A character with a ponytail and a backpack stands on a rocky shore, looking out at a dramatic volcanic landscape under a sunset sky. The scene is filled with jagged rock formations and distant structures.

게임 엔진

LEC 07 게임플레이 프레임워크



한국공학대학교
TECH UNIVERSITY OF KOREA

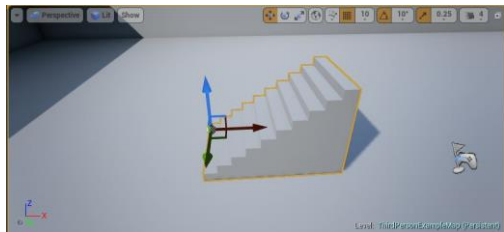
이대현 교수

LEC 06 복습

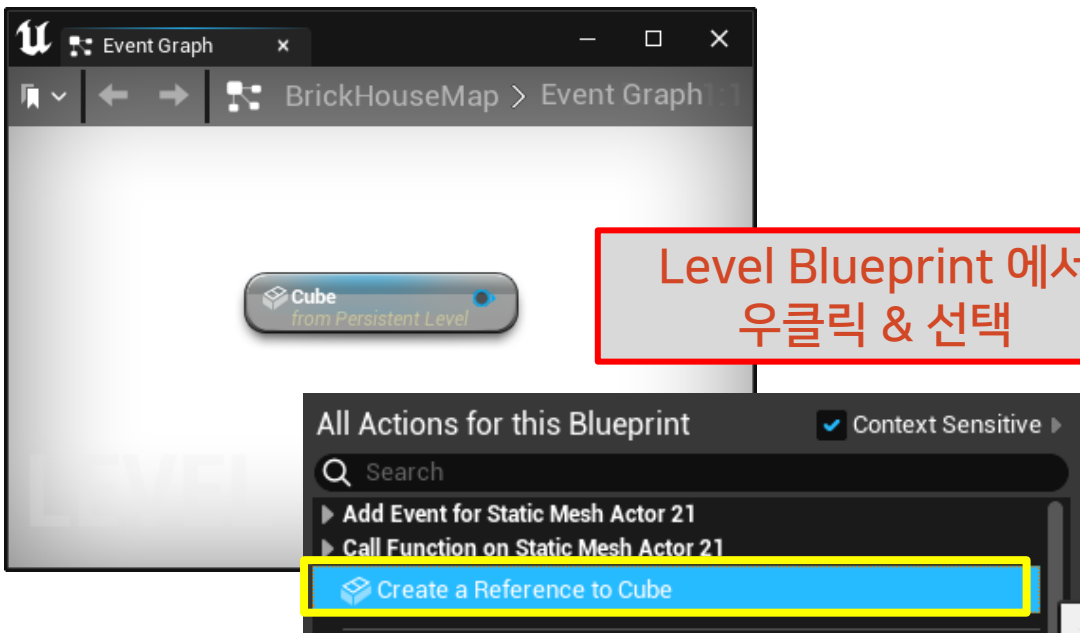
레벨 블루프린트

- 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.

액터 레퍼런싱 방법 #1



뷰포트에서 액터 선택



Level Blueprint 에서
우클릭 & 선택


All Actions for this Blueprint

☒ Context Sensitive

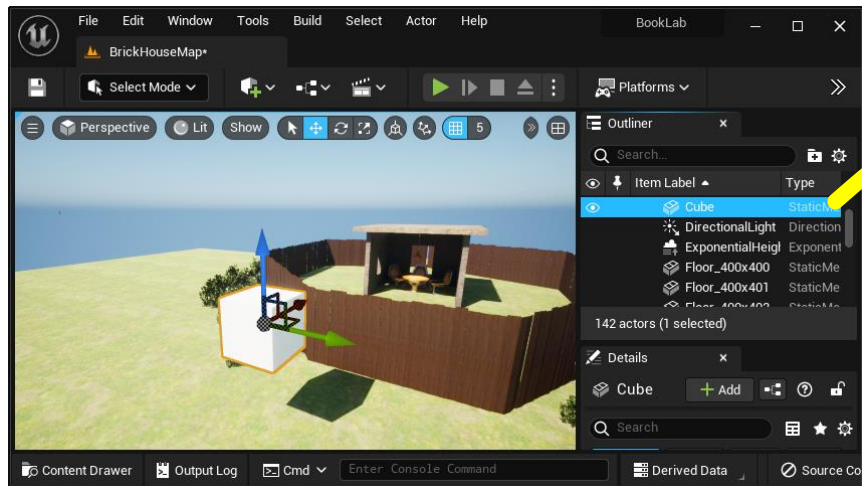
Search

▶ Add Event for Static Mesh Actor 21

▶ Call Function on Static Mesh Actor 21

 Create a Reference to Cube

액터 레퍼런싱 방법 #2

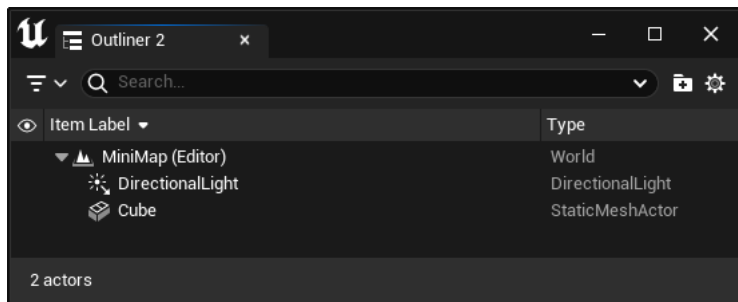


World Outliner 에서 액터를 드래그하여,
레벨 블루프린트 이벤트 그래프 내로 이동

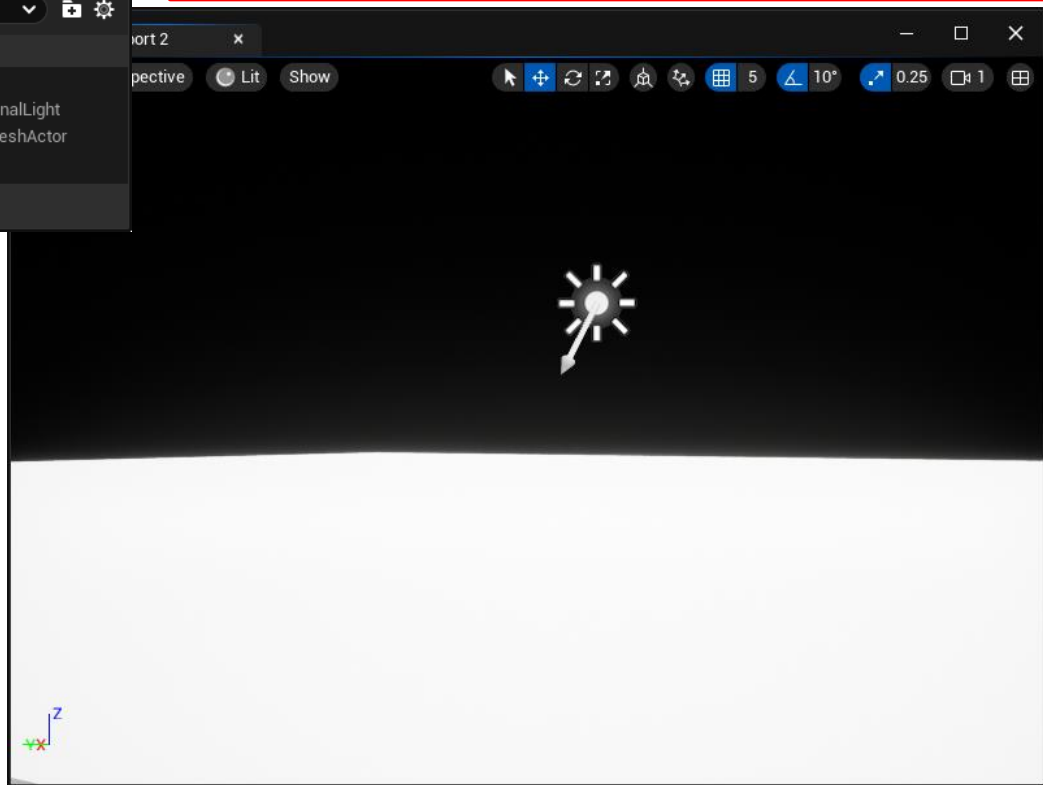
목차

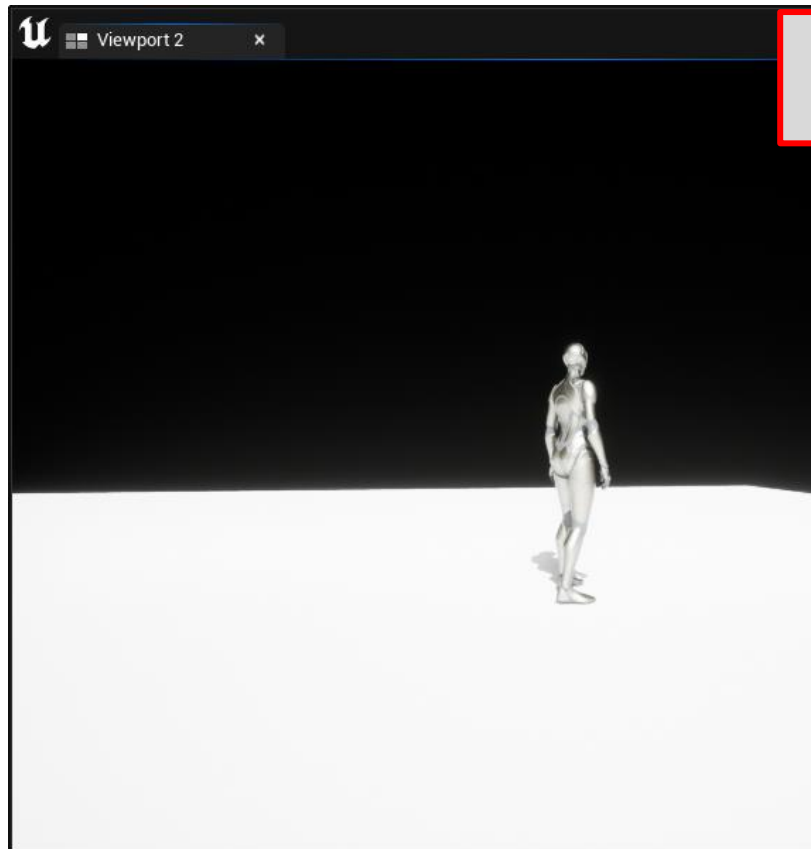
- 게임 플레이 프레임워크
- Default Pawn 클래스
- Aircraft 조종 실습

MiniMap

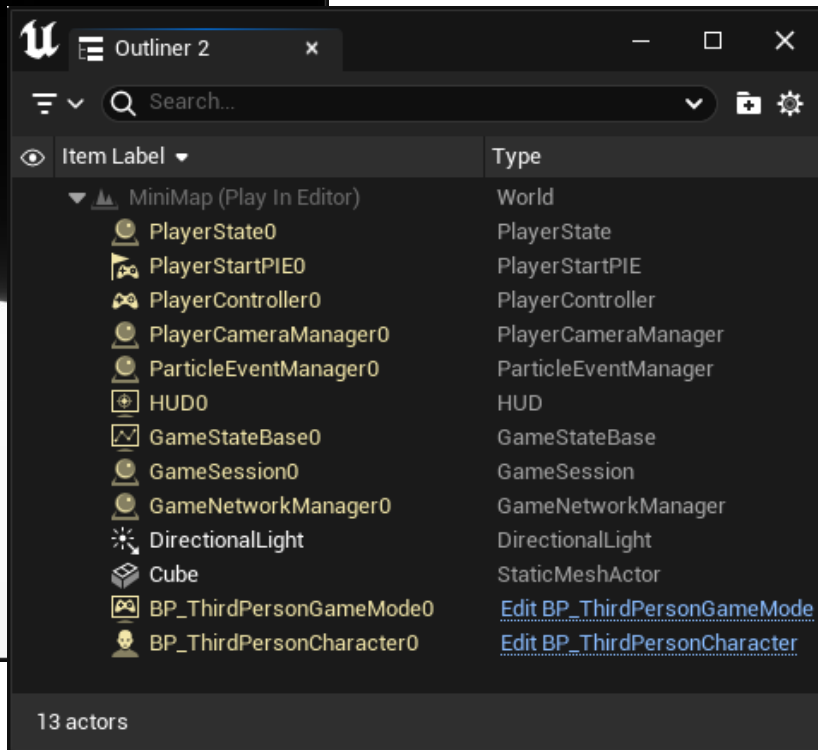


단 두 개의 액터로만 구성된 레벨





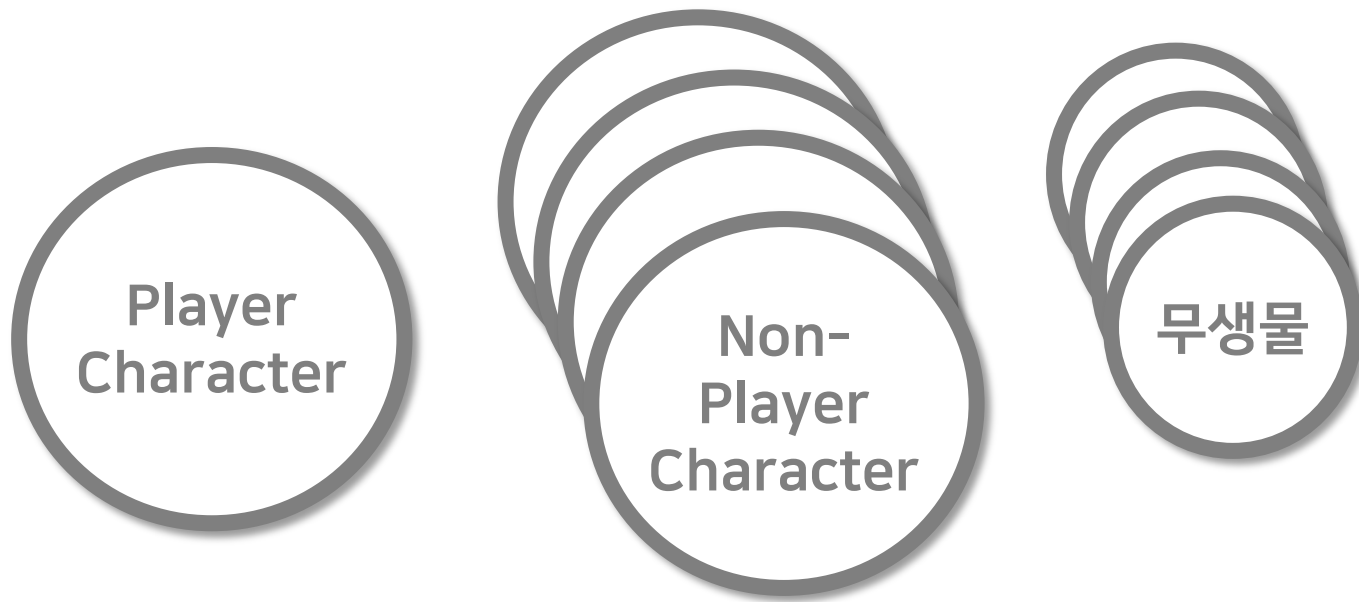
Play 하면, 많은 액터들이 내부적으로 생성된다. -> 엔진이 생성하는 것임.



