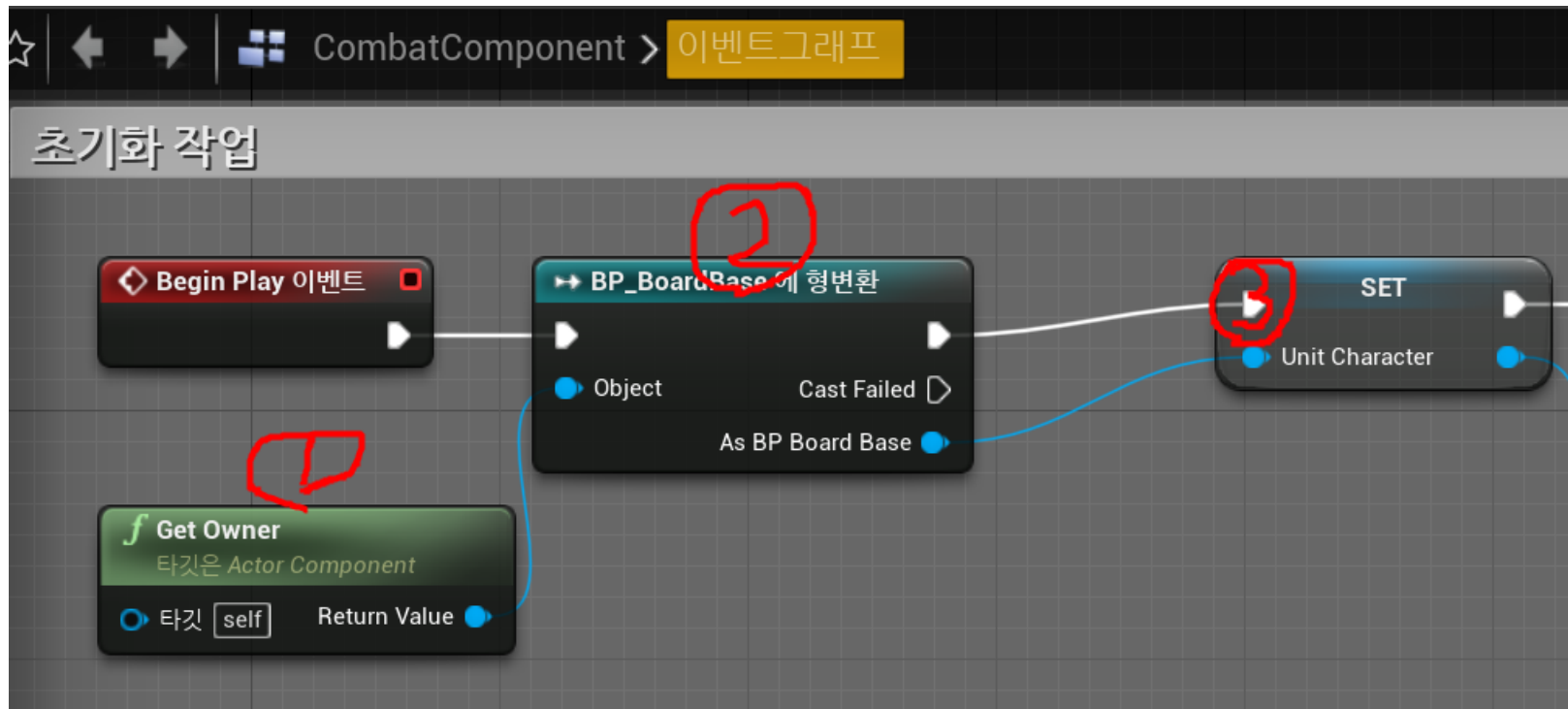
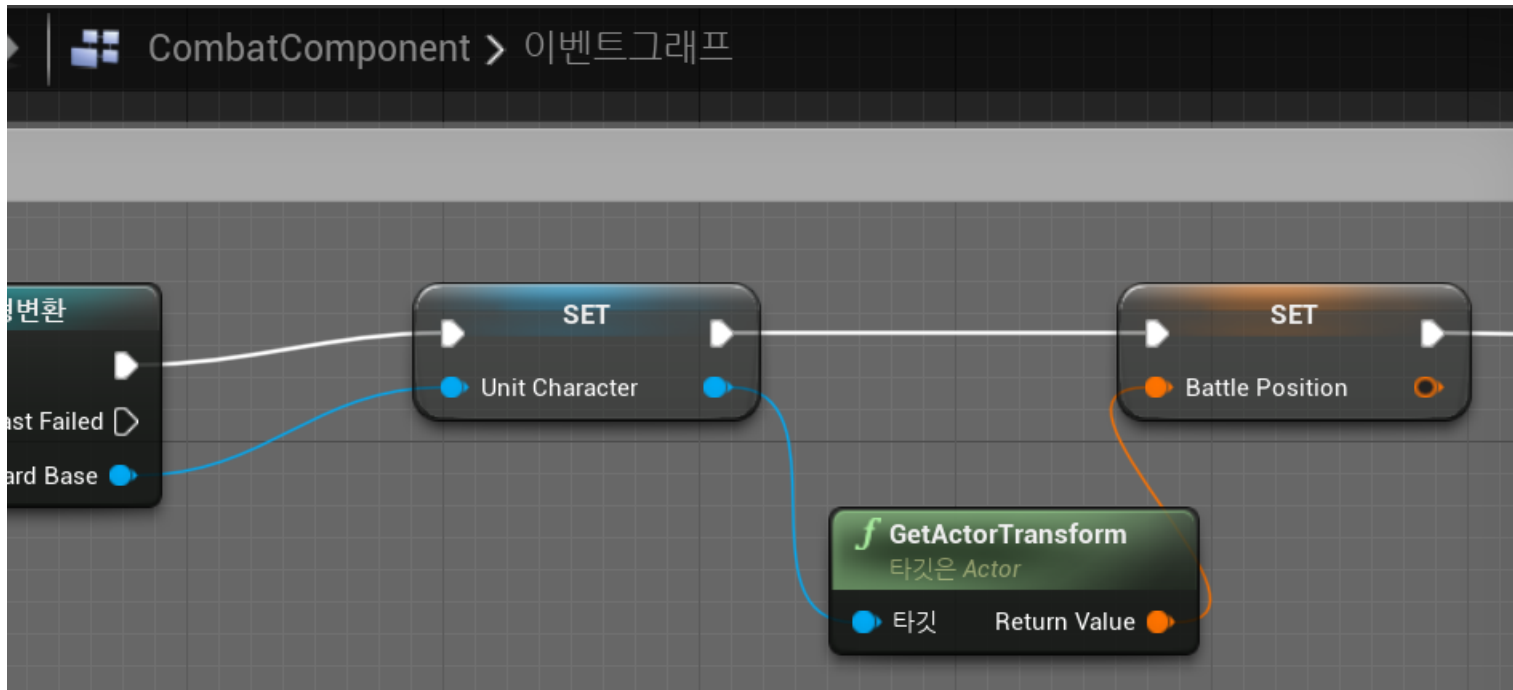


# 목차

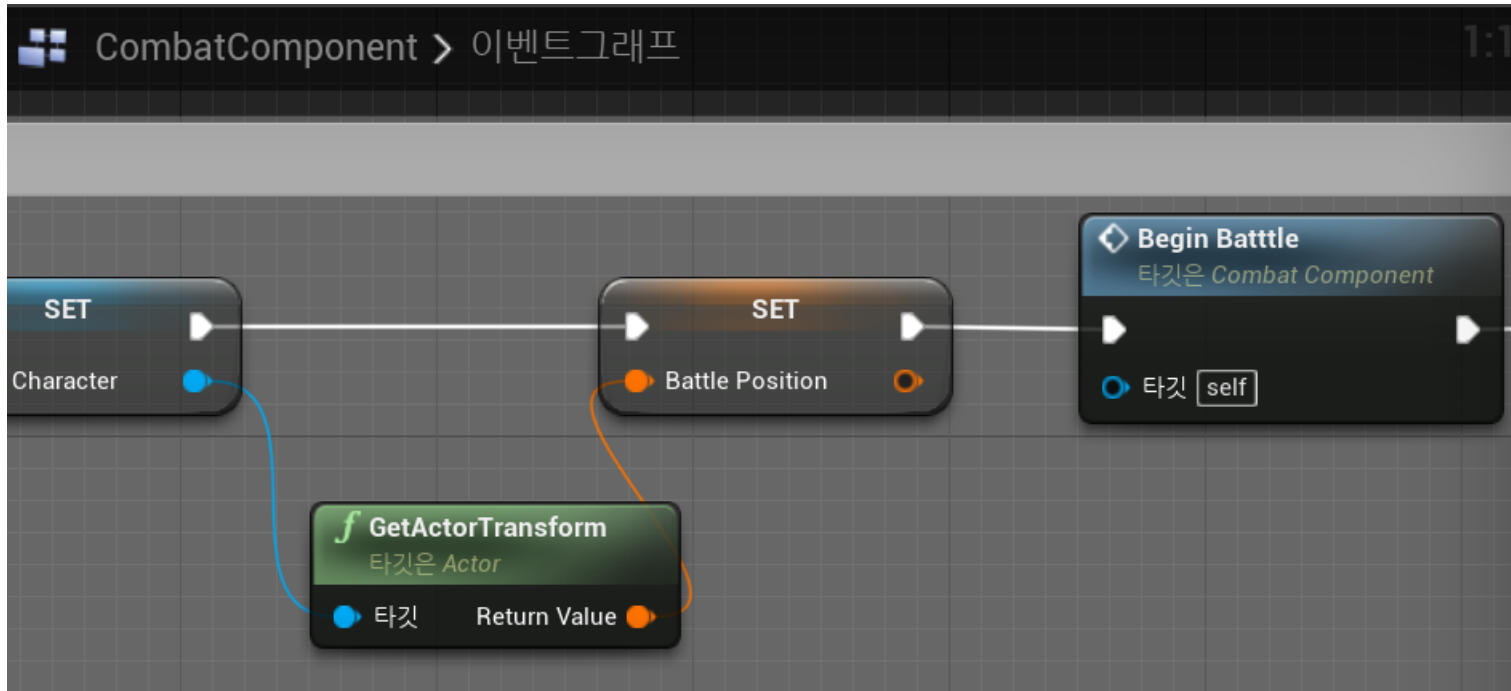
1. **CombatComponent 처리**
2. **BattlePosition 처리**
3. **Actor를 상속받은 Camera\_TopDown 생성**
4. **Actor를 상속받은 DynamicCarmera를 생성한다.**
5. **Interface DynamicCameraInterface를 생성한다.**
6. **DynamicCamera 처리**
7. **BattlePosition과 Camera\_TopDown, DynamicCamera를 레벨에 추가한다.**



