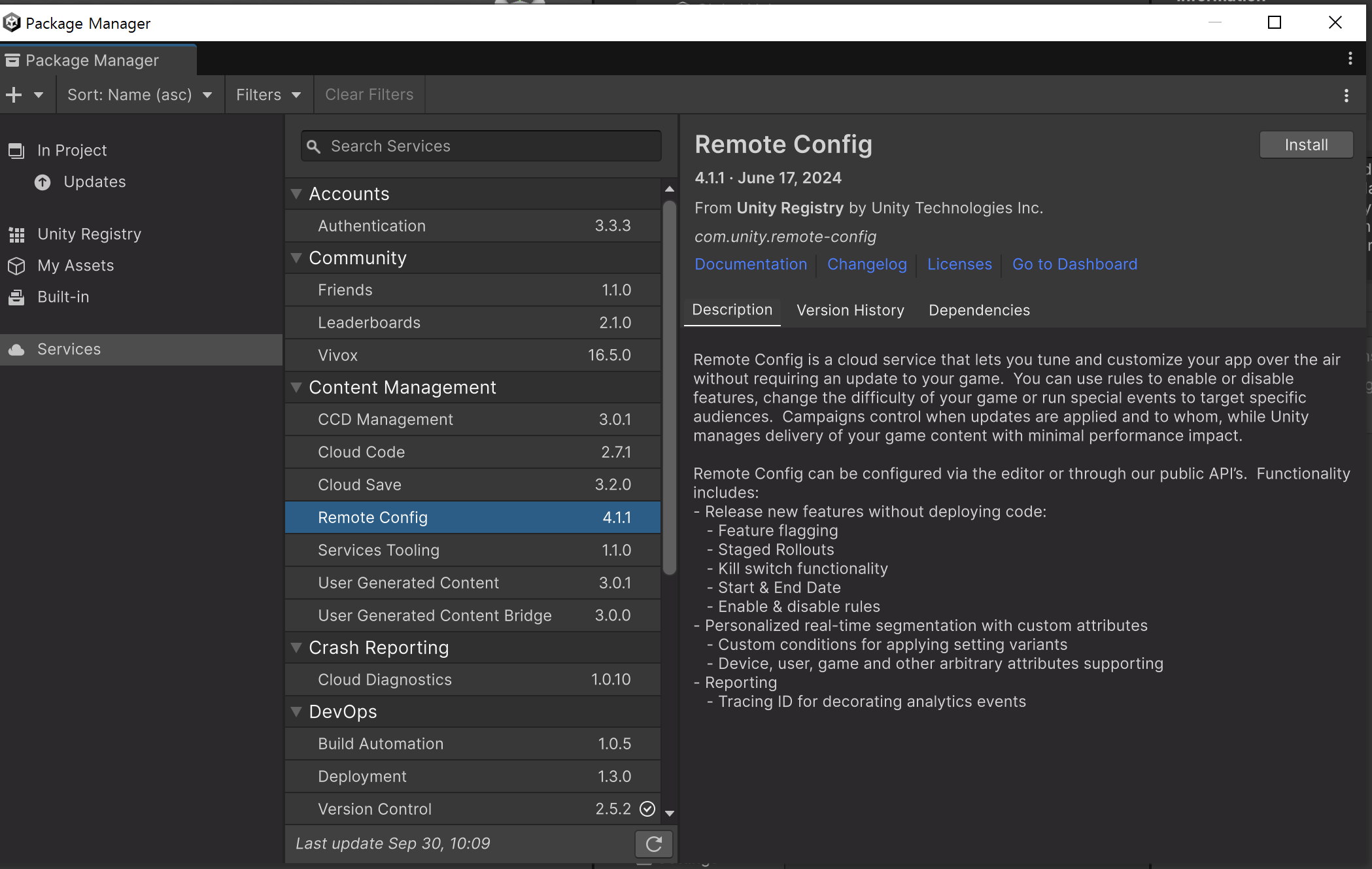
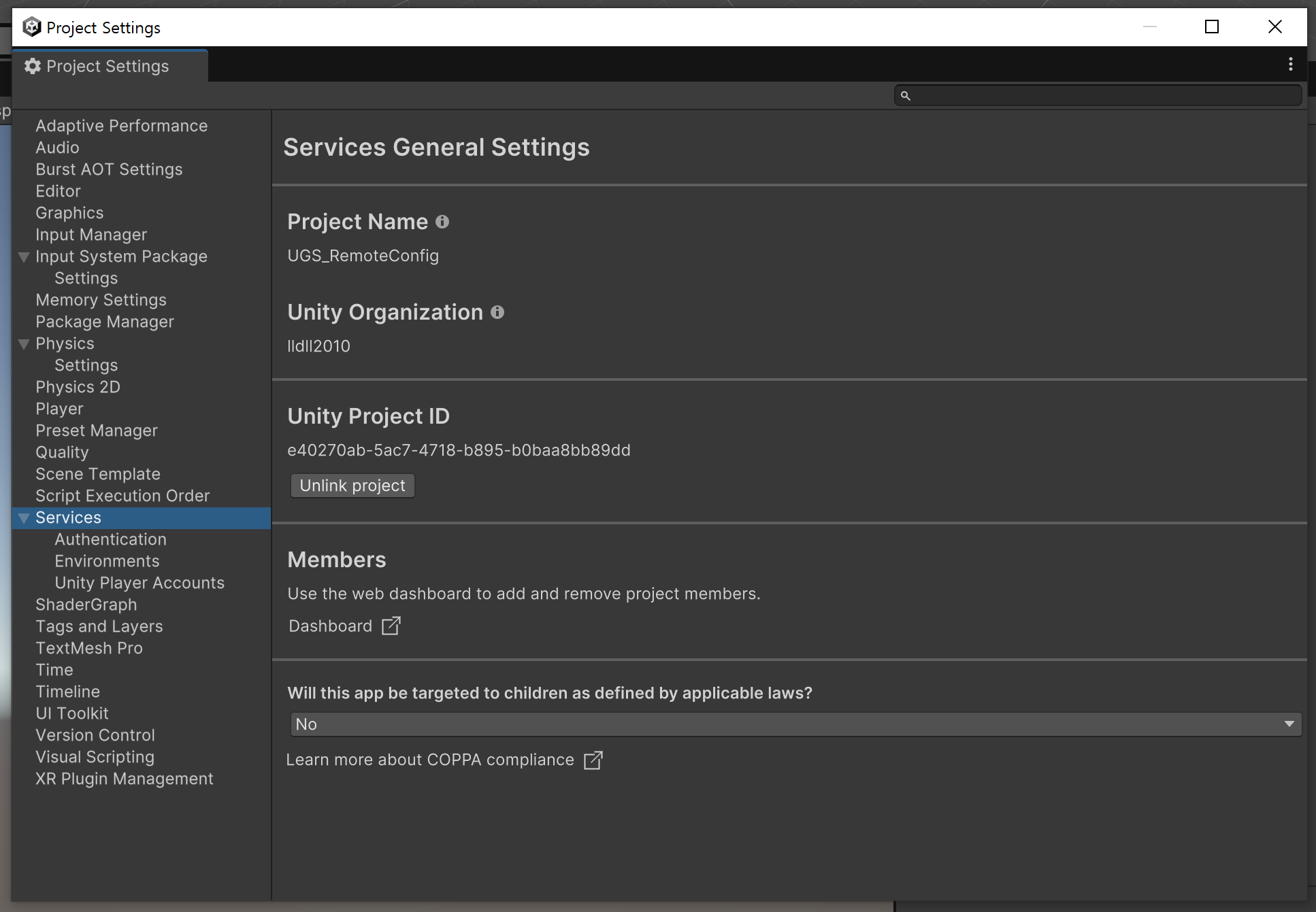


