A cinematic screenshot from The Last of Us Part II. A character, Ellie, is seen from behind, standing on a rocky, desolate landscape. She is looking towards a ruined city with tall, jagged spires and skeletal remains of buildings. The sky is a mix of dark, heavy clouds and a bright, orange-hued sunset or sunrise, creating a dramatic and somber atmosphere. The text '게임 엔진' is overlaid in the upper right area.

게임 엔진

LEC 06 레벨 블루프린트



한국공학대학교
TECH UNIVERSITY OF KOREA

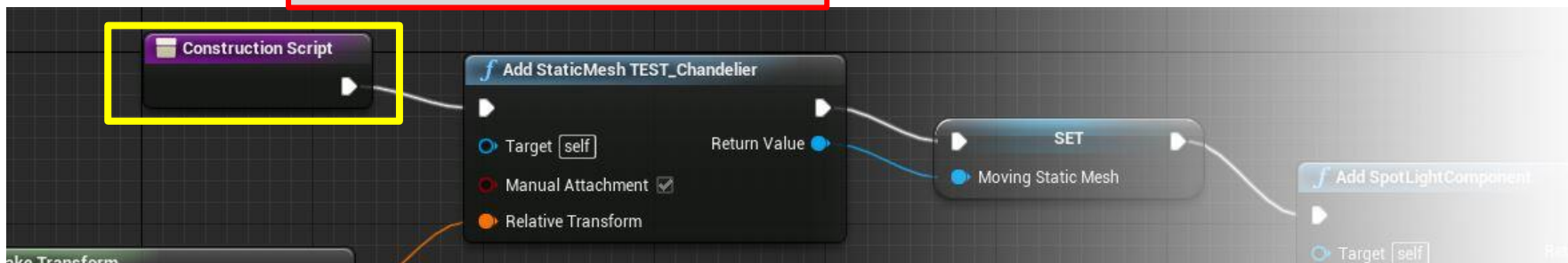
이대현 교수

LEC 05 복습

컨스트럭션 스크립트(Construction Script)

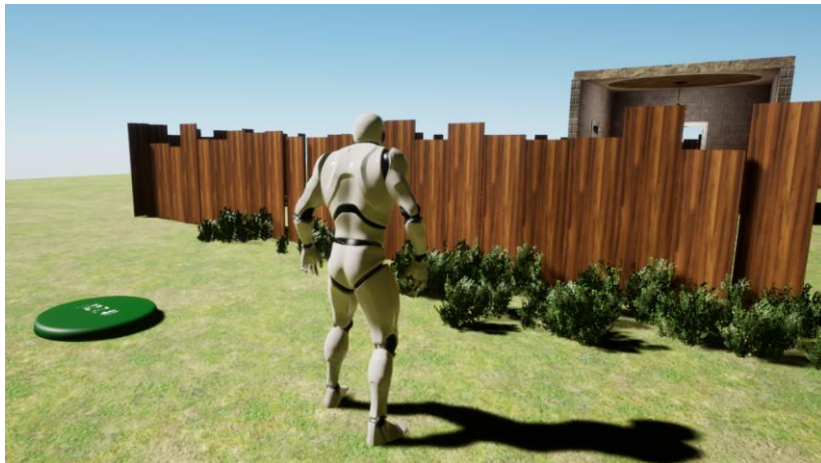
- 블루프린트 클래스의 구성을 프로그래밍적으로 하기 위한 스크립트.
 - 블루프린트 안에 컴포넌트들을 효과적으로, 자동적으로 담음.

Construction Script 노드를
시작점으로 하는 함수를 작성



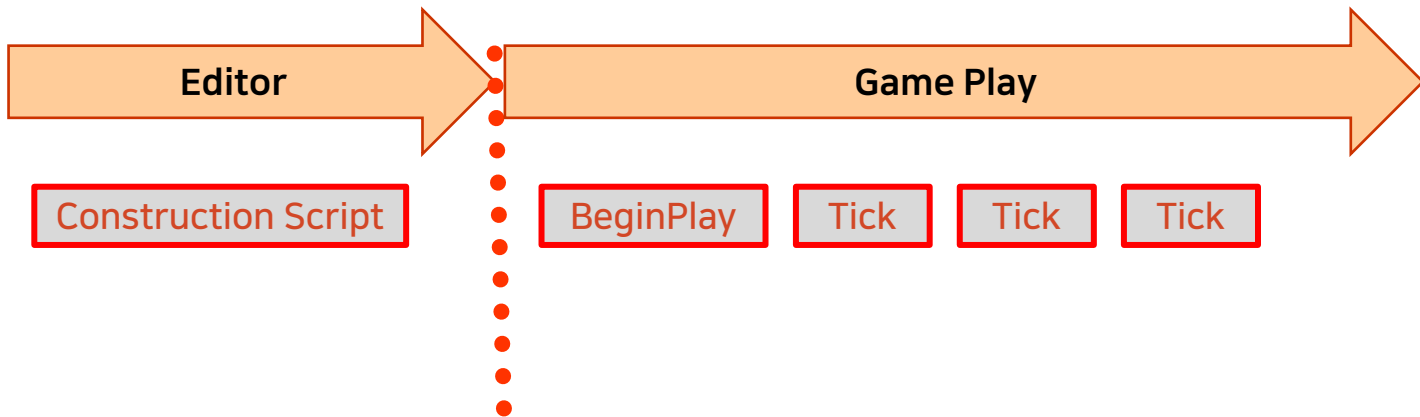
컨스터렉션 스크립트 활용

- 대량의 컴포넌트들로 구성된 블루프린트 클래스의 개발
 - 예) 수풀, 담장
- 유연한 블루프린트 클래스의 개발
 - 각기 다른 속성을 갖는 액터들을 생성할 수 있음.
 - 예) 키가 다른 적군 NPC
- 게임 플레이 시작 전에 준비되어야 할 요소를 갖추는데 활용
- 레벨 구성의 자동화



컨스트럭션 스크립트의 실행

- 컨스트럭션 스크립트는 에디터 상에서만 실행되고 반영됨.
 - 액터가 생성되는 시점(레벨 상에 액터를 끌어다가 놓을 때), 또는 액터의 속성 변경이 일어날 때(이동, 회전 등등)마다 실행됨.
 - 실제로 게임 플레이가 시작되면, BeginPlay event 가 호출되어, 기본적인 초기화를 수행하고, 이어서 게임 루프에서 반복적으로 계속해서 Tick event를 호출함.