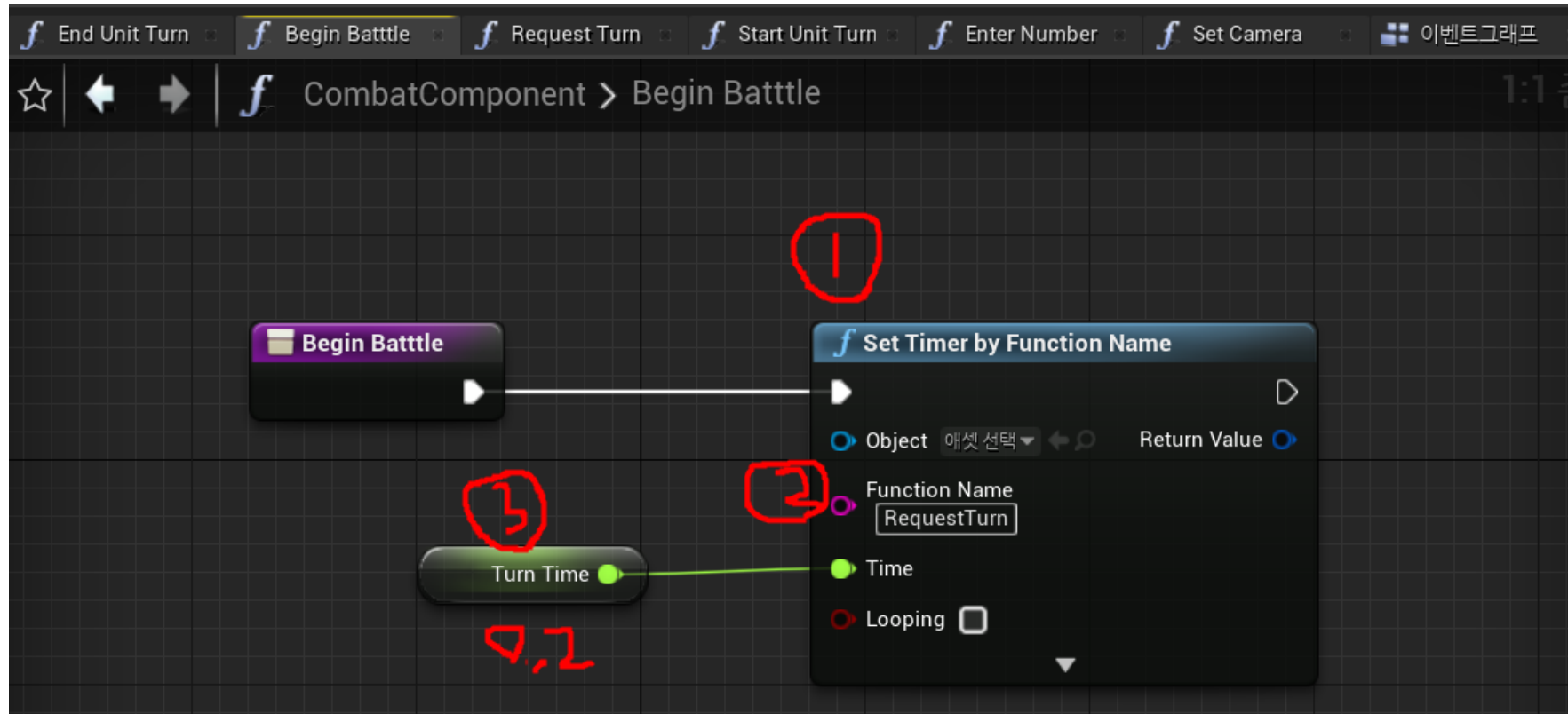
1. CombatComponent에 들어가서 BeginPlay 이벤트에 BP\_BoardBase를 형변환하고 변수로 승격해서 UnitCharacter를 생성한다.



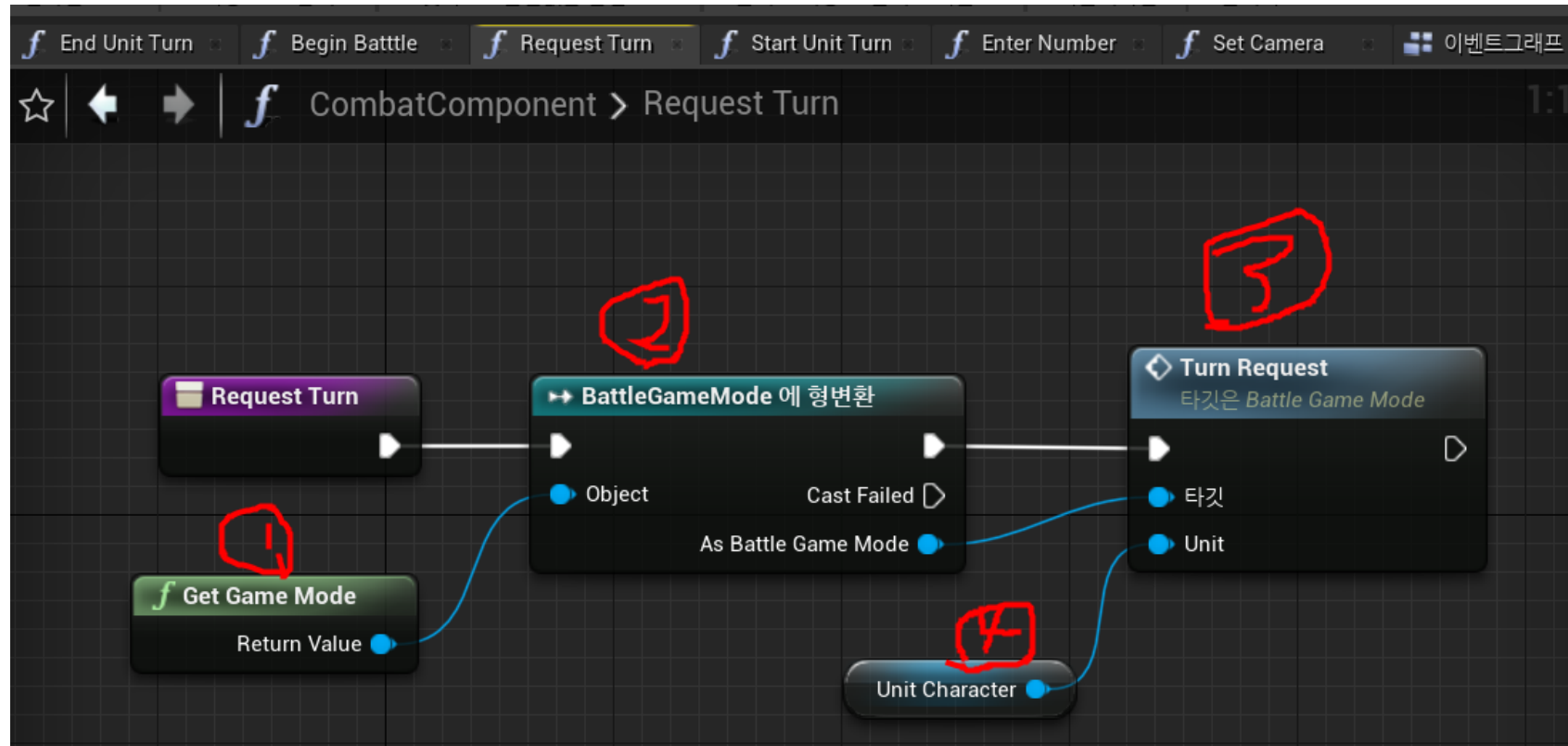
1.1 UnitCharacter에 GetActorTransform을 호출하고 결과값을 변수로 승  
격해서 BattlePosition을 생성한다.



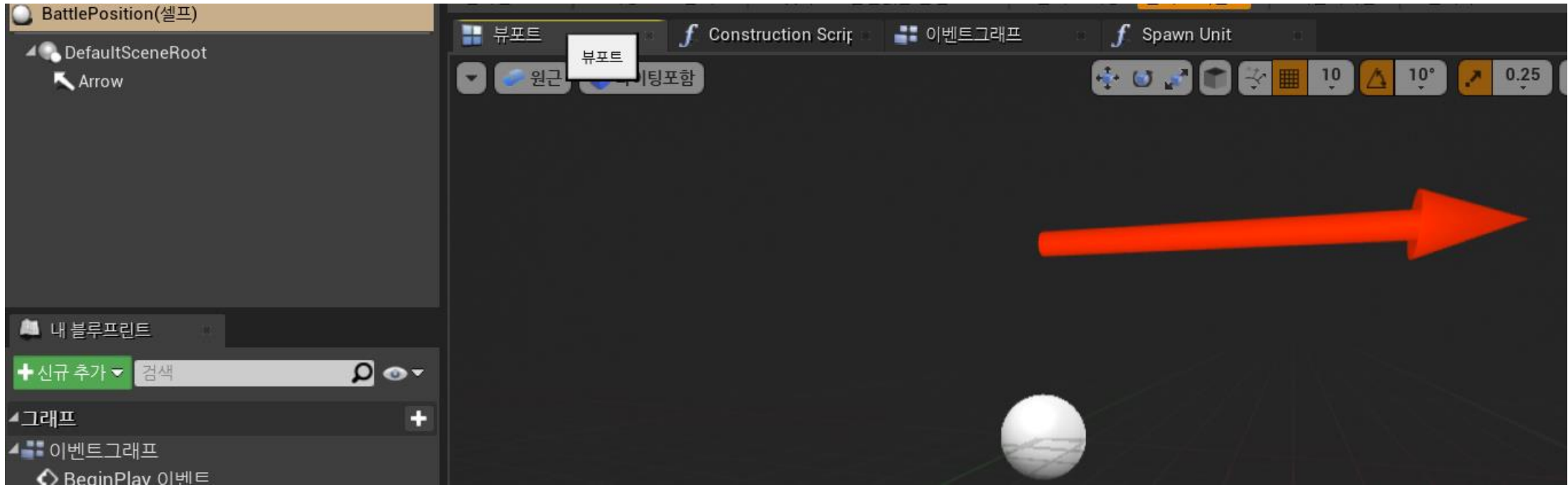
1.2 BeginBattle 함수를 호출한다.