폴더 정리

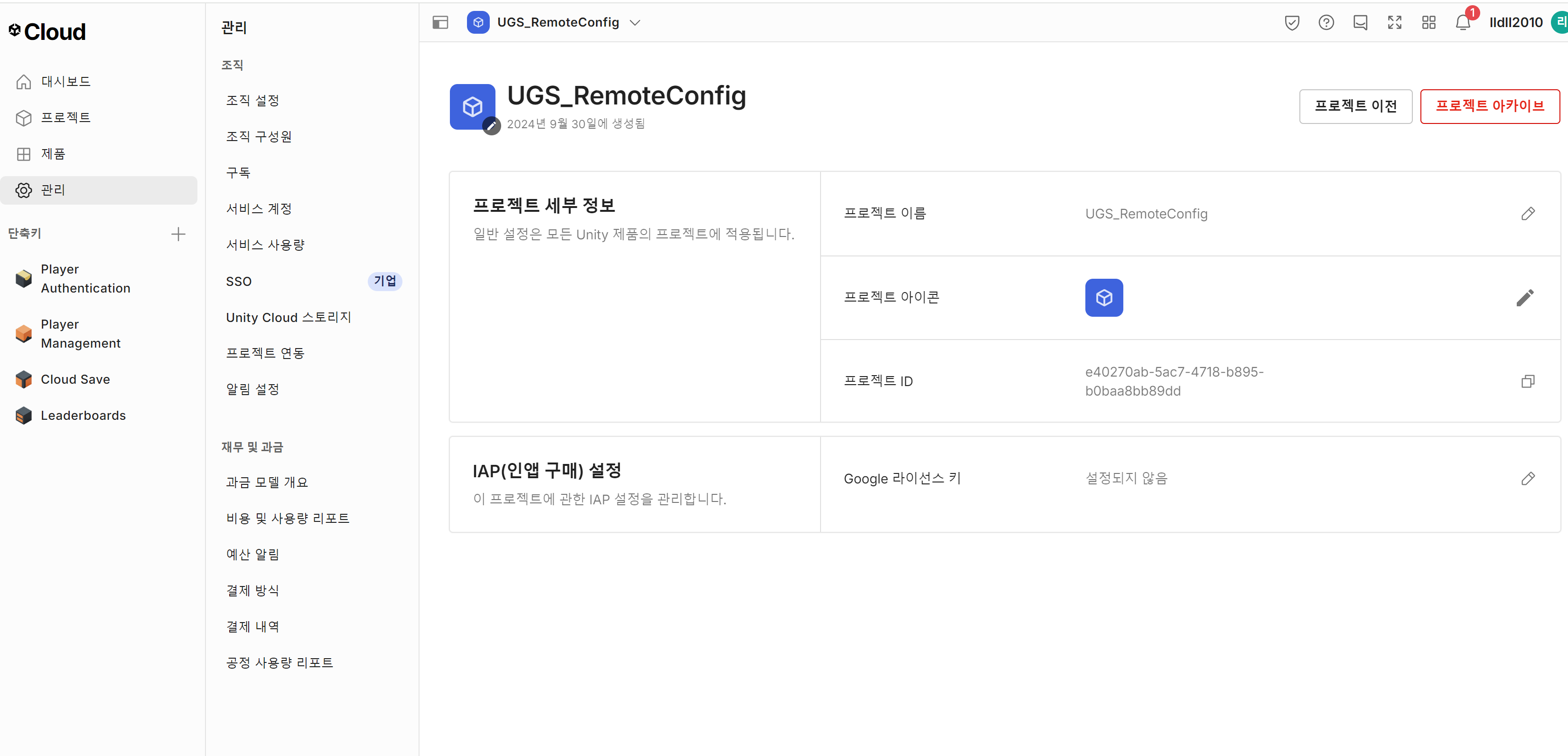


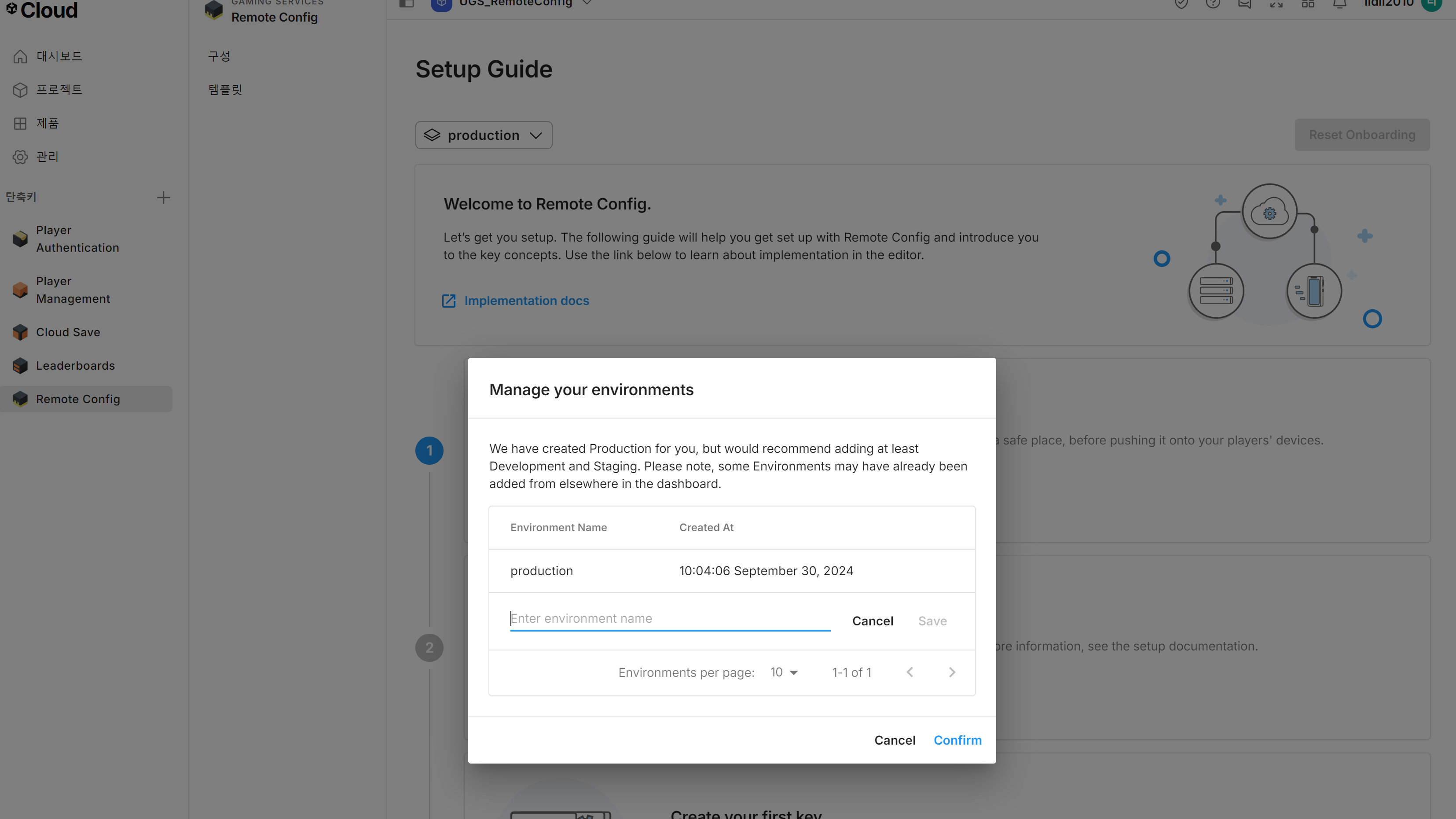
패키지 매니저 (컨트롤 + 0) remote Config, DevOps에 deployment 설치



