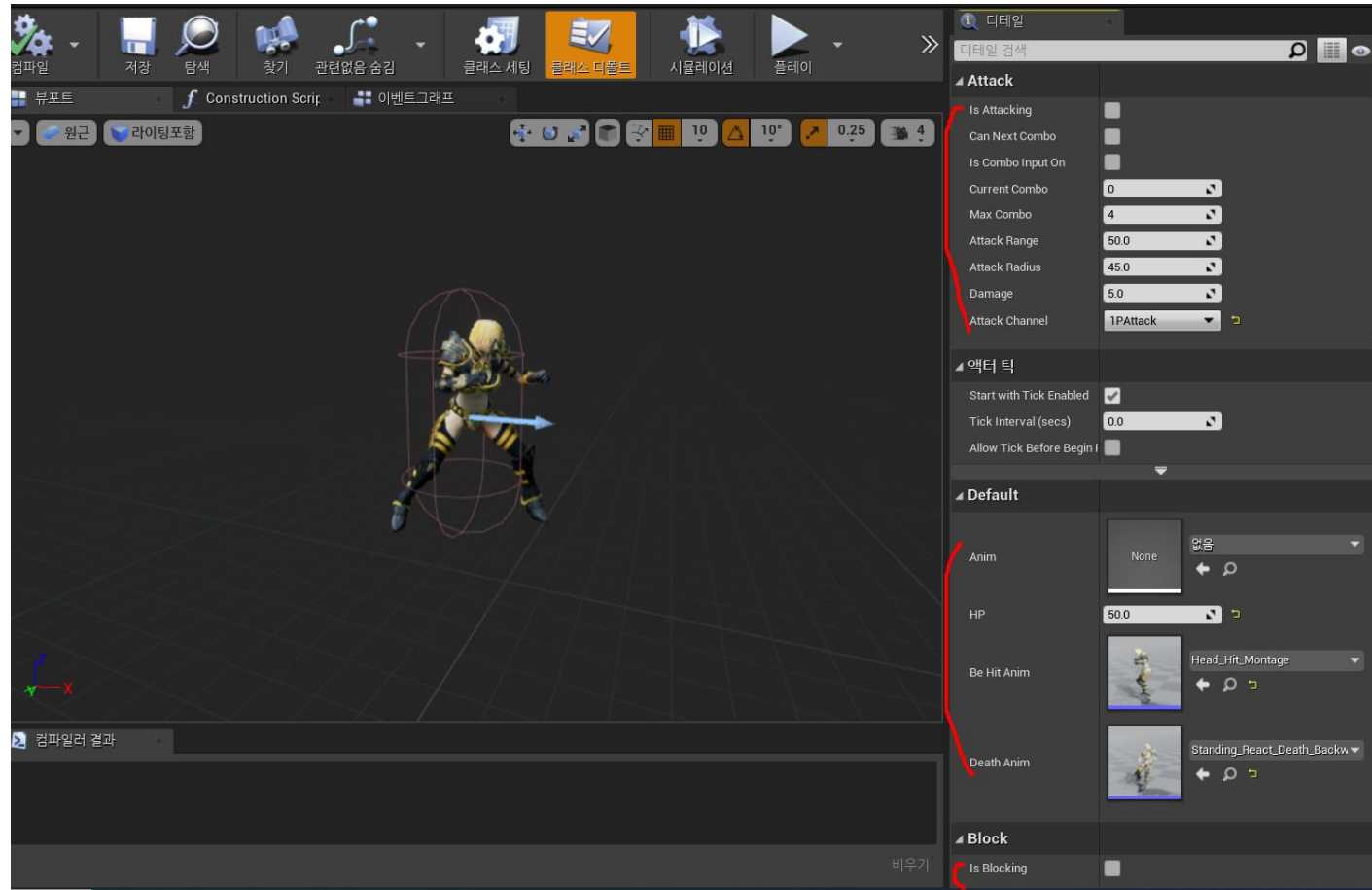
2.12 UI 프리셋의 ABCharacter 오브젝트 설정을 변경한다.



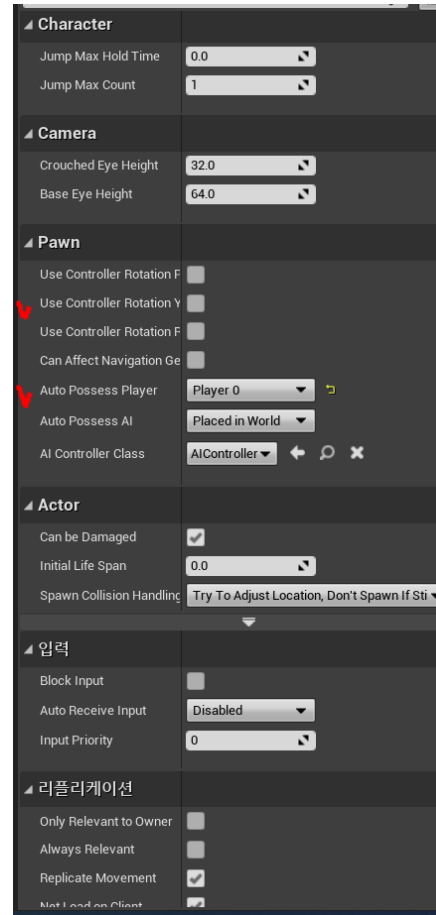
3. PublicCharacter를 상속받은 MariaPlayer_BP를 생성한다.



3.1 메시에 들어가서 애니메이션 블루프린트와 해당 메시를 넣는다.