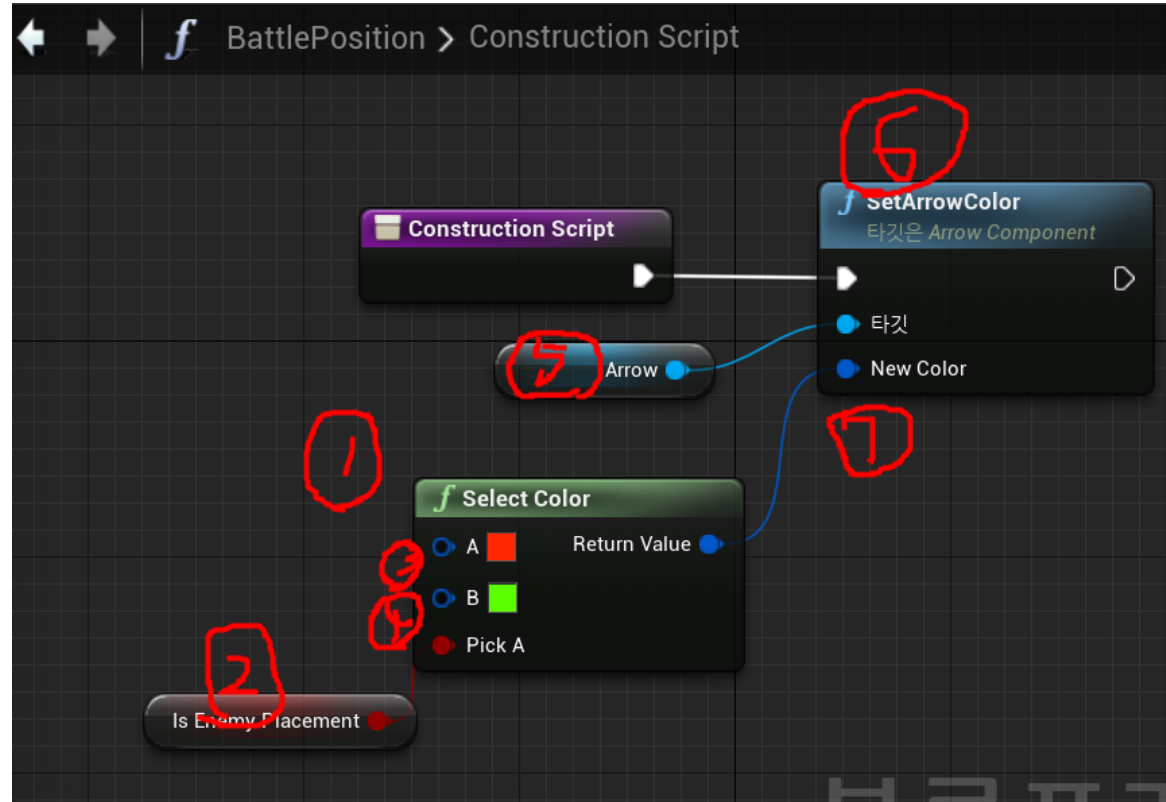
1.3 BeginBattle 함수에 들어가서 SetTimerbyFuctionName을 호출하고, FuctionName에 RequestTurn을 넣고 Time을 변수로 승격해서 Turn Time을 생성하고 0.2로 설정한다.



1.4 RequestTurn 함수에 들어가서 BattelGameMode를 형변환하고 TurnRequest 함수를 호출한다.



2. BattlePosition에 들어가서 Arrow를 추가한다.



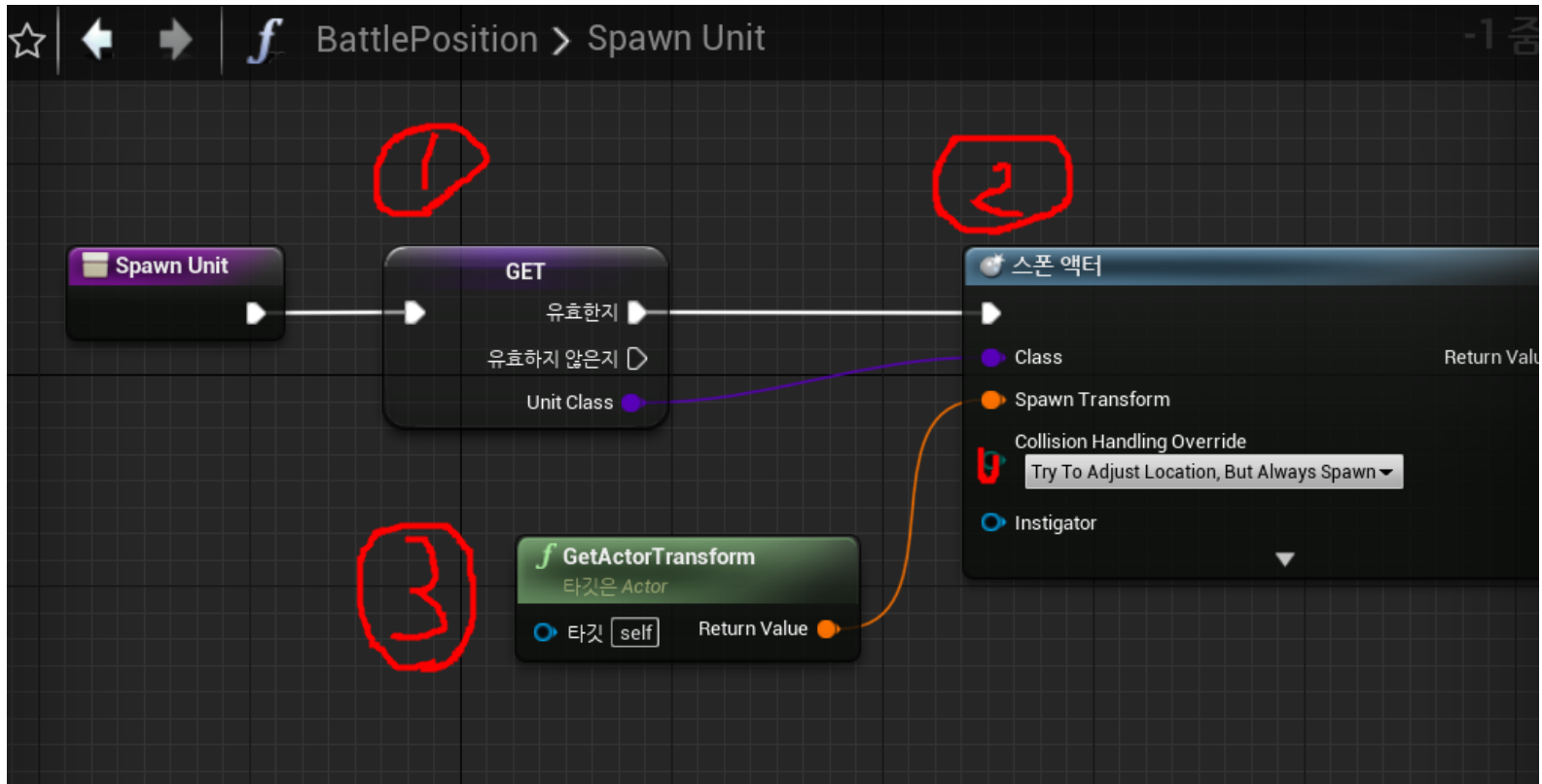
2.1 ConstructionScript에 들어가서 SelectColor를 호출하고 PickA에 변수로 승격해서 IsEnemyPlacement를 생성.



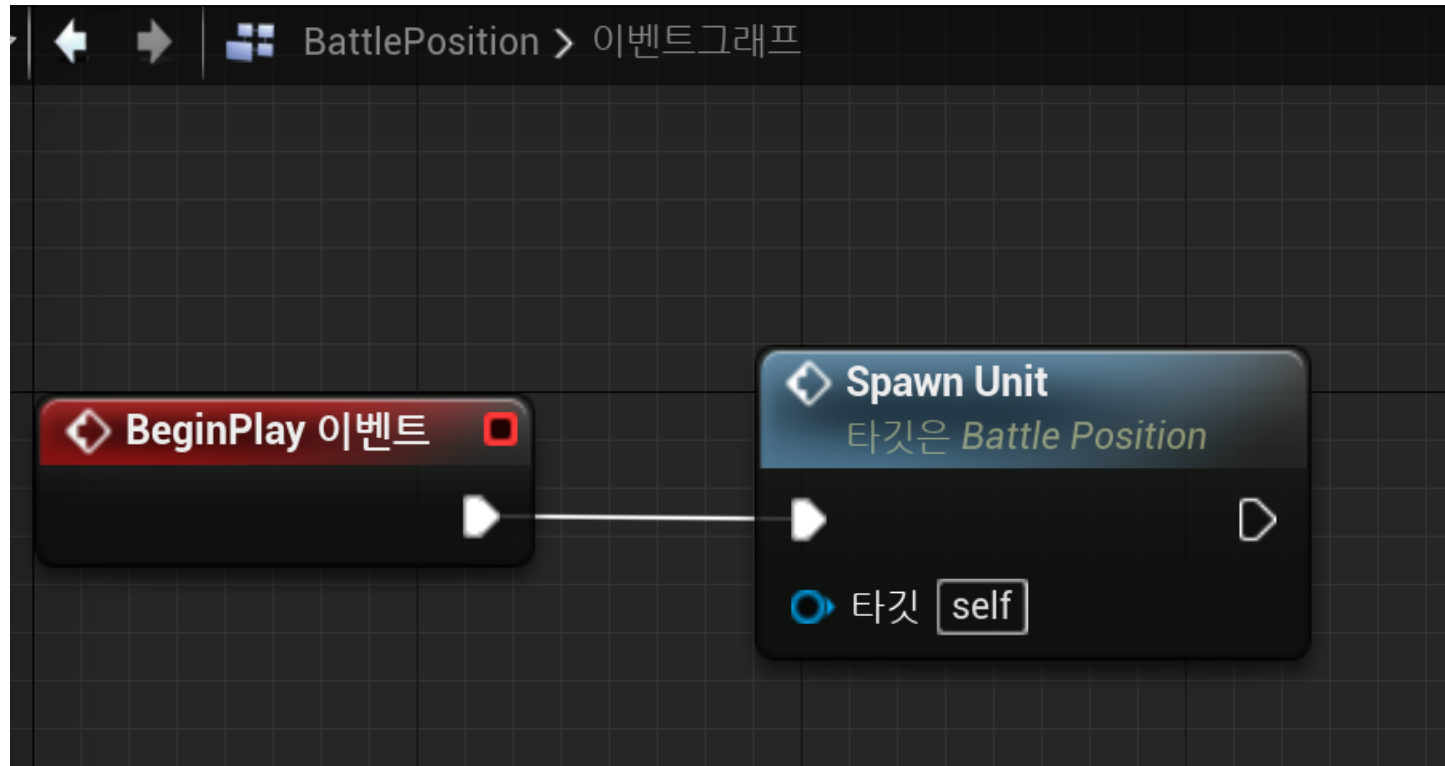
Variable	
변수 이름	UnitClass
변수 유형	BP Board Base
인스턴스 편집가능	<input checked="" type="checkbox"/>
블루프린트 읽기 전용	<input type="checkbox"/>
툴팁	
스폰시 노출	<input type="checkbox"/>
프라이빗	<input type="checkbox"/>
시네마틱에 노출	<input type="checkbox"/>
카테고리	Default
리플리케이션	None
리플리케이션 조건	None