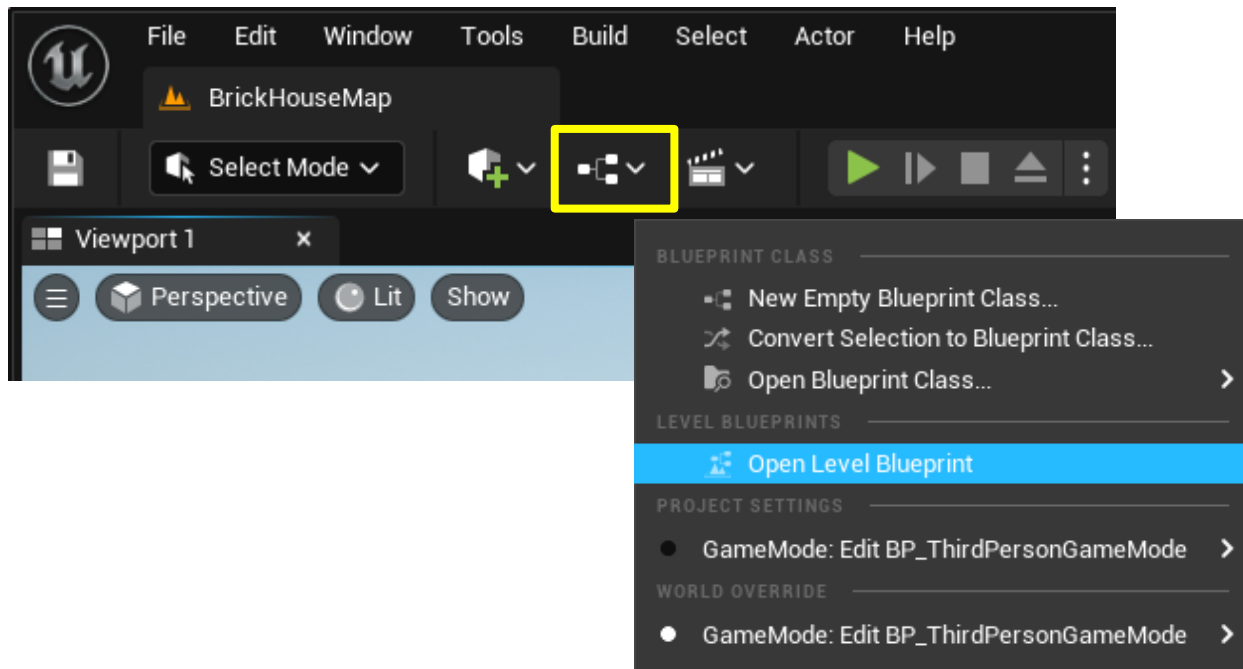
목차

- 레벨 블루프린트 개요
- 액터 레퍼런싱 방법
- 실습 - 금화 획득 게임 제작

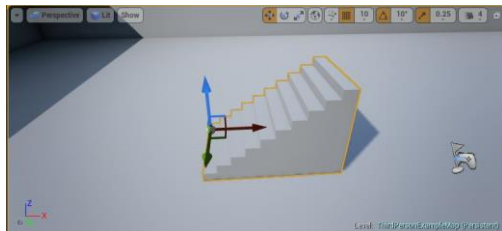
레벨 블루프린트

- Level 상에 존재하는 액터들간의 상호 작용을 스크립트로 처리.
- 각 Level 마다, 한 개의 레벨 블루프린트 존재.
- 액터들을 “직접적으로” 제어할 수 있음.(C++에서 전역 변수를 쓰는 것처럼).
- 예전 언리얼 버전 (3.0 이전)에서는 자주 사용됐음.
- 현재는 빠르게 결과를 확인하고 싶을 때(프로토타입, 액터 임시 테스트 등) 사용.
- 키보드 또는 마우스 입력 등을 직접적으로 처리할 수 있음.

