A character with a prosthetic arm 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



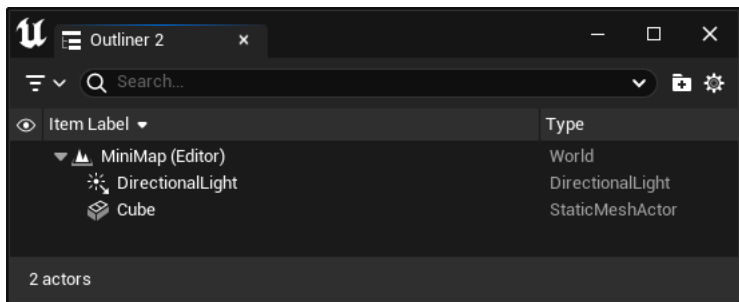
한국공학대학교
TECH UNIVERSITY OF KOREA

이대현 교수

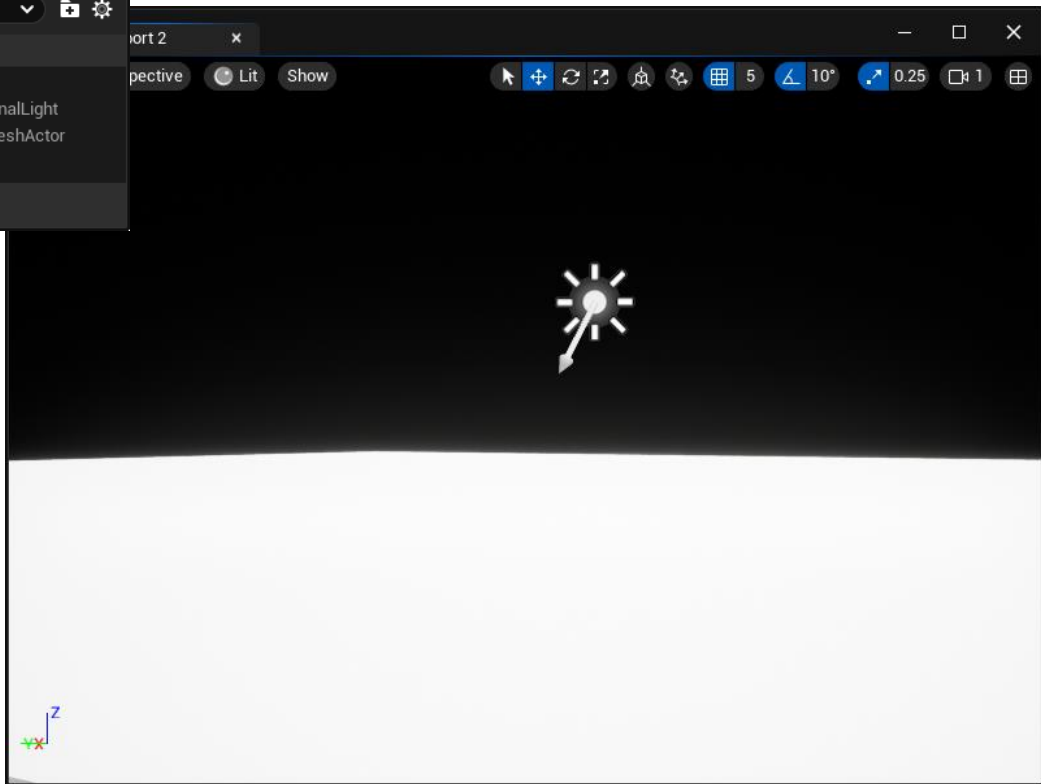
목차

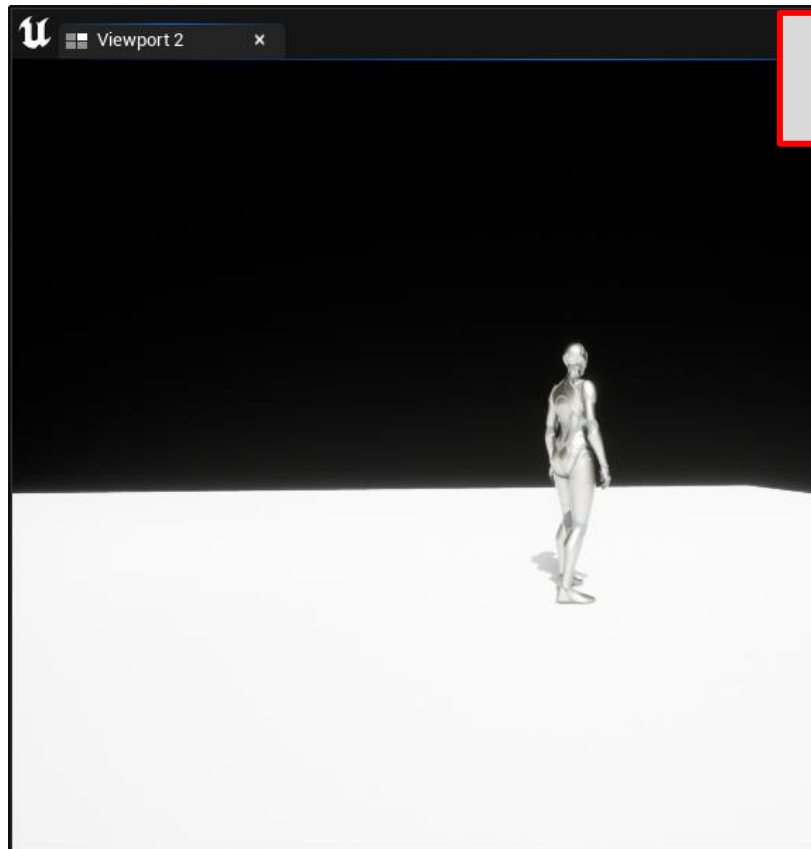
- 게임 플레이 프레임워크
- Default Pawn 클래스
- Aircraft 조종 실습

MiniMap

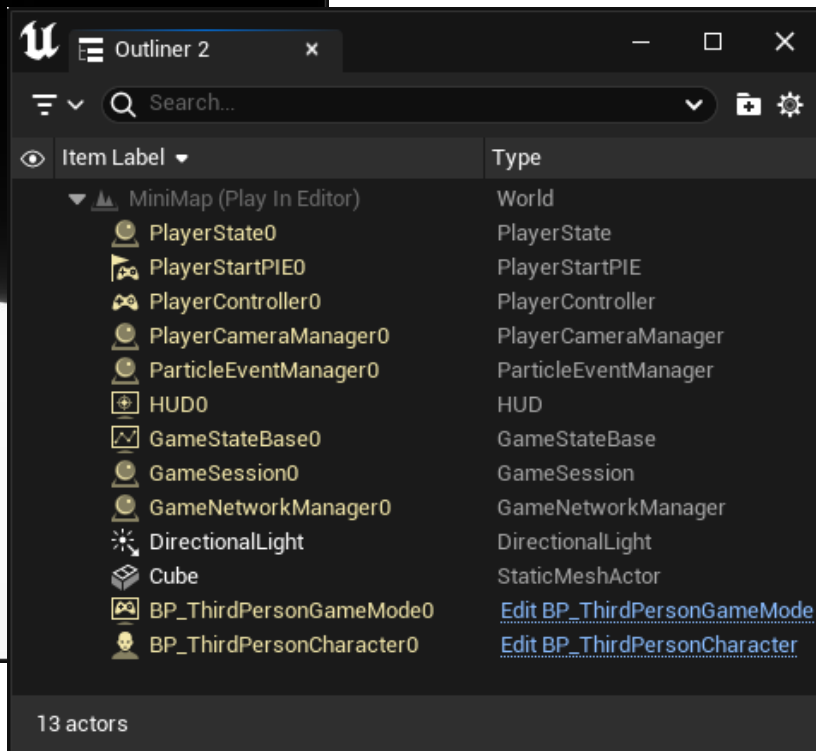


단 두 개의 액터로만 구성된 레벨





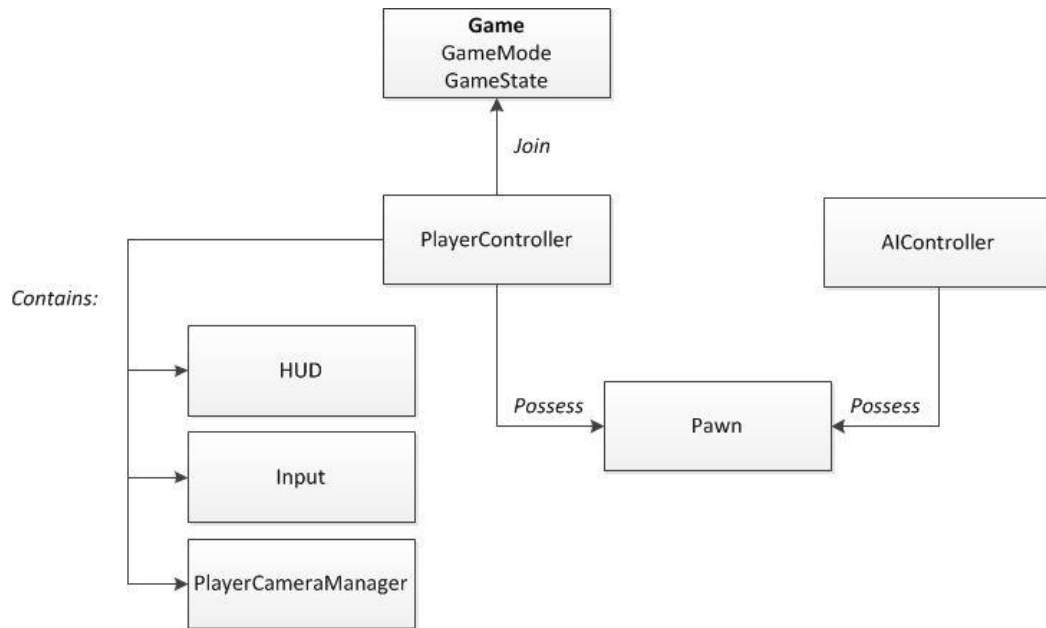
Play 하면, 많은 액터들이 내부적으로 생성된다. -> 엔진이 생성하는 것임.



게임 플레이 프레임워크

■ 게임 실행에 필요한 각 요소들의 역할과 구성 관계

- 액터
- 카메라
- 폰
- 컨트롤러
- HUD
- 입력
- 게임 모드



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.