기본값	
Unit Class	None

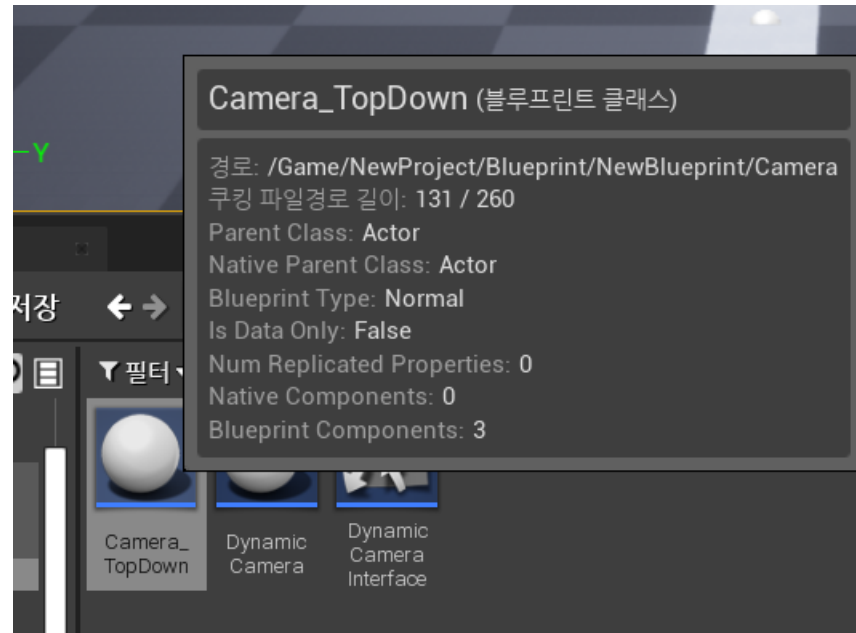
2.2 BP\_BoardBase 변수 UnitClass를 생성한다.(인스턴스 편집기능에 체크한다.)



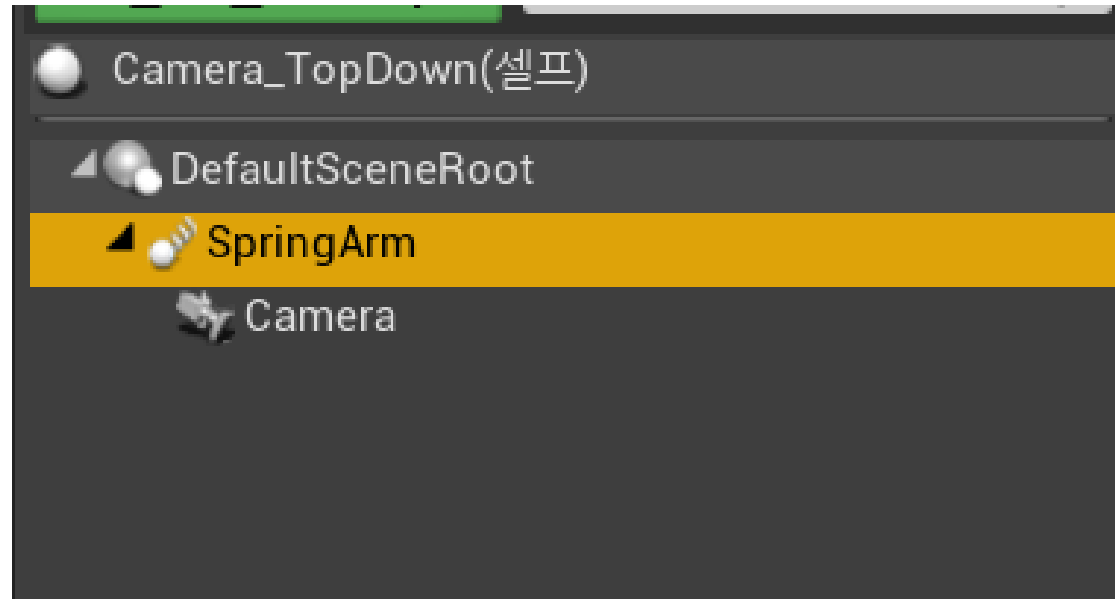
2.3 SpawnUnit 함수를 생성하고, UnitClass를 스폰한다.



## 2.4 BeginPlay 이벤트에 SpawnUnit 함수 호출



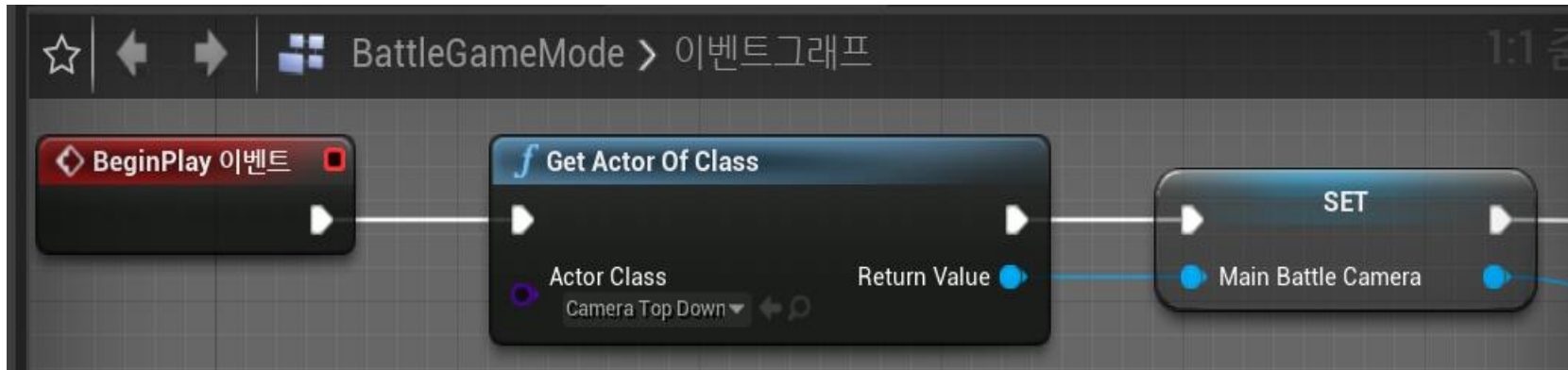
### 3. Actor를 상속받은 Camera\_TopDown 생성



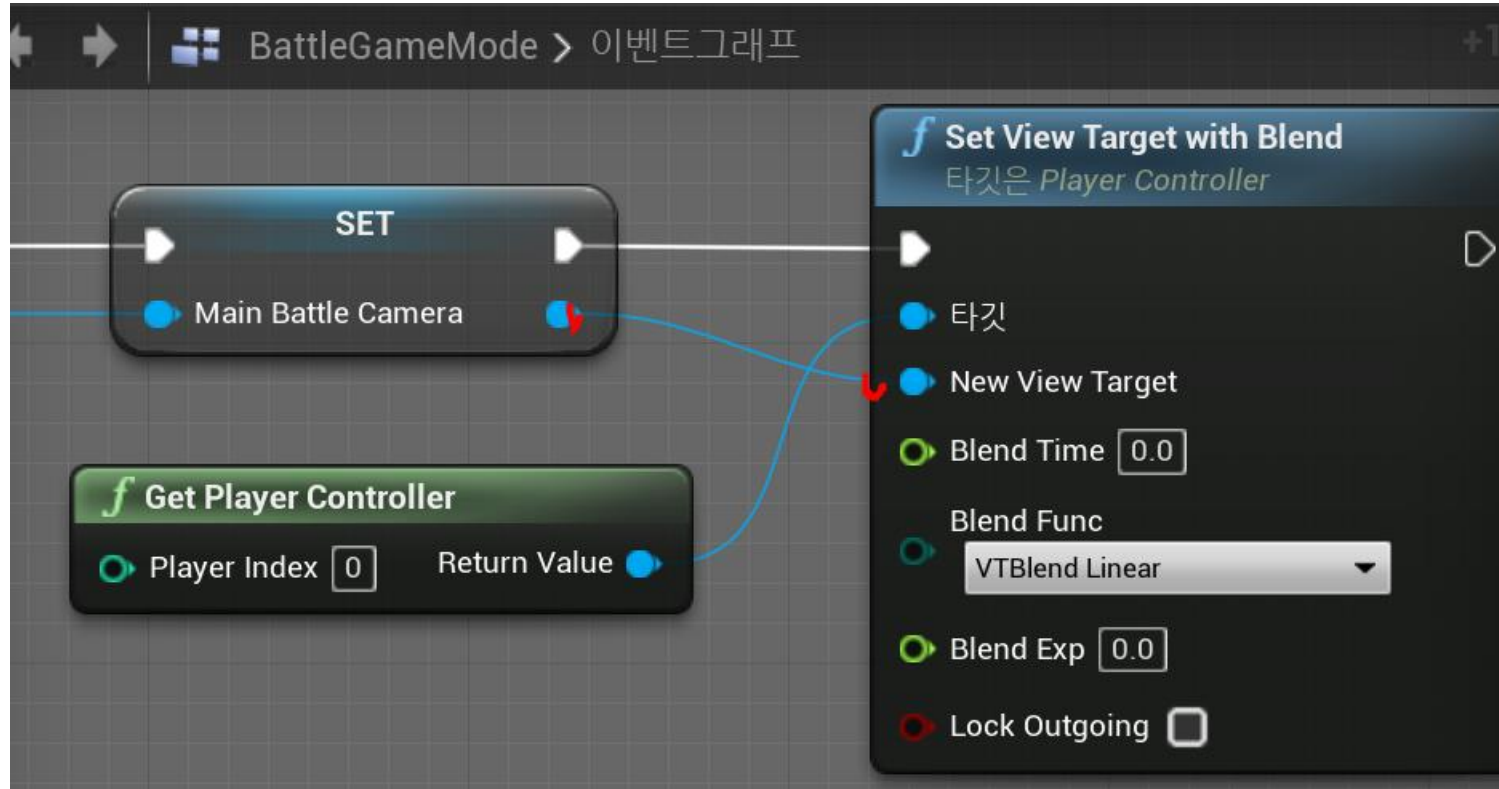
### 3.1 Camera\_ToDown에 들어가서 컴포넌트 추가

Variable	
변수 이름	SpringArm
틀립	
카테고리	디폴트
Editable when Inherited	<input checked="" type="checkbox"/>
트랜스폼	
위치	X 0.0 Y 0.0 Z 70.0
회전	X 0.0° Y -40.0° Z 0.0°
스케일	X 1.0 Y 1.0 Z 1.0
소켓	
부모 소켓	None
Camera	
Target Arm Length	300.0
Socket Offset	X 0.0 Y 0.0 Z 0.0
Target Offset	X 0.0 Y 0.0 Z 0.0
Camera Collision	
Probe Size	12.0
Probe Channel	Camera
Do Collision Test	<input checked="" type="checkbox"/>