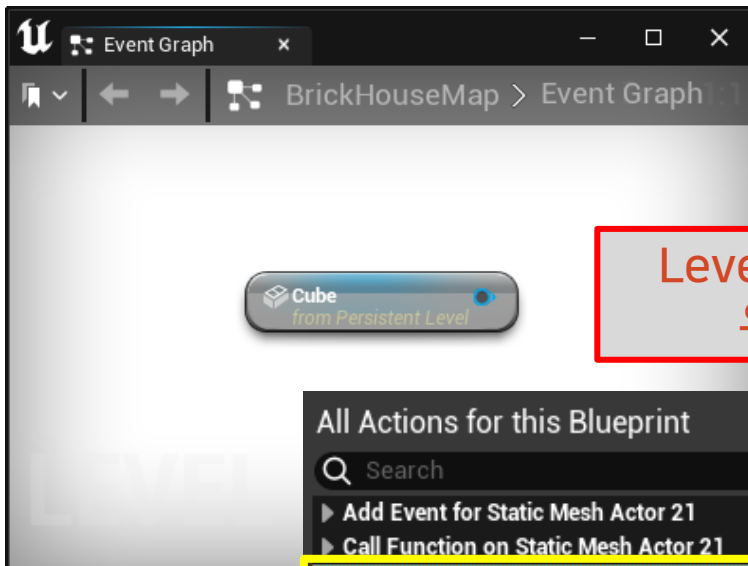
Level Blueprint Open



액터 레퍼런싱 방법 #1



뷰포트에서 액터 선택



Level Blueprint 에서
우클릭 & 선택

All Actions for this Blueprint

☒ Context Sensitive

Search

▶ Add Event for Static Mesh Actor 21

▶ Call Function on Static Mesh Actor 21

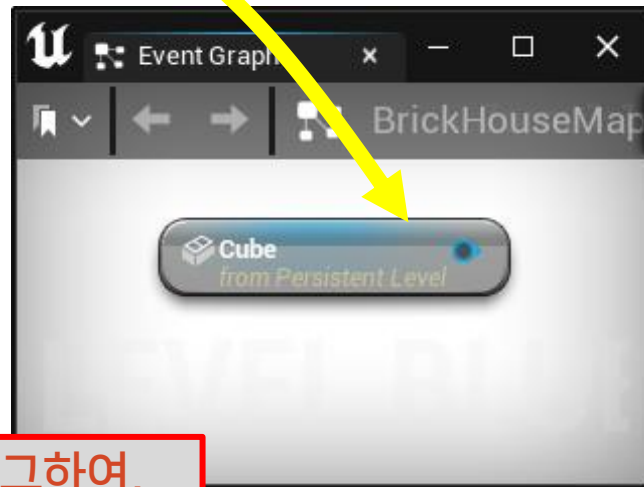
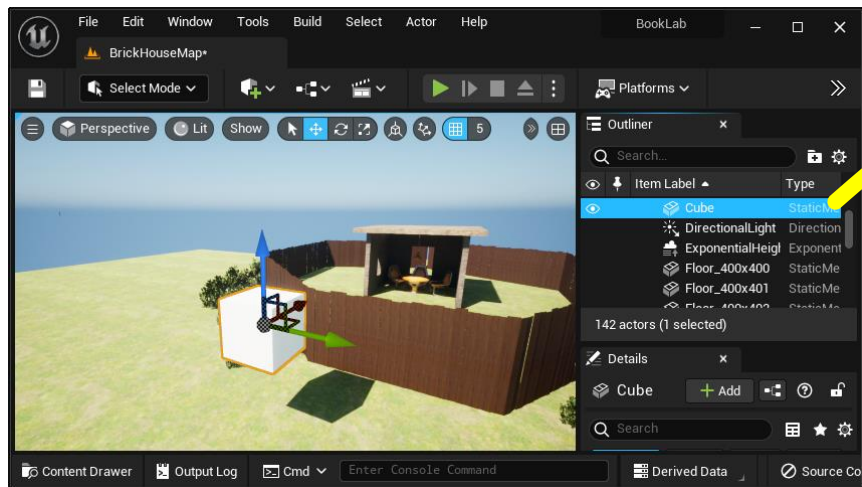
 Create a Reference to Cube

BP

main

BP

액터 레퍼런싱 방법 #2



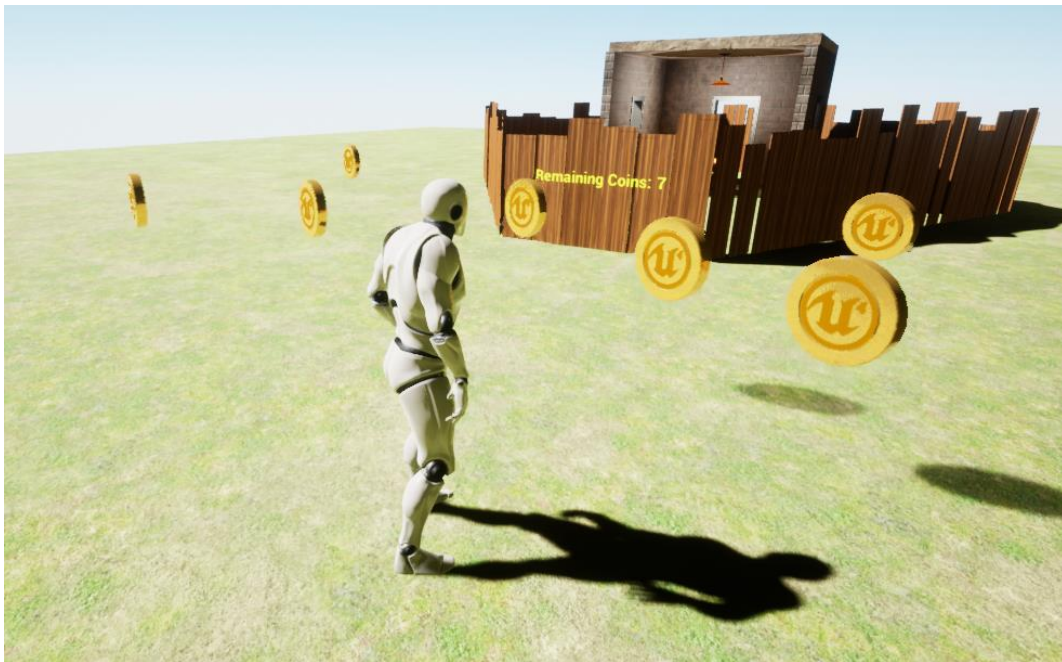
World Outliner 에서 액터를 드래그하여,
레벨 블루프린트 이벤트 그래프 내로 이동



실습 골드 코인 획득

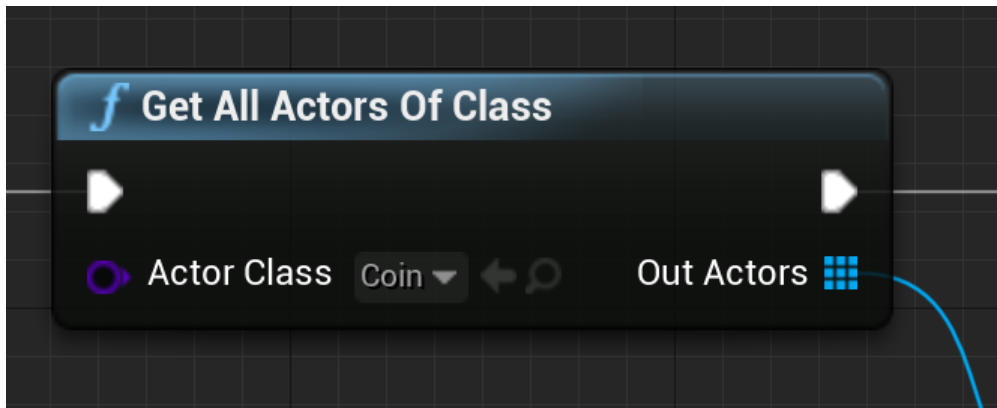
실습 목표

- 코인 획득 후 문을 부수고 집안으로 들어가는 캐릭터 구현.
- 레벨 블루프린트의 효과적 활용 방법.