PlayerController

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



Pawn와 PlayerController의 관계

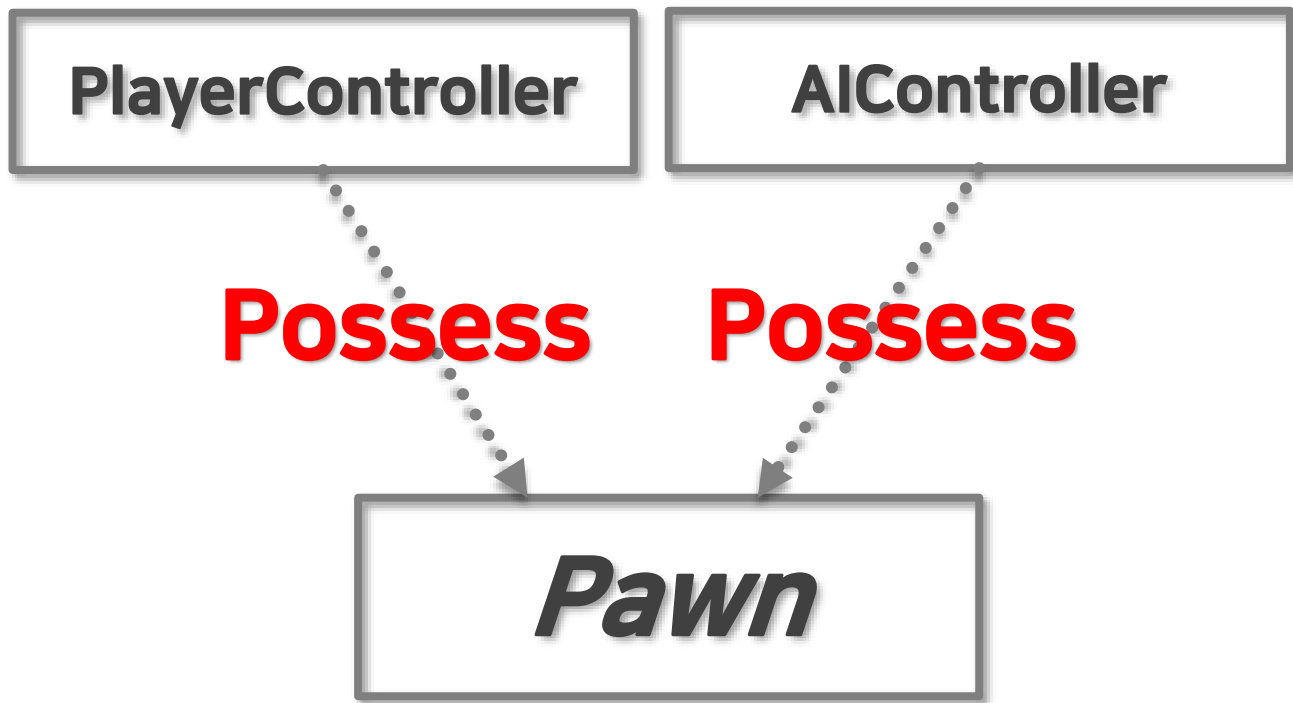


컨트롤러

캐릭터

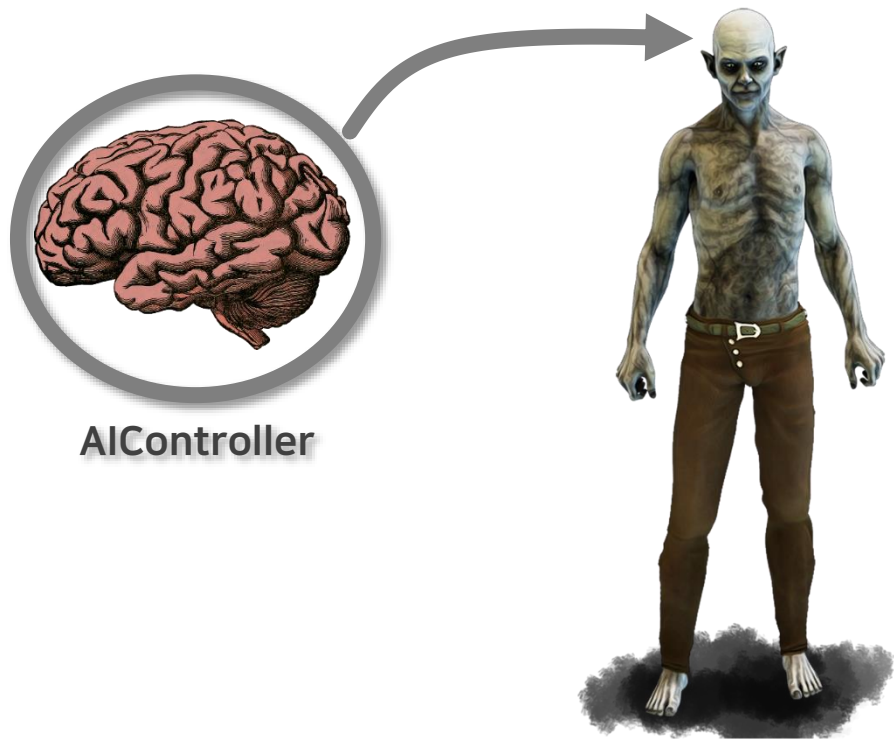


게임 플레이 프레임워크 핵심 구조 - 지배(Possess) 구조



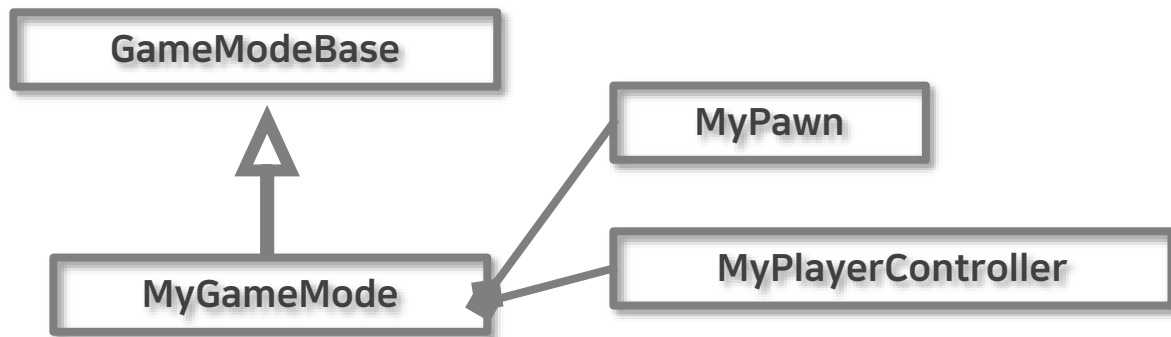
AIController

- NPC 를 제어하는 인공지능 컨트롤러

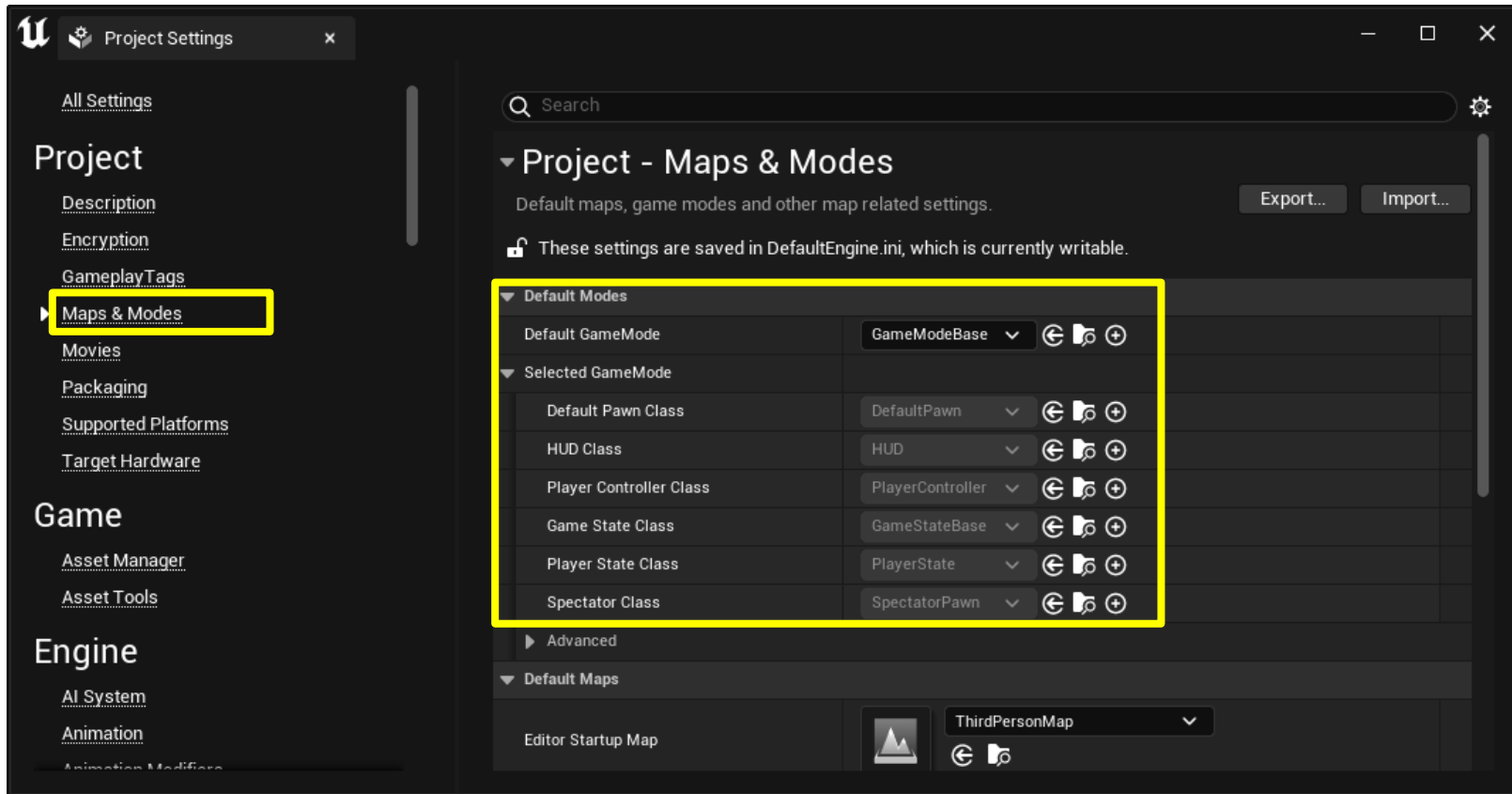


GameMode

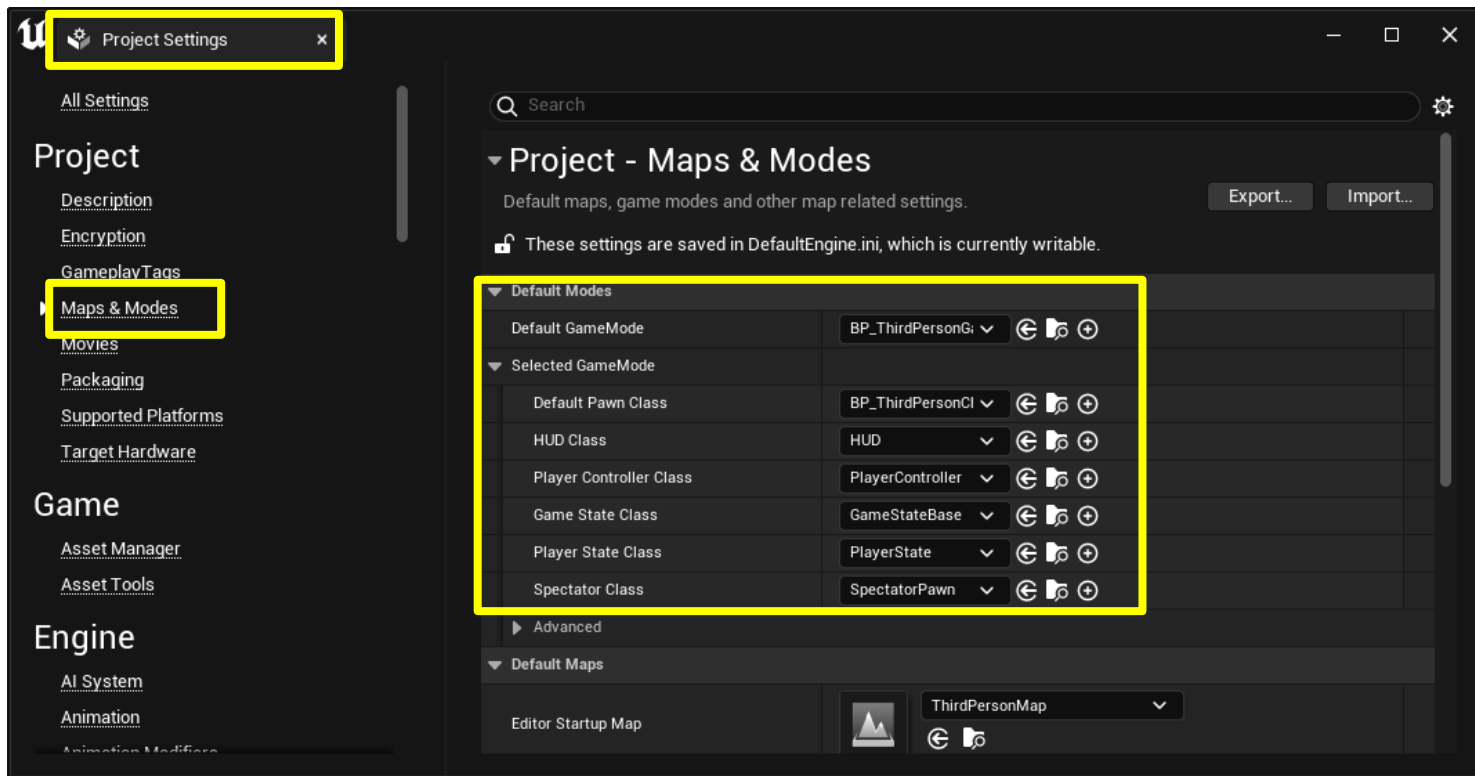
- 프레임워크를 구성하는 클래스들을 지정.
- 레벨마다 한 개의 GameMode가 설정됨.