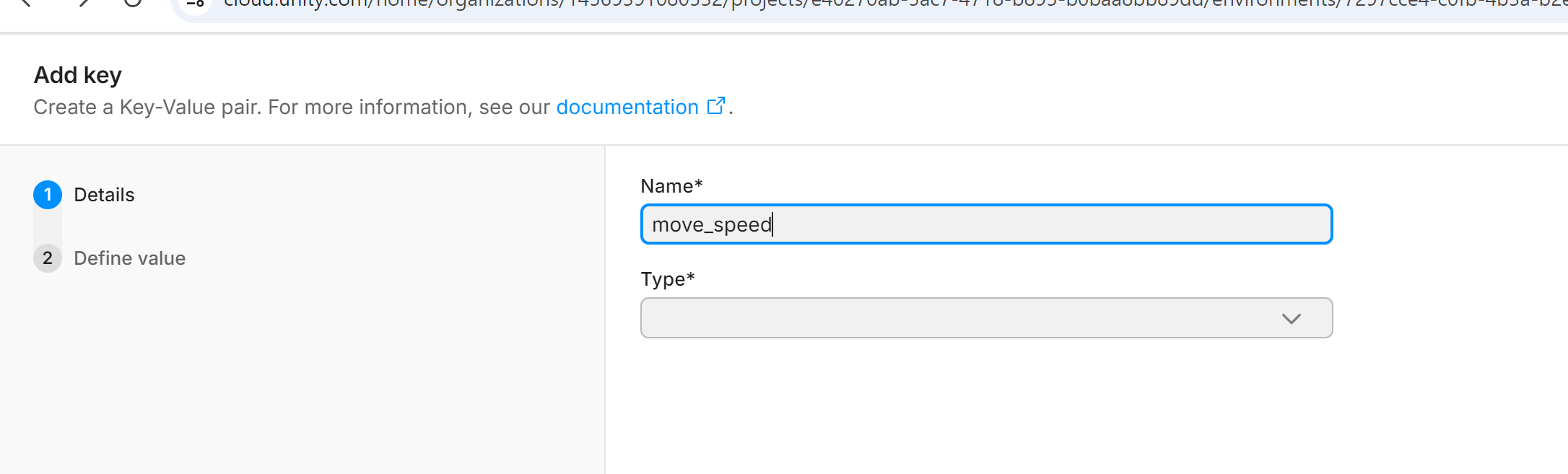
프로젝트 세팅스의 서비스에서 보면 연결되어있음

Dashborad

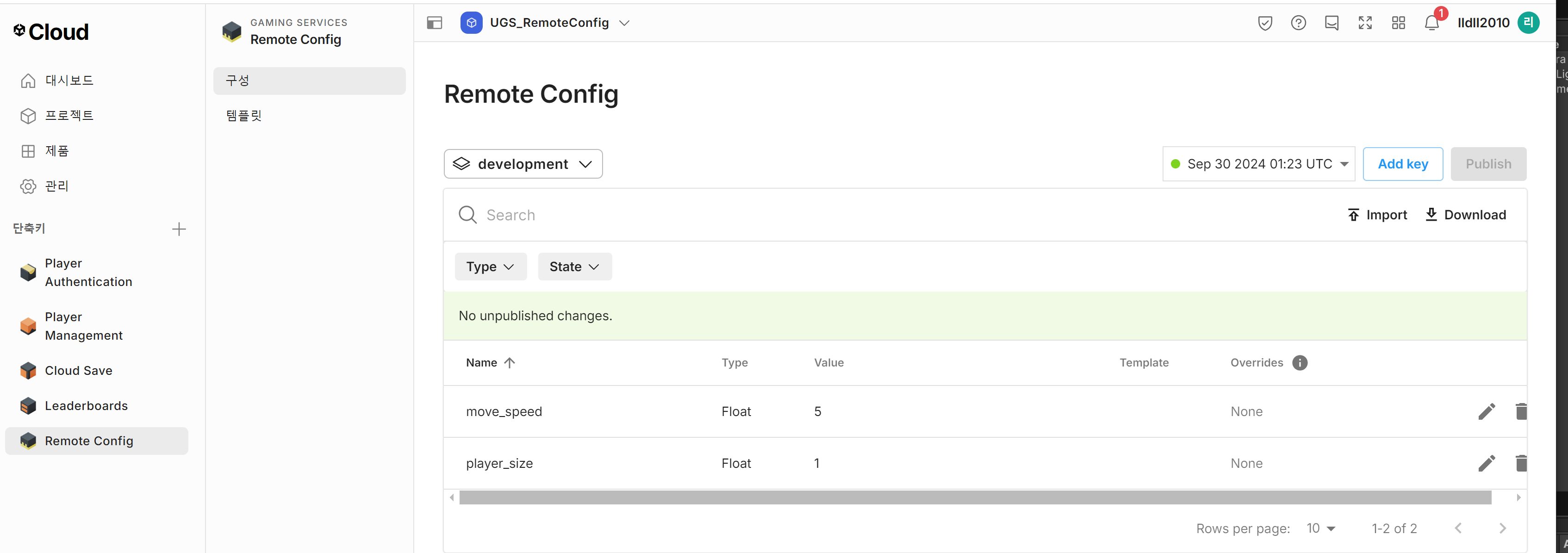