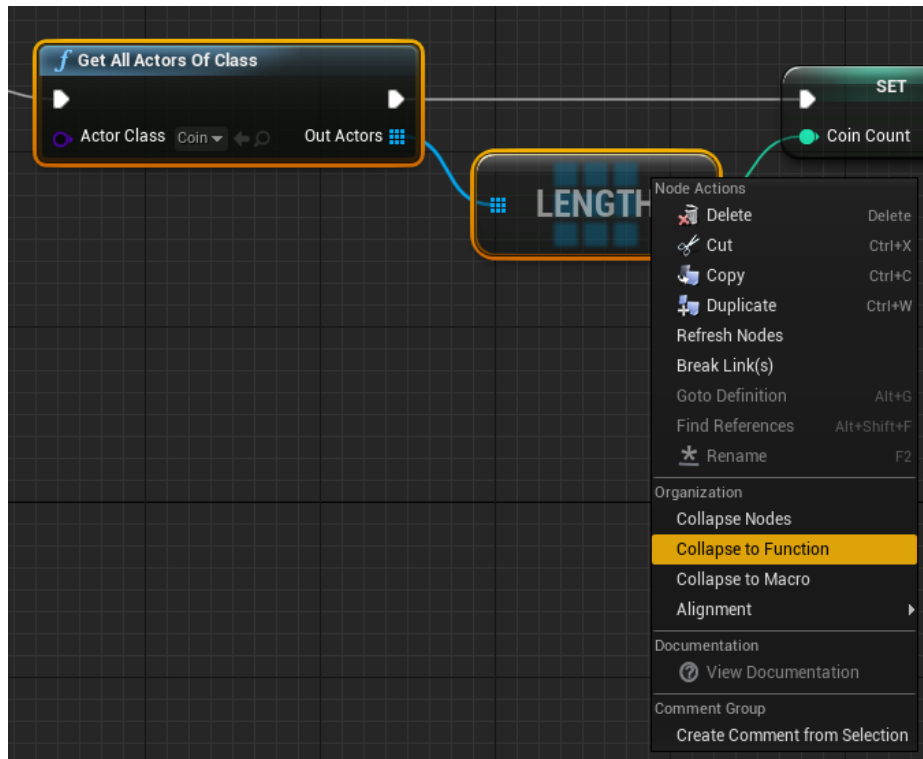
Get All Actors of Class 노드

- 레벨 안에 들어있는 특정한 클래스의 액터들을 모두 가져옴.
- 입력: 클래스 이름
- 출력: 액터들의 “배열”



함수로 접기 (Collapsing to Function)

- 자주 사용되는 노드들을 묶어서 함수로 변환.



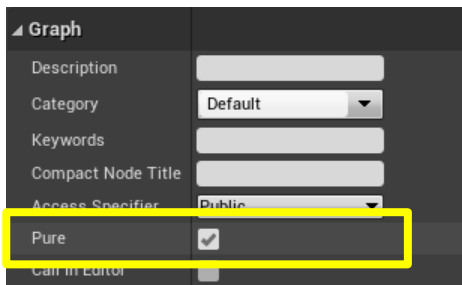
순수(pure) 함수 vs. 비순수(impure) 함수

■ Pure 함수

- 다른 액터들이나 변수의 상태를 변경하지 않는 함수.
- 현재의 상태에 기반하여, 계산하거나 처리하는 함수.

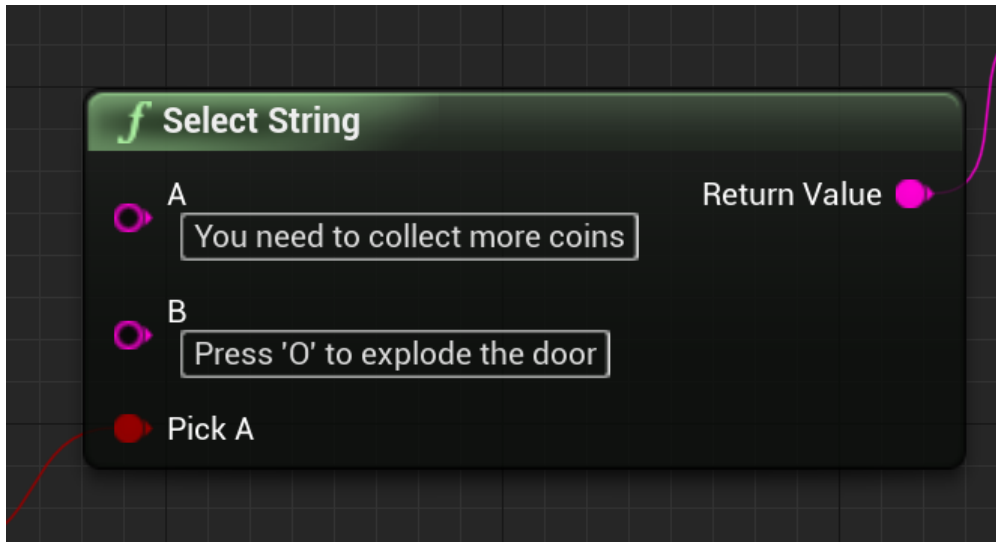
■ Impure 함수

- 상태를 자유롭게 변경하는 함수.