Project의 기본 게임 모드 설정



Third Person 샘플 프로젝트의 게임 모드



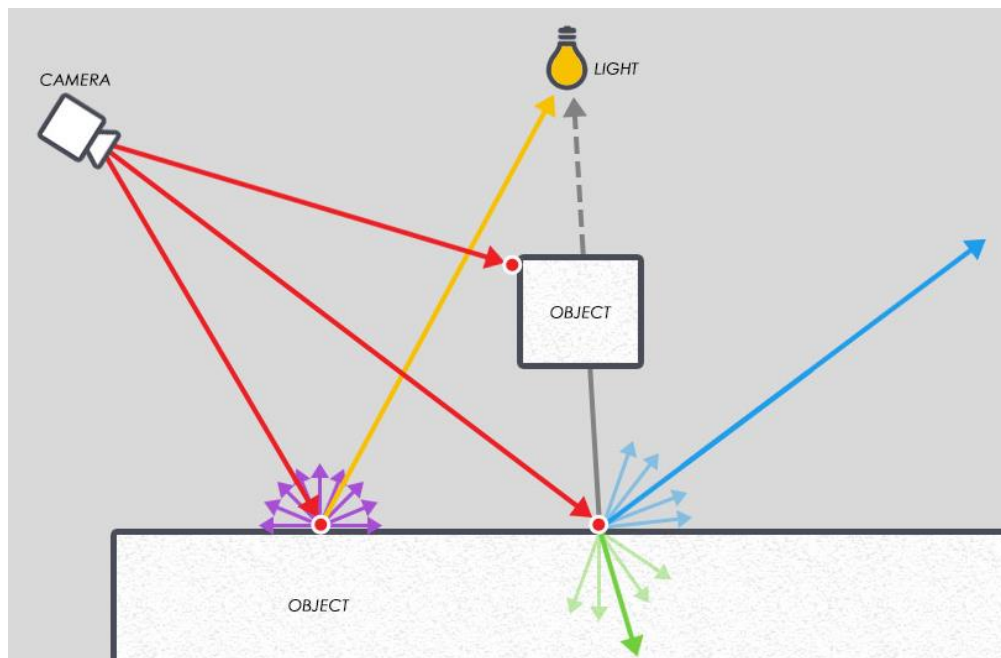


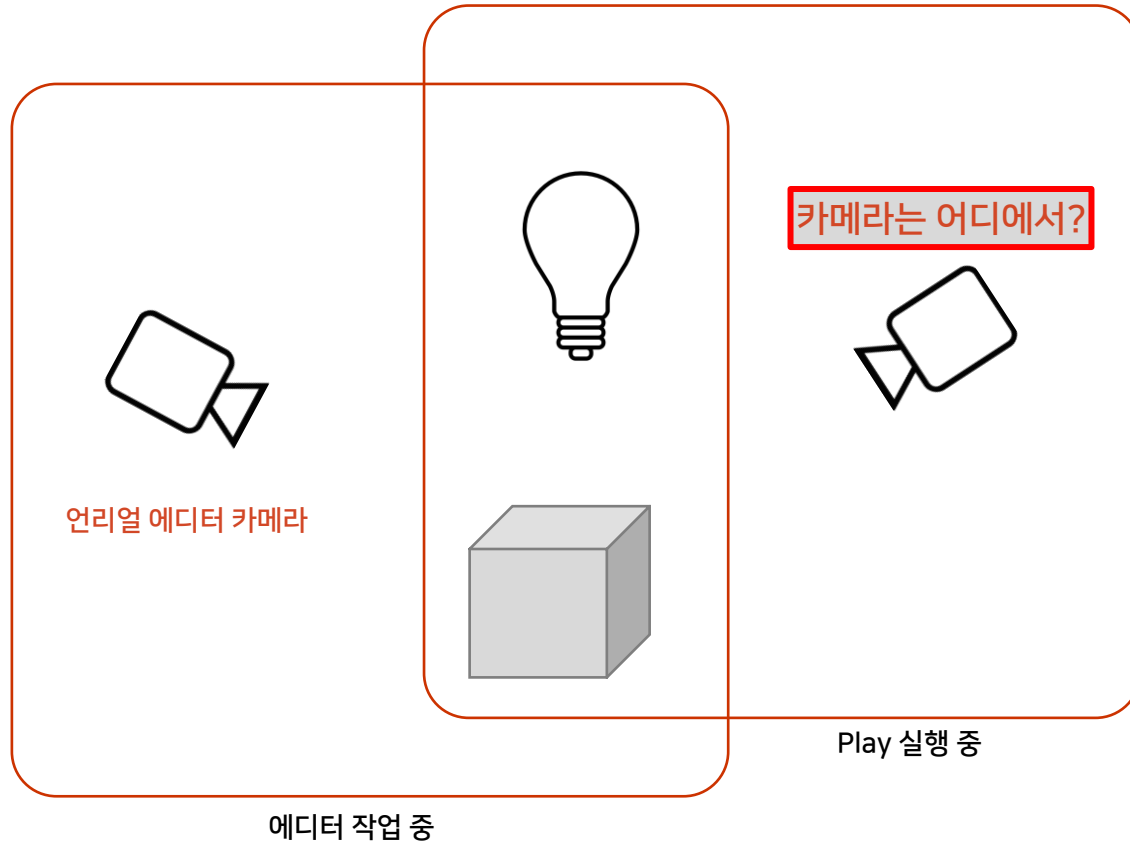
실습 LAB

게임 모드 구성 실습

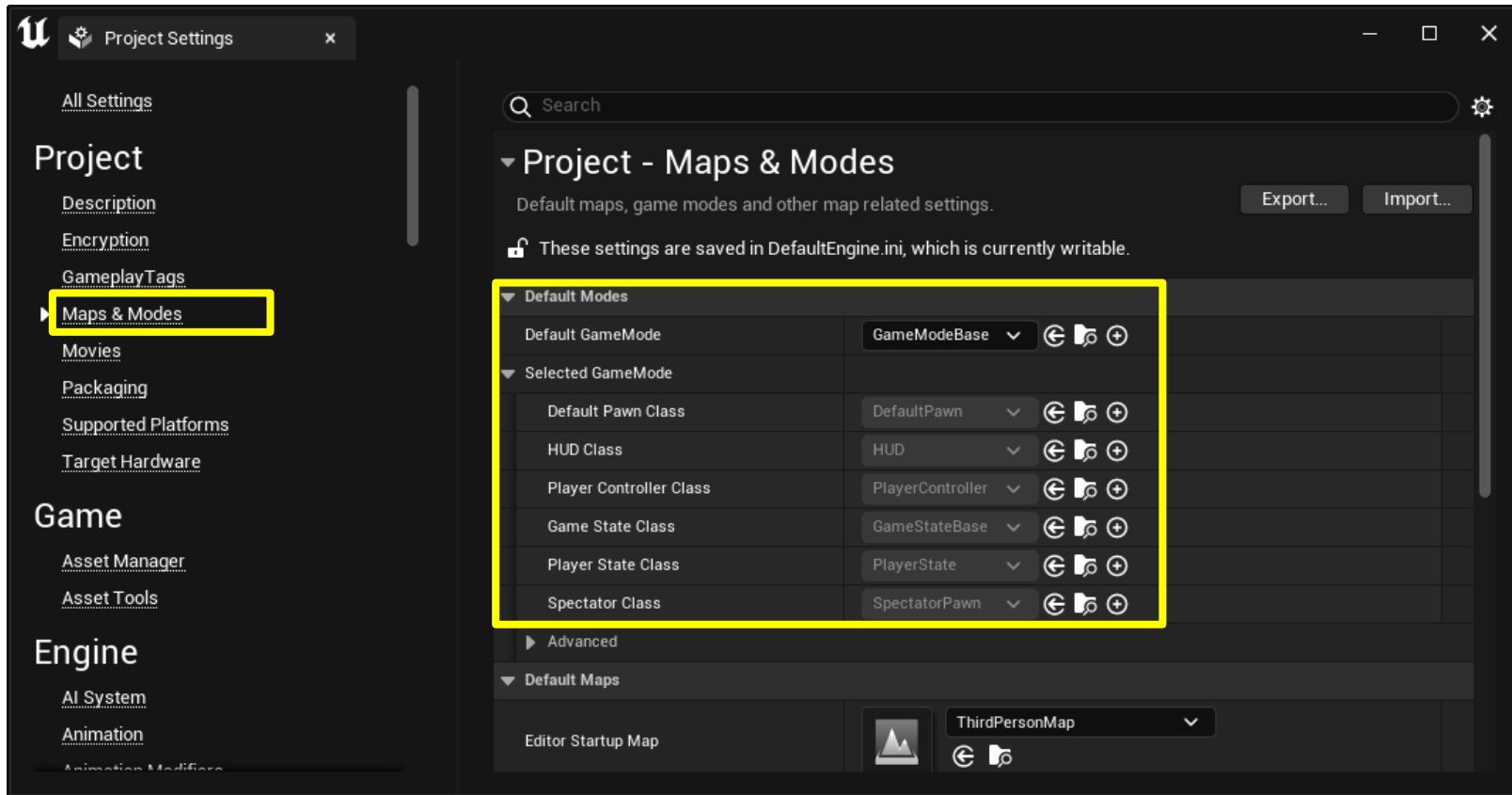
3D 렌더링 기본 구성 요소

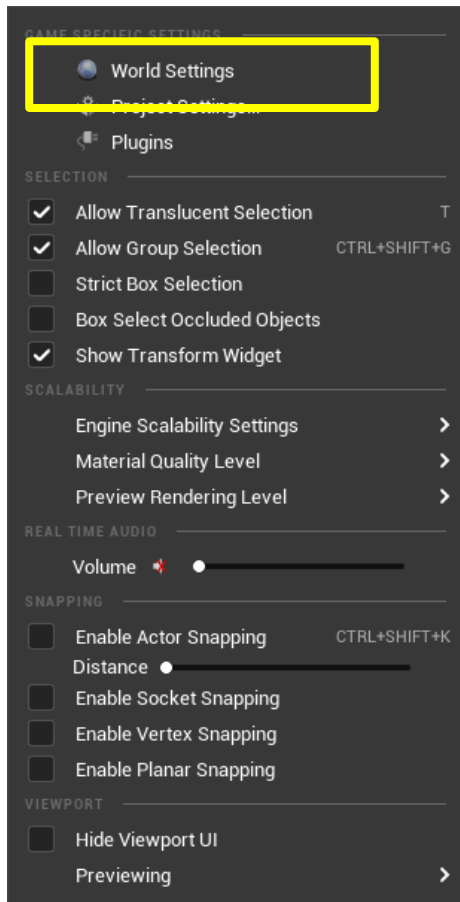
- 씬(Scene) - 여러개의 객체(Object)들로 구성
- 광원(Light)
- 카메라(Camera)