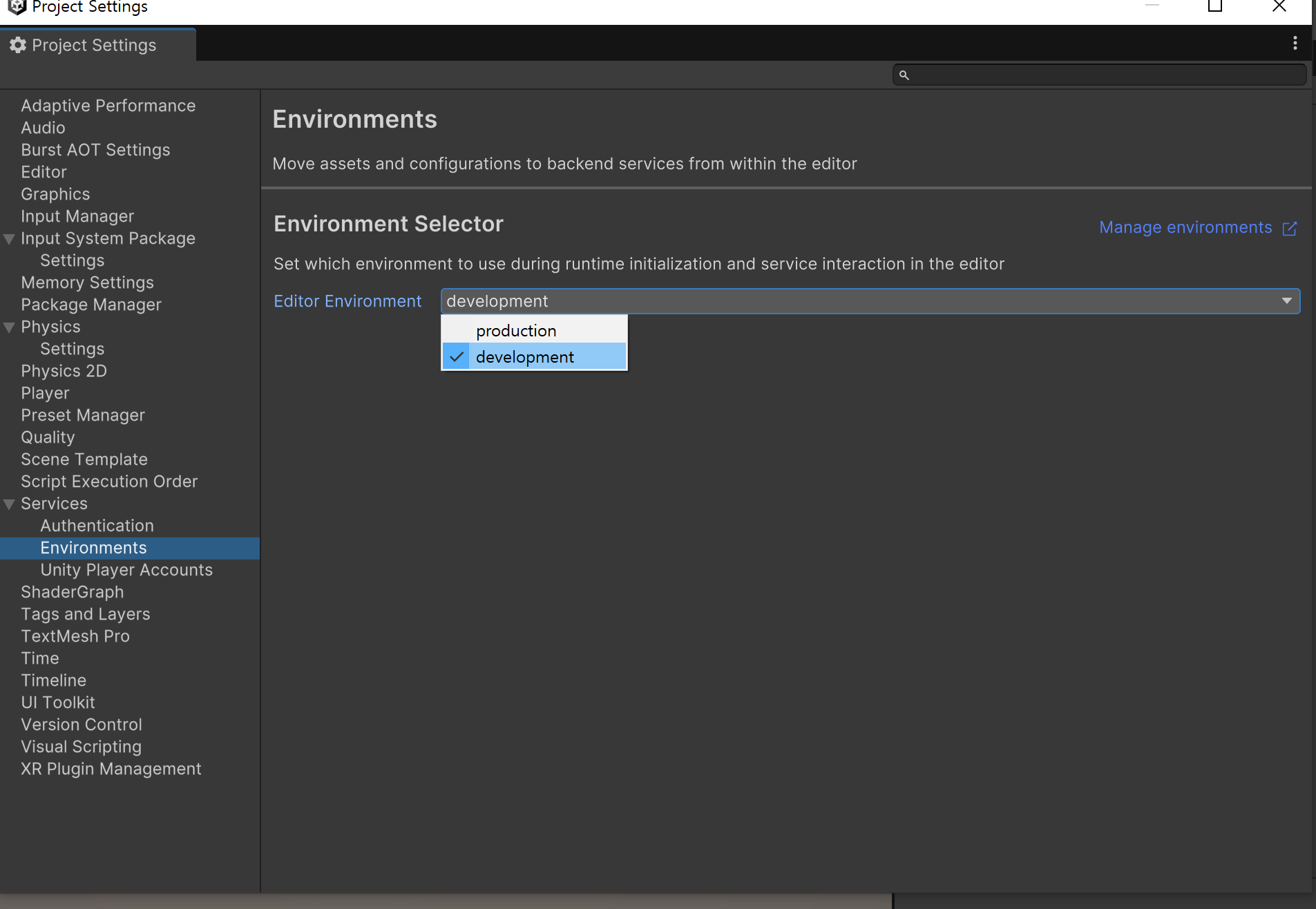
Product 의 remote config의 setup guide 찾으면 creat 있음