### 3.2 SpringArm의 값을 변경한다.

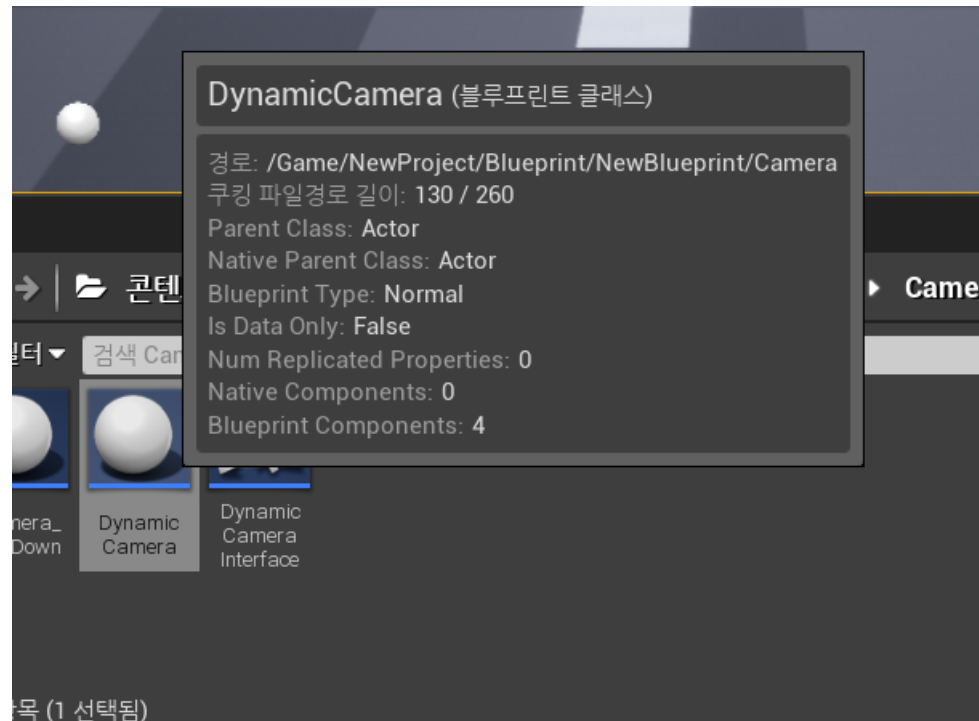


**3.3 BattleGameMode에 들어가서 Camera\_TopDown을 가져오고 변수로 승격해서 MainBattleCamera를 생성한다.**

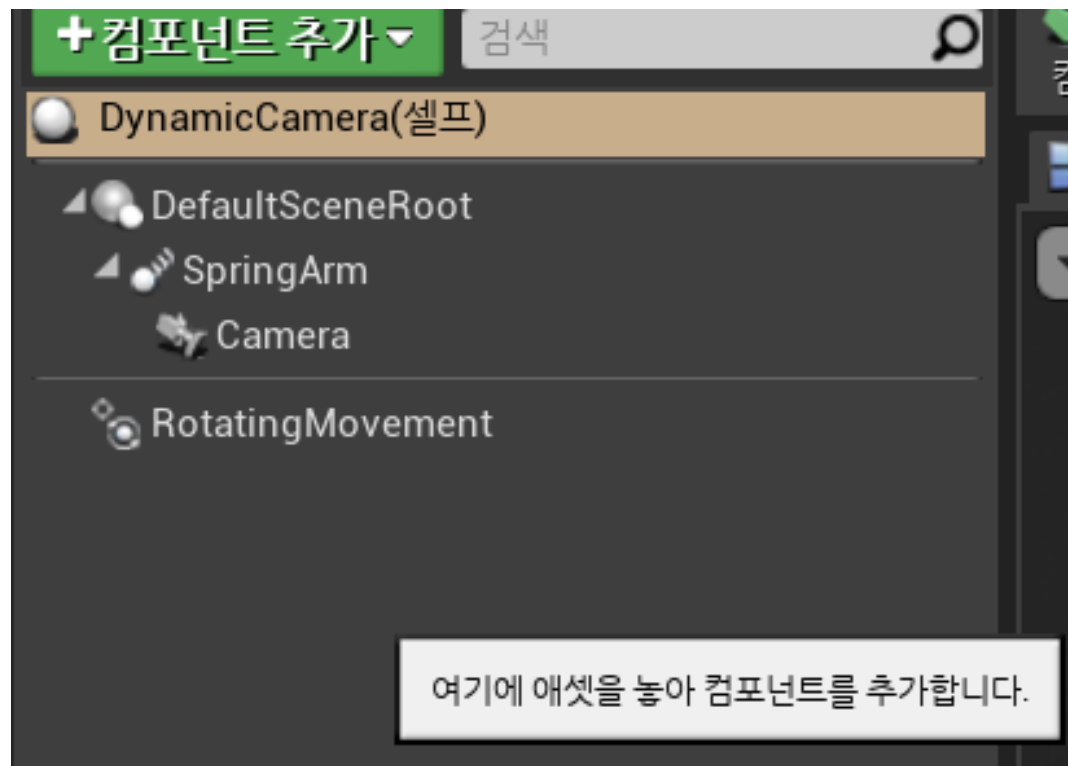


3.4 GetPlayerController를 호출하고, SetViewTargetWithBlend를 호출한다.

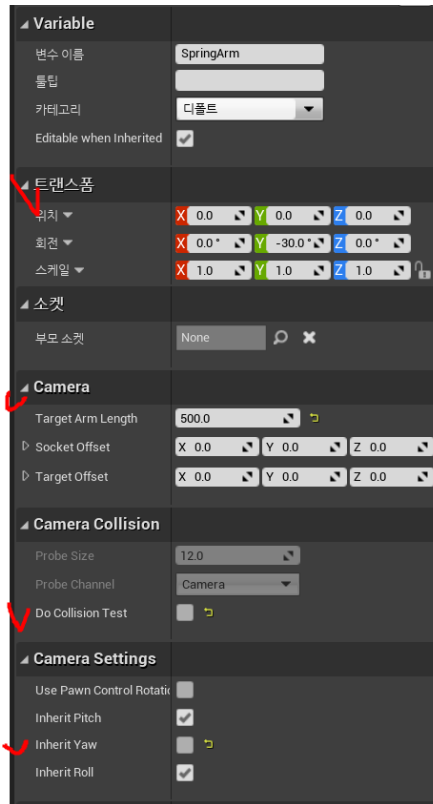




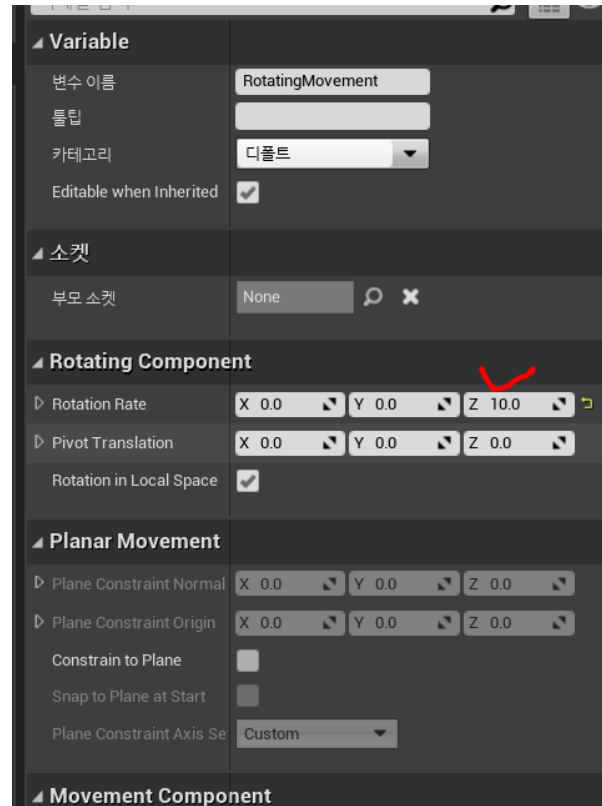
4. Actor를 상속받은 DynamicCarmera를 생성한다.



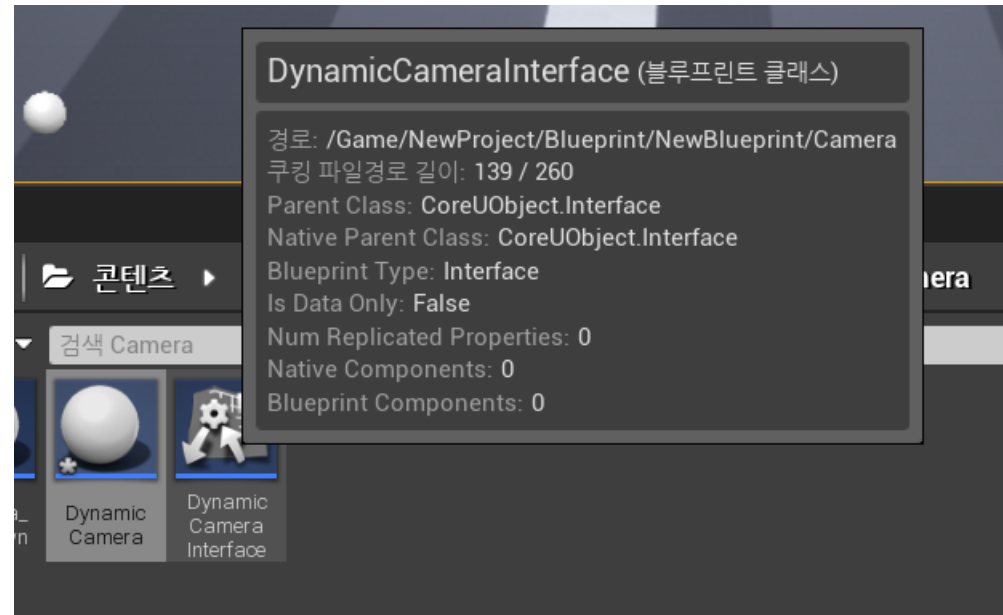
## 4.1 컴포넌트를 추가한다.