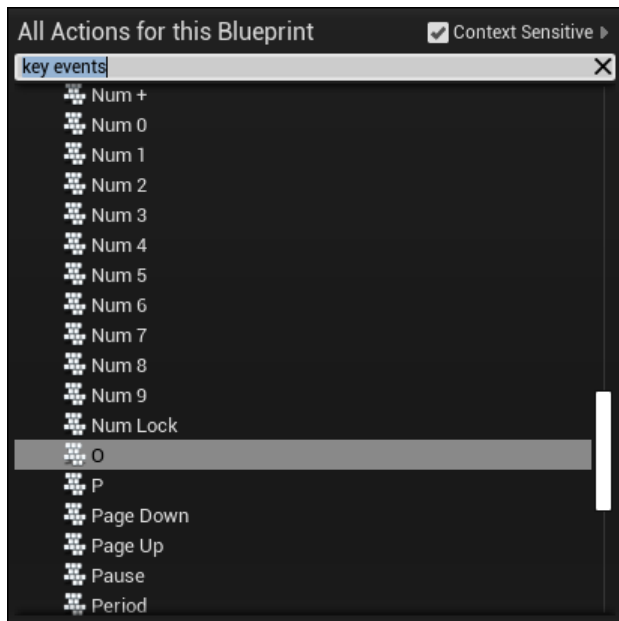
Select String 노드

- 두개의 문자열 중 하나 선택

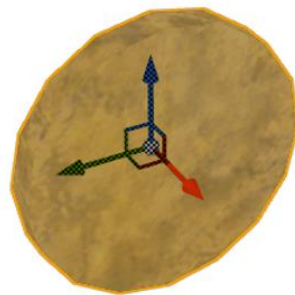
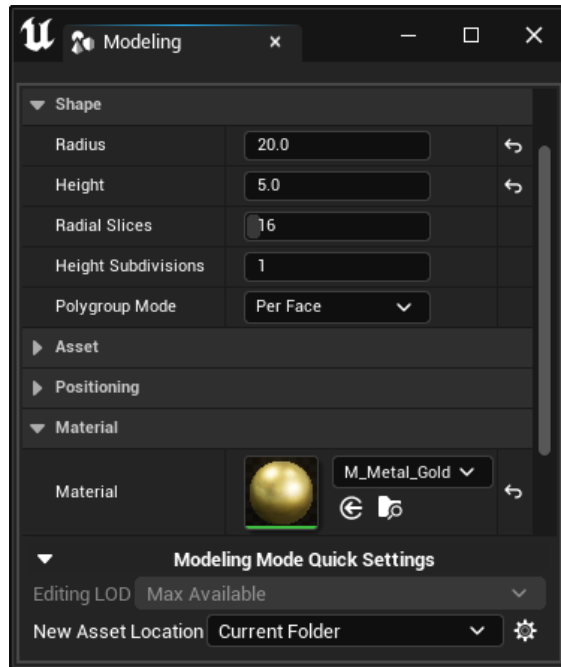
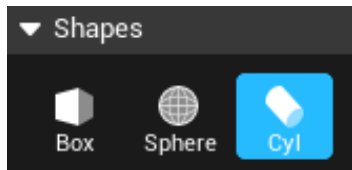


키 입력 체크

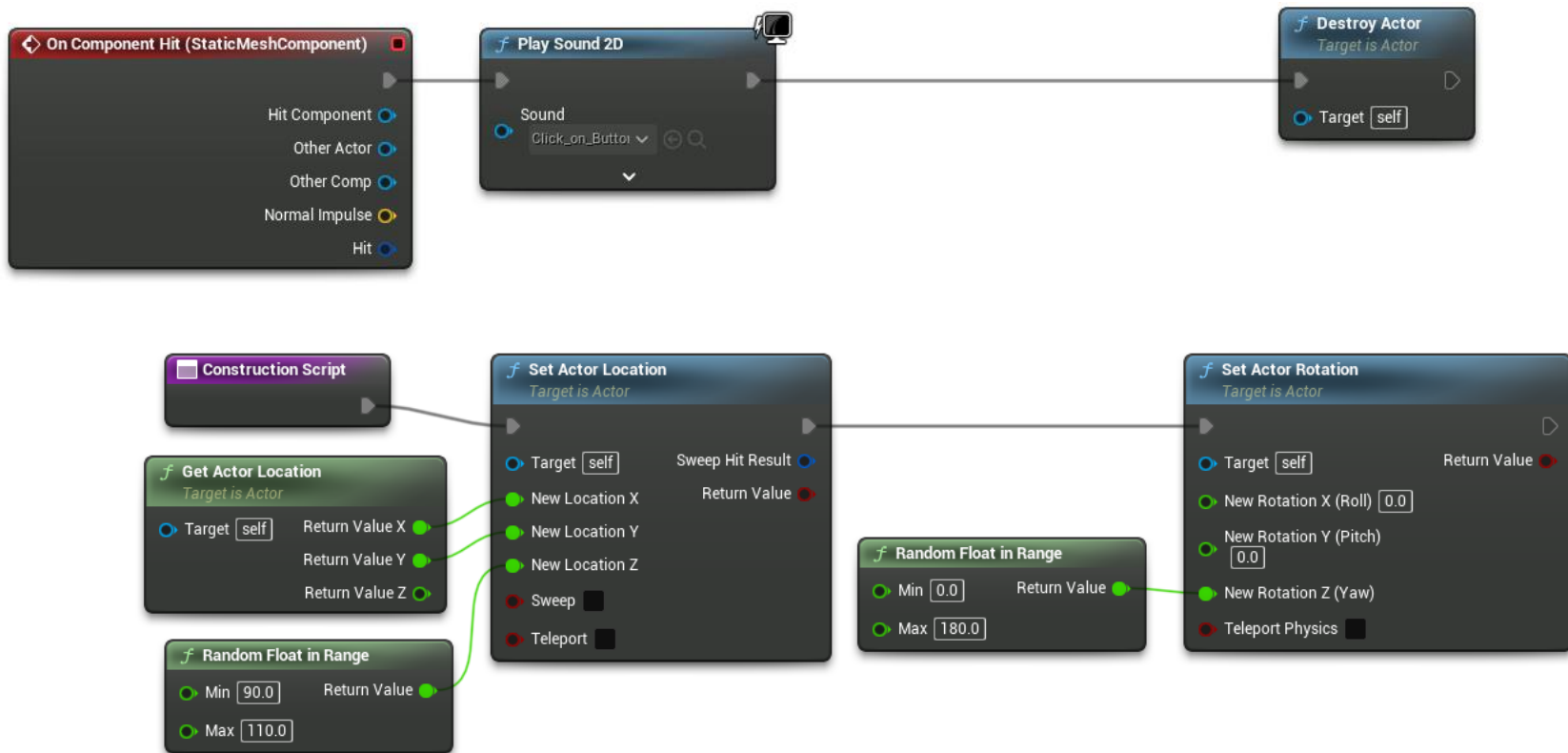
- 레벨 블루프린트 상에서는 키 이벤트를 쉽게 처리할 수 있음.