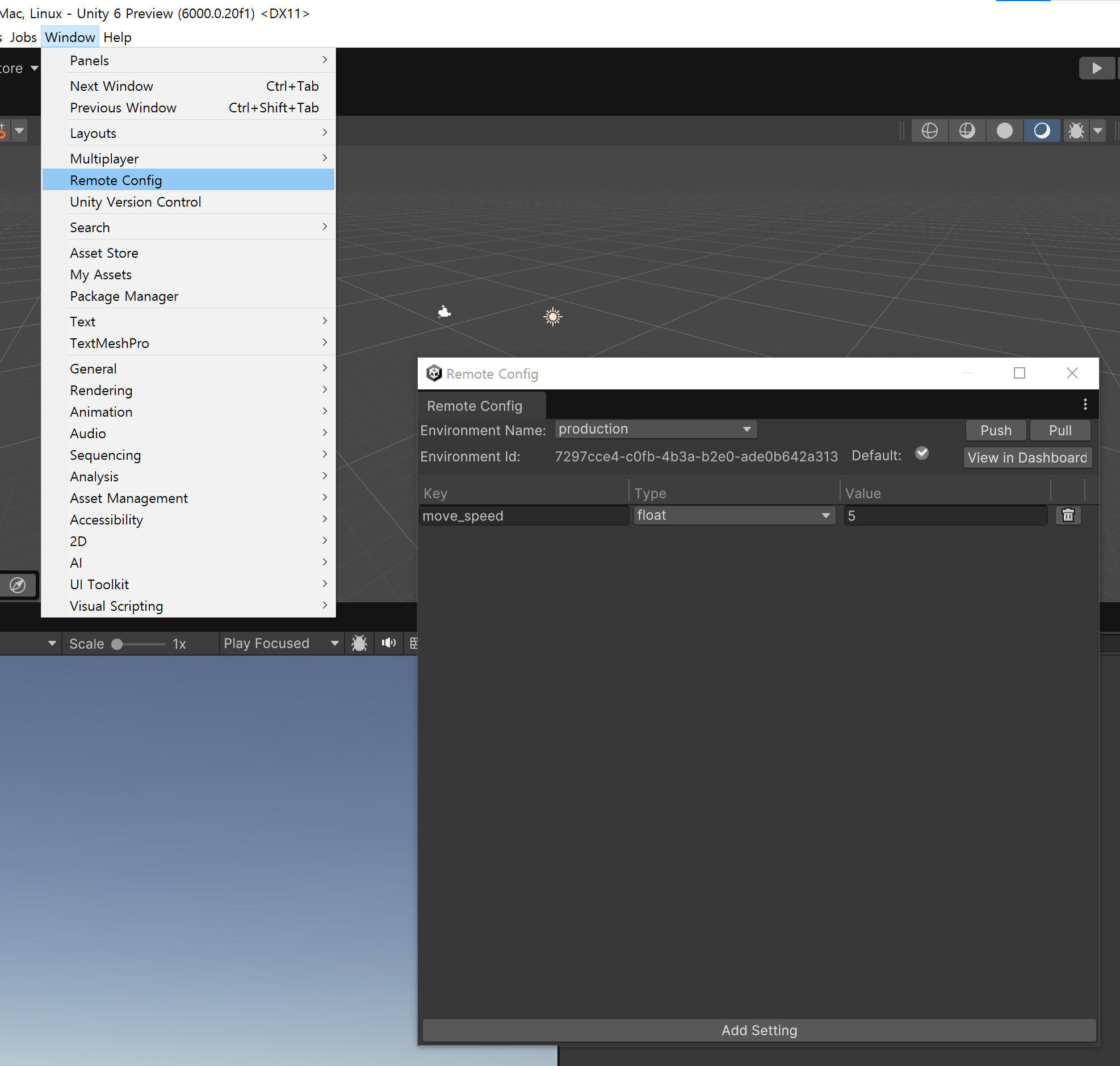


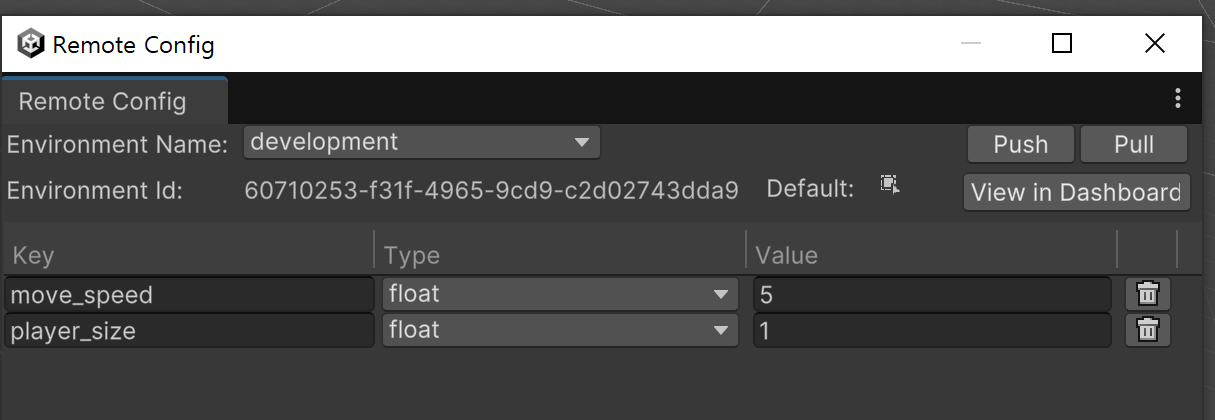
서버에서 사용하는 파라미터는 가능하면 소문자, \_ 써서 구분



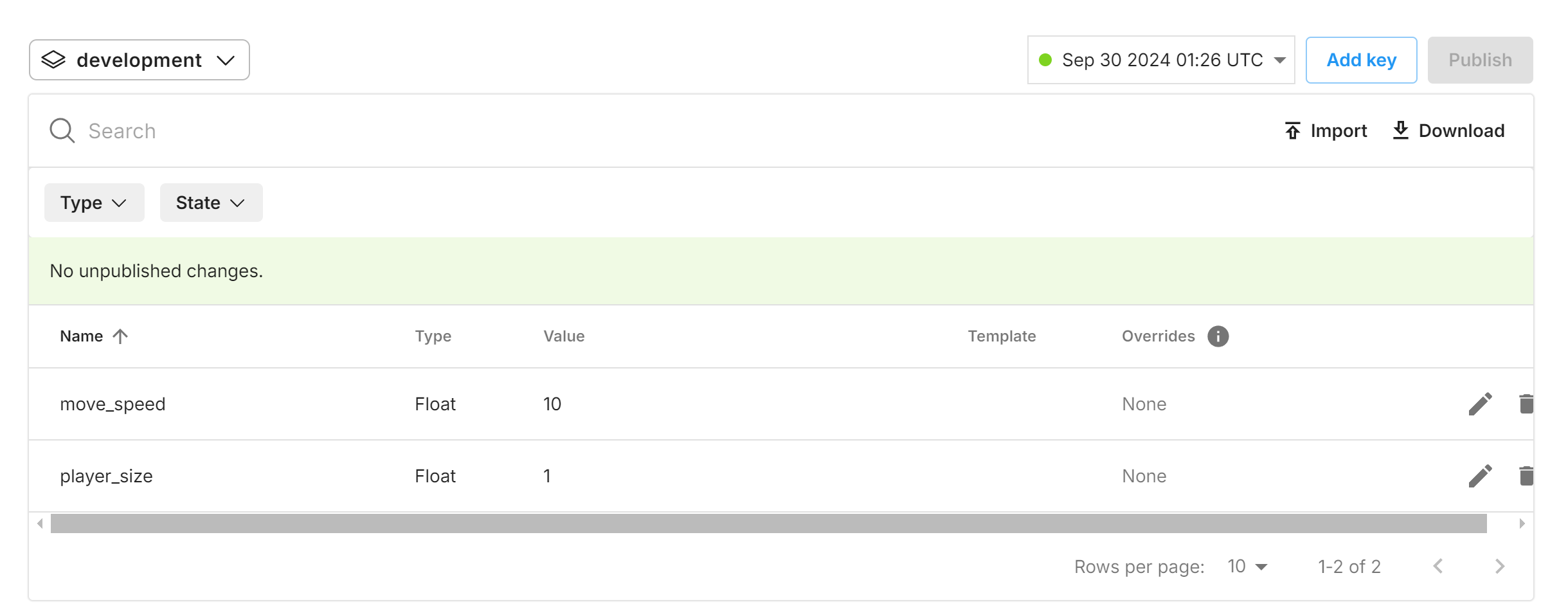