## 4.2 SpringArm의 설정을 변경한다.



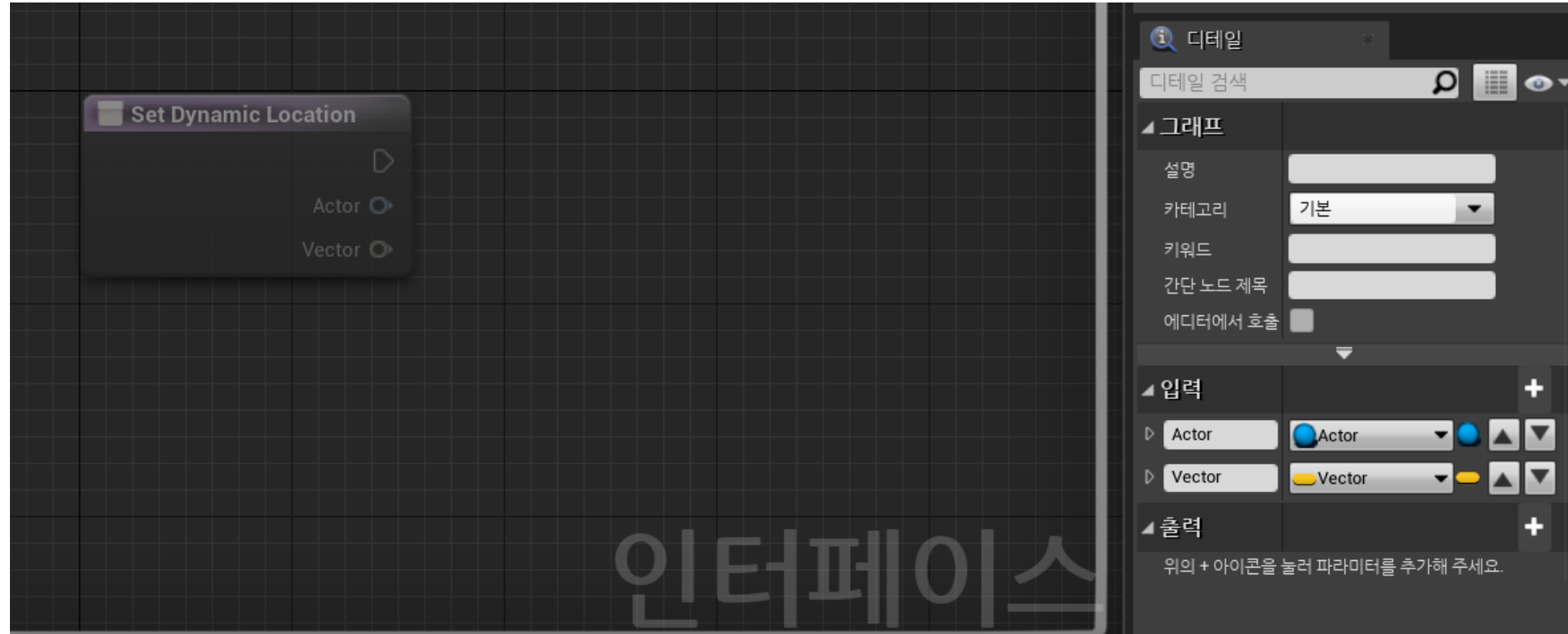
## 4.3 RotatingMovement의 RotatingComponent의 설정을 변경한다.



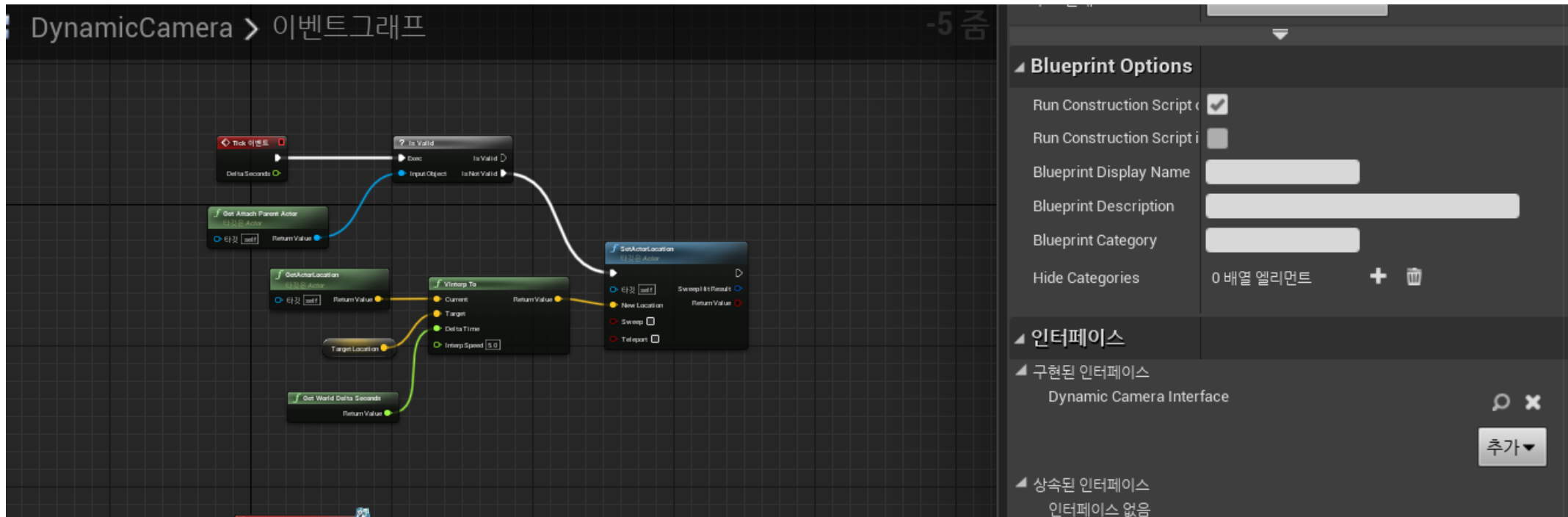
**5. Interface DynamicCameraInterface를 생성한다.**