3.2 클래스 디폴트에 들어가서 부모에게 받은 변수들을 수정한다.

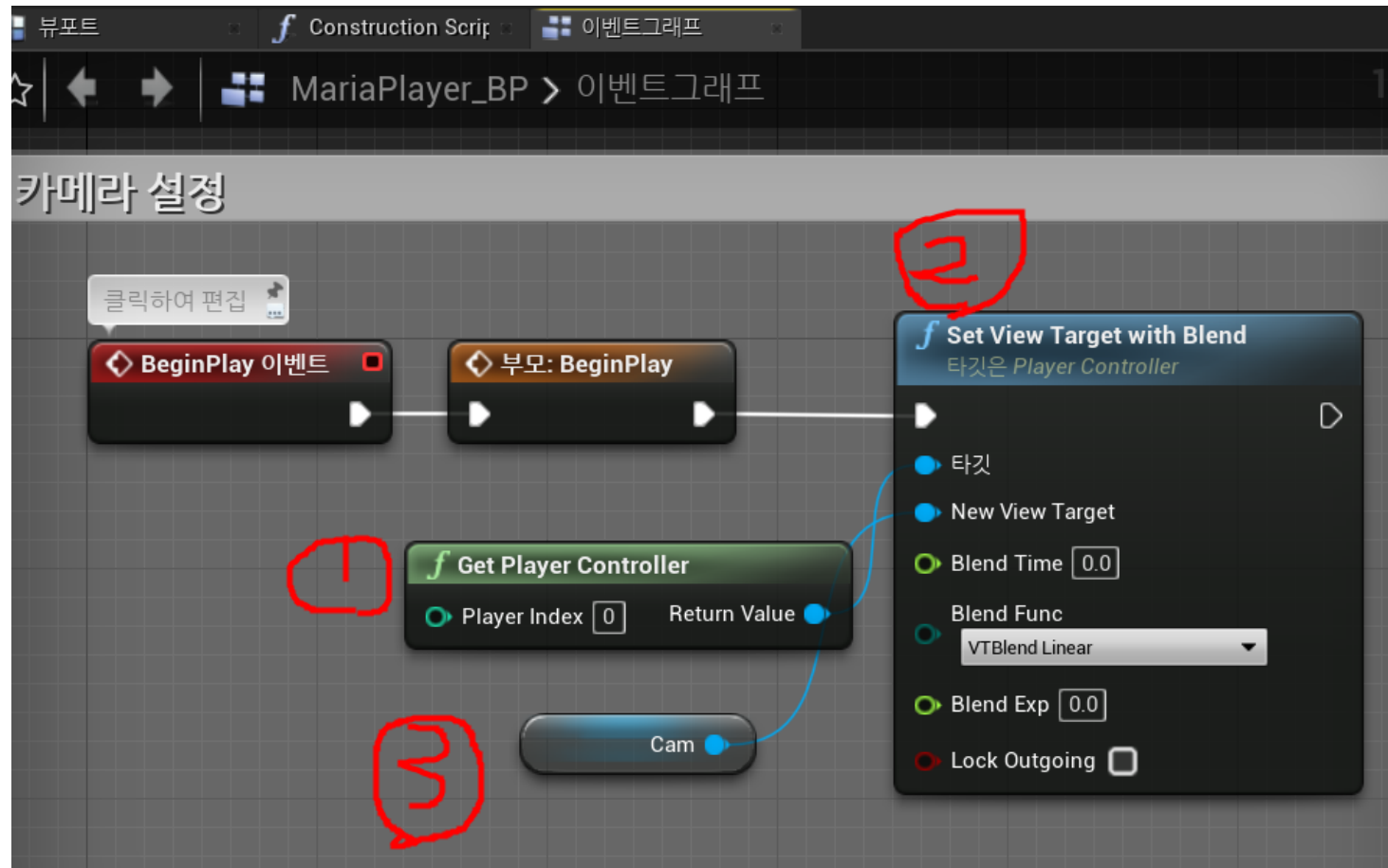


3.2.1 Pawn에 들어가서 변수 수정

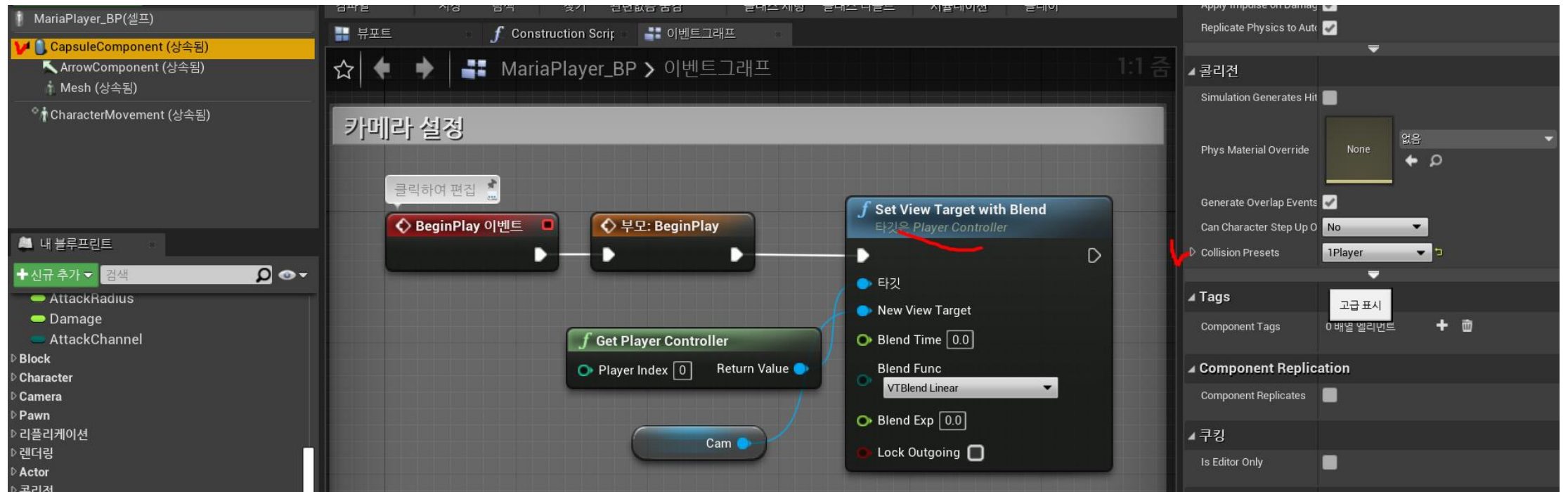
Variable	
변수 이름	Cam
변수 유형	Camera Actor
인스턴스 편집가능	☒
블루프린트 읽기 전용	☐
툴팁	
스폰시 노출	☐
프라이빗	☐
시네마틱에 노출	☐
카테고리	Default
리플리케이션	None
리플리케이션 조건	None