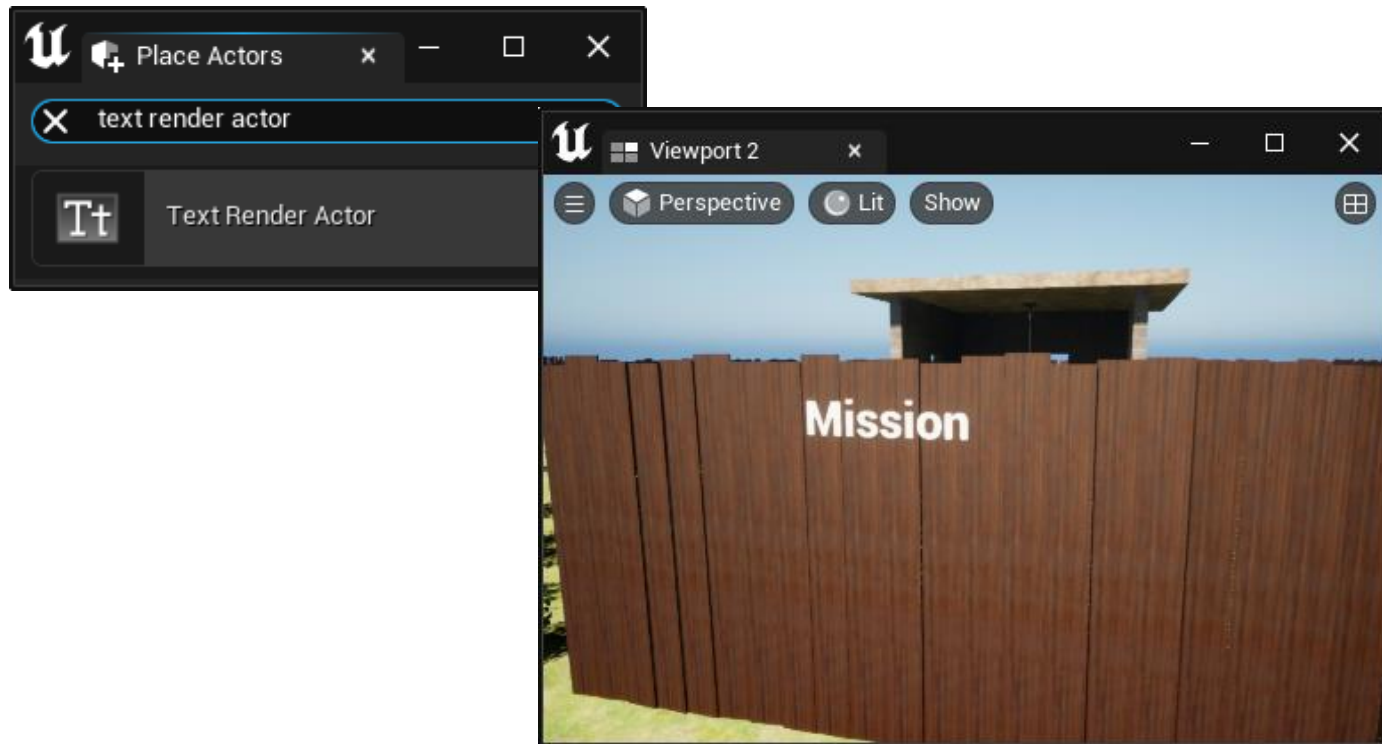
금화 3D 모델



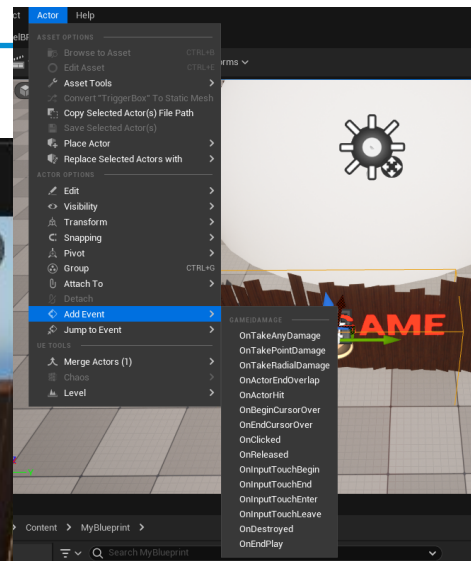
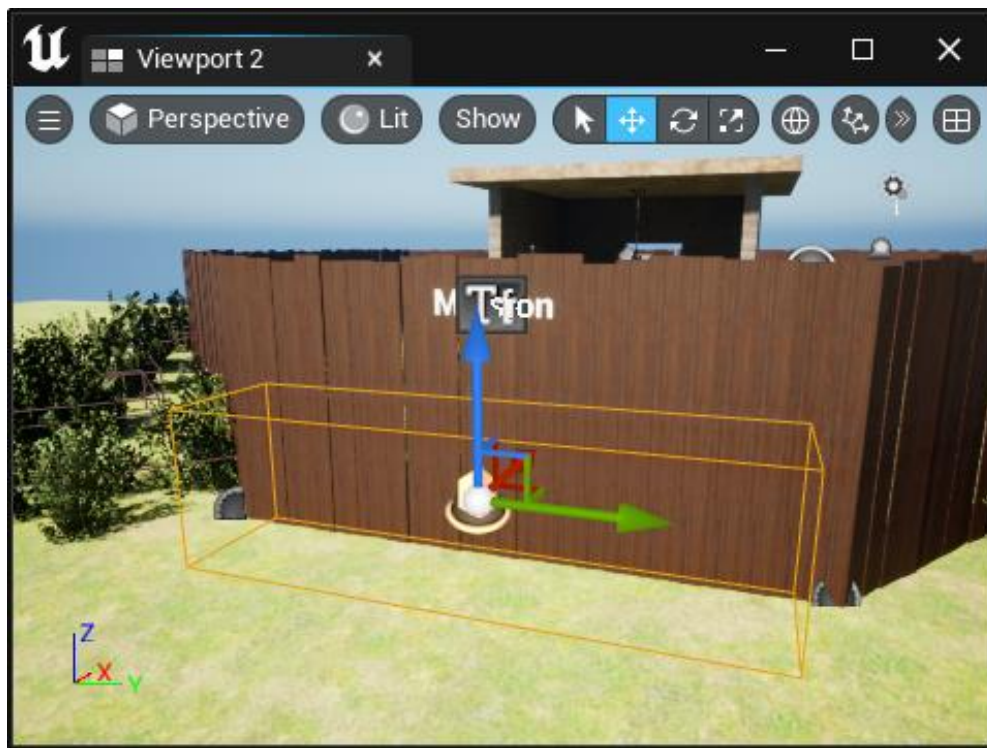
금화 Blueprint