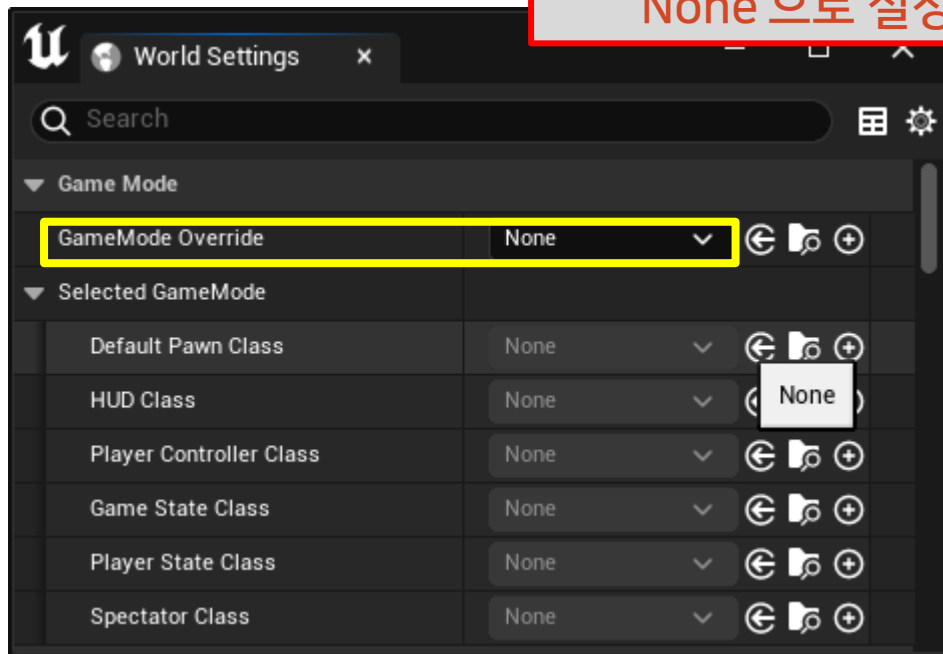


Project의 기본 게임 모드 확인 - GameModeBase



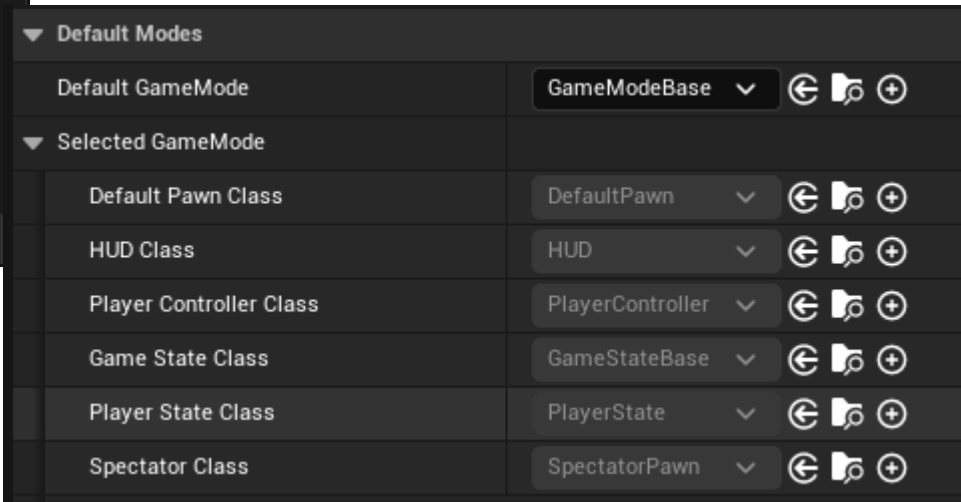


World Settings 의
GameMode 확인
'None'으로 설정

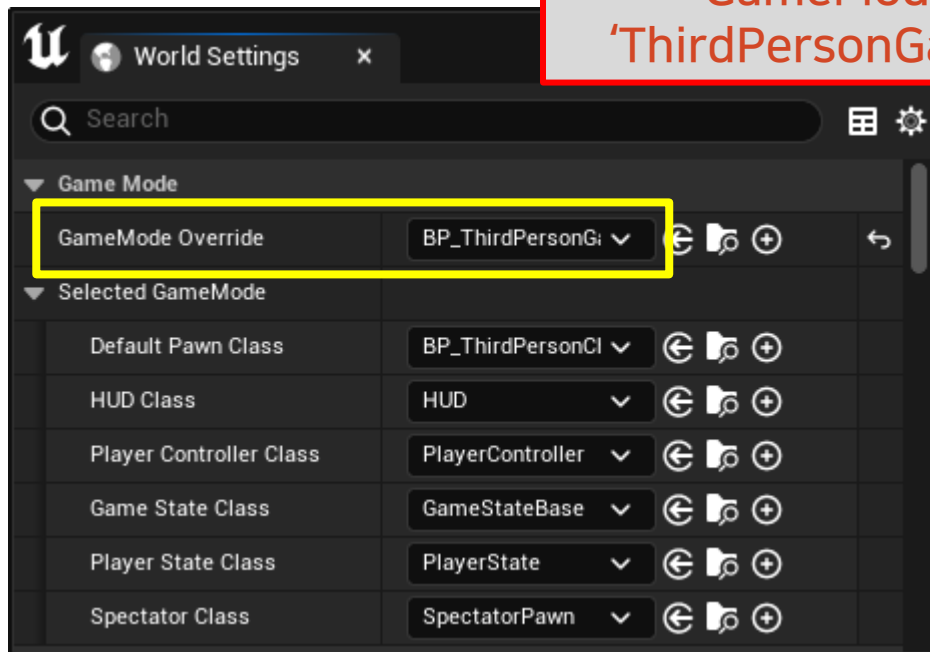


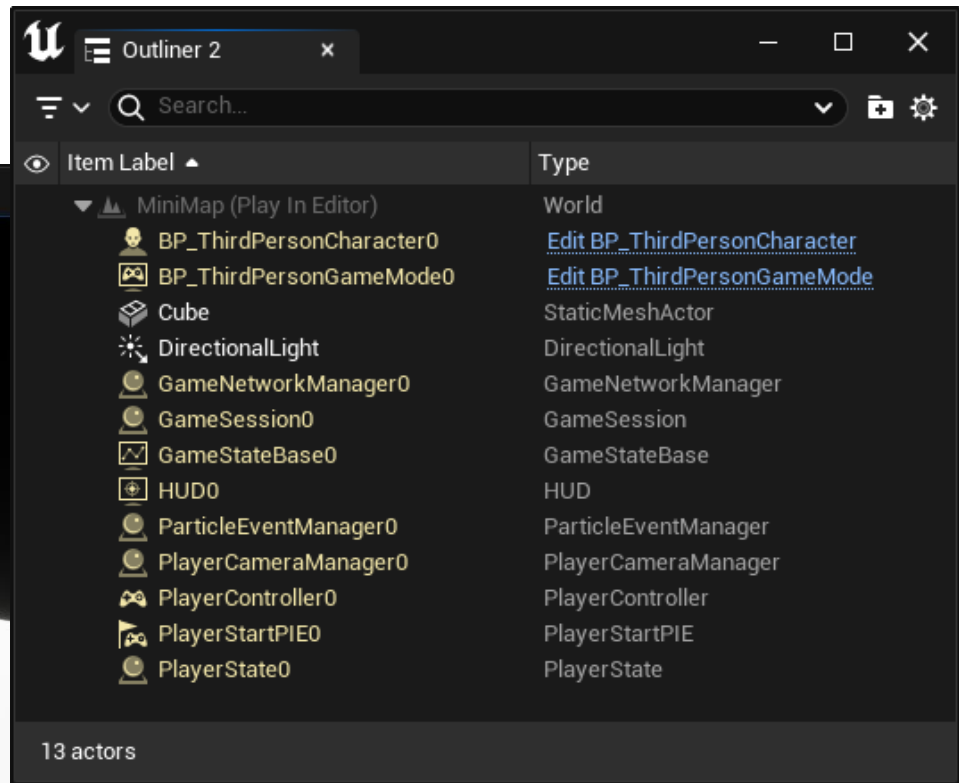
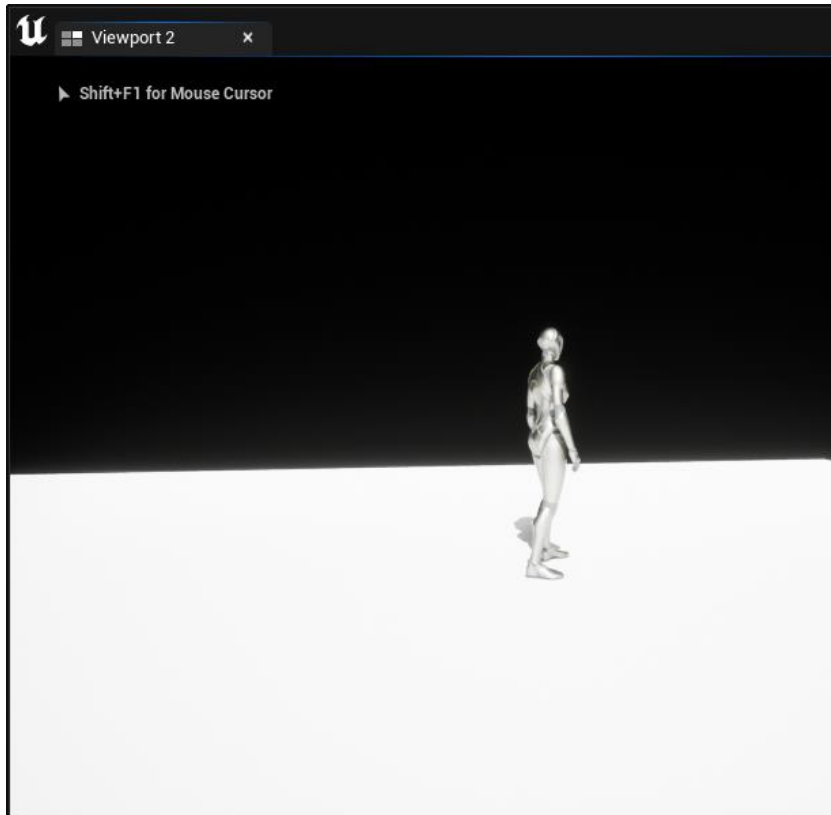


GameModeBase 액터가 생성됨.
 GameModeBase에서 지정한 클래스를 이용하여, 각종 액터들이 생성됨.
 DefaultPawn이 생성됨.
 PlayController 는 PlayerCameraManager를 갖게 됨.
 PlayerCameraManager는 자동생성된 CameraActor를 갖게 됨.



GameMode Override 를
'ThirdPersonGameMode'로 설정.