기본값	
Cam	없음

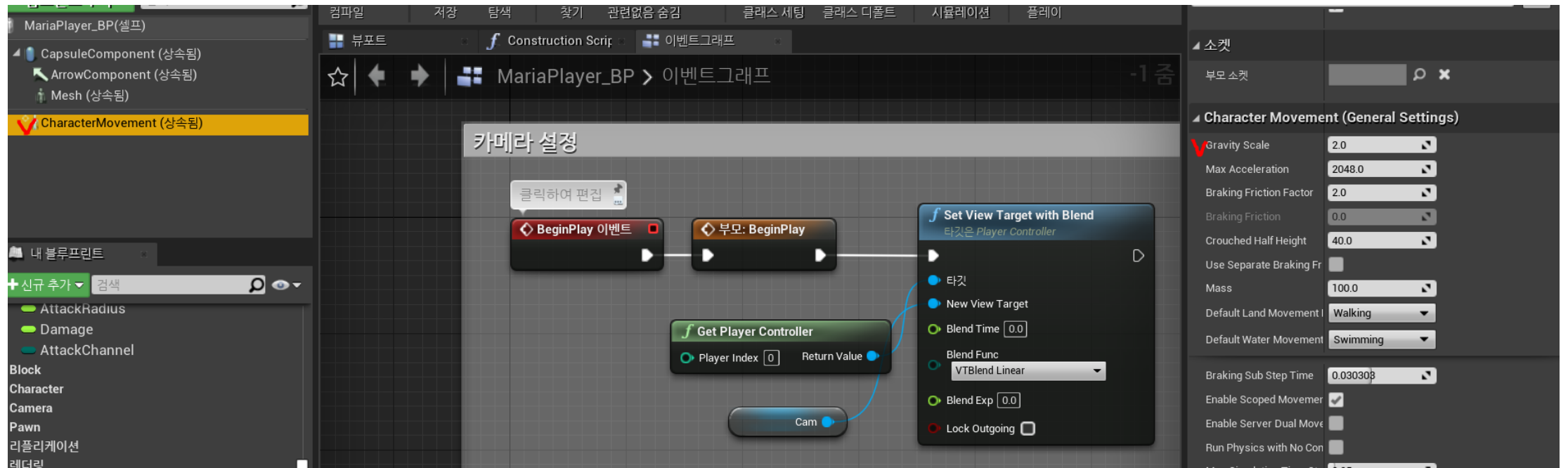
3.3 CameraActor 형 Cam 변수를 추가한다.



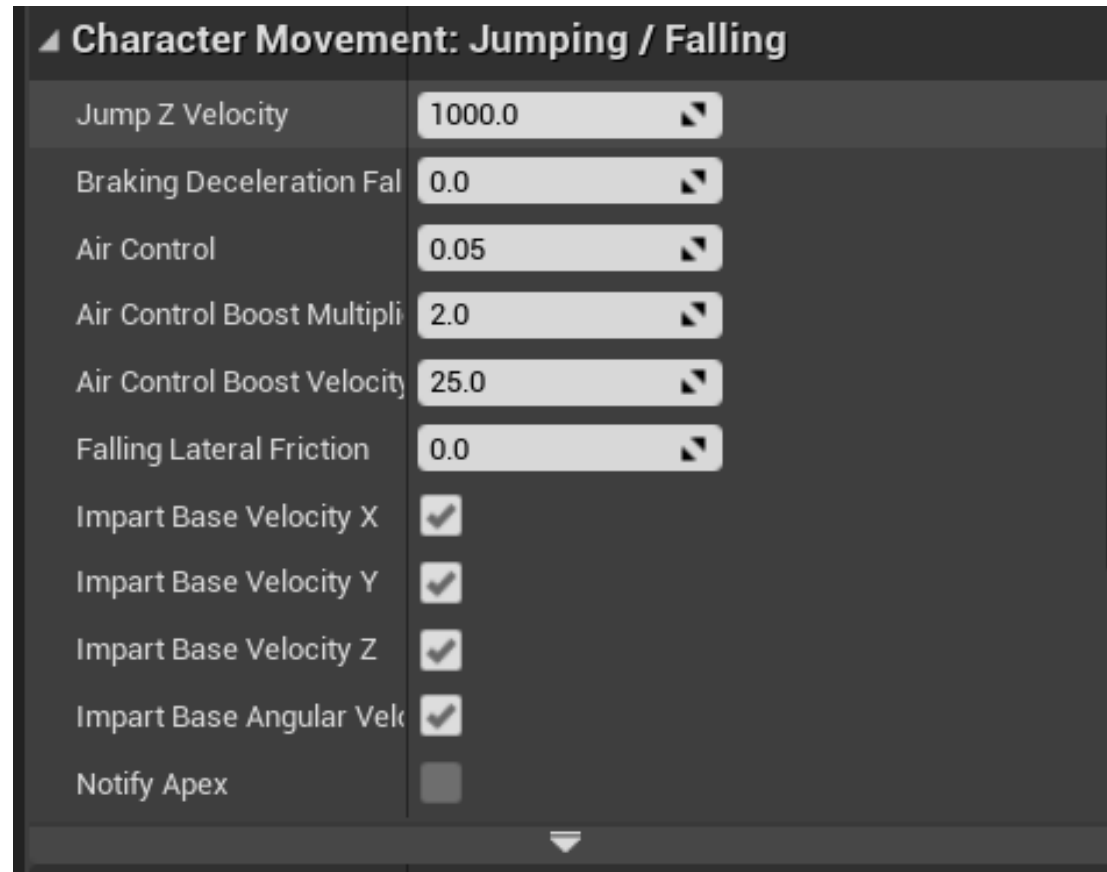
3.4 BeginPlay에 Cam을 타케팅하도록 한다.