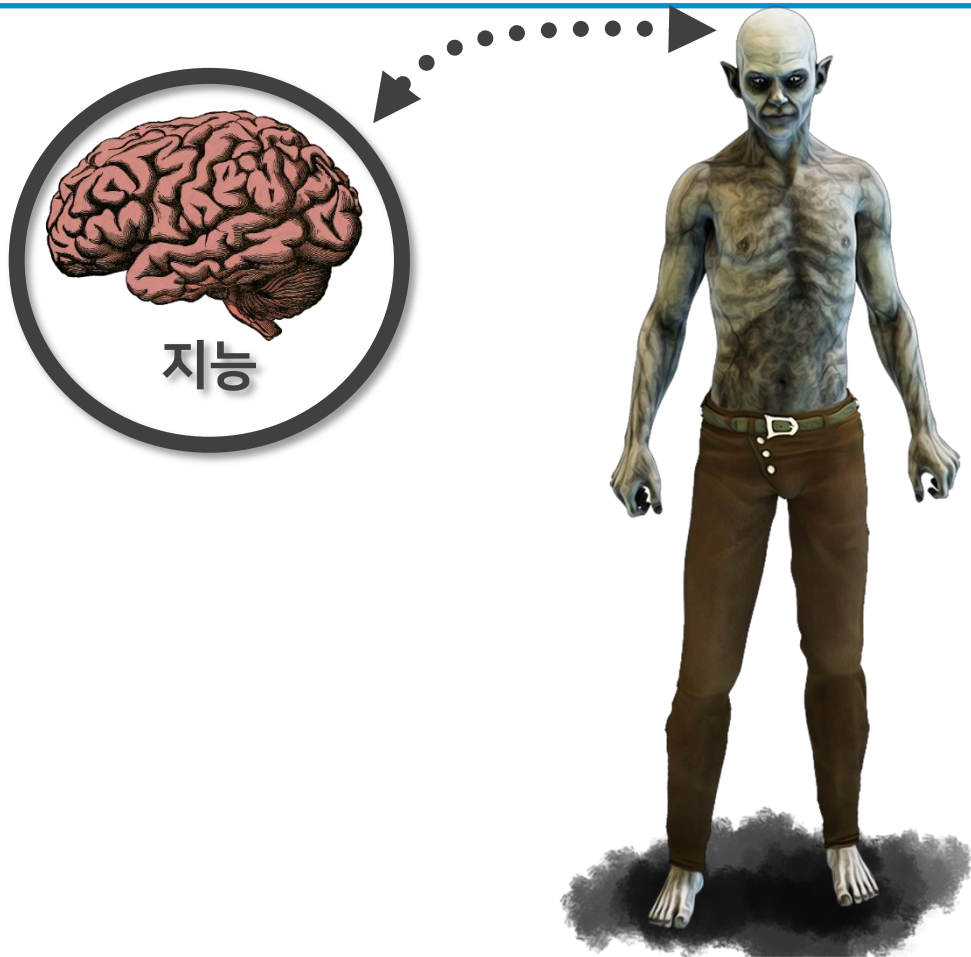
컴퓨터 게임이란?

가상 세계에서 객체들의 상호작용을
시뮬레이션하고 그 결과를 보여주는 시스템

게임 내 객체들의 종류



좀비는 왜 움직일까?



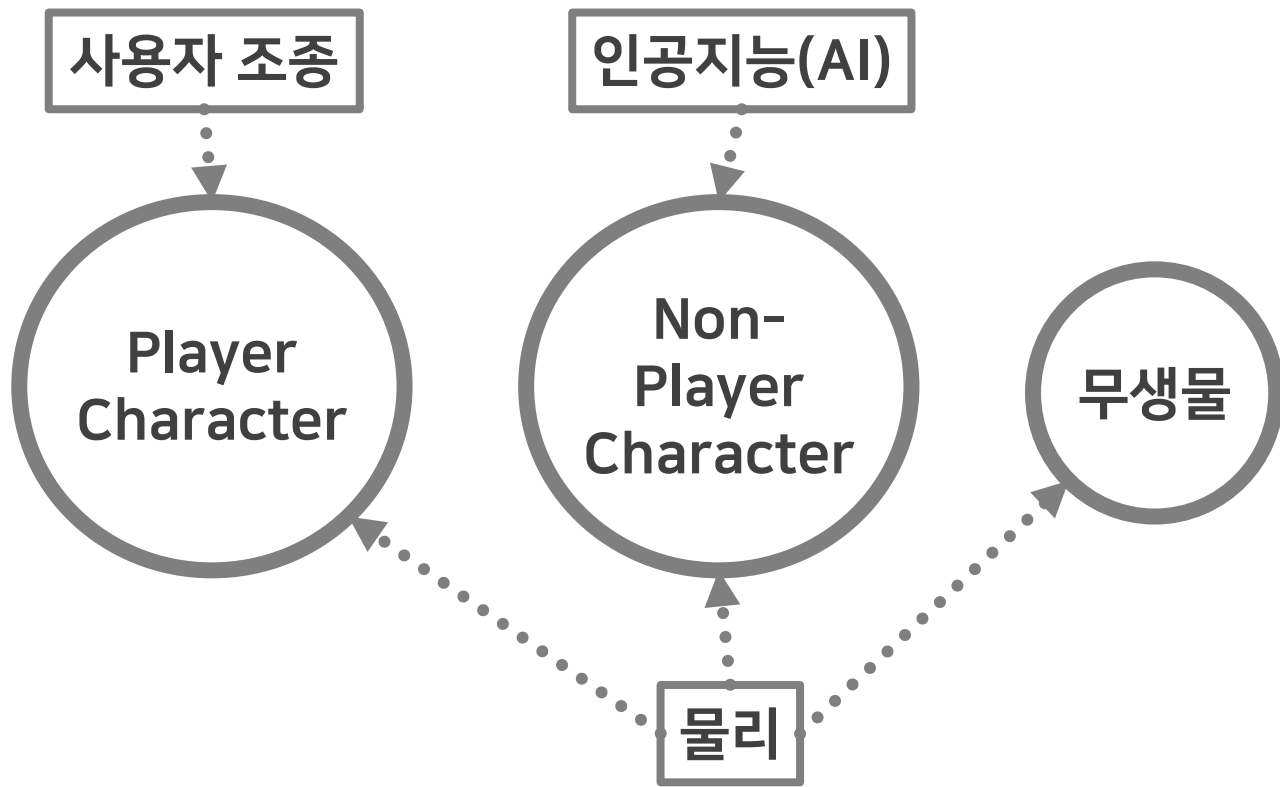
적군은 왜 추락할까?



주인공 캐릭터는 왜 움직일까?



게임 객체의 제어

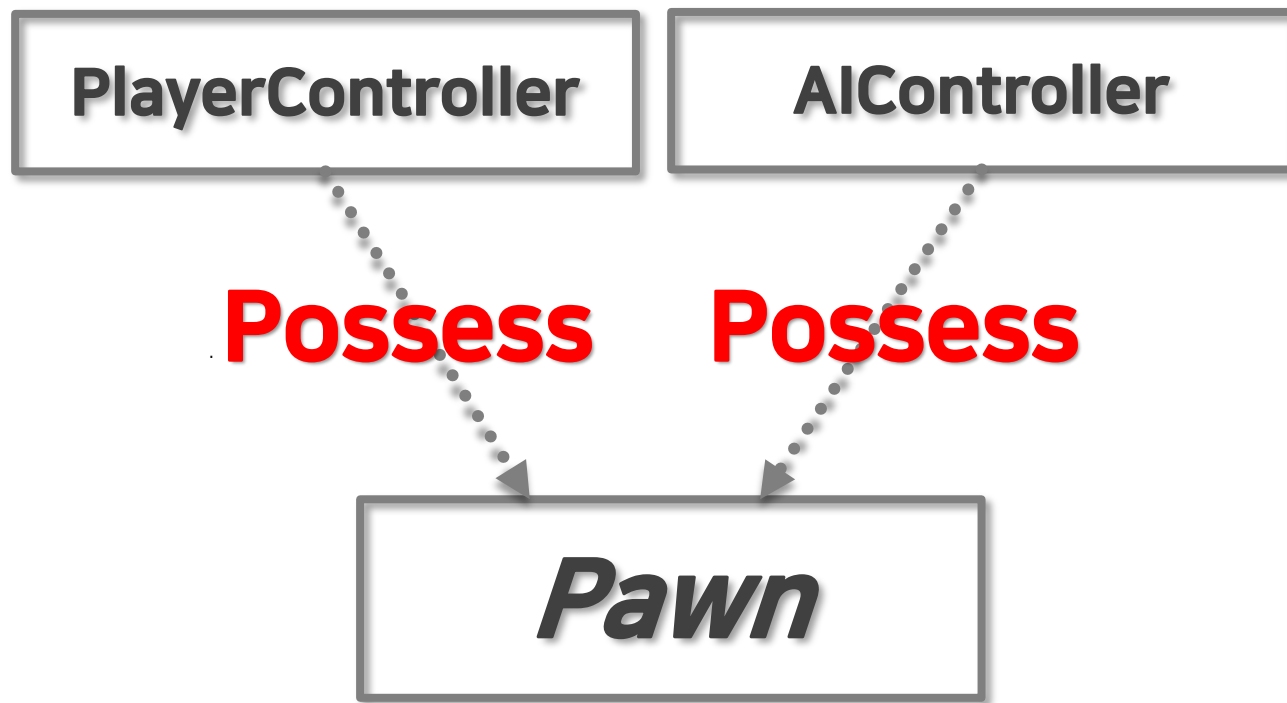


게임 객체 들의 상호 작용을 시뮬레이션하는
소프트웨어 구조

Pawn

- 모든 게임 (Live) 객체들의 베이스 클래스.
 - Q: 모든 객체들의 베이스 클래스는??
- 플레이어, NPC, ...
- 객체의 시각적, 물리적 표현을 담당.
- 객체들의 (비시각적, 비물리적) 상태는?
 - Player : PlayerState 에 저장.
 - NPC: AIController 에 저장.

언리얼 엔진 게임 플레이 프레임워크 - 지배(Possess) 구조



PlayerController

- 게임 플레이어와 게임 월드를 연결시키는 인터페이스.
- Pawn 뿐만 아니라, 카메라, HUD 도 제어.



PlayerState

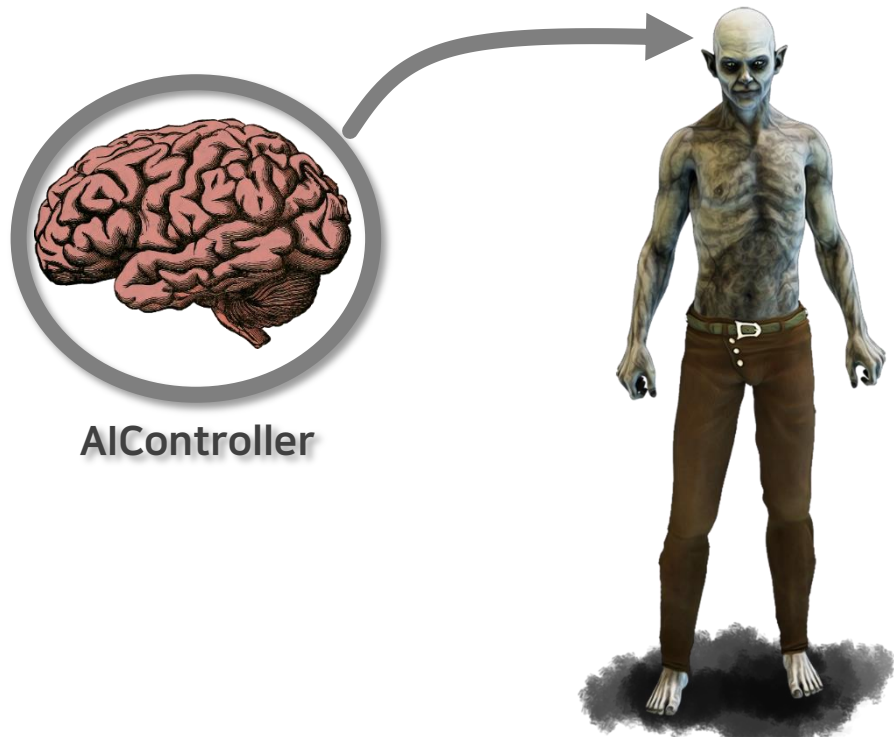
- 플레이어의 게임 내 현재 상태를 저장

- 이름
- 점수
- 레벨
- 생명력



AIController

- NPC 를 제어하는 인공지능 컨트롤러



HUD

- 게임 플레이 도중 화면 위에 놓이는 정보.
- 게임의 현재 상태를 플레이어에게 전달.