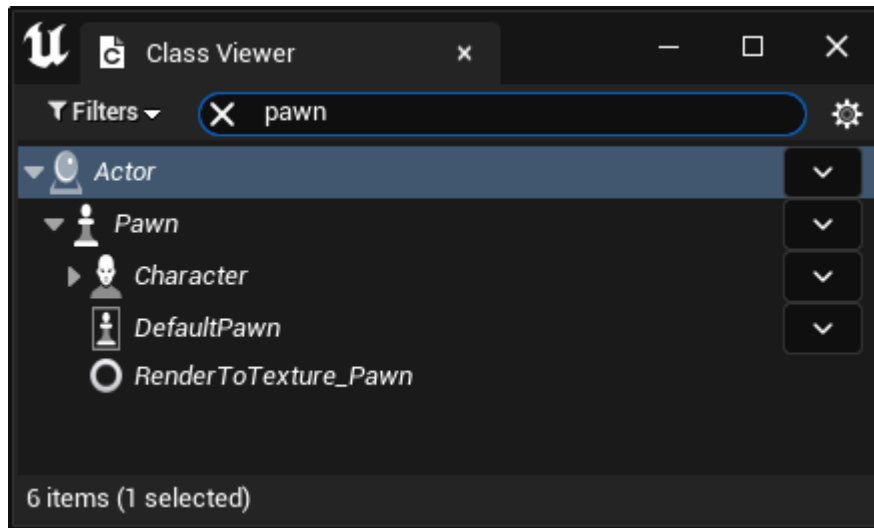


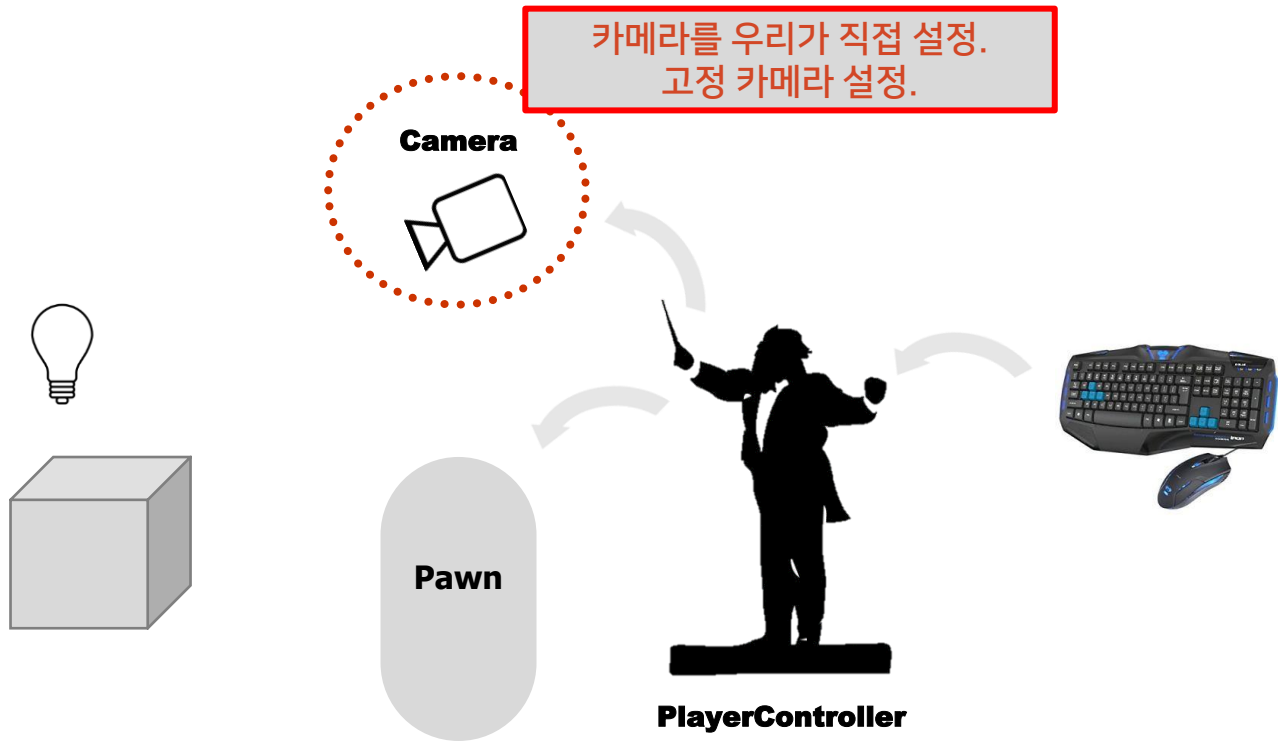


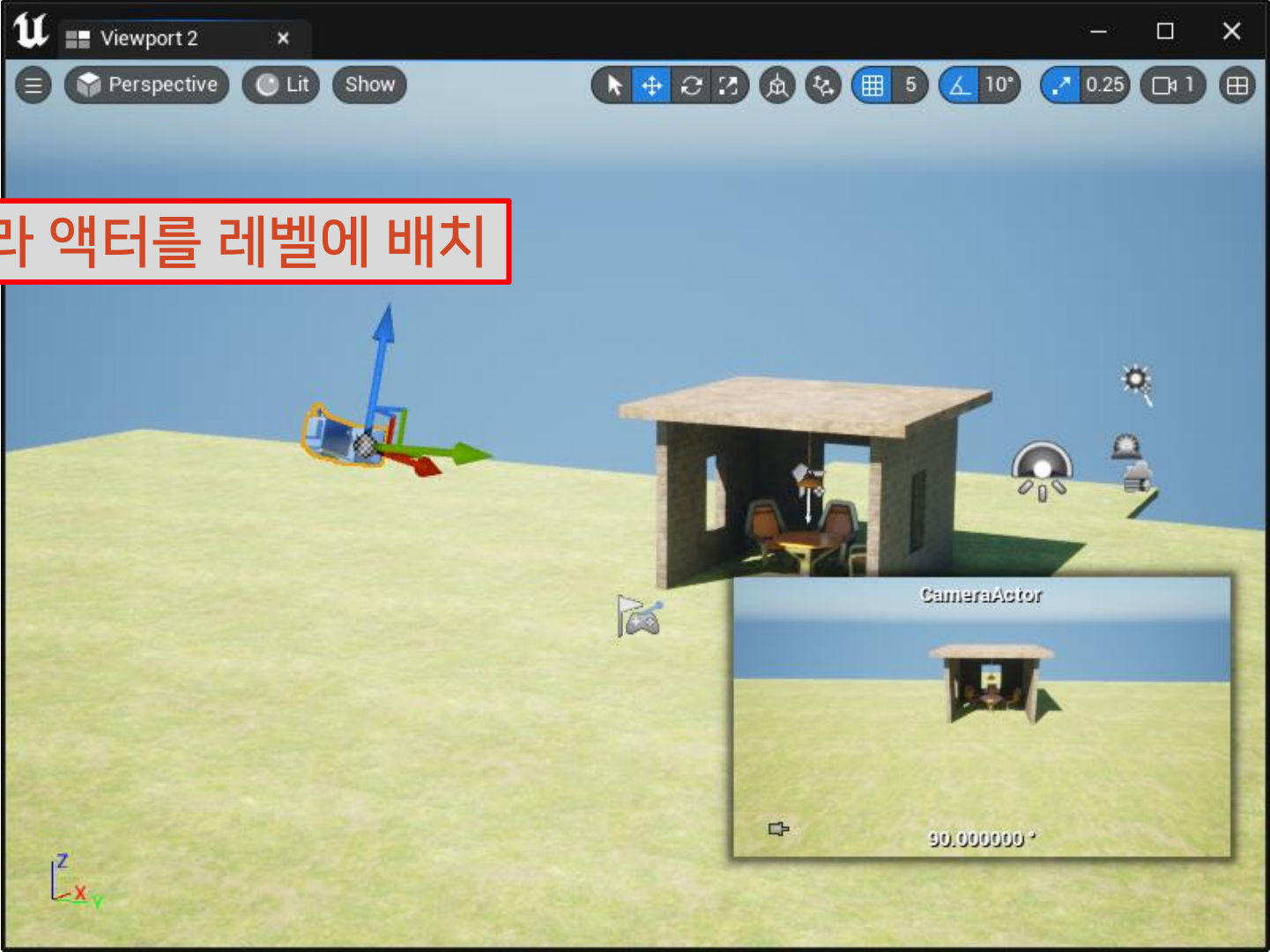
실습 LAB

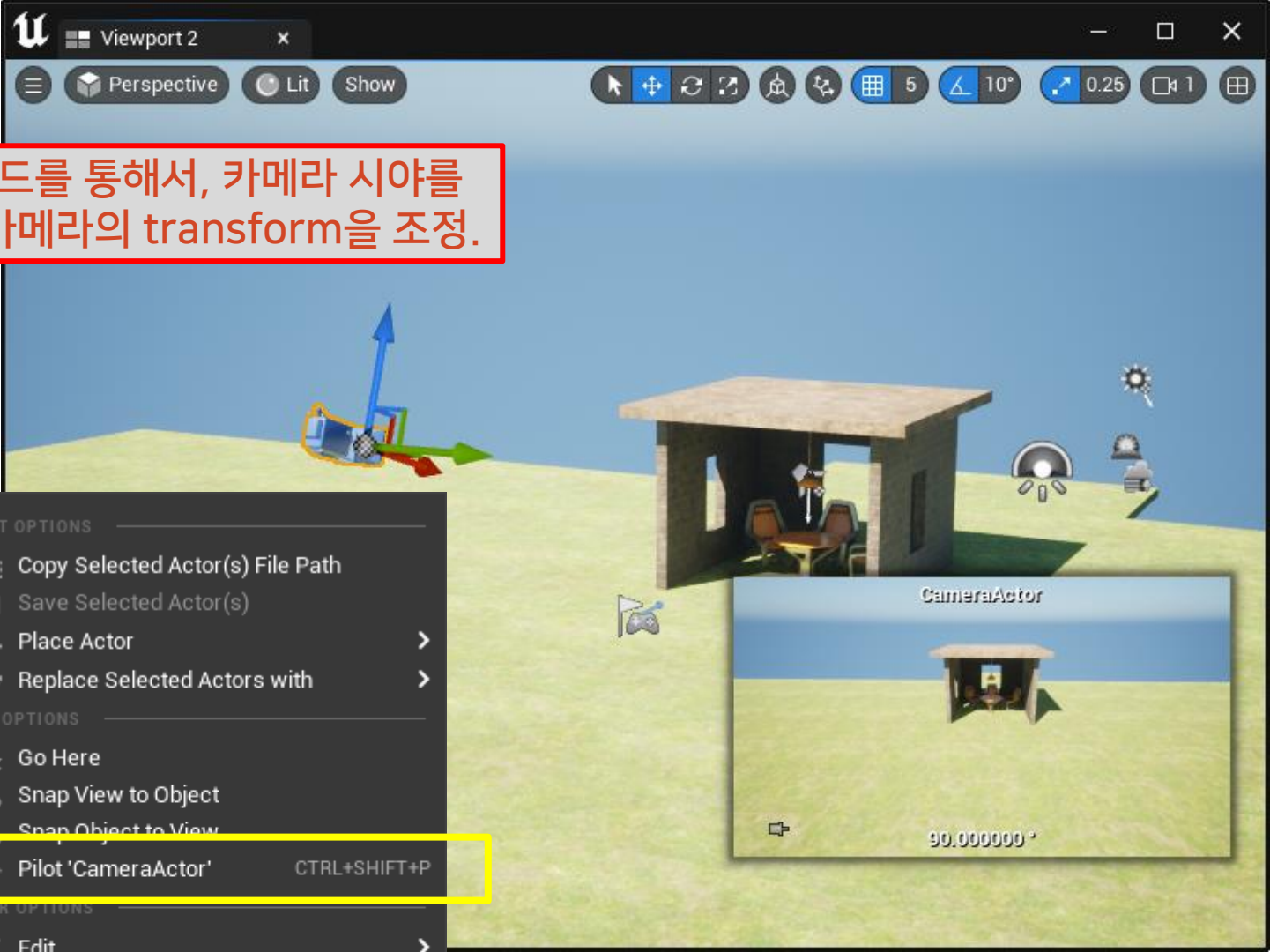
고정 카메라 설정

Actor – Pawn – Character, DefaultPawn



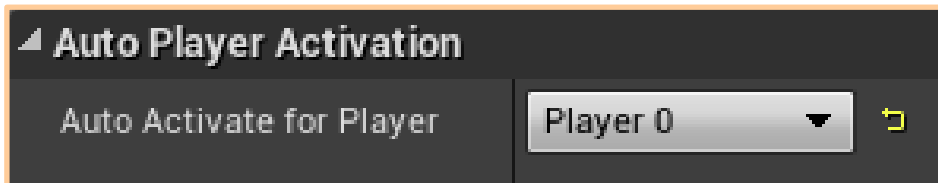






Pilot 모드를 통해서, 카메라 시야를 보면서, 카메라의 transform을 조정.





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.