3.5 CapsuleComponent의 콜리전에 들어가서 CollisionPresets에 1Player를 넣는다.



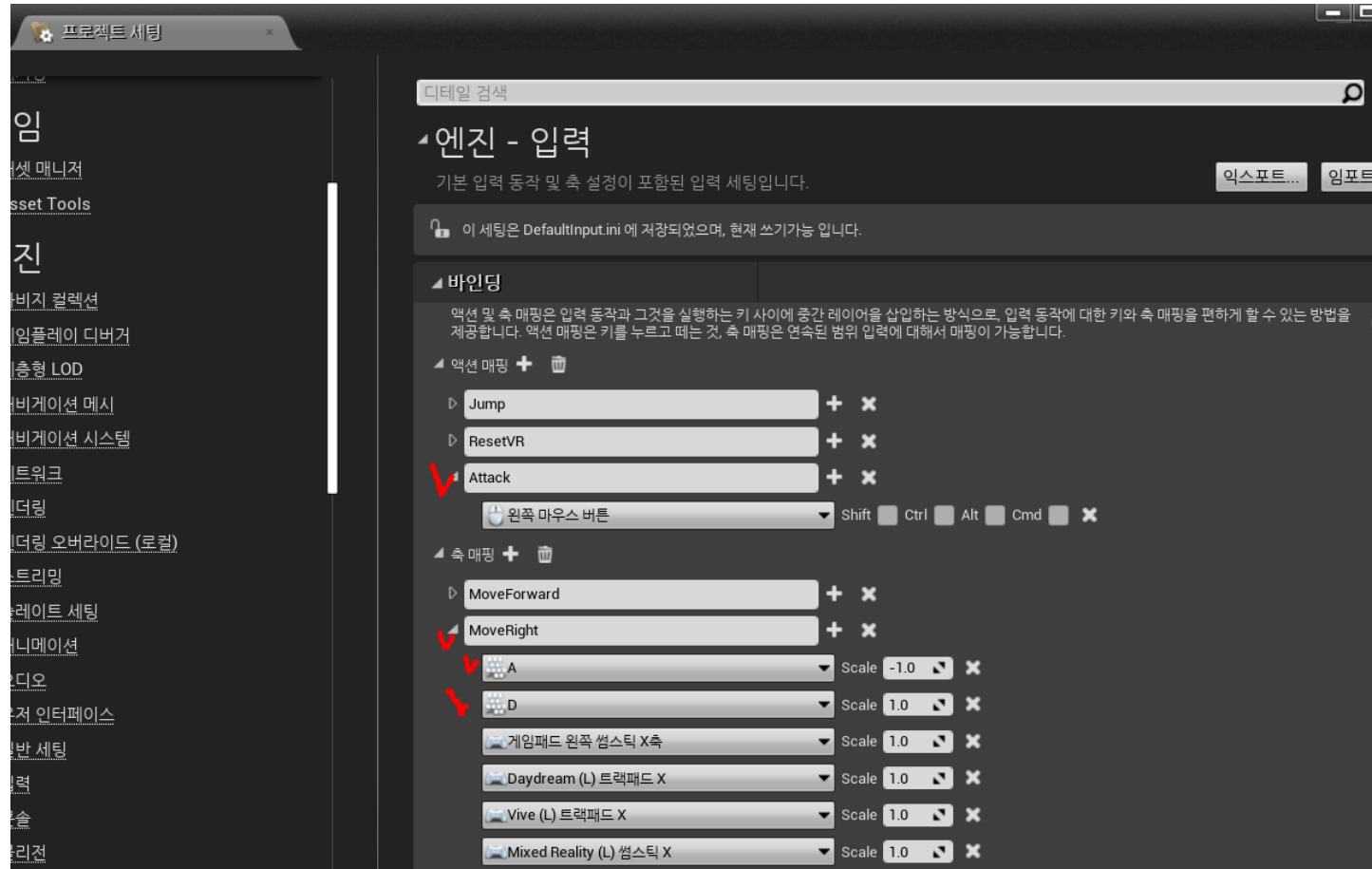
3.6 CharacterMovement에 들어가서 GravityScale을 수정한다.