GameState

- 게임에 접속 중인 모든 클라이언트들이 알아야 하는 공통 정보 저장.
 - 접속된 플레이어 목록
 - 게임의 팀 점수
 - 오픈 월드 게임에서 완료한 미션
- 연결된 모든 클라이언트들에 복제(Replication)됨.

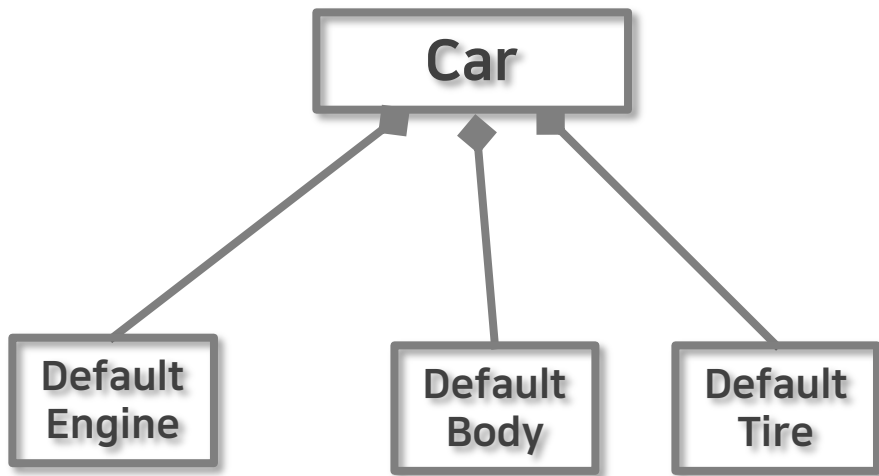
게임 플레이 프레임워크 핵심 클래스들

- Pawn
- PlayerController
- HUD
- GameState
- PlayerState

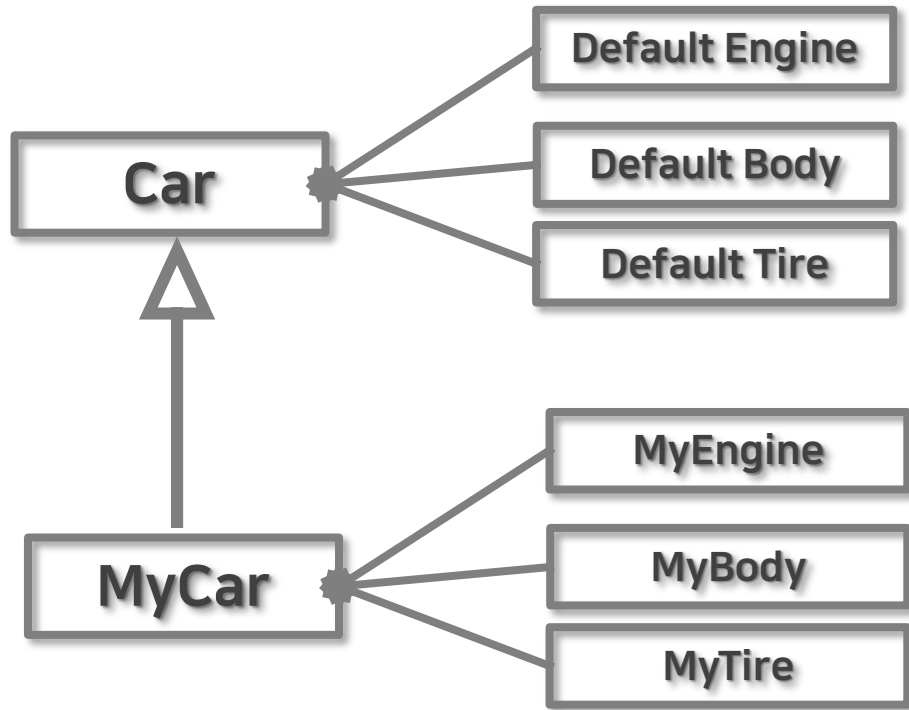
이것들을 개별 게임에 맞게 변경하려면?

소프트웨어 사용자 확장

- 구성(Composition)과 상속(Inheritance)를 이용

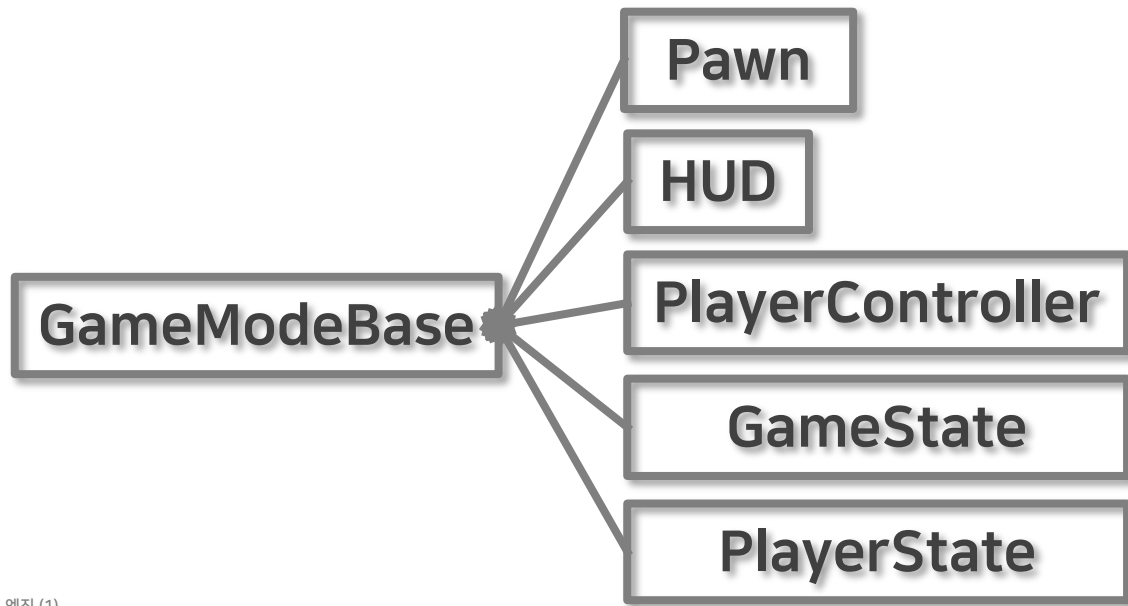


부모 클래스로부터 구성 방식을 상속받고, 부품을 바꿔 끼운다.



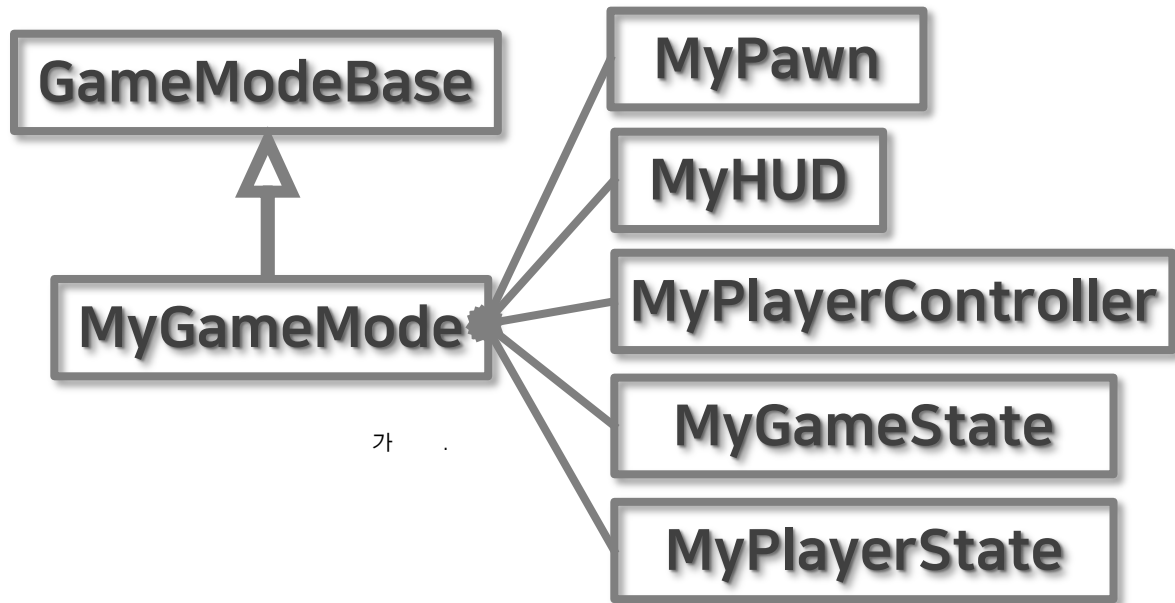
GameMode

- 핵심 클래스를 구성(Composition)품으로 담는 컨테이너
- 레벨마다 한 개의 GameMode가 설정됨.
- 게임의 기본 규칙 설정 - 플레이어 수, 플레이어의 참가 방식 등.
- 프레임워크를 구성하는 클래스들을 지정.



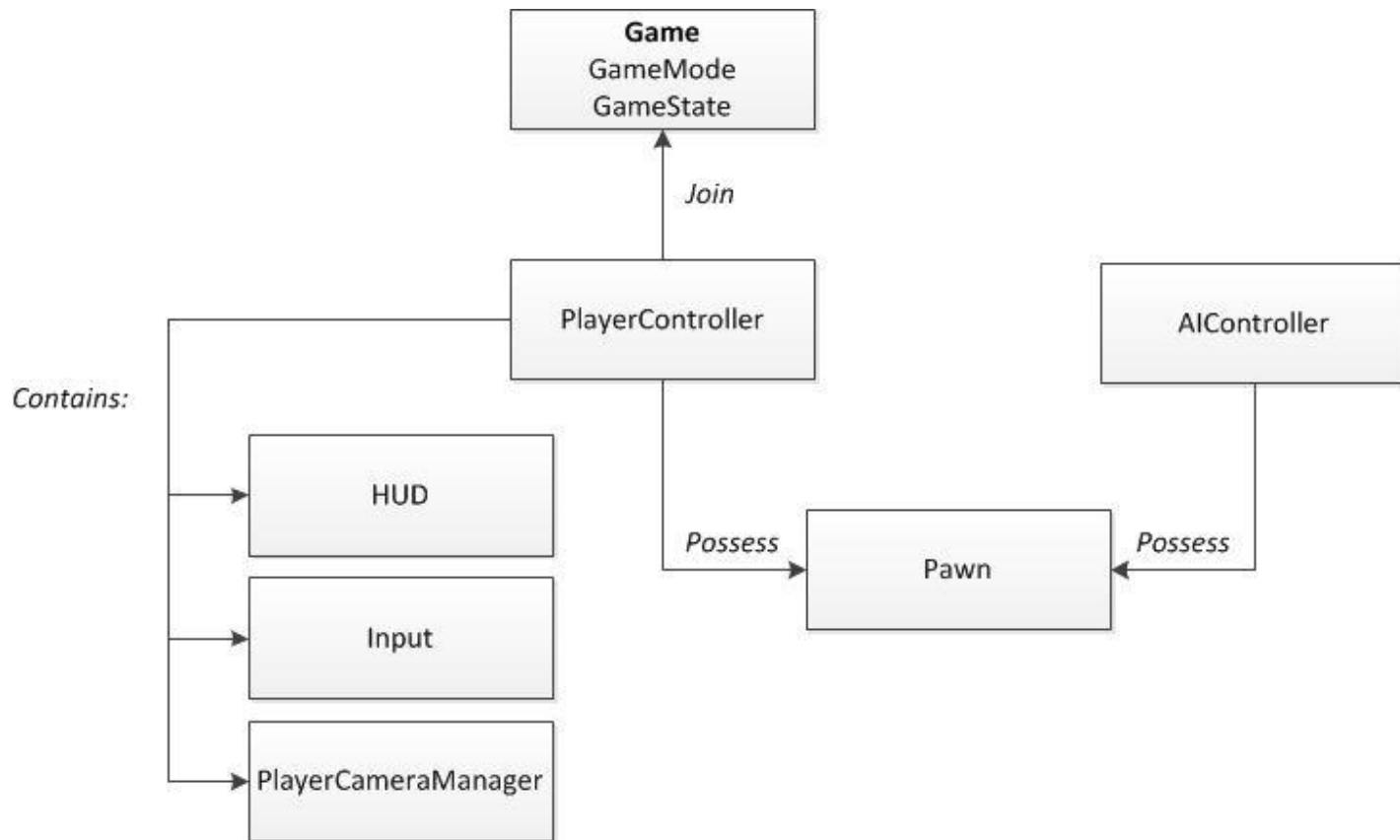
GameMode

- GameModeBase의 자식 게임 모드를 만들어서, 독자적인 게임 모드 설정.