Add key 해서 player\_size 추가

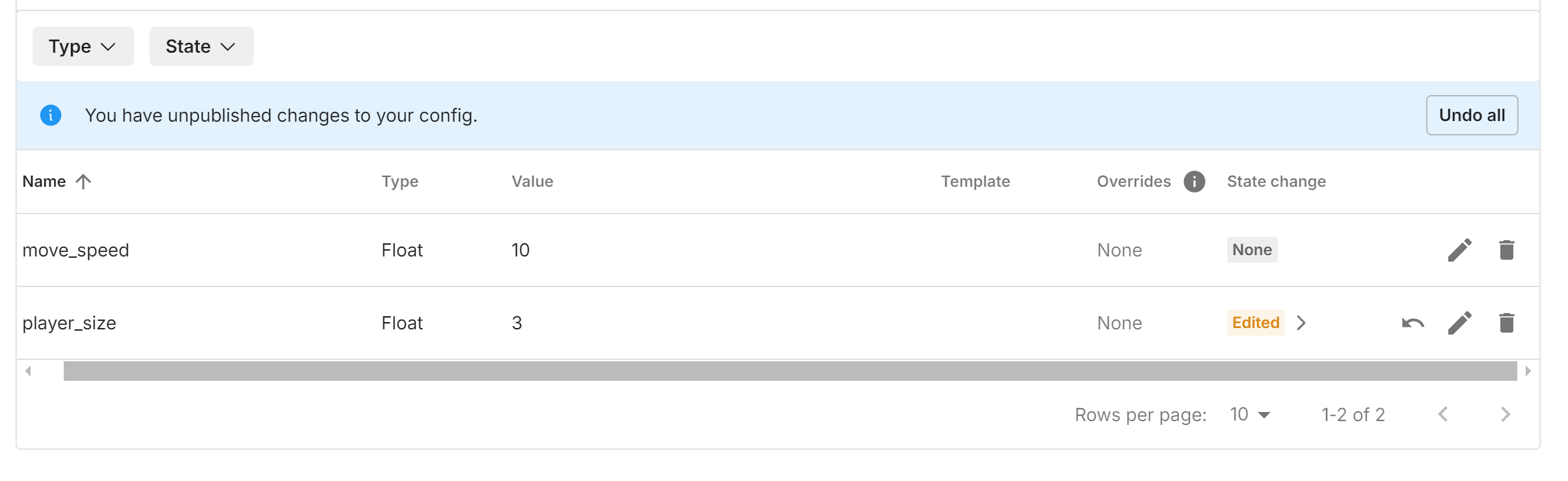




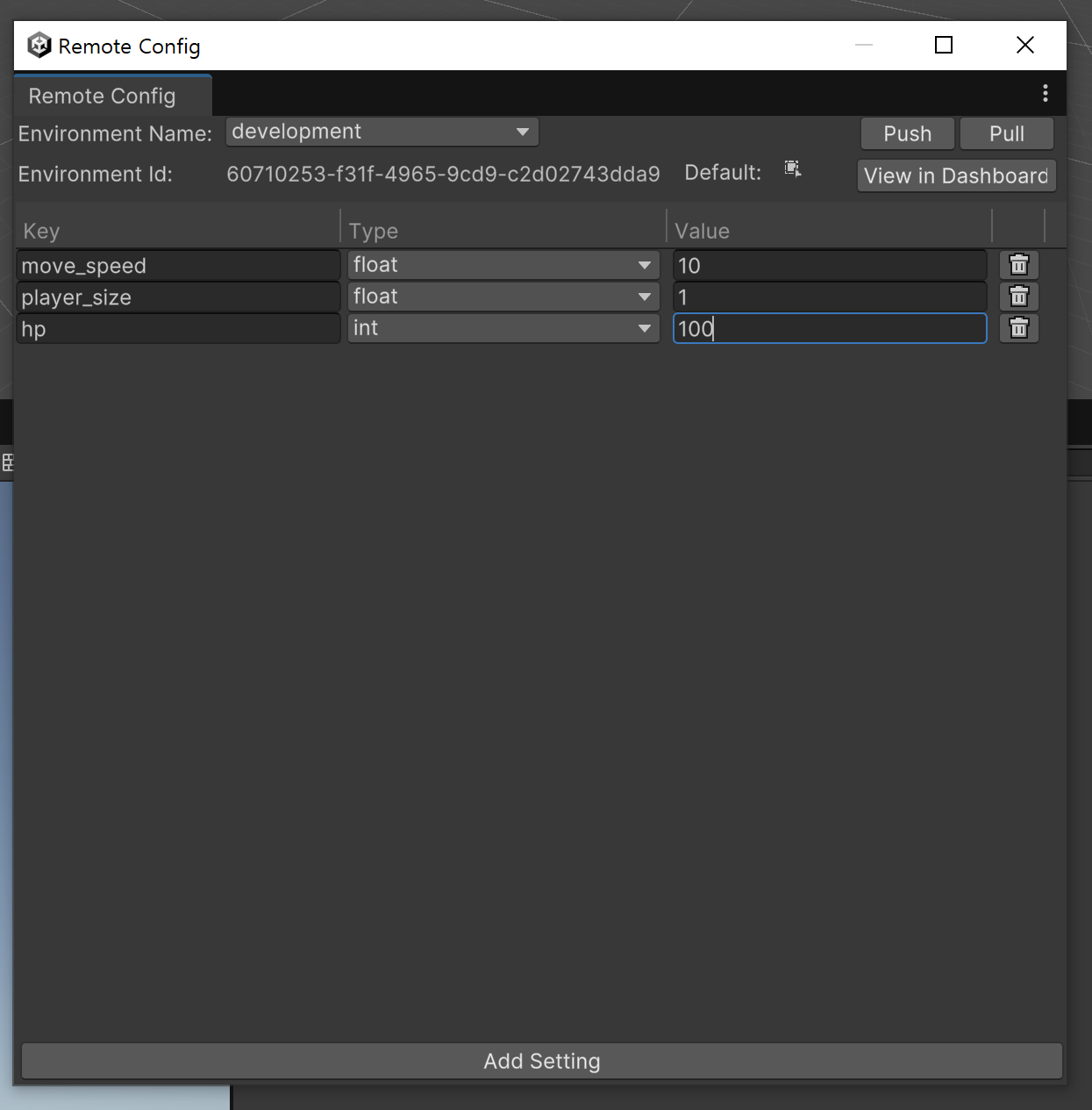


속성값을 실시간으로 변경, push 하면 저장할 수 있다.