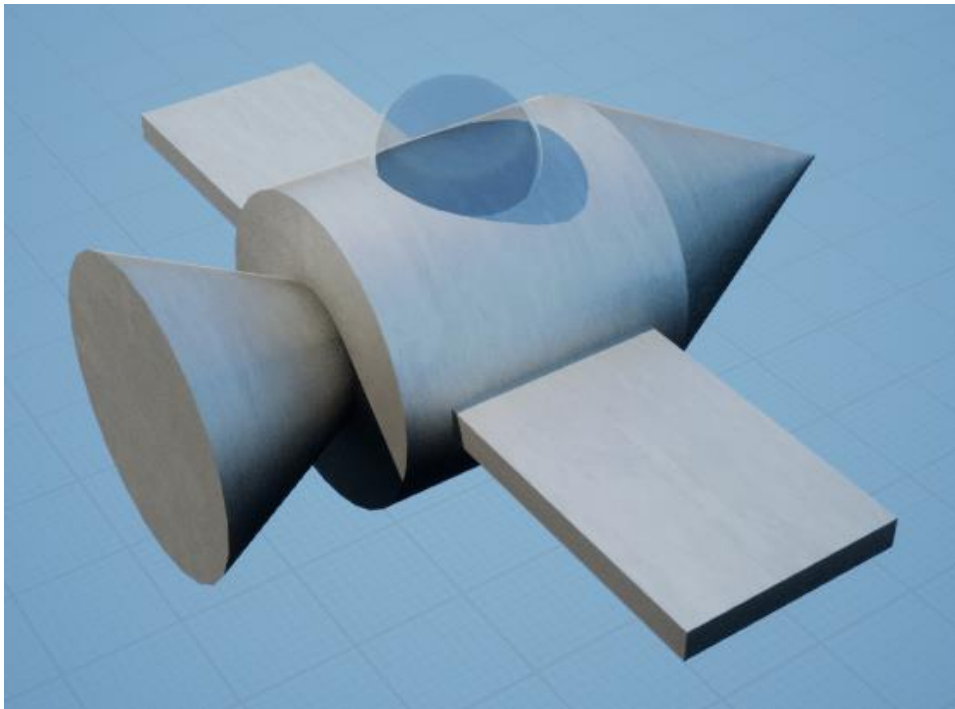


실습 LAB

Aircraft Blueprint 제작 및 조종

Aircraft Blueprint

- WASD 키를 통해서 전후진, 좌우 이동

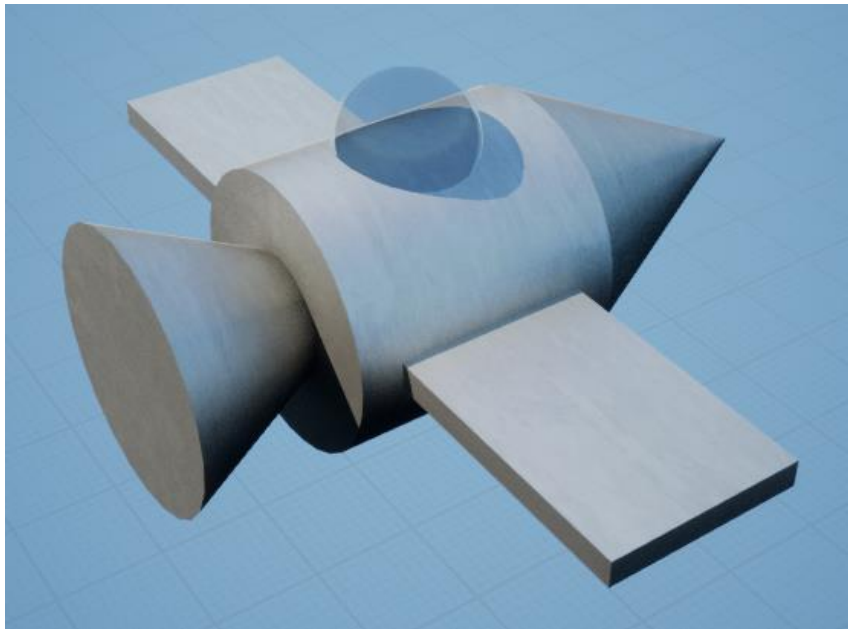


블루프린트 설계

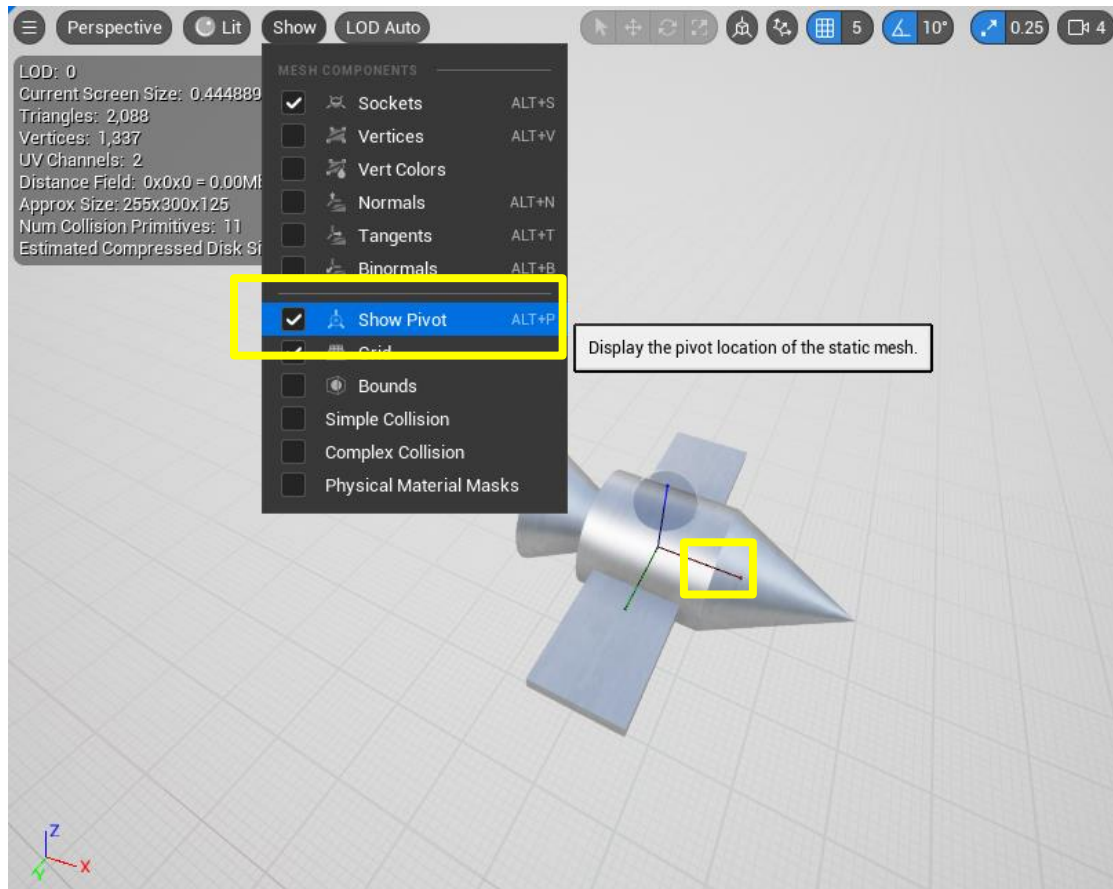
- 그대로 사용할 수 있는 블루프린트 있는가? NO
- 비슷한 블루프린트가 있는가? YES
 - "DefaultPawn"
- 추가할 것만 있는가? YES
 - Static Mesh 모양만 추가 !!!

Aircraft Static Mesh 만들기

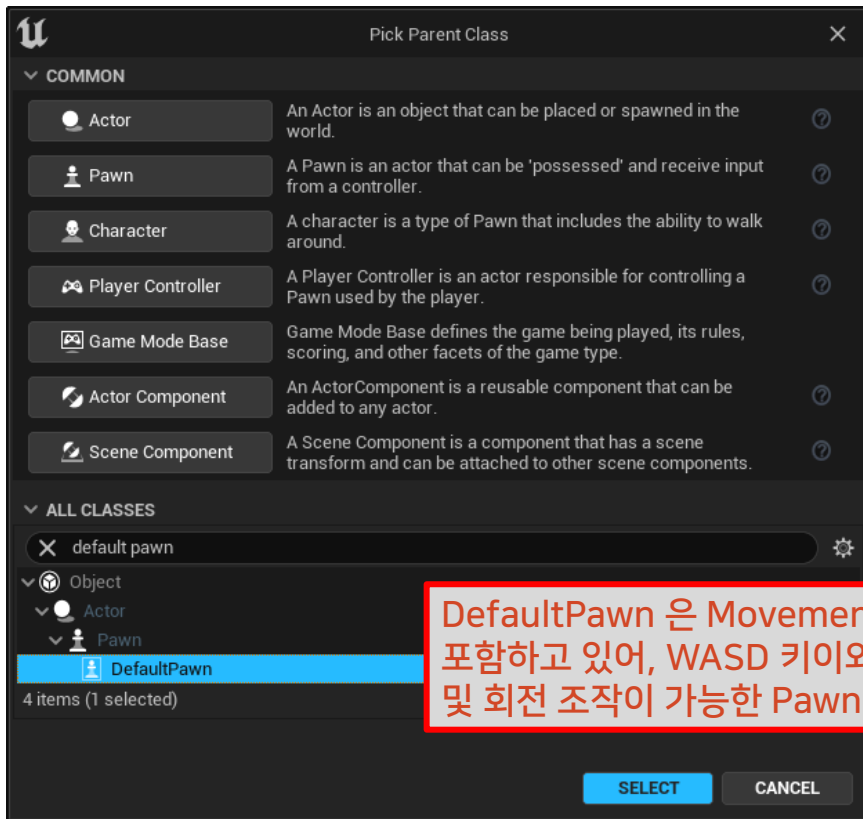
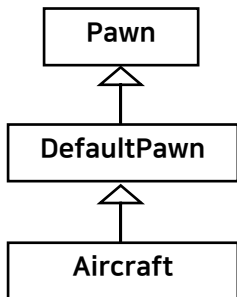
- Cube, Sphere, Cylinder, Cone 을 이용해서, 비행기 모양을 만듦.
- 액터 머징을 통해 단일 메시로 만듦. - SM_Aircraft