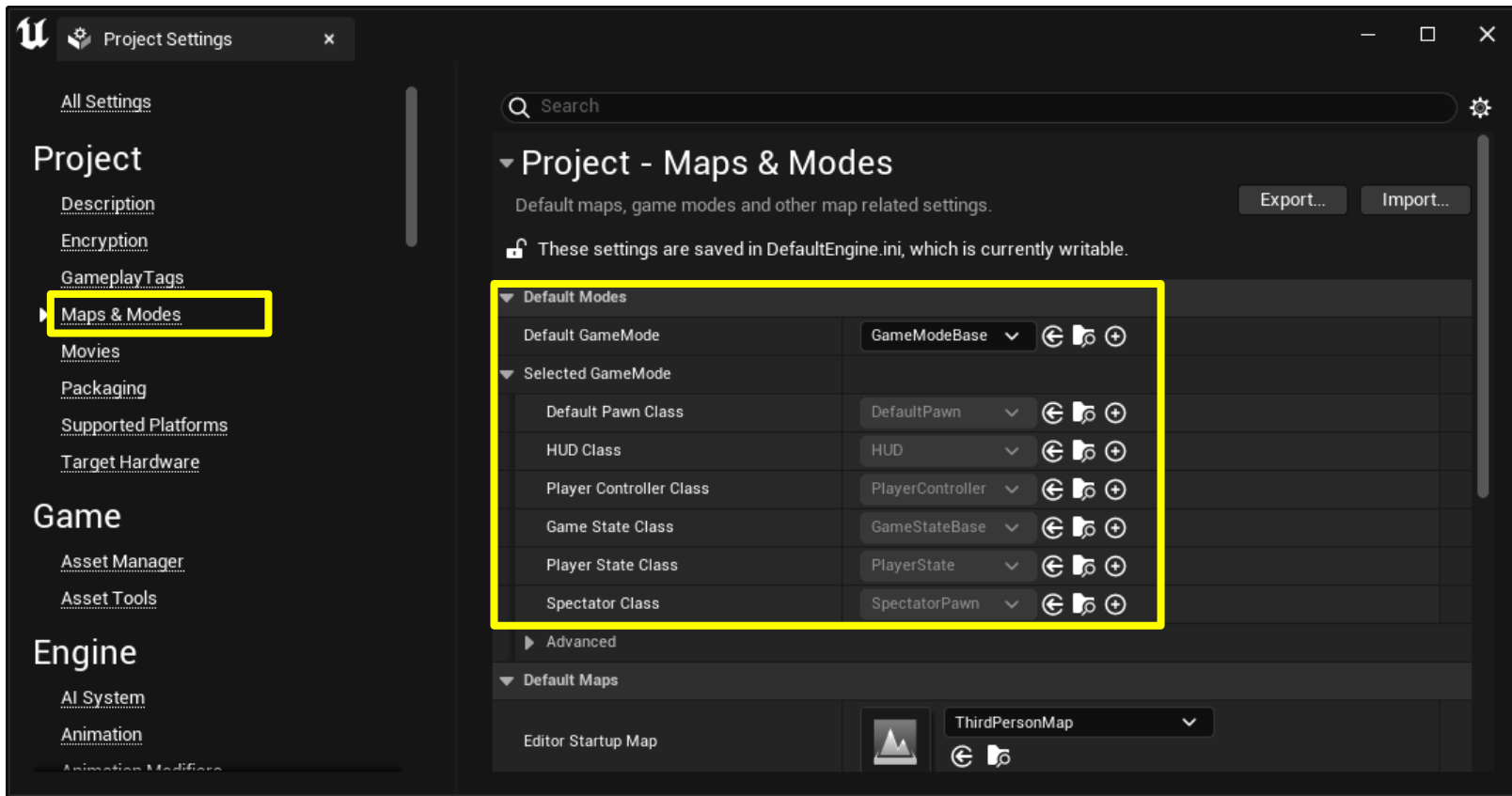


가

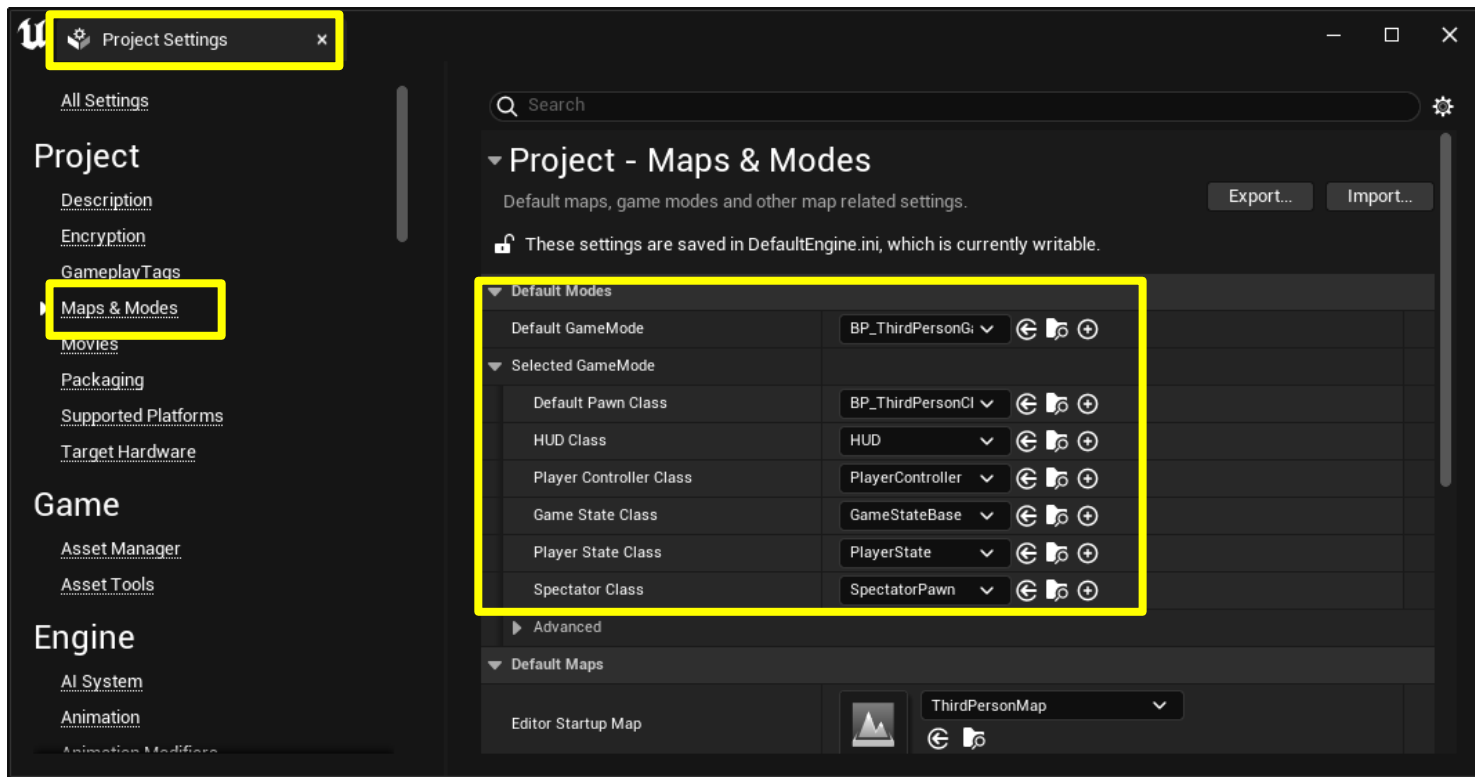
언리얼 엔진 핵심 클래스 관계도



Project의 기본 게임 모드 설정



Third Person 샘플 프로젝트의 게임 모드



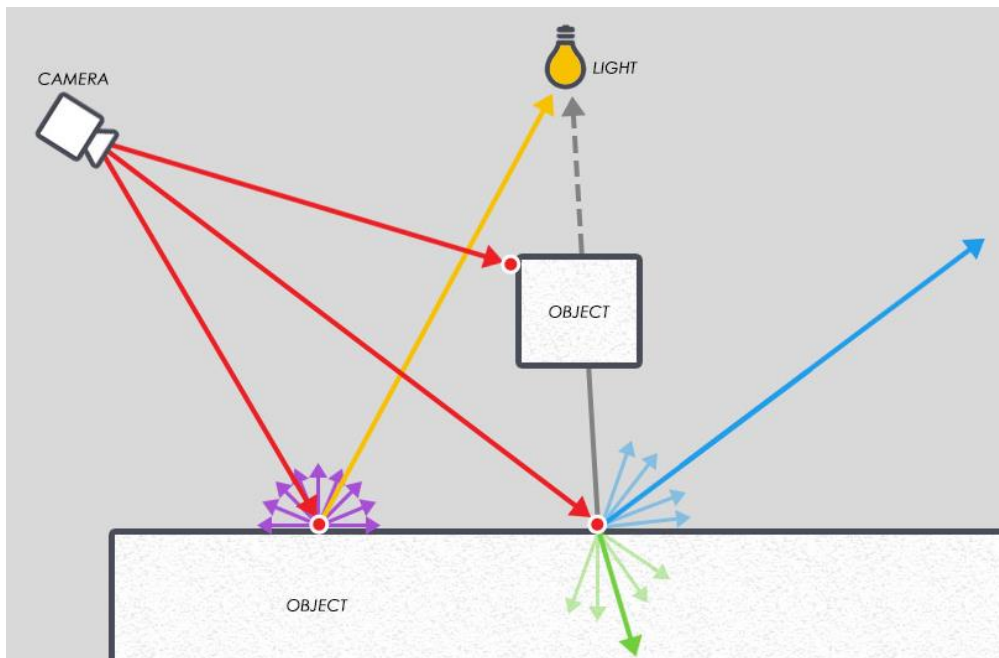


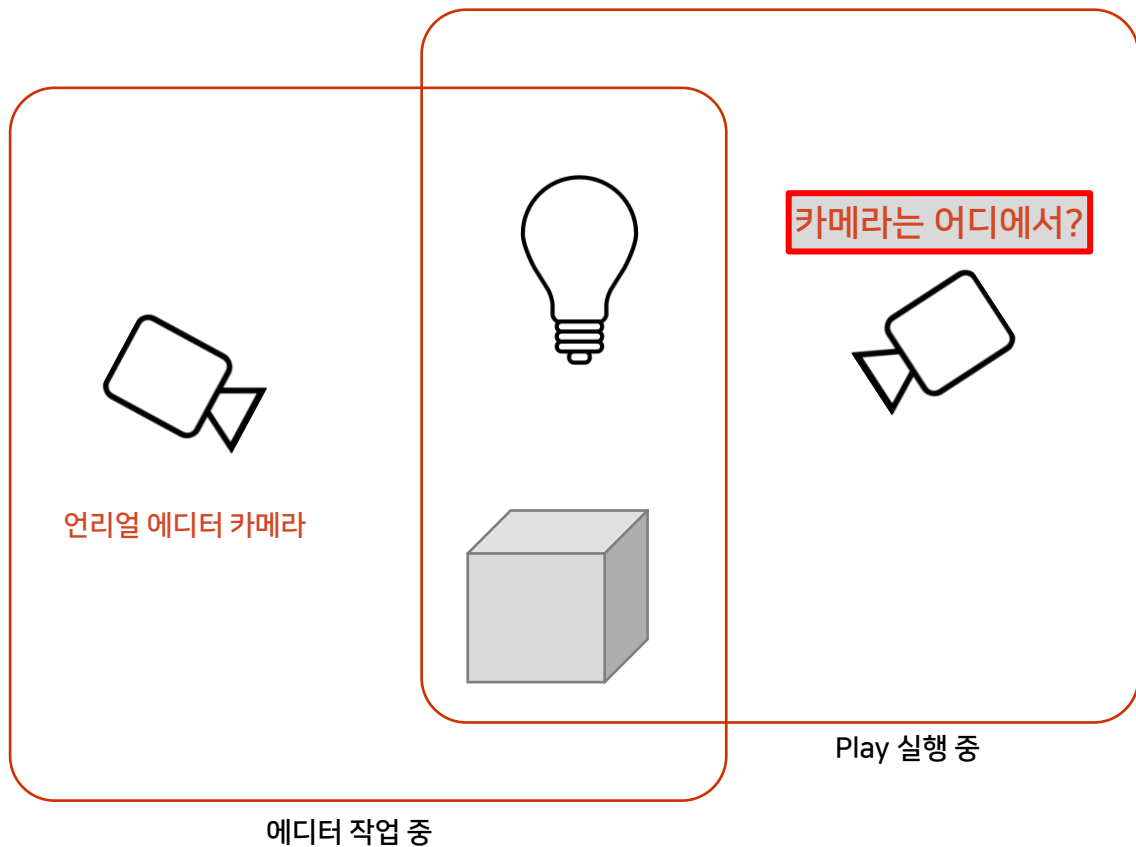
실습 LAB

게임 모드 구성 실습

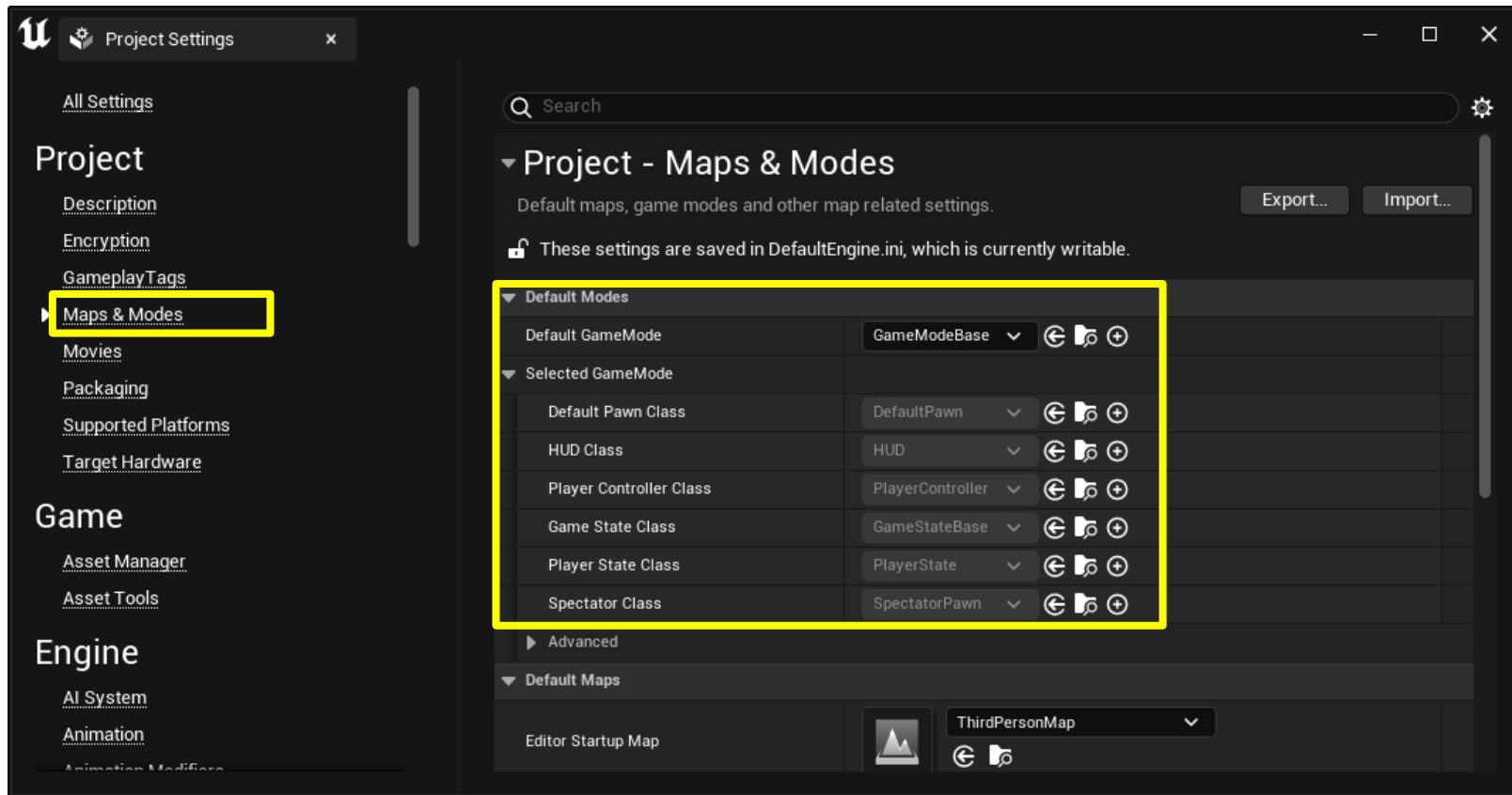
3D 렌더링 기본 구성 요소

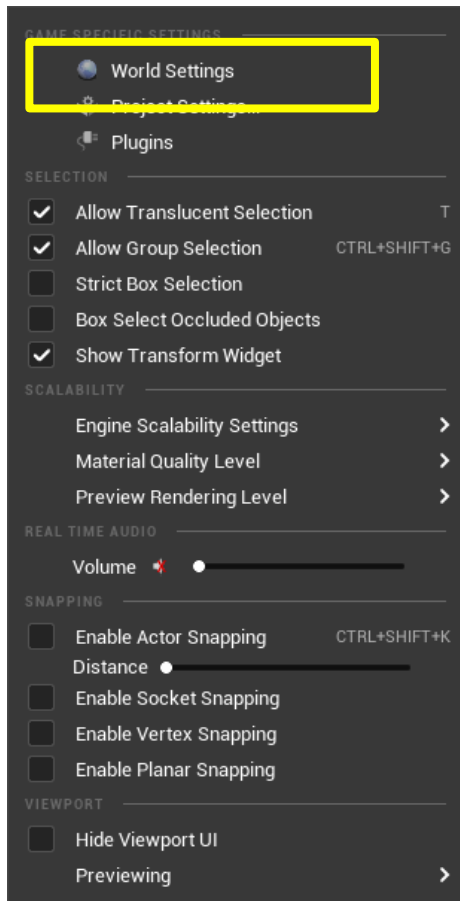