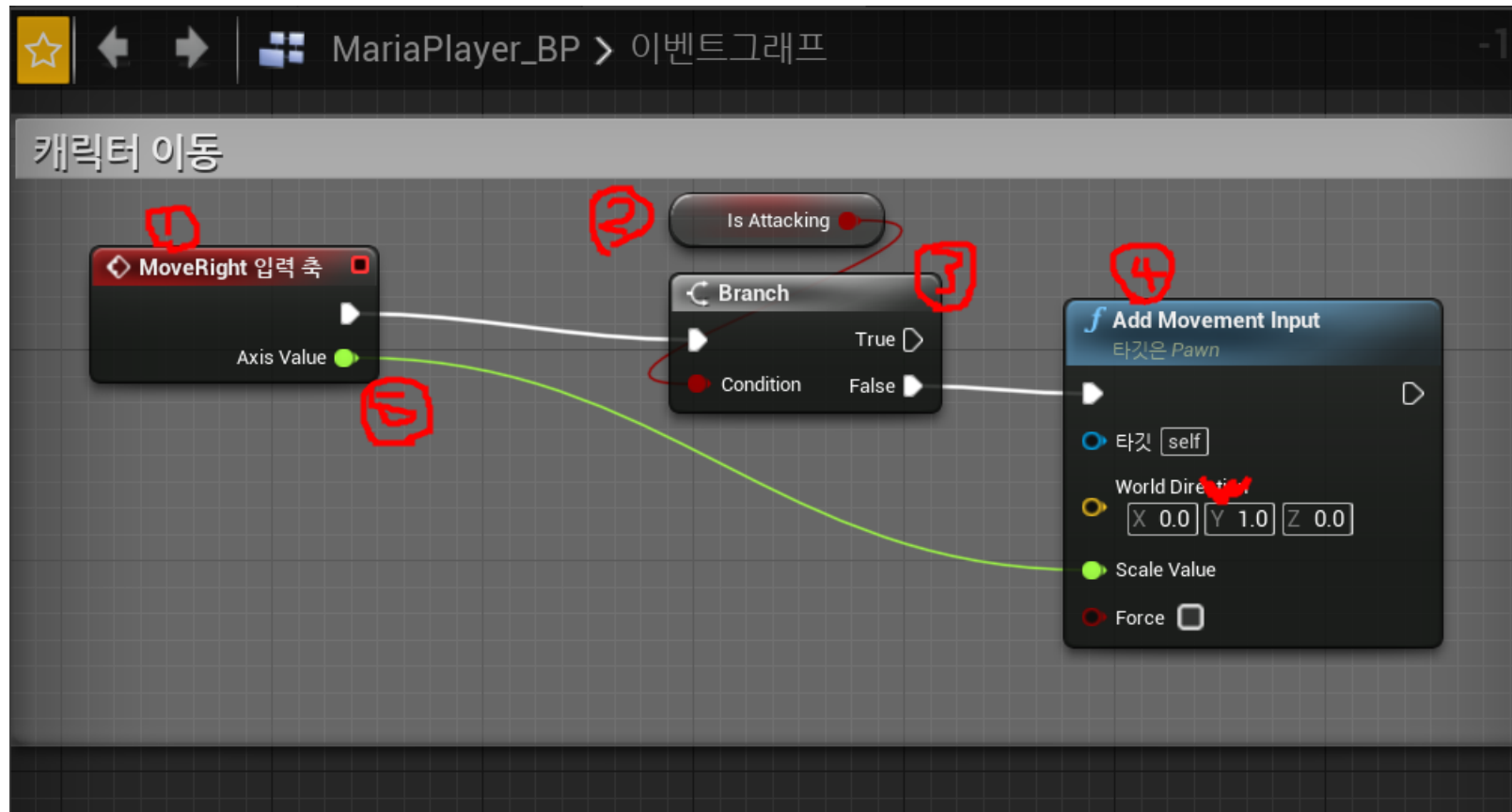
3.6.1 JumpZVelocity의 값을 수정한다.