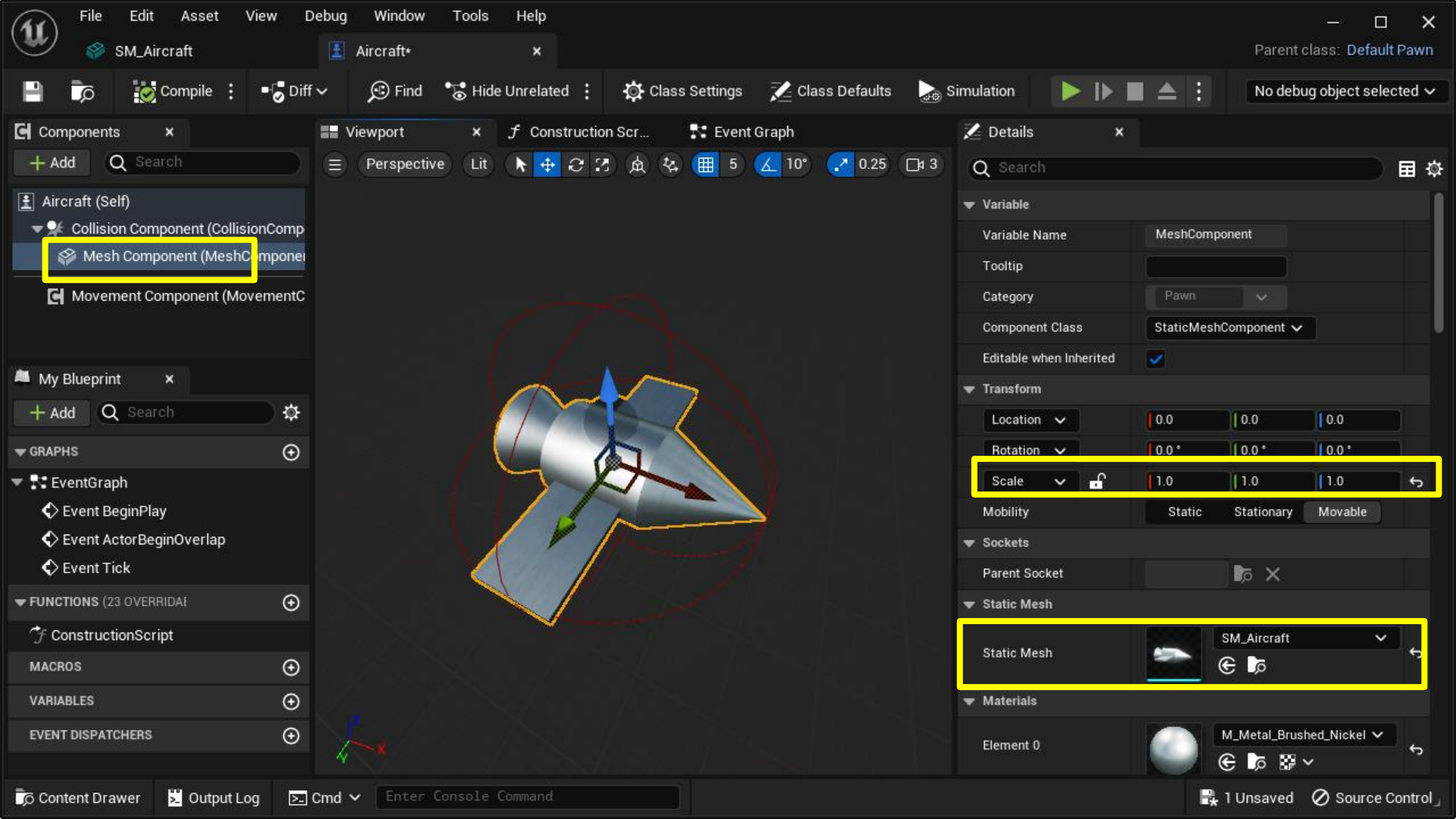


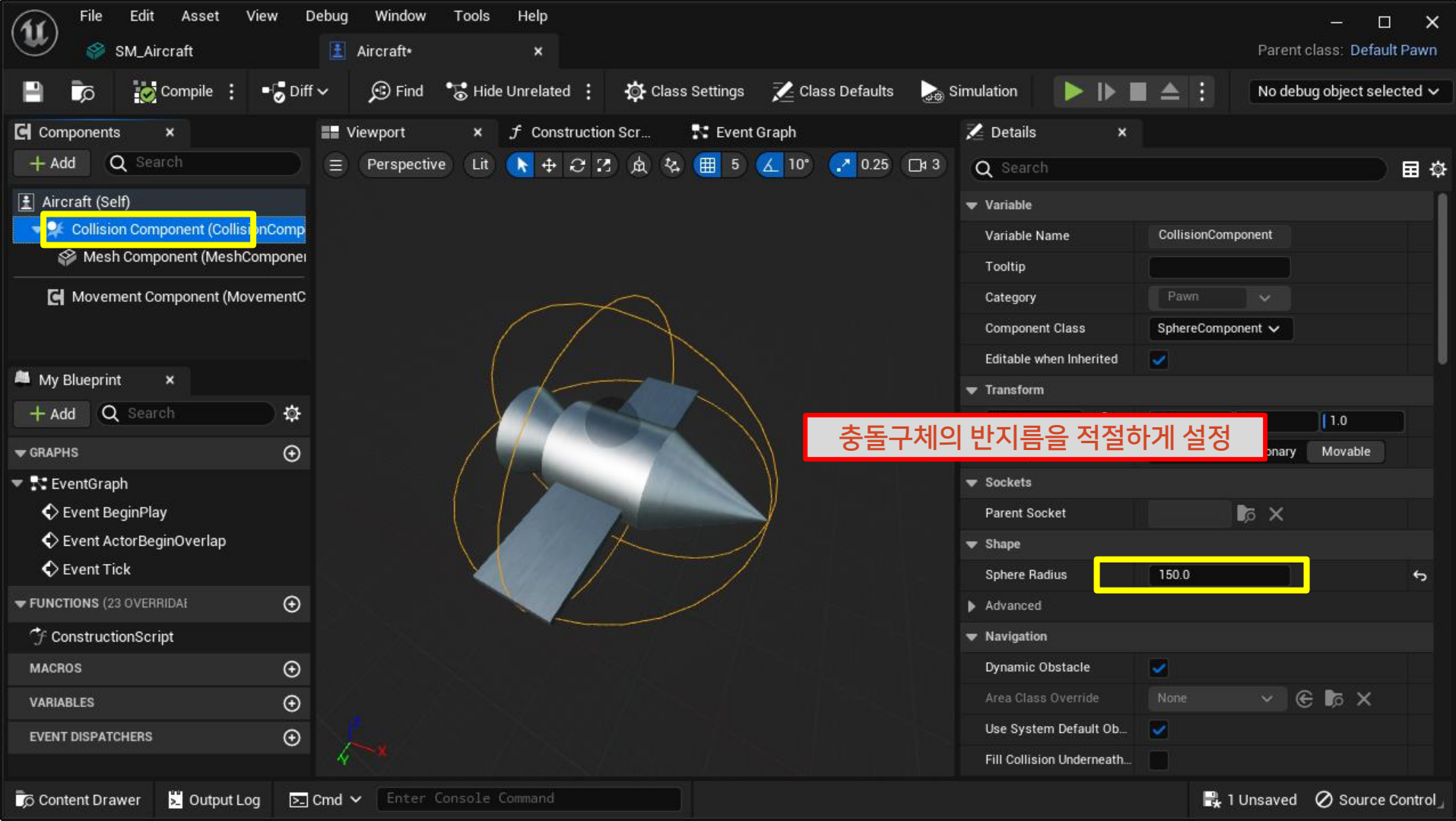
Forward Direction – X – Red 방향 확인



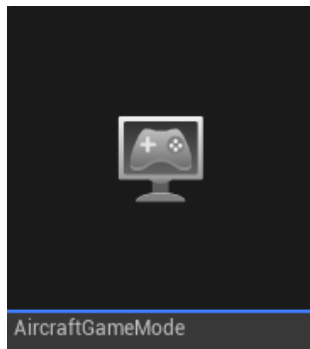
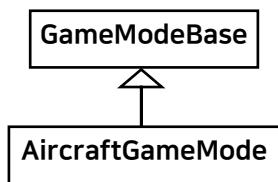
DefaultPawn 을 Parent로 하여, Blueprint를 만듦.







게임 모드 만들기 : Blueprint 클래스로 만듦.



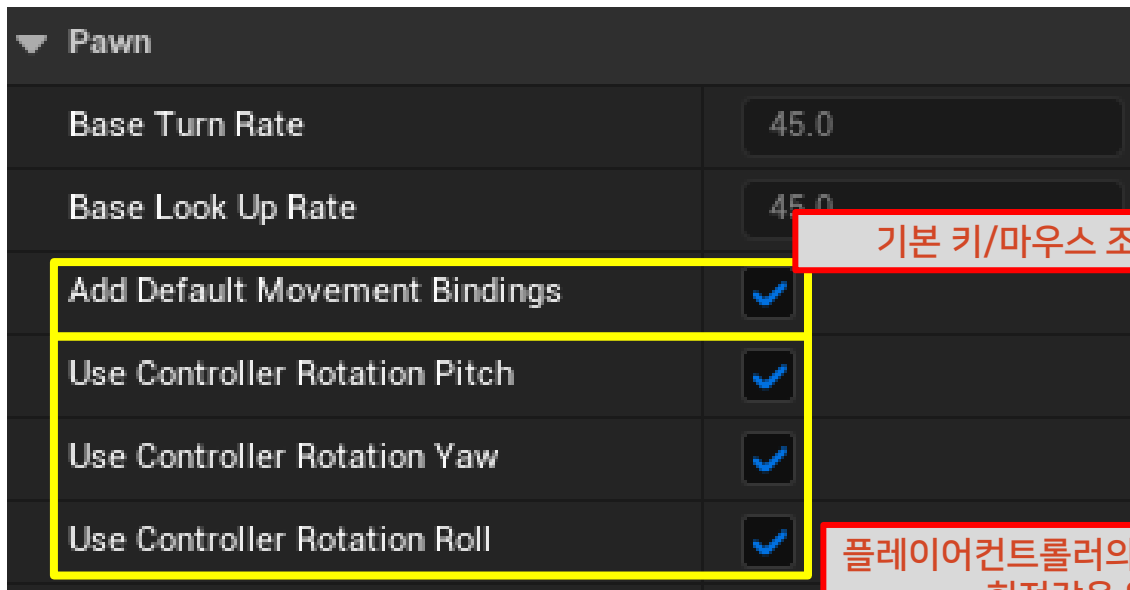