- 씬(Scene) - 여러개의 객체(Object)들로 구성
- 광원(Light)
- 카메라(Camera)



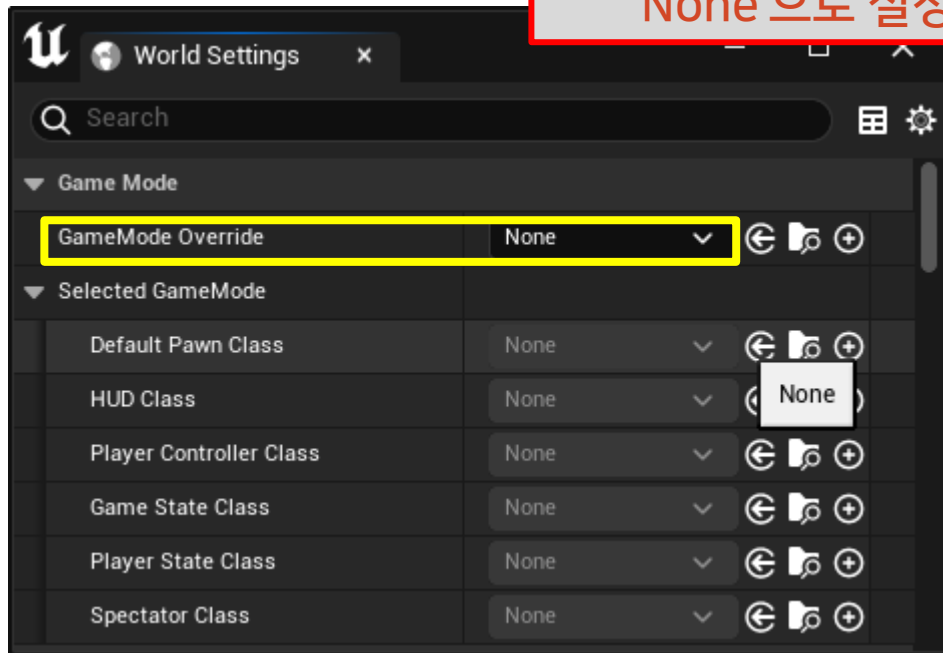


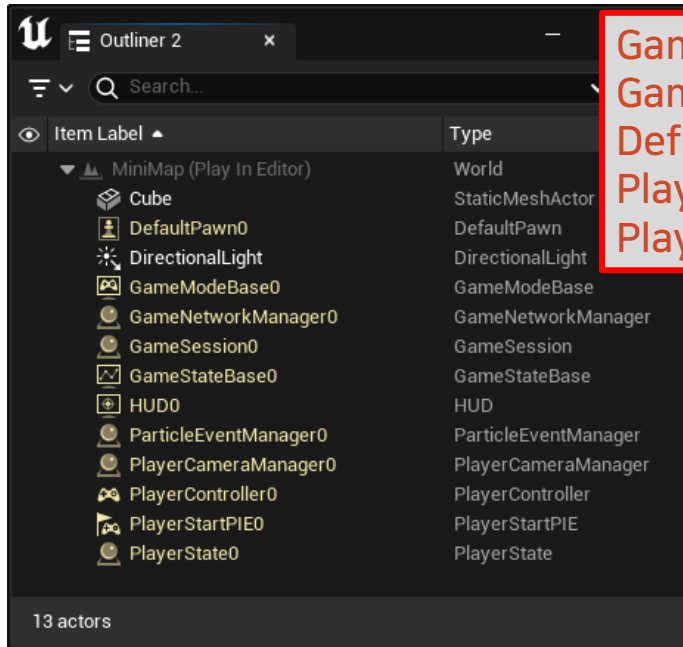
Project의 기본 게임 모드 확인 - GameModeBase



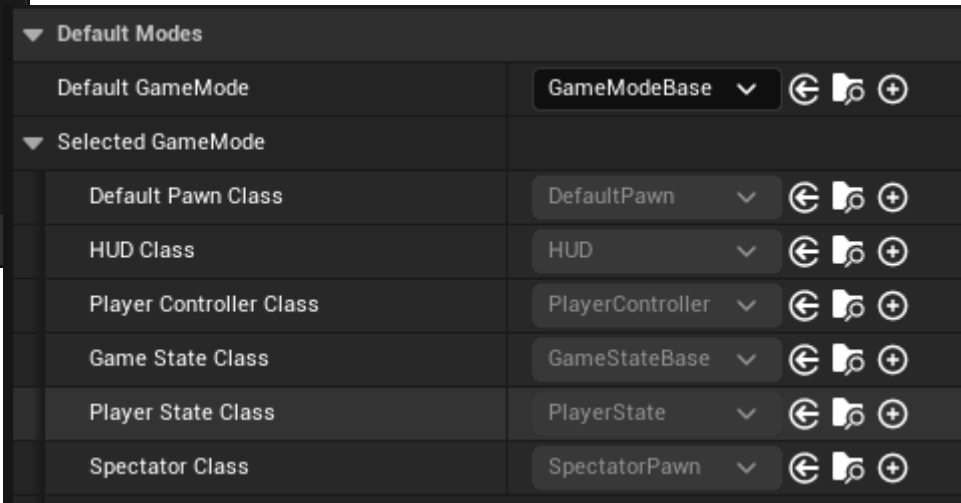


World Settings 의
GameMode 확인
'None'으로 설정

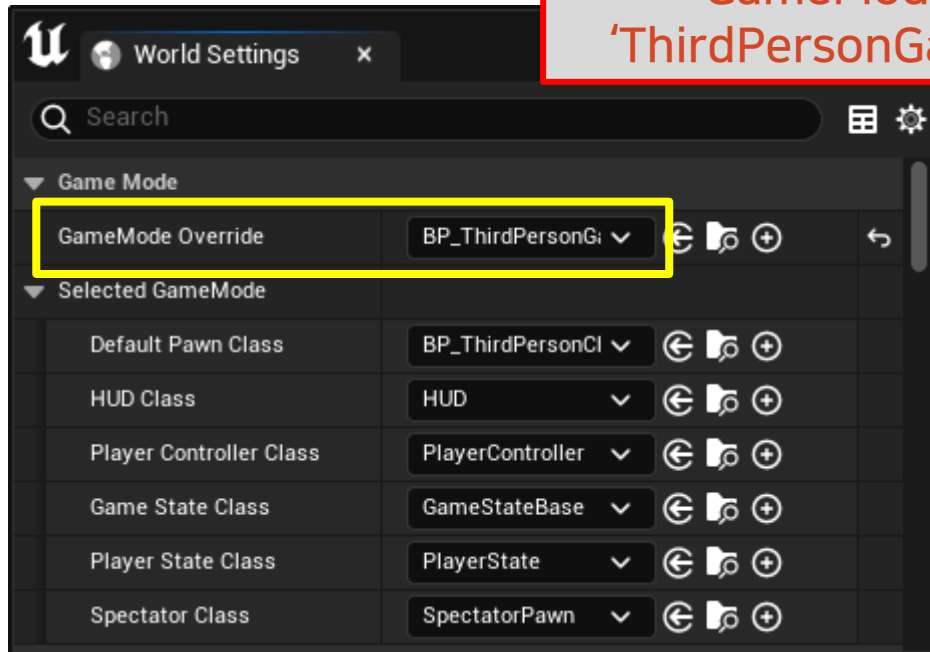


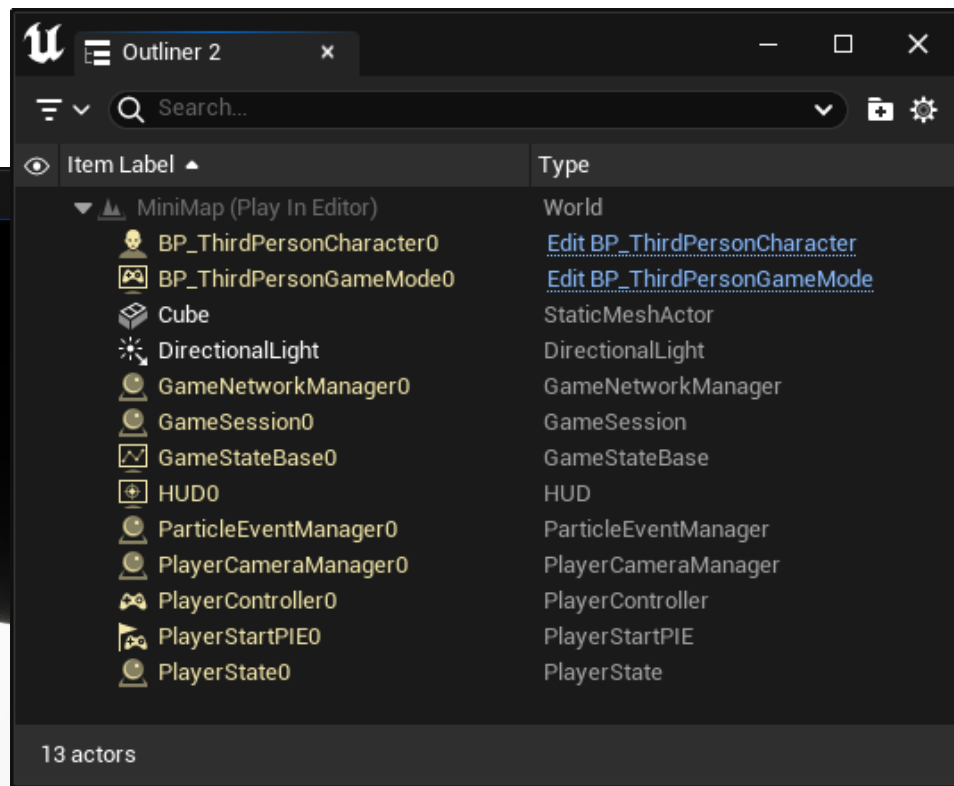
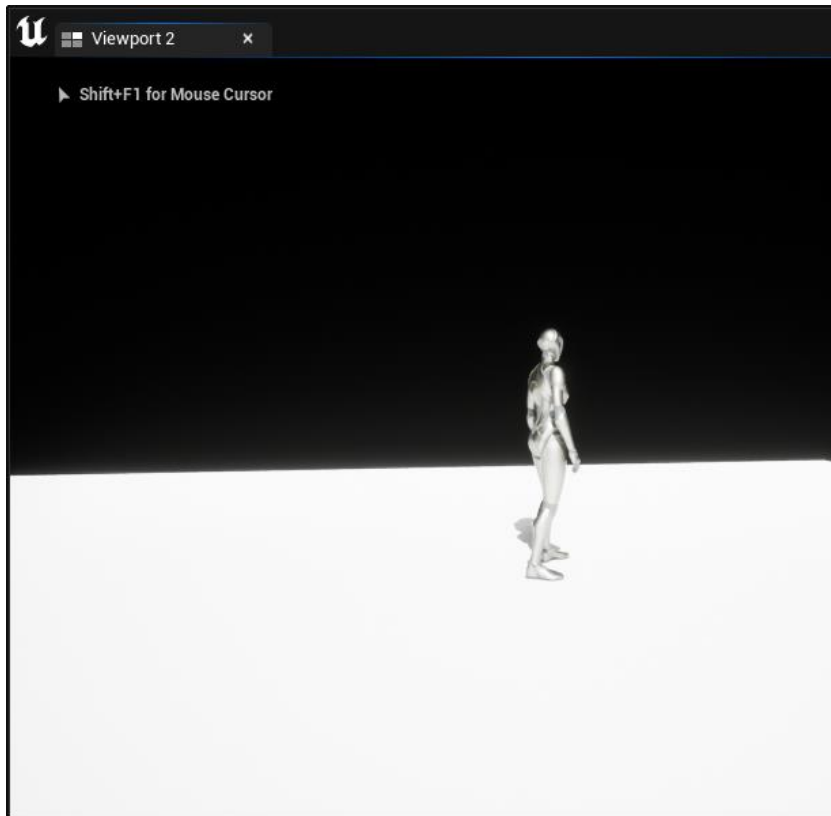