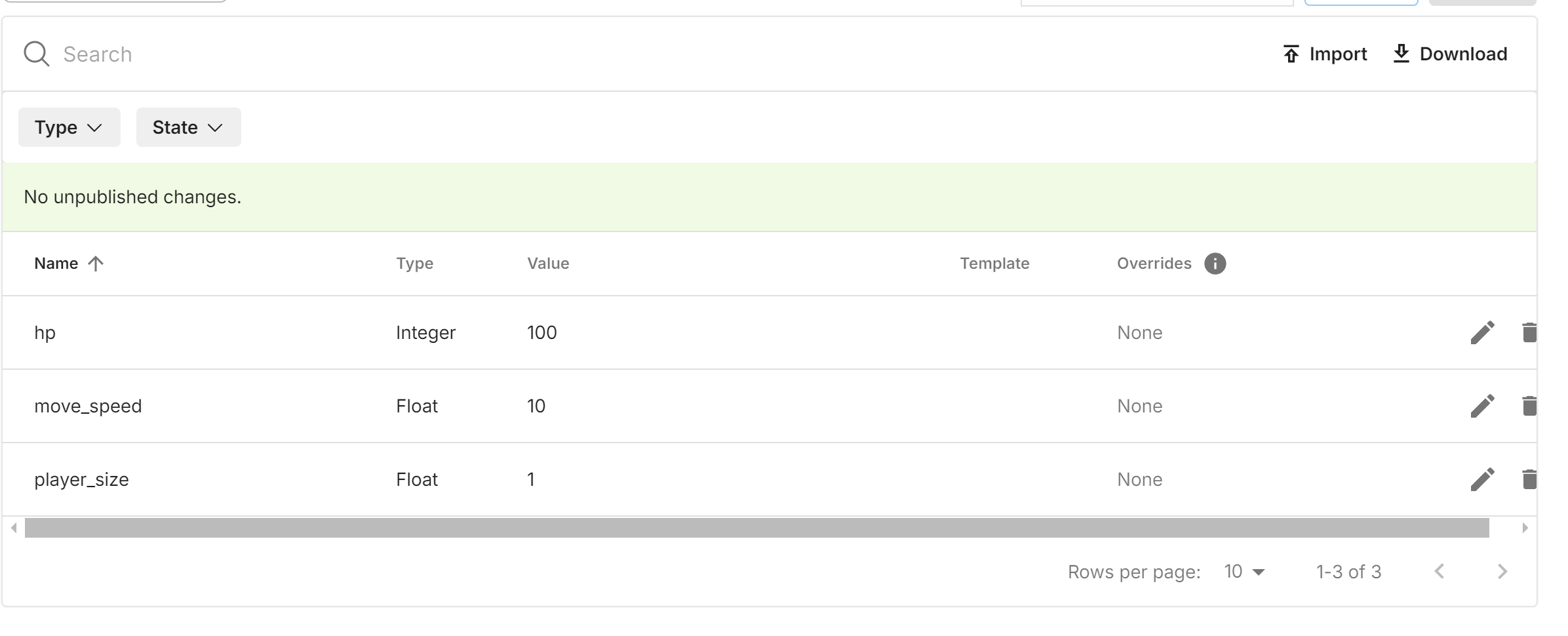


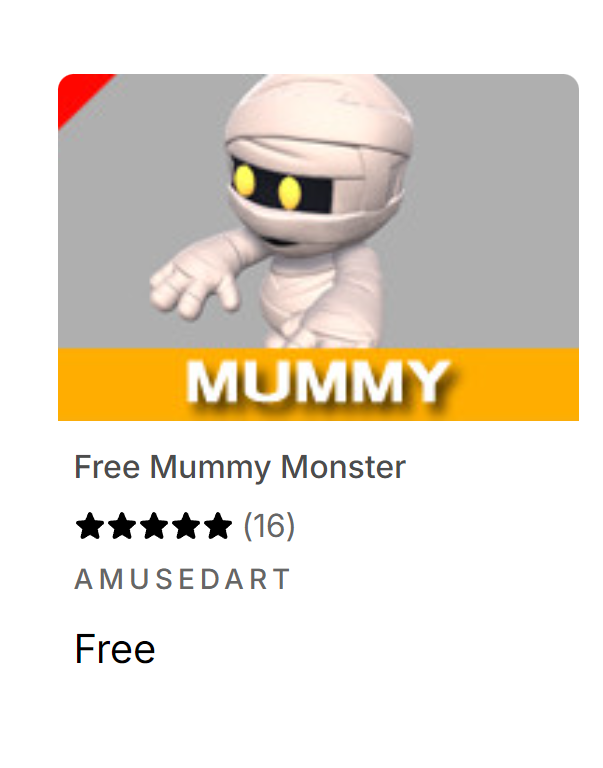


동일하게 dashboard 상에서도 변경 가능, 롤백 하고싶으면 화살표 누르면 됨



아래에 add setting 해서 새로운 값 입력도 가능