3.6.2 RotationRate의 Z값을 수정하고, OrientRotationtoMovement를 수정한다.



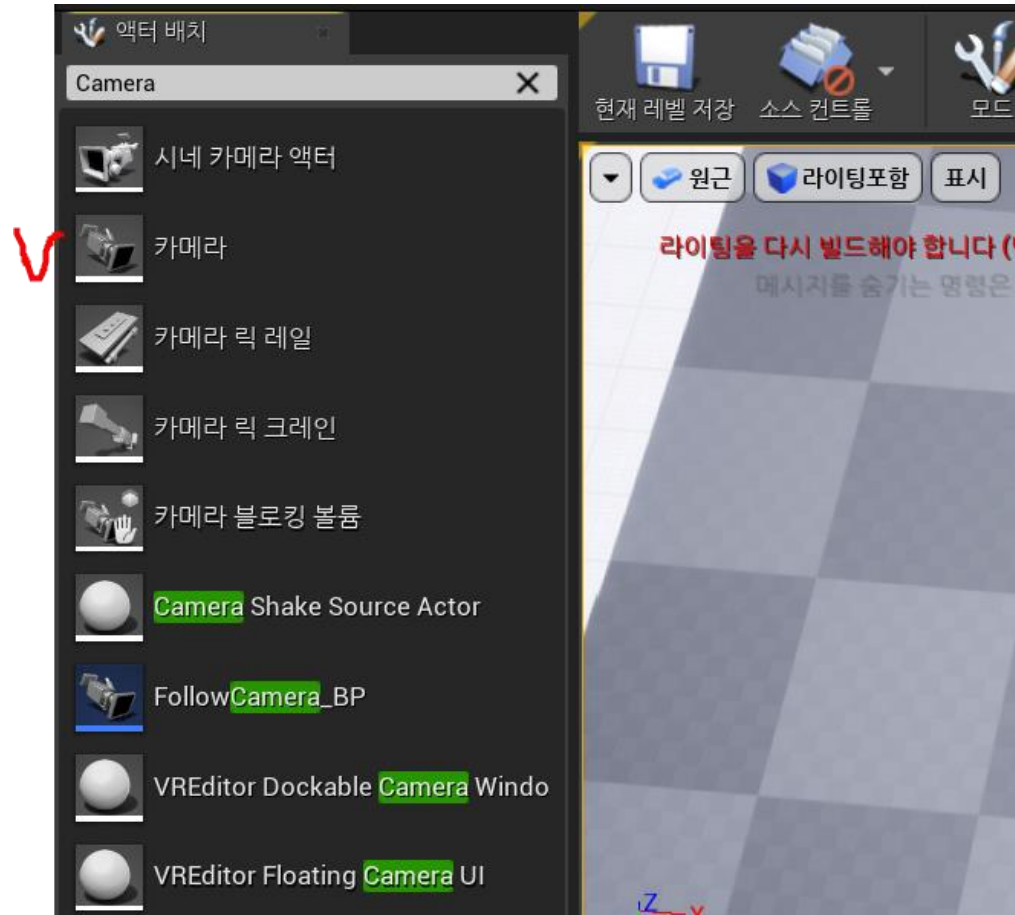
3.7 프로젝트 세팅, 입력에 들어가서 Attack, MoveRight를 추가한다.