GameModeBase 액터가 생성됨.
GameModeBase에서 지정한 클래스를 이용하여, 각종 액터들이 생성됨.
DefaultPawn이 생성됨.
PlayController는 PlayerCameraManager를 갖게 됨.
PlayerCameraManager는 자동생성된 CameraActor를 갖게 됨.



GameMode Override 를
'ThirdPersonGameMode'로 설정.



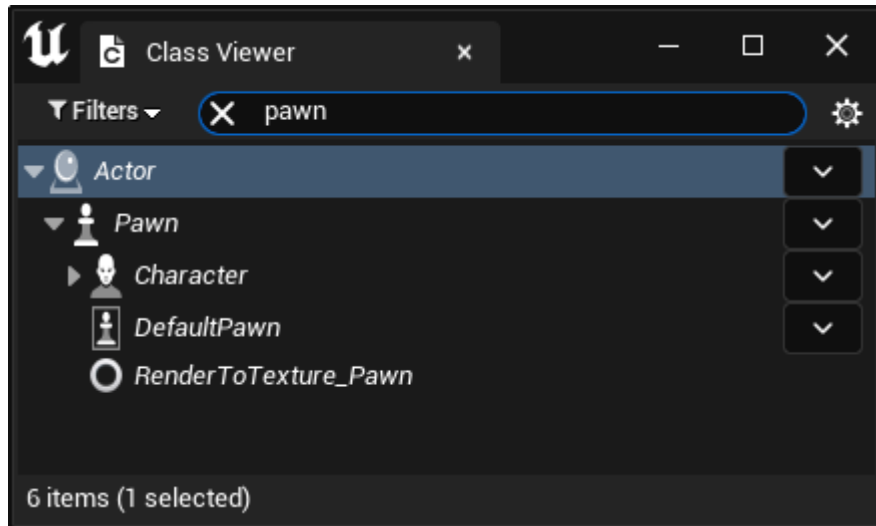


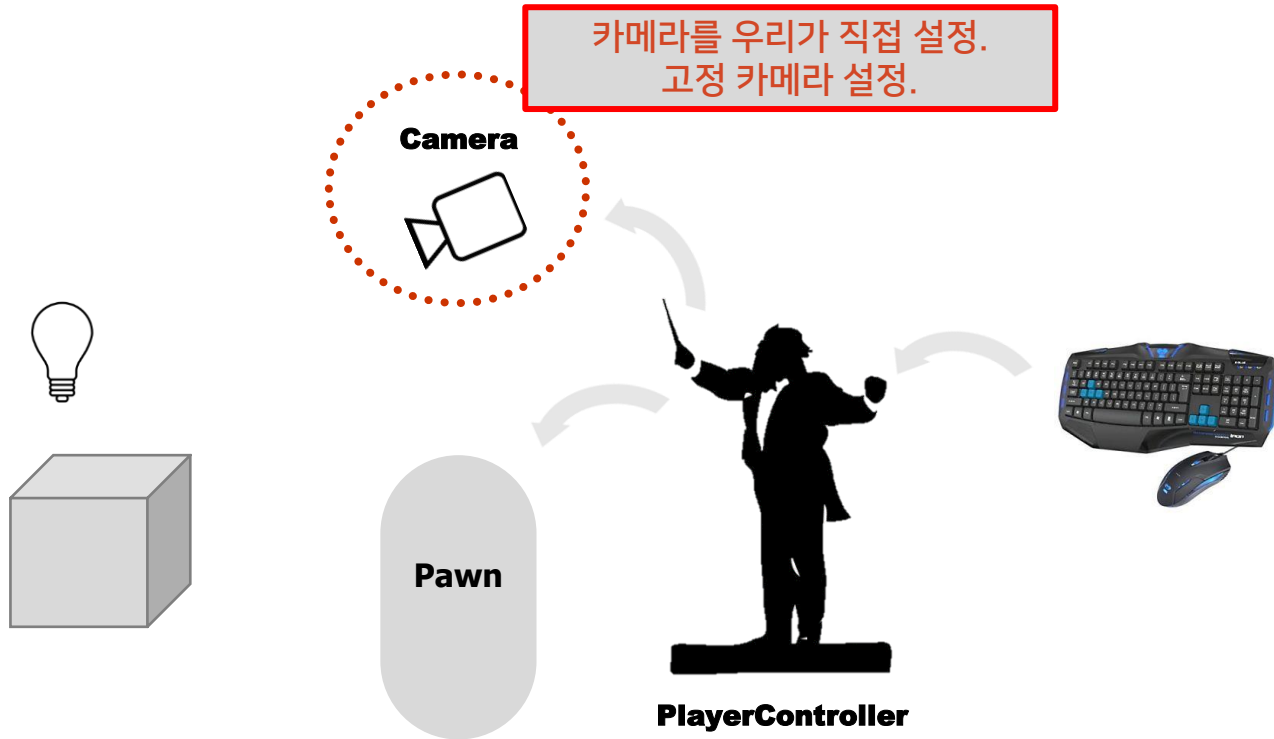


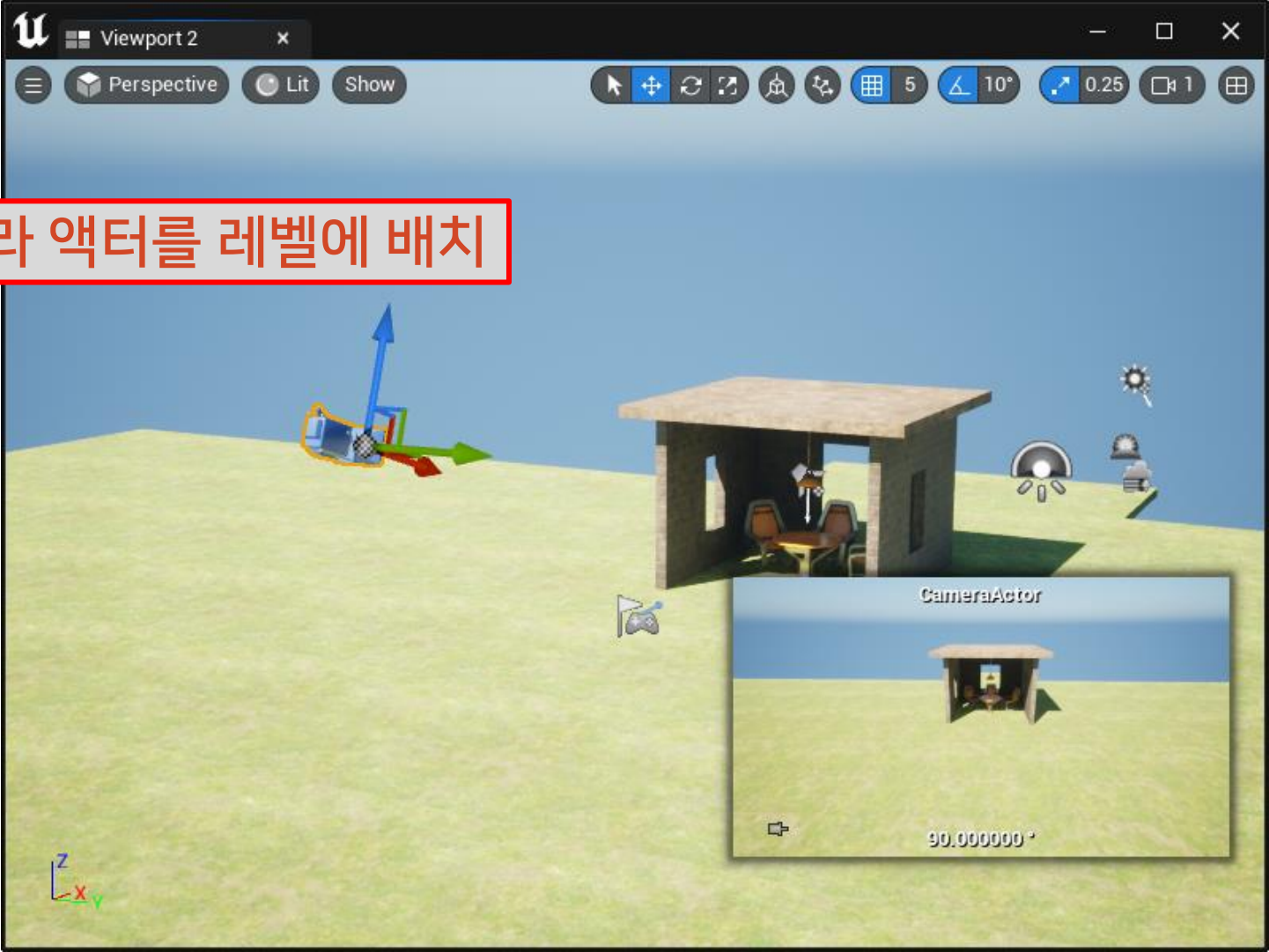
실습 LAB

고정 카메라 설정

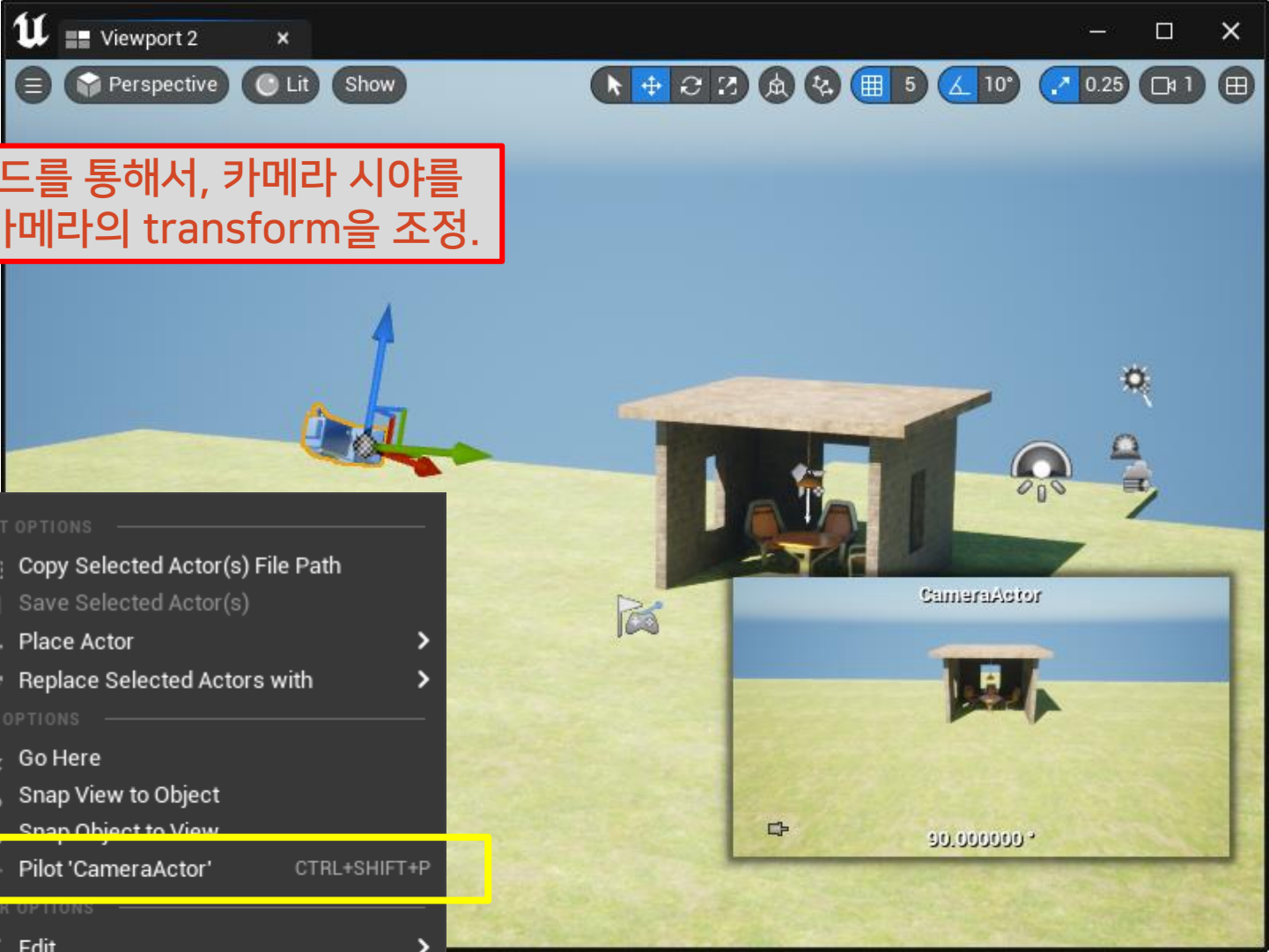
Actor – Pawn – Character, DefaultPawn





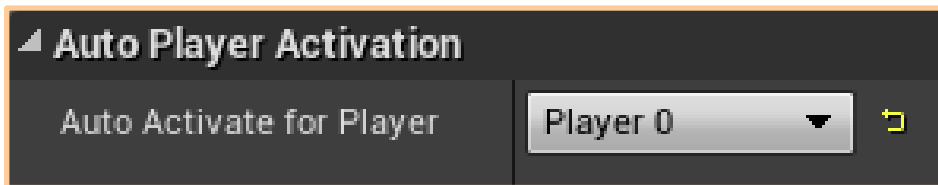


카메라 액터를 레벨에 배치



Pilot 모드를 통해서, 카메라 시야를 보면서, 카메라의 transform을 조정.