Mission Text 배치



문 앞에 트리거 박스 배치



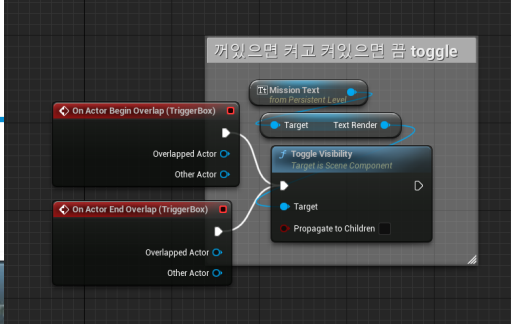
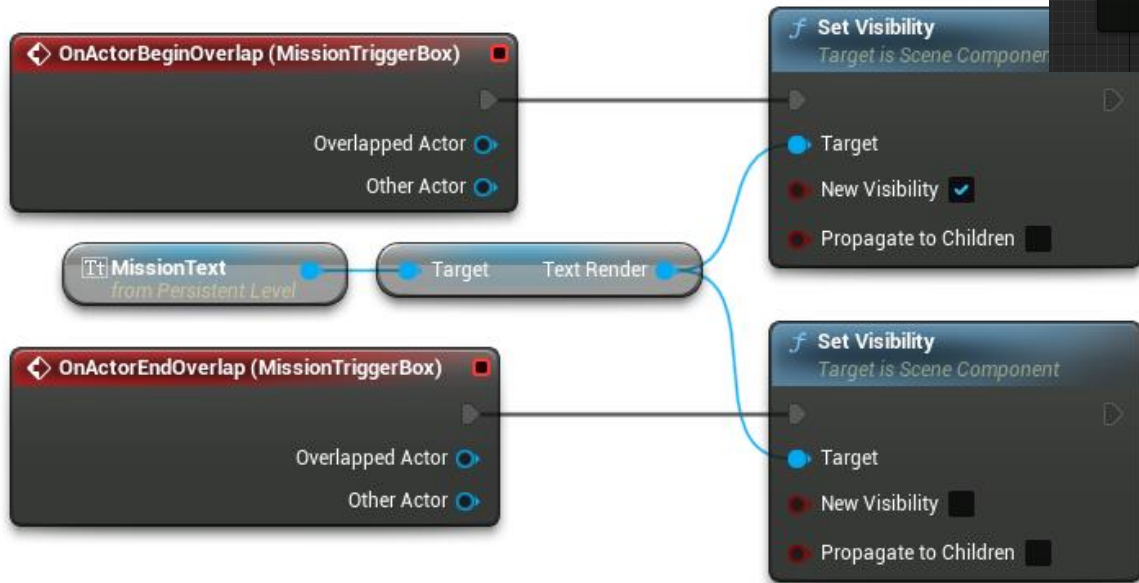
Actor

가 가 .
가

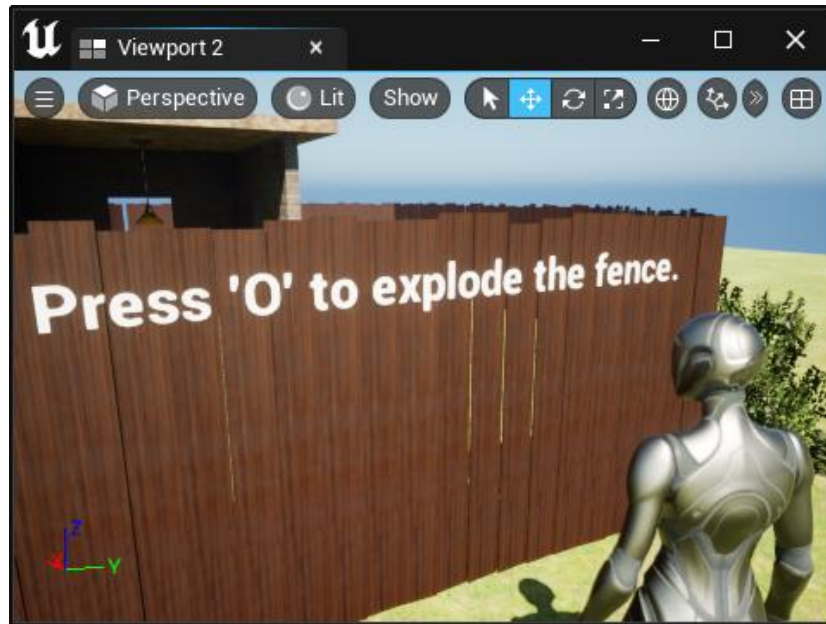
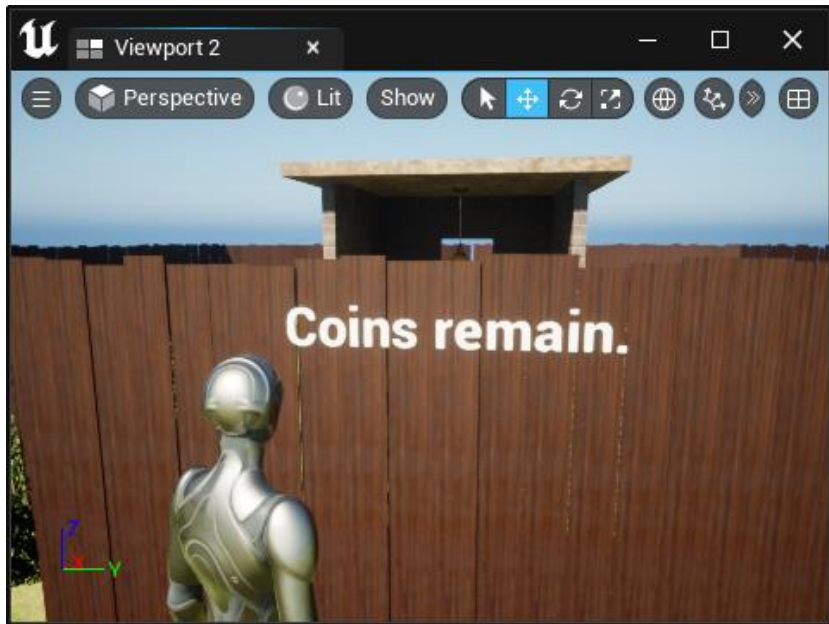
Mission Text 표시