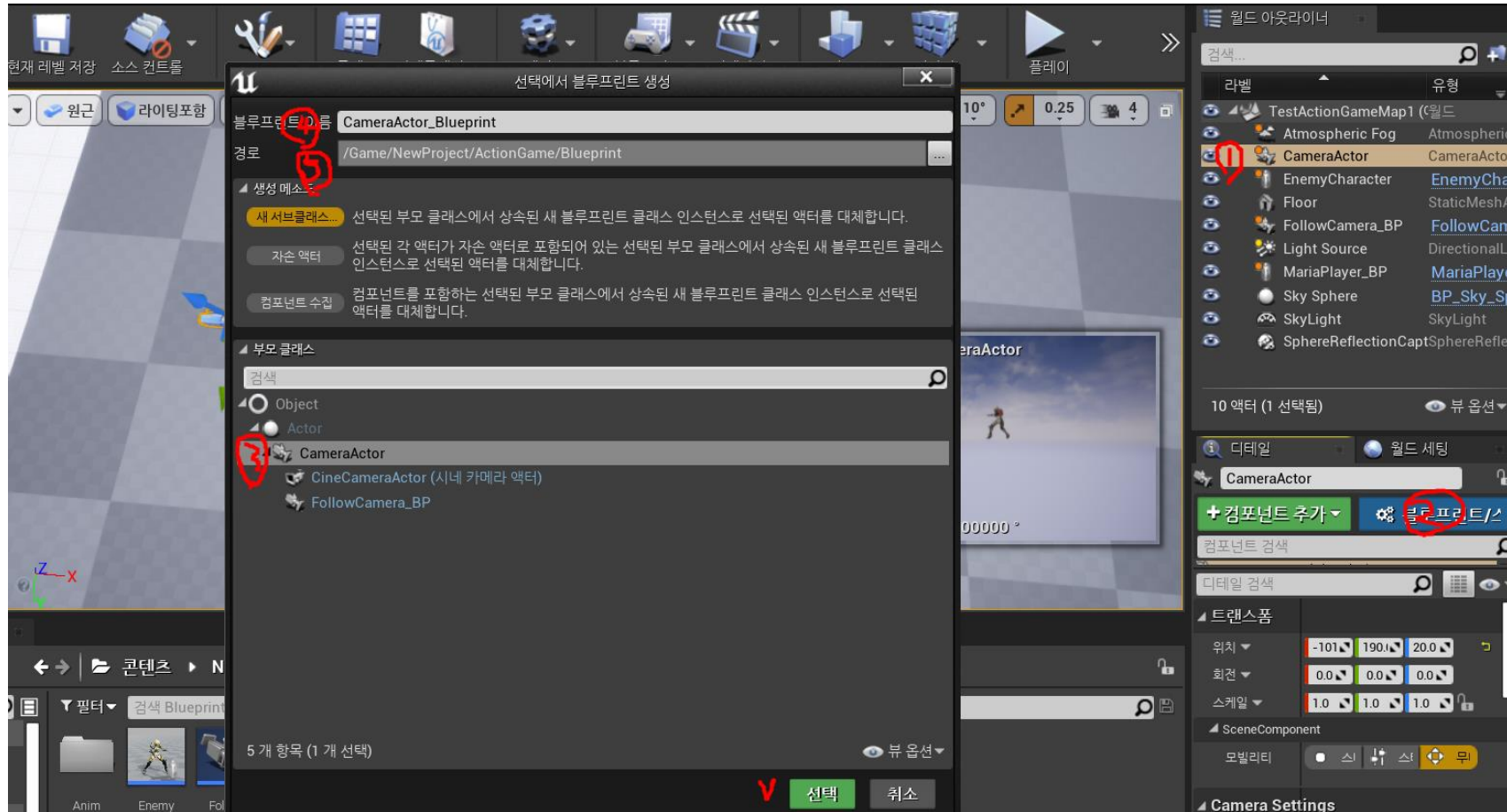
3.7.1 캐릭터 이동 처리



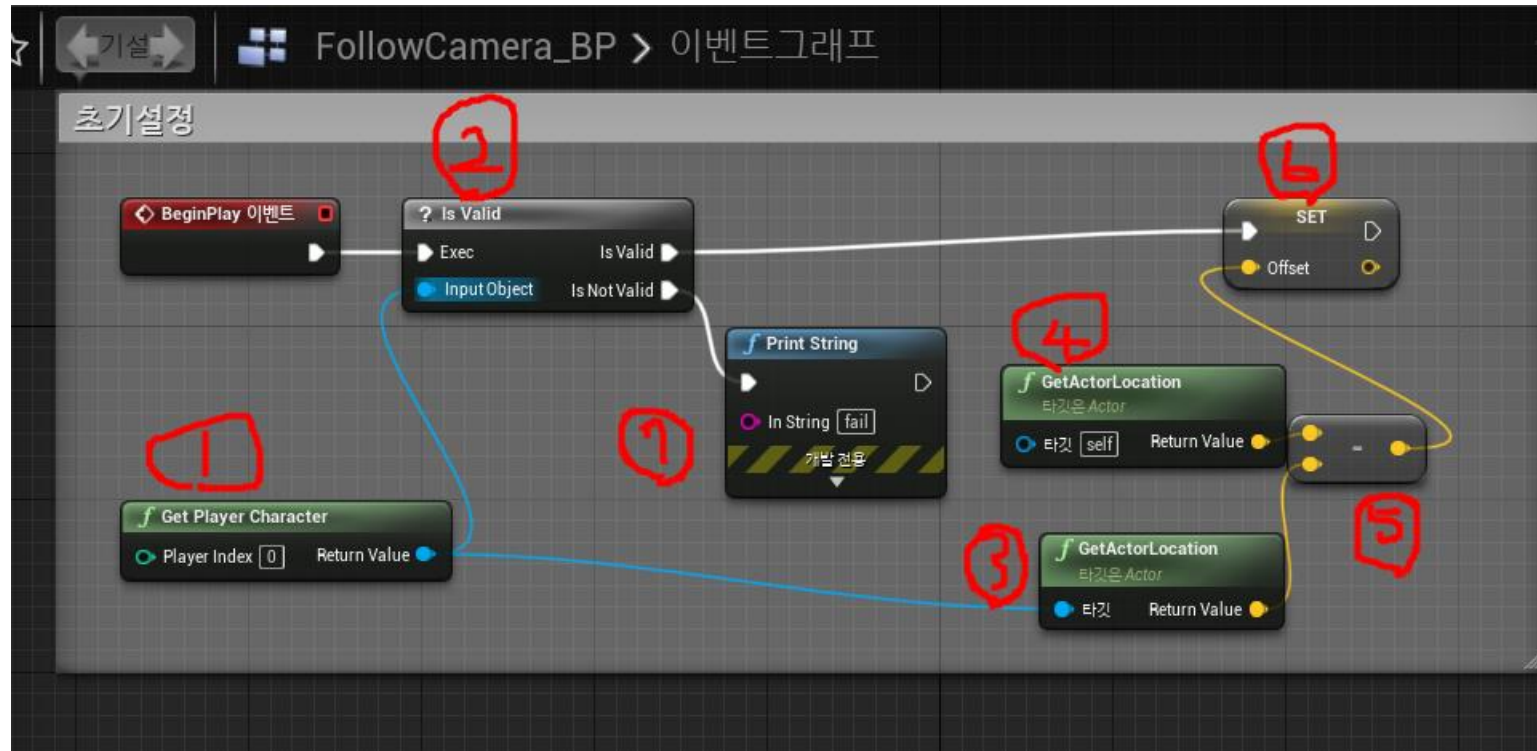
3.7.2 캐릭터 공격을 처리한다.



4. 카메라를 검색해서 카메라를 레벨에 배치한다.