PlayerController의 카메라로 지정.
지정하지 않으면, PlayerController는 자동
생성된 내부 카메라를 메인 카메라로 활용함.



Play 실행 하면, 카메라의 위치가 고정된
상태에서, 키 입력에 따라서
"DefaultPawn"이 이동함.

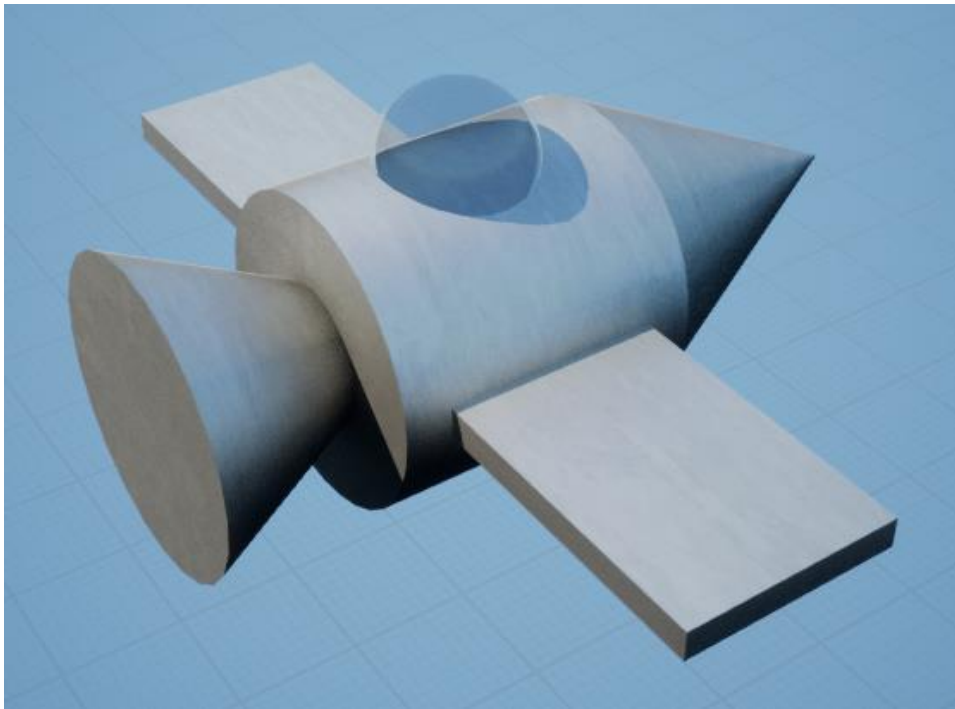


실습 LAB

Aircraft Blueprint 제작 및 조종

Aircraft Blueprint

- WASD 키를 통해서 전후진, 좌우 이동

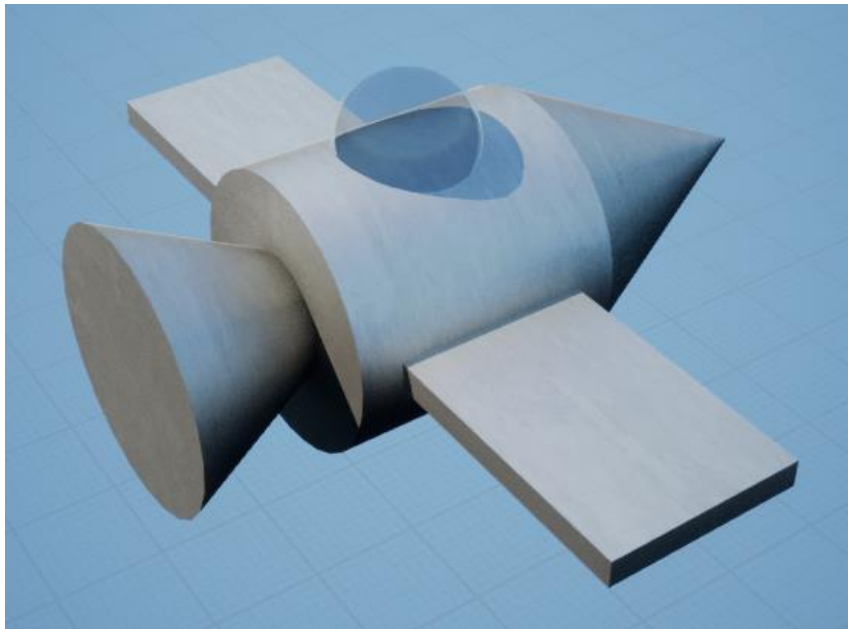


블루프린트 설계

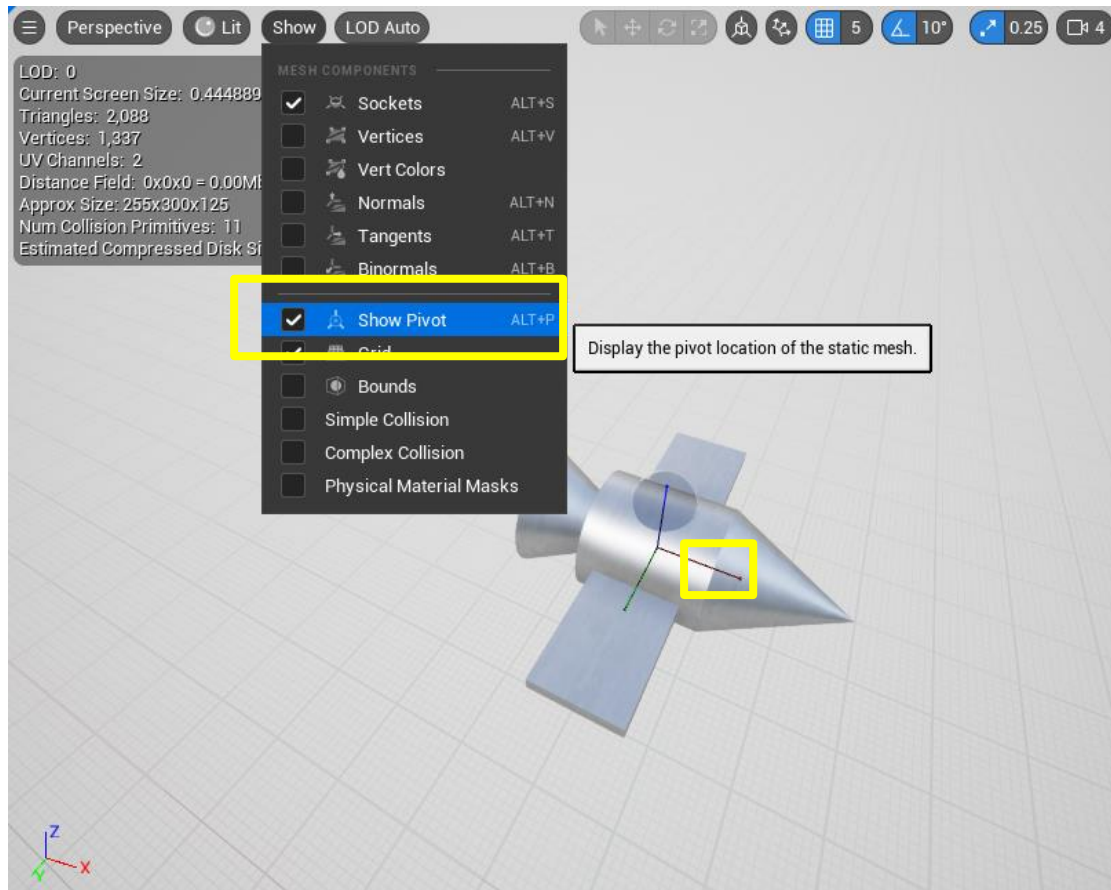
- 그대로 사용할 수 있는 블루프린트 있는가? NO
- 비슷한 블루프린트가 있는가? YES
 - "DefaultPawn"
- 추가할 것만 있는가? YES
 - Static Mesh 모양만 추가 !!!

Aircraft Static Mesh 만들기

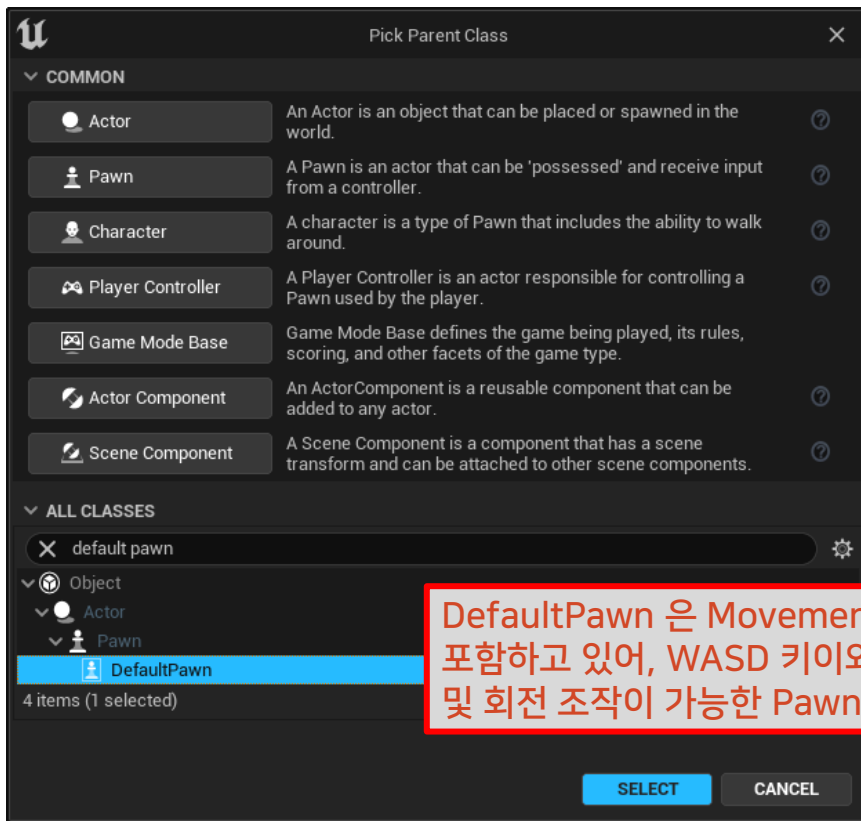
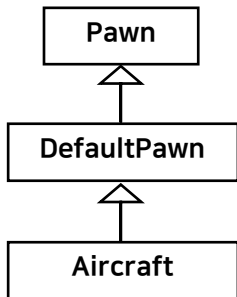
- Cube, Sphere, Cylinder, Cone 을 이용해서, 비행기 모양을 만듦.
- 액터 머징을 통해 단일 메시로 만듦. - SM_Aircraft