5.1 SetOrbit 함수를 추가하고, 입력에 boolean 변수 IsOrbiting을 추가한다.

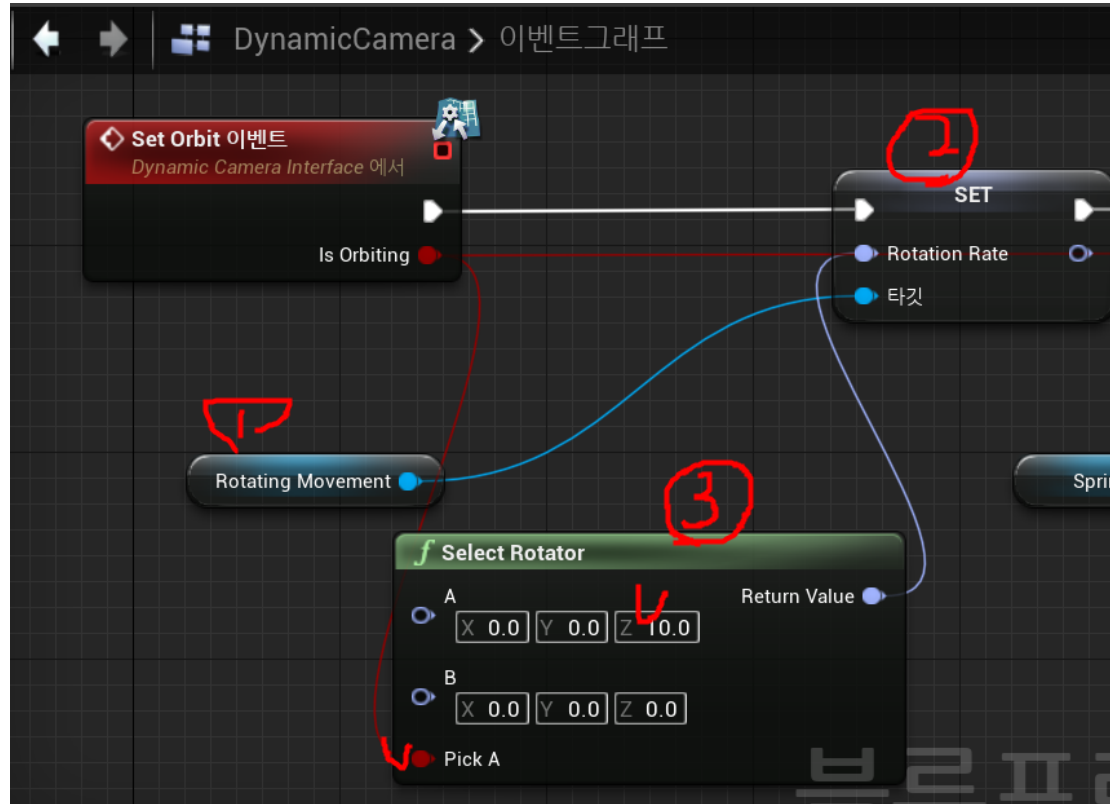


5.2 SetDynamicLocation을 생성하고 입력에 Actor 형 변수 Actor와 Vector 형 변수 Vector를 추가한다.

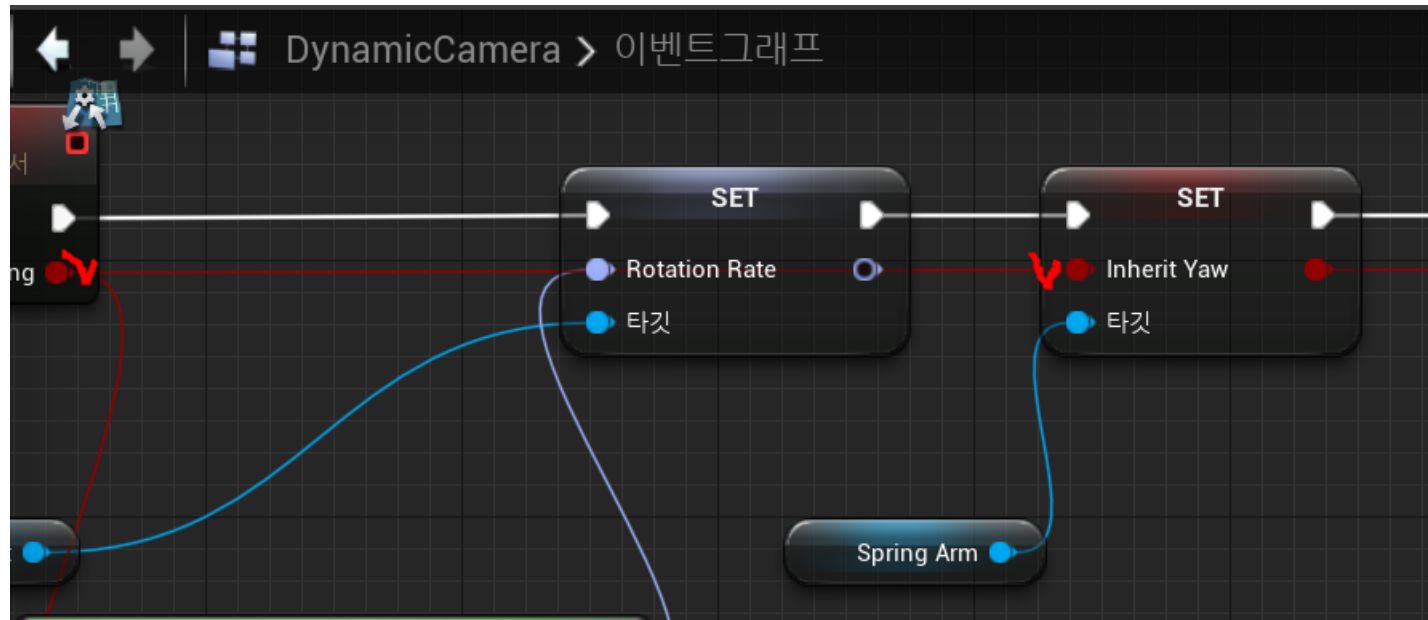


6. DynamicCamera에 들어가서 DynamicCameraInterface를 추가한다.

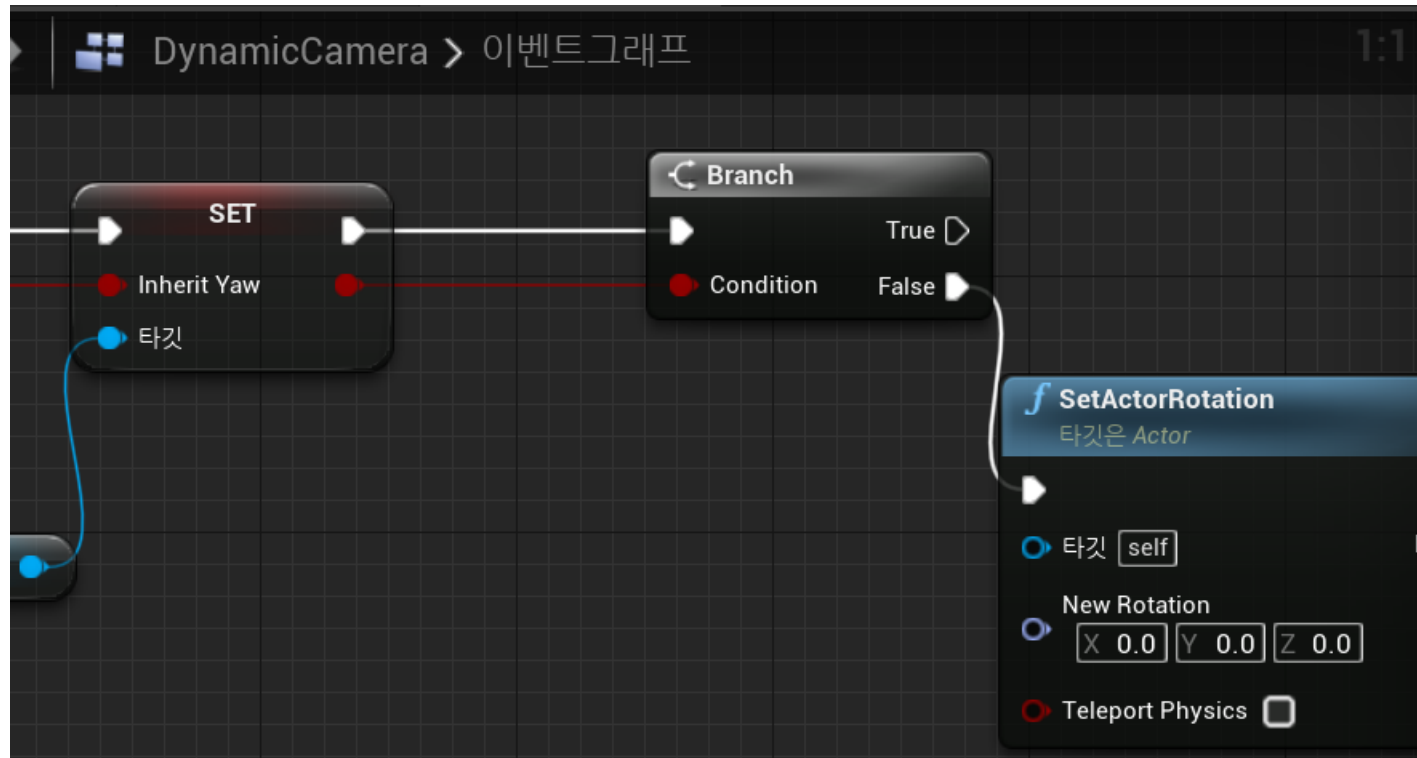




6.1 SetOrbit를 구현한다. 먼저 RotatingMovement의 RotationRate를 호출하고 SelectRotator의 값을 넣는다.



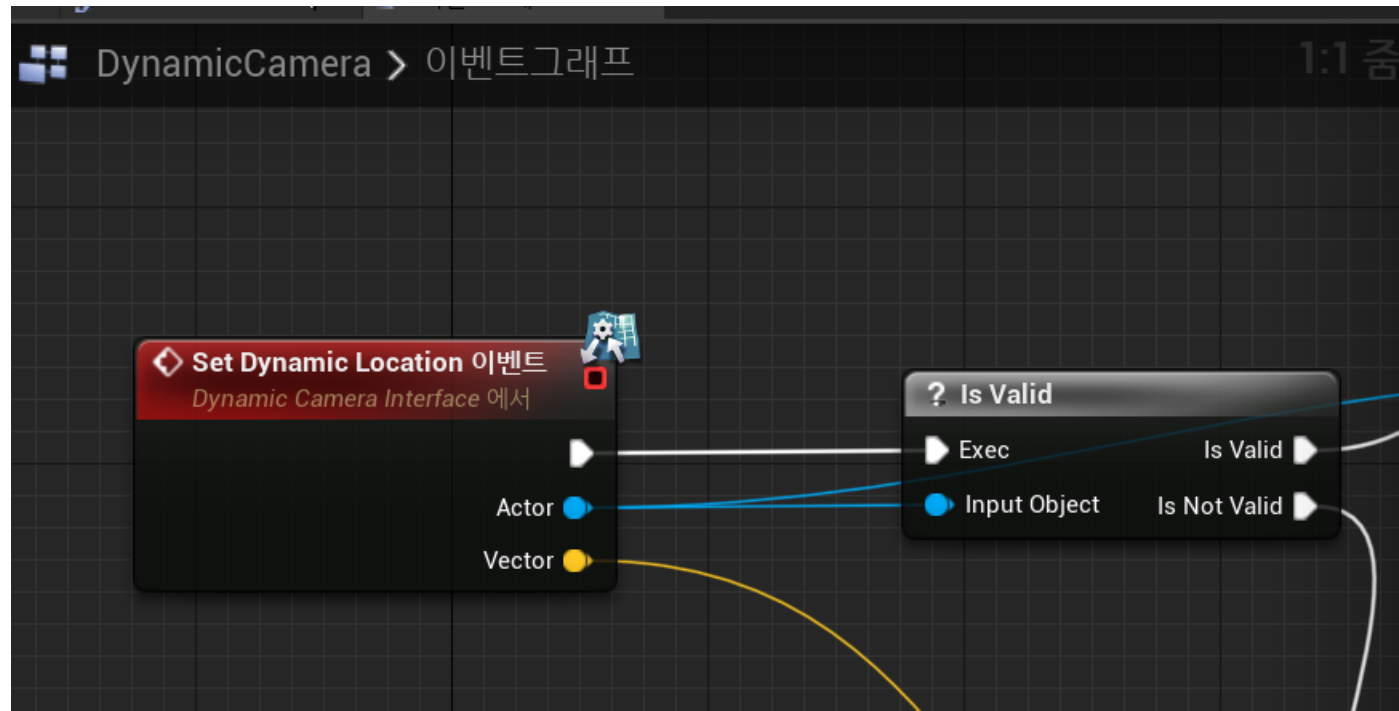
6.1 SpringArm의 InheritYaw SET 하고, IsOribiting에 연결한다.



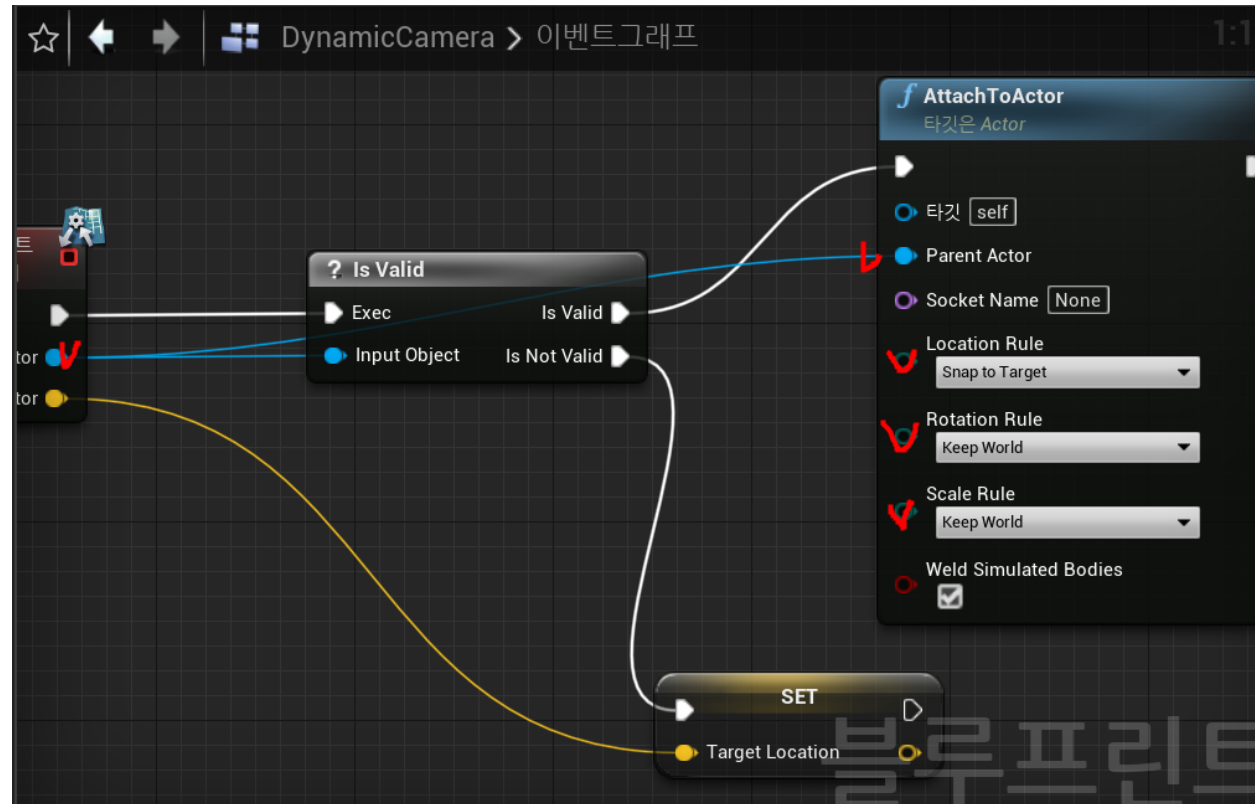
6.2 InheritYaw의 값을 Branch에 연결하고 False 이면 SetActorRotation을 호출한다.