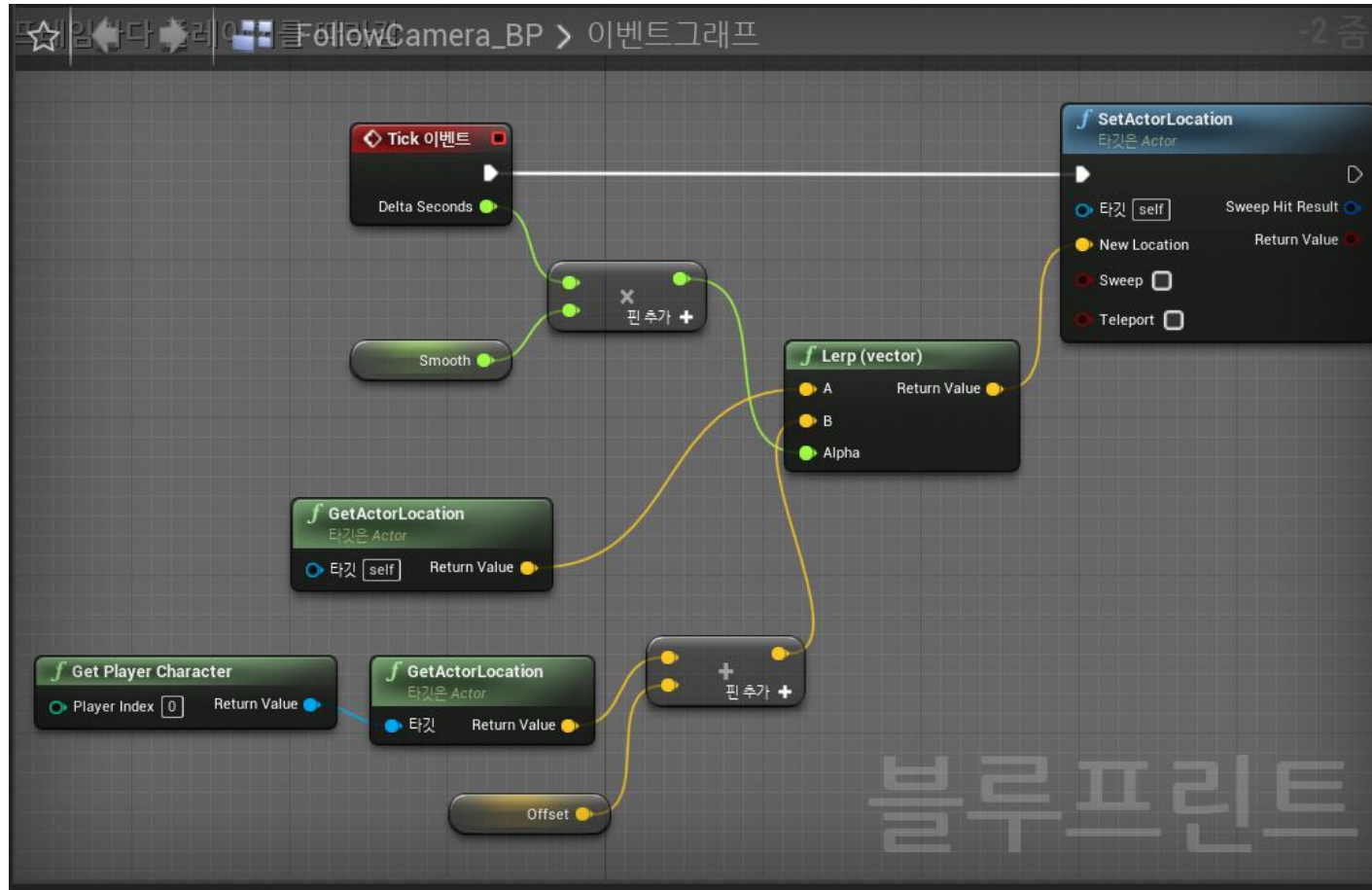
4.1 레벨에 배치된 카메라를 블루프린트로 편집하기 위해서 블루프린트를 생성한다.



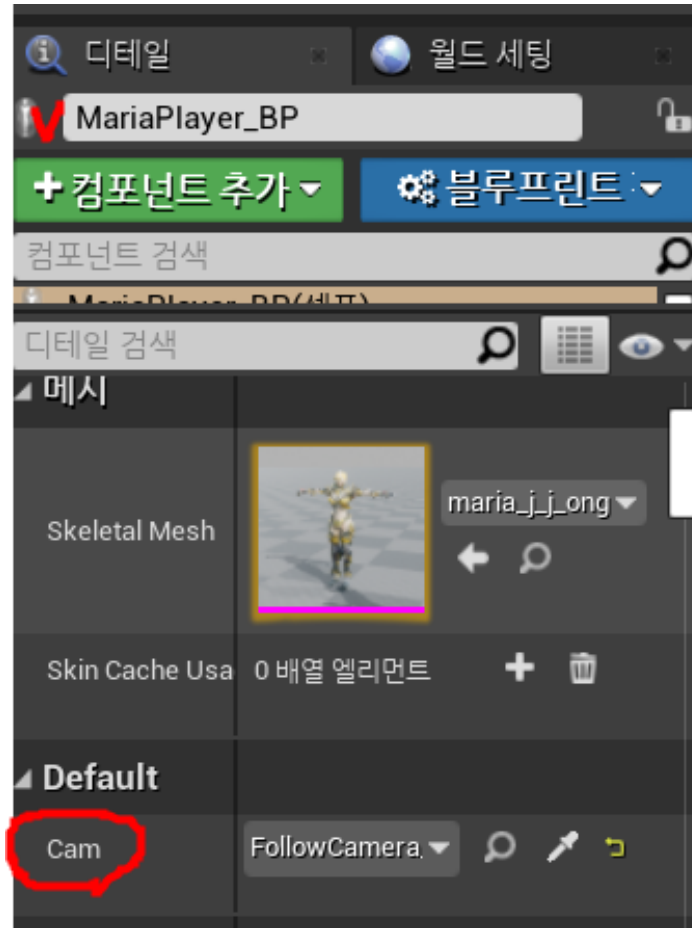
4.2 생성된 블루프린트에 들어가서 Vector 계산을 변수로 승격해서 Offset을 생성한다.

Variable	
변수 이름	Smooth
변수 유형	플로트
인스턴스 편집가능	☐
블루프린트 읽기 전용	☐
툴팁	
스폰시 노출	☐
프라이빗	☐
시네마틱에 노출	☐
카테고리	Default
슬라이더 범위	
값 범위	
리플리케이션	None
리플리케이션 조건	None
기본값	
Smooth	6.0

4.3 float형 변수 Smooth 추가



4.4 매 프레임마다 캐릭터의 위치가 바뀌면 캐릭터의 위치로 이동하게 한다.



4.5 레벨에 배치된 캐릭터의 Cam 변수에 FollowCamera를 추가한다.



4.6 레벨에 배치된 플레이어 캐릭터



4.7 레벨에 배치된 FollowCamera