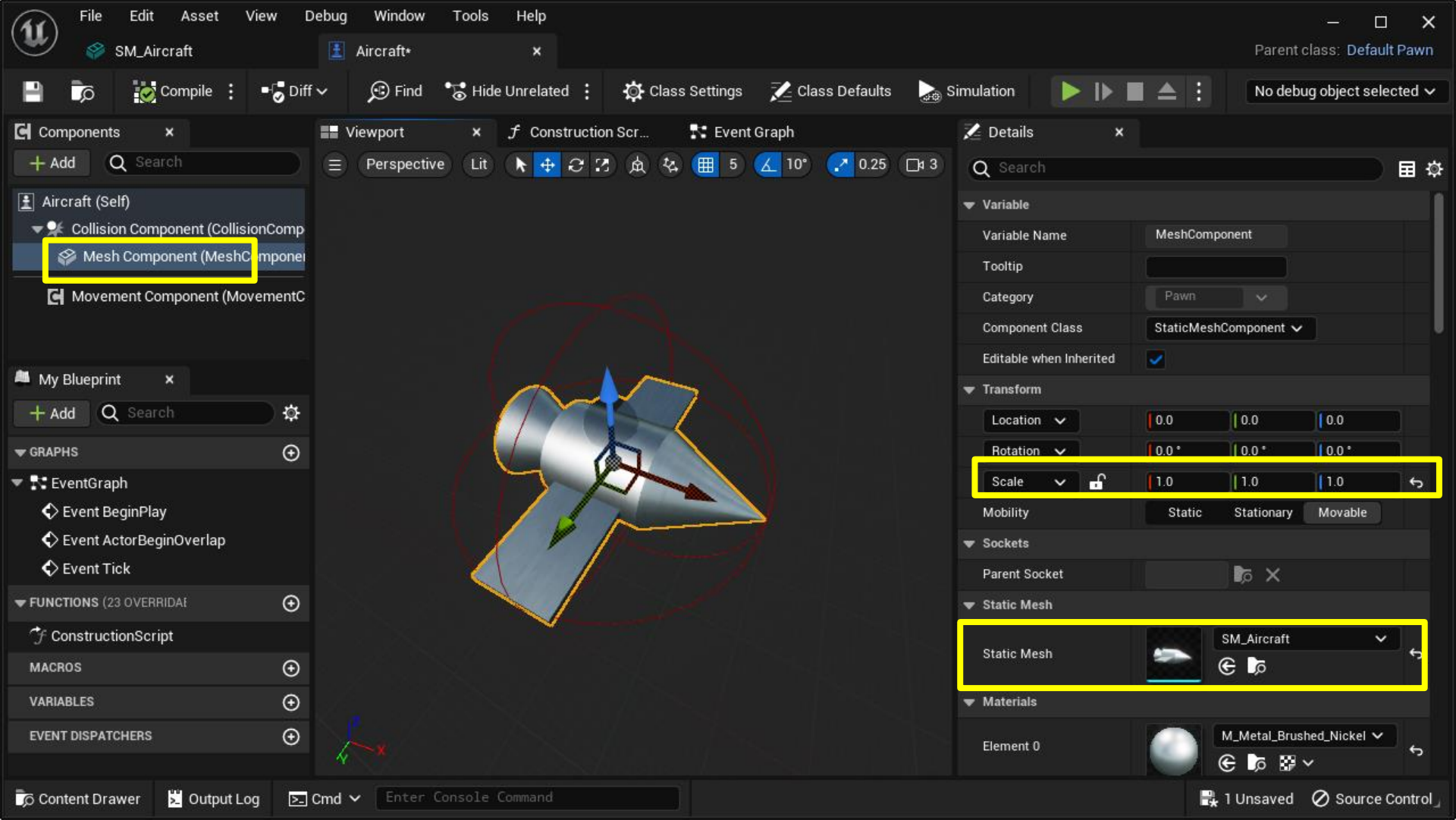
Forward Direction - X - Red 방향 확인

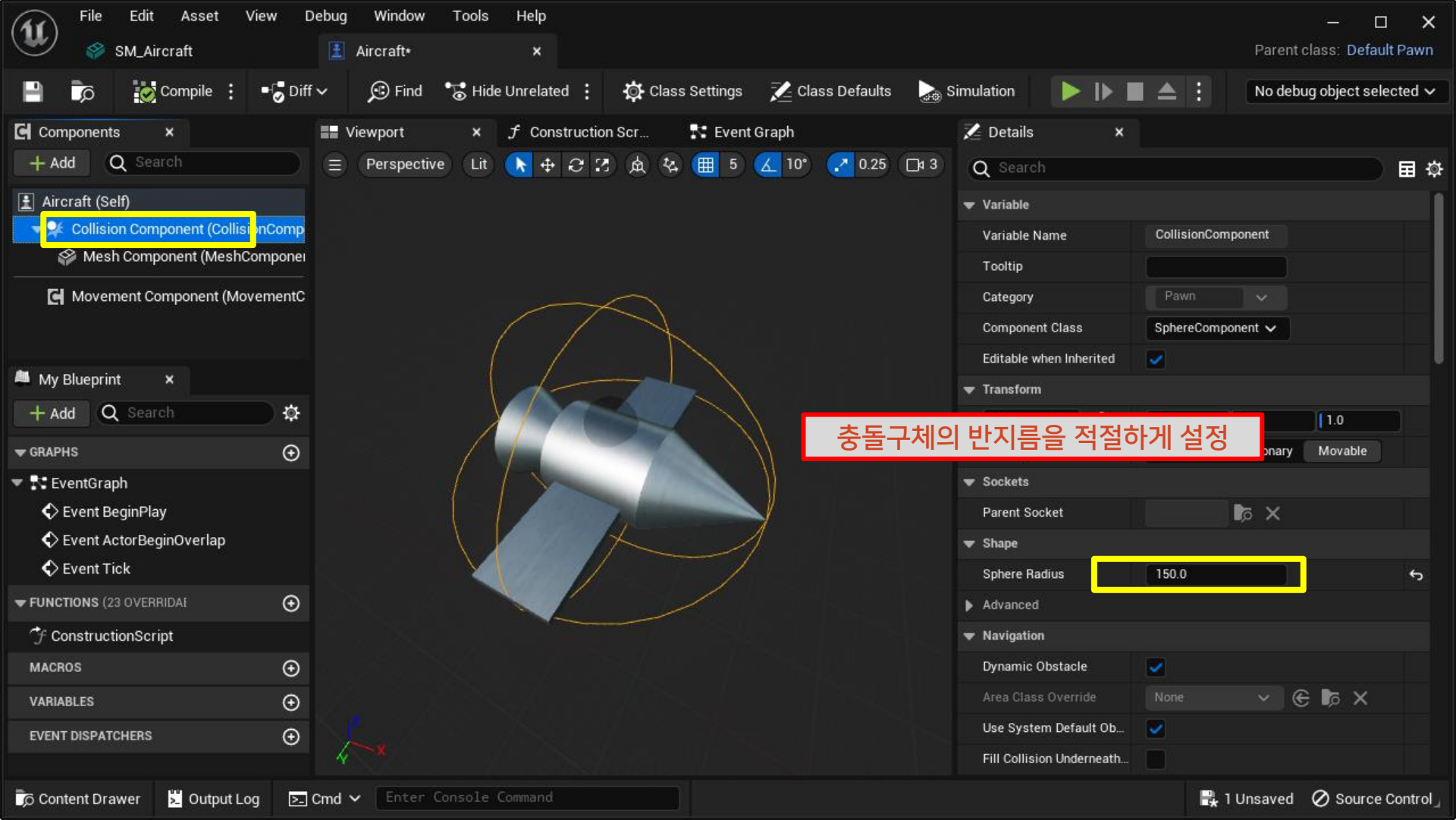


DefaultPawn 을 Parent로 하여, Blueprint를 만듦.

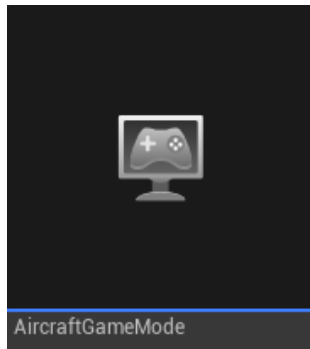
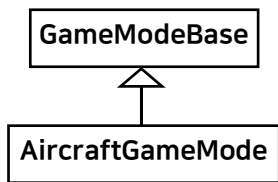


DefaultPawn 은 MovementComponent를 기본으로 포함하고 있어, WASD 키와 마우스에 따라 기본 이동 및 회전 조작이 가능한 Pawn임.





게임 모드 만들기 : Blueprint 클래스로 만듦.



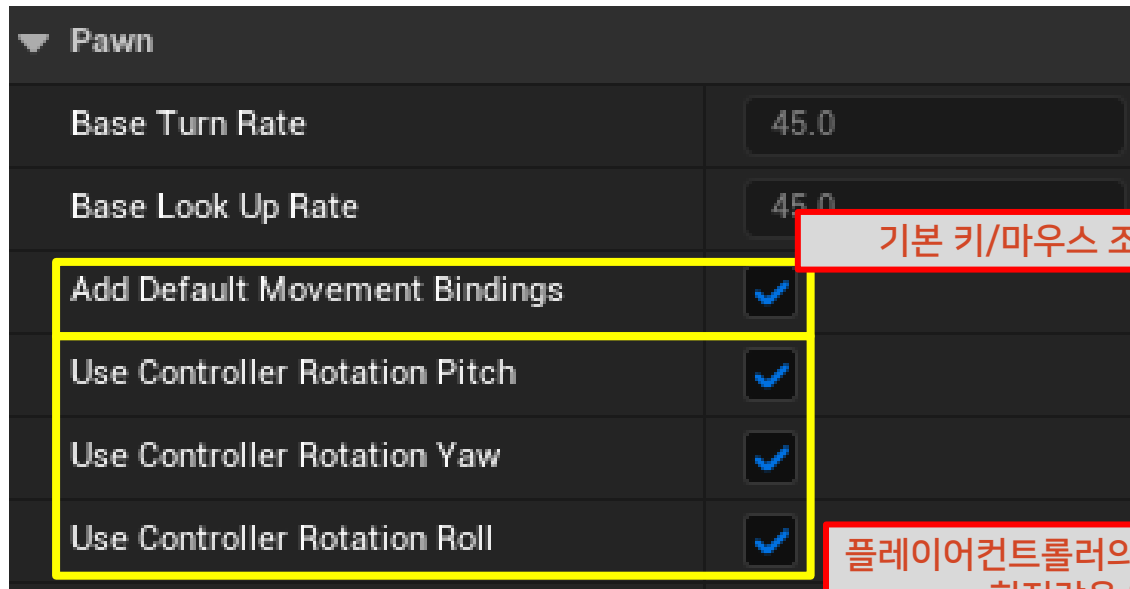
GameModeBase를 베이스클래스로
하는 블루프린트 클래스를 만듦.

AircraftGameMode

▼ Classes						
Game Session Class	GameSession ▼	↶	📁	✕		
Game State Class	GameStateBase ▼	↶	📁	⊕		
Player Controller Class	PlayerController ▼	↶	📁	⊕		
Player State Class	PlayerState ▼	↶	📁	⊕		
HUD Class	HUD ▼	↶	📁	⊕	✕	
Default Pawn Class	Aircraft ▼	↶	📁	⊕	✕	↶
Spectator Class	SpectatorPawn ▼	↶	📁	⊕		

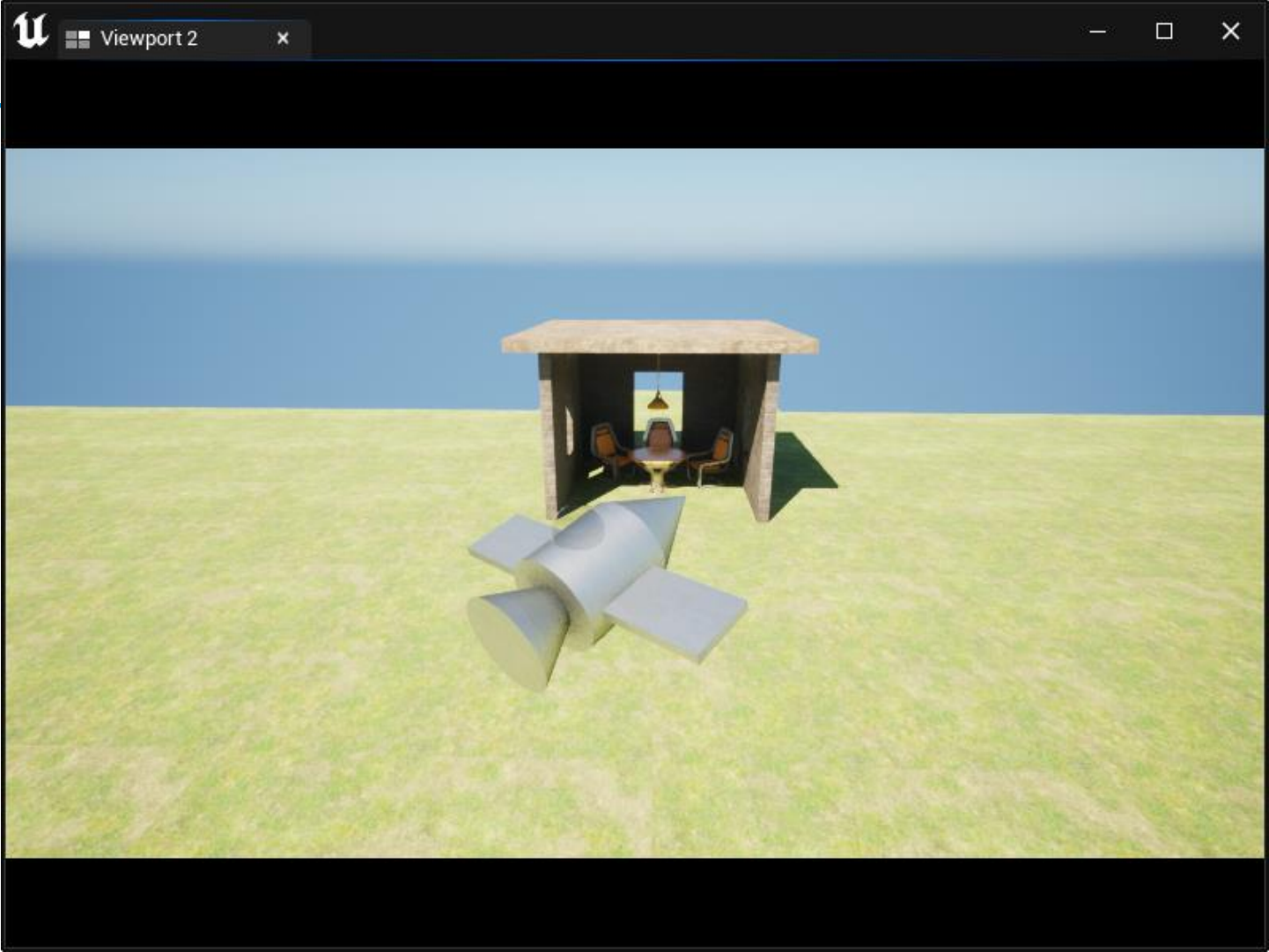
'Default Pawn Class'를
Aircraft 로 지정 !

Aircraft Blueprint 의 Class Defaults 설정



기본 키/마우스 조작 자동 적용

플레이어컨트롤러의 회전값과 Pawn의 회전값을 일치시킴.



언리얼 엔진 핵심 클래스 다이어그램