toggle

가



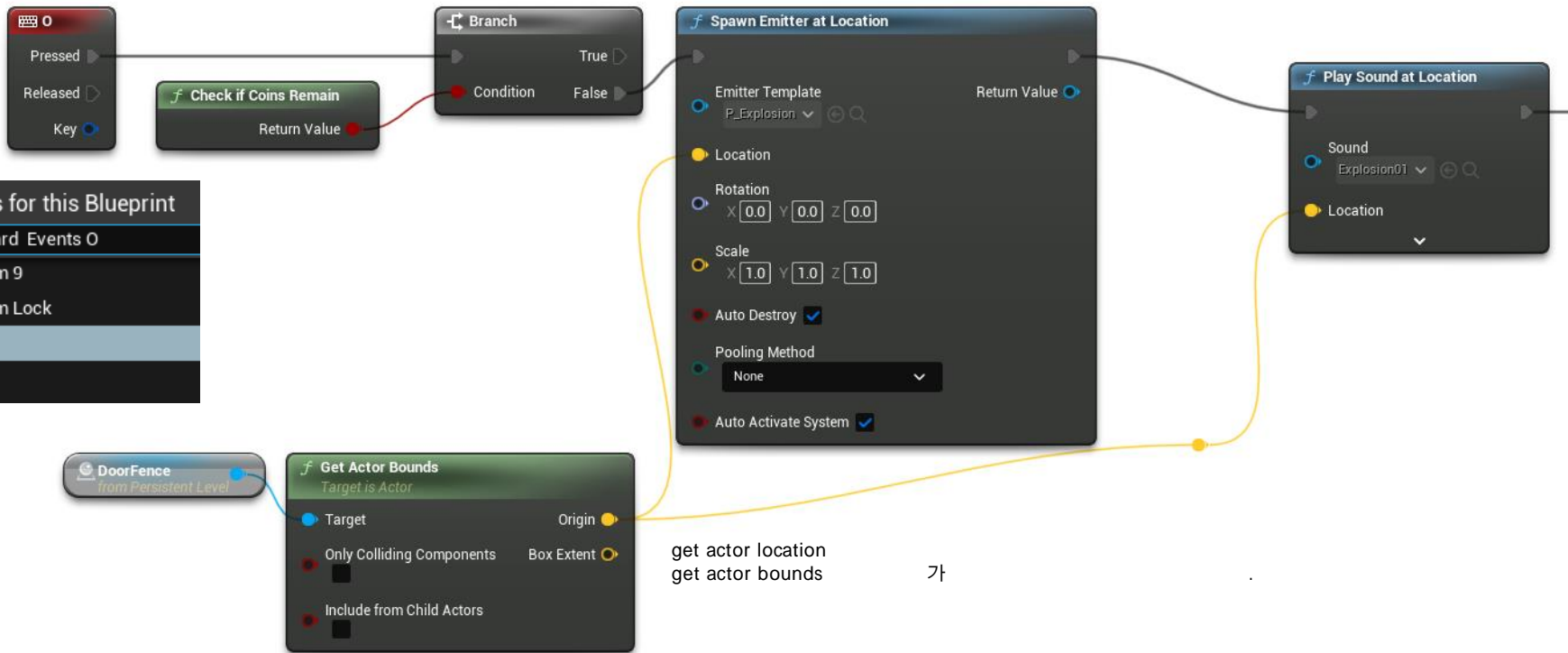
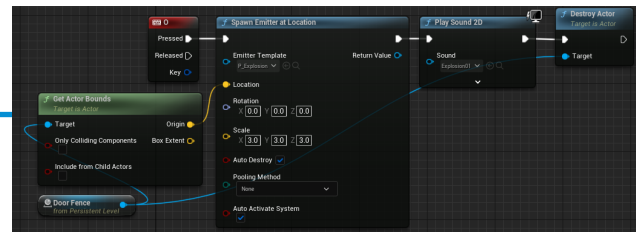
코인 개수에 따른 mission text 표시



코인 개수에 따른 mission text 표시



'O' 키를 눌러서 문 폭파

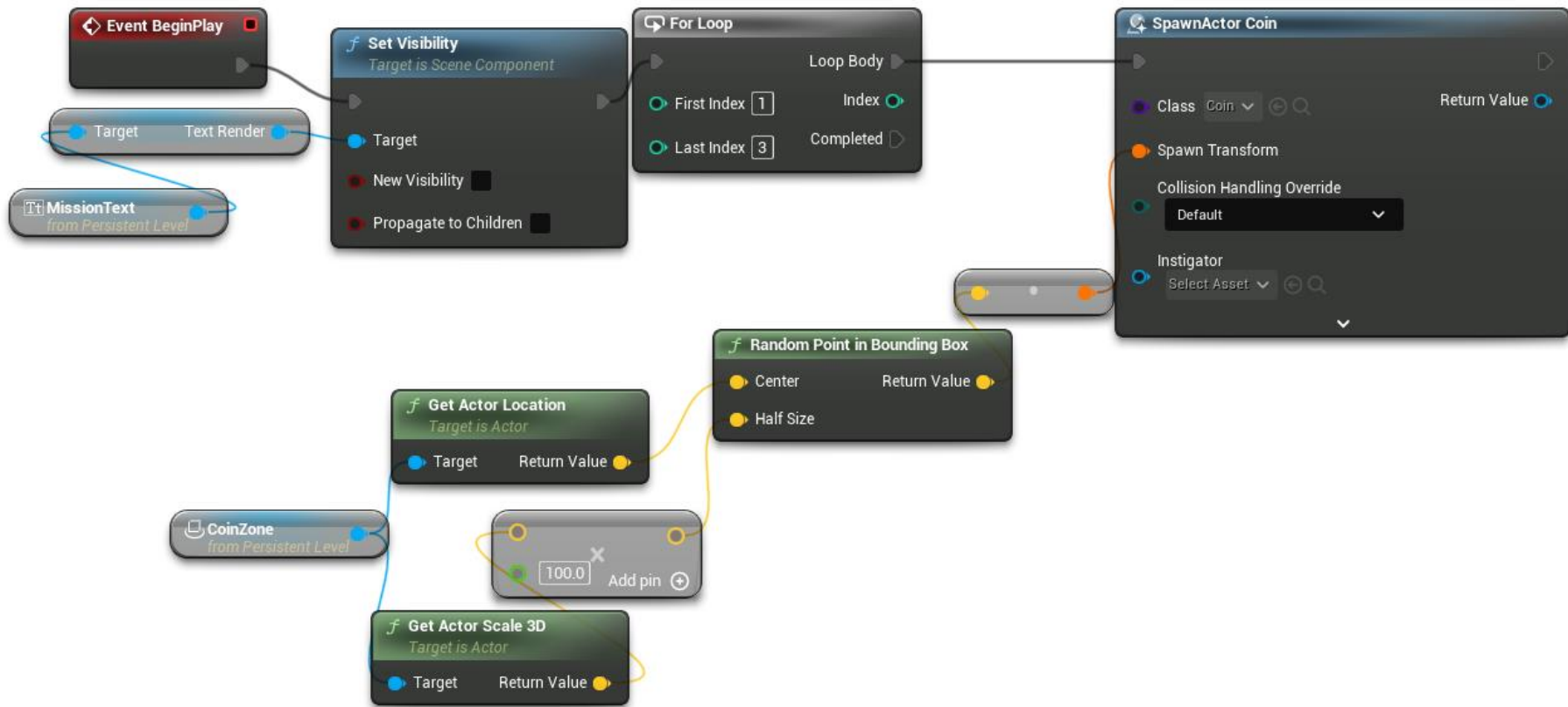


get actor location
get actor bounds

가



랜덤 코인 생성과 배치



코인 개수 체크 함수