GameModeBase를 베이스클래스로
하는 블루프린트 클래스를 만듦.

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 ▼	↶	📁	⊕		

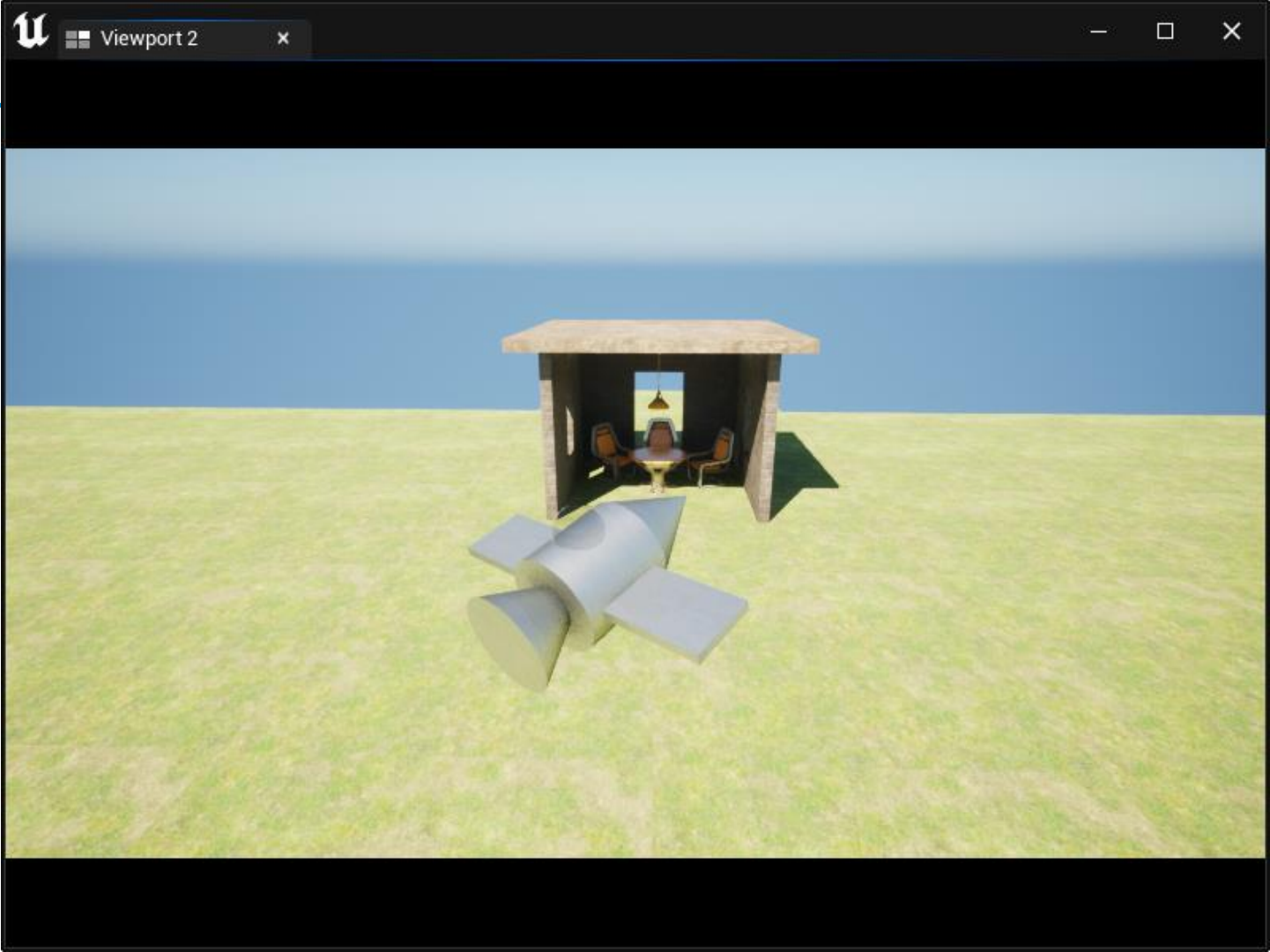
'Default Pawn Class'를
Aircraft 로 지정 !

Aircraft Blueprint 의 Class Defaults 설정



기본 키/마우스 조작 자동 적용

플레이어컨트롤러의 회전값과 Pawn의 회전값을 일치시킴.



언리얼 엔진 핵심 클래스 다이어그램