Variable	
변수 이름	TargetLocation
변수 유형	Vector
인스턴스 편집가능	<input type="checkbox"/>
블루프린트 읽기 전용	<input type="checkbox"/>
툴팁	
3D 위젯 표시	<input type="checkbox"/>
스폰시 노출	<input type="checkbox"/>
프라이빗	<input type="checkbox"/>
시네마틱에 노출	<input type="checkbox"/>
카테고리	Default
리플리케이션	None
리플리케이션 조건	None
기본값	
Target Location	X 0.0 Y 0.0 Z 0.0
X	0.0
Y	0.0
Z	0.0

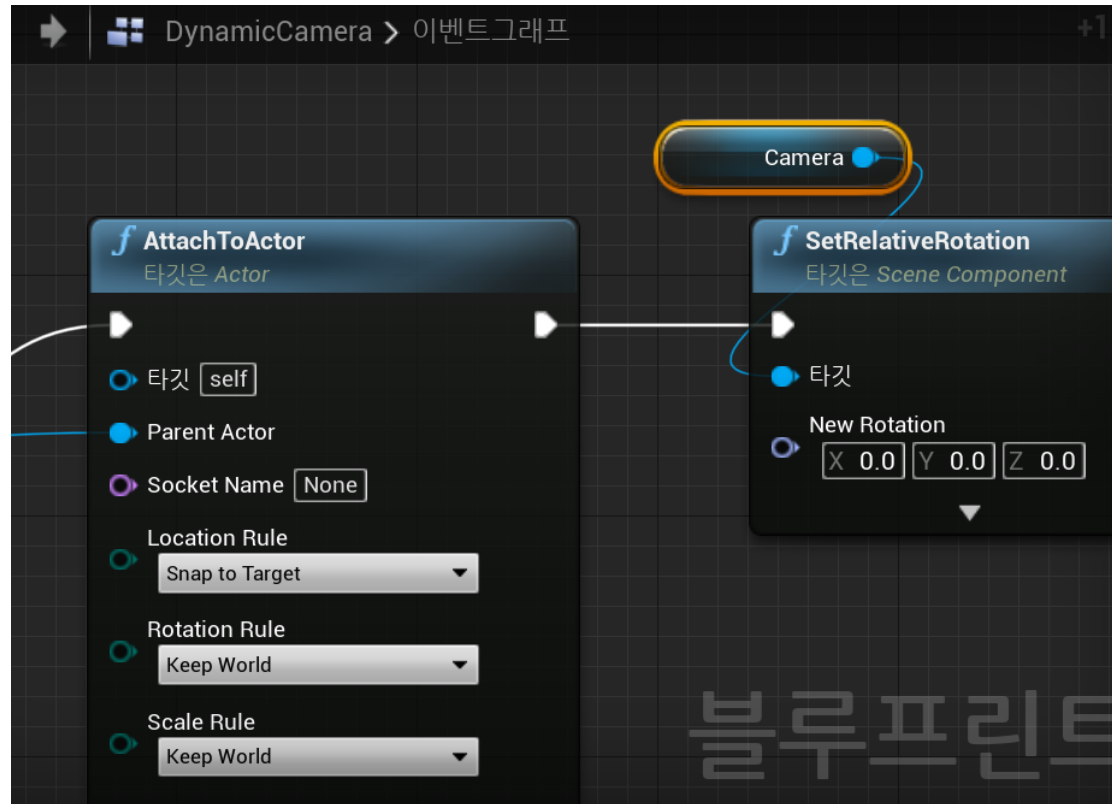
## 6.3 Vector 형 변수 TargetLocation을 추가한다



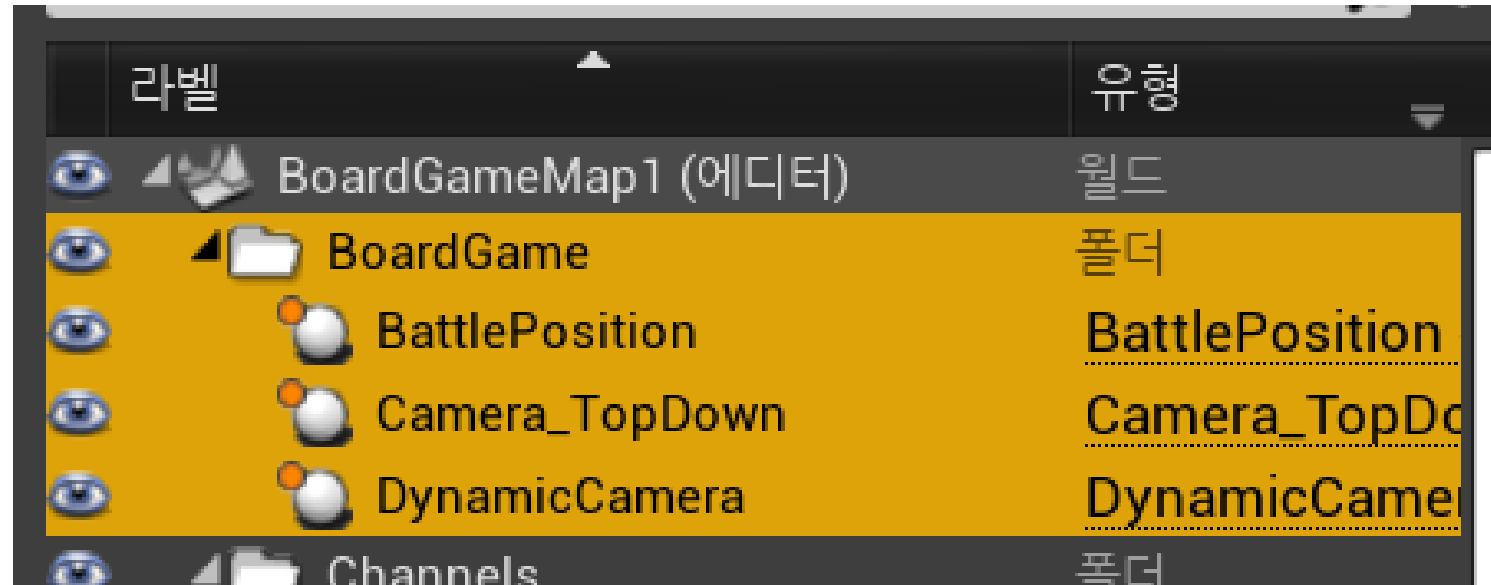
6.4 SetDynamicLocation을 구현한다. Actor가 유효한지 검사한다.



6.5 유효하면 AttackToActor를 호출하고, 유효하지 않으면 TargetLocation을 SET 한다.

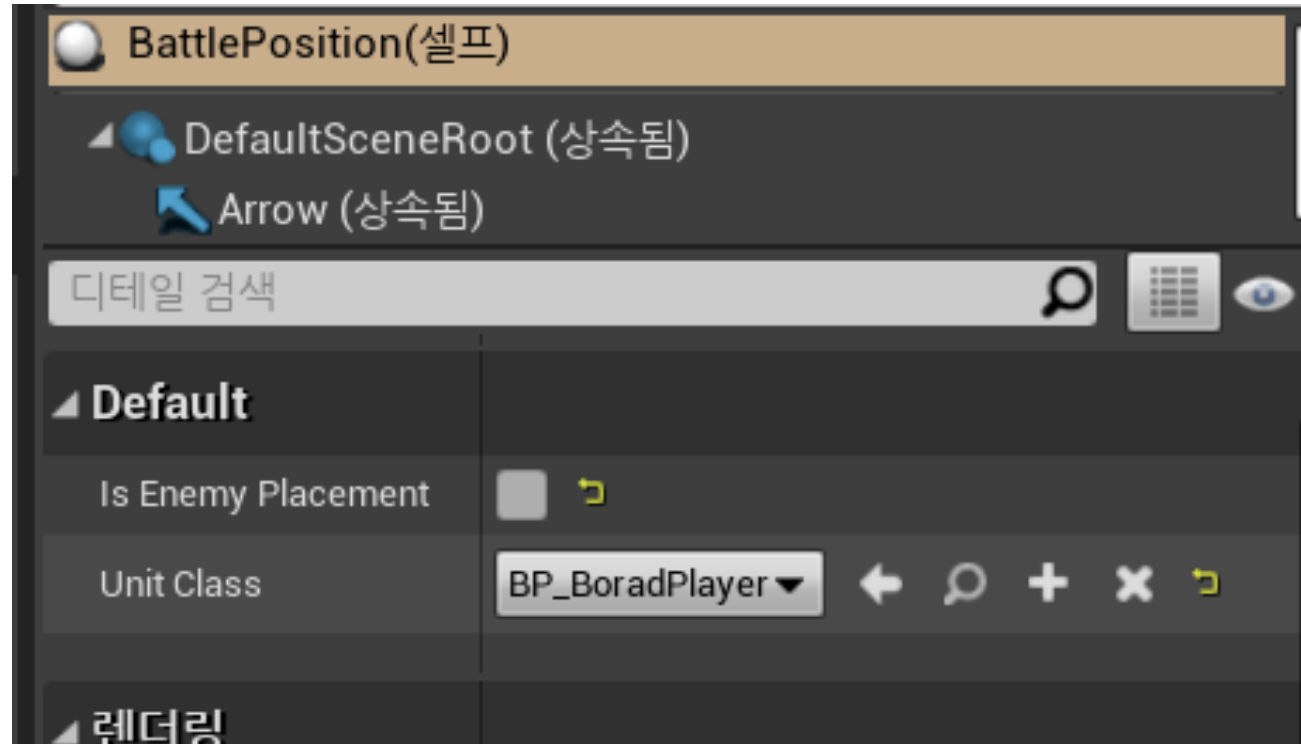


6.6 Camera의 SetRelativeRotation을 호출한다.



7. BattlePosition과 Camera\_TopDown, DynamicCamera를 레벨에 추가한다.





7.1 BattlePositon의 UnitClass에 BP\_BoardPlayer를 추가한다.