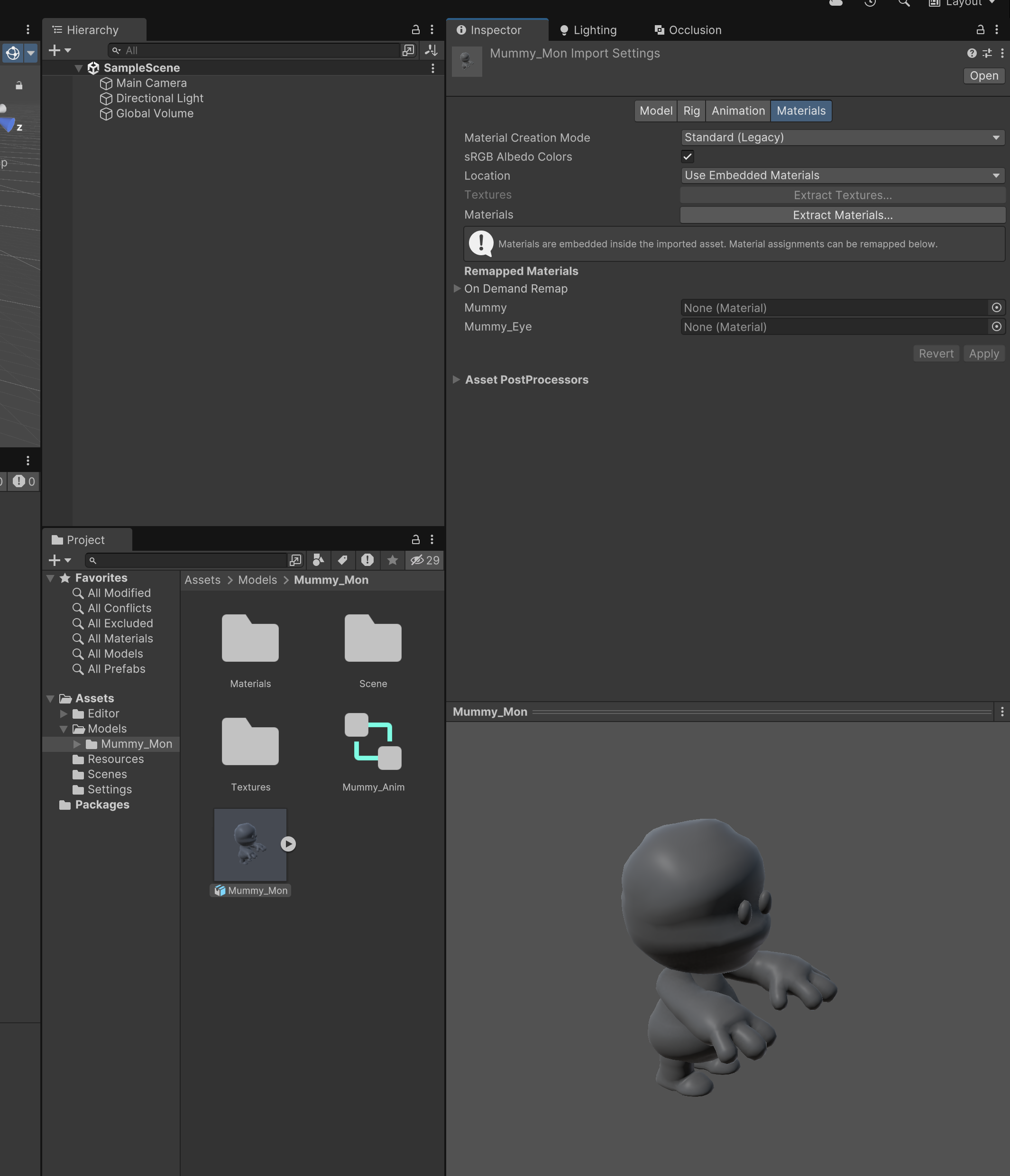




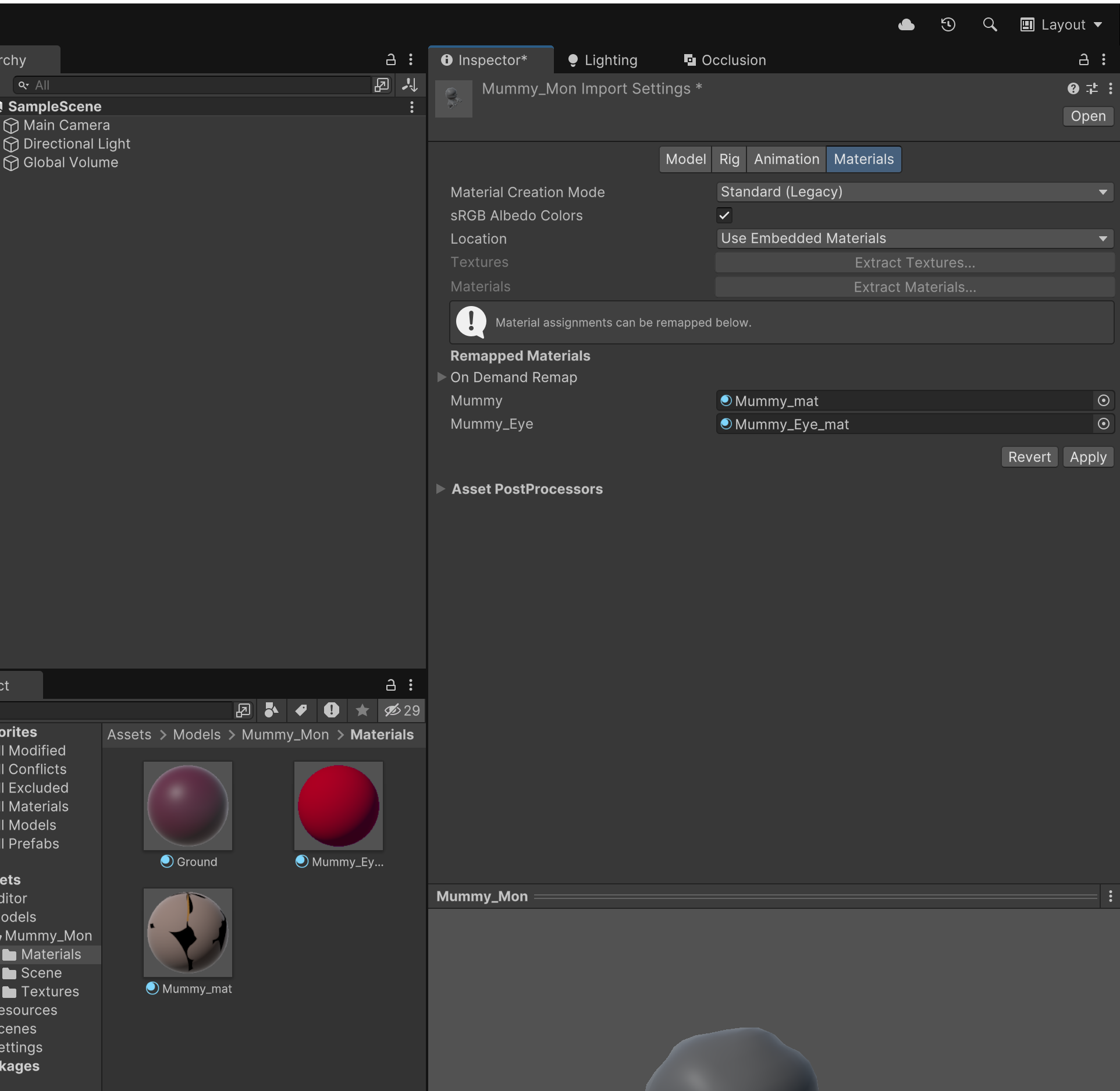
에셋스토어에서 mummy 검색, 추가 render pipeline convert 해주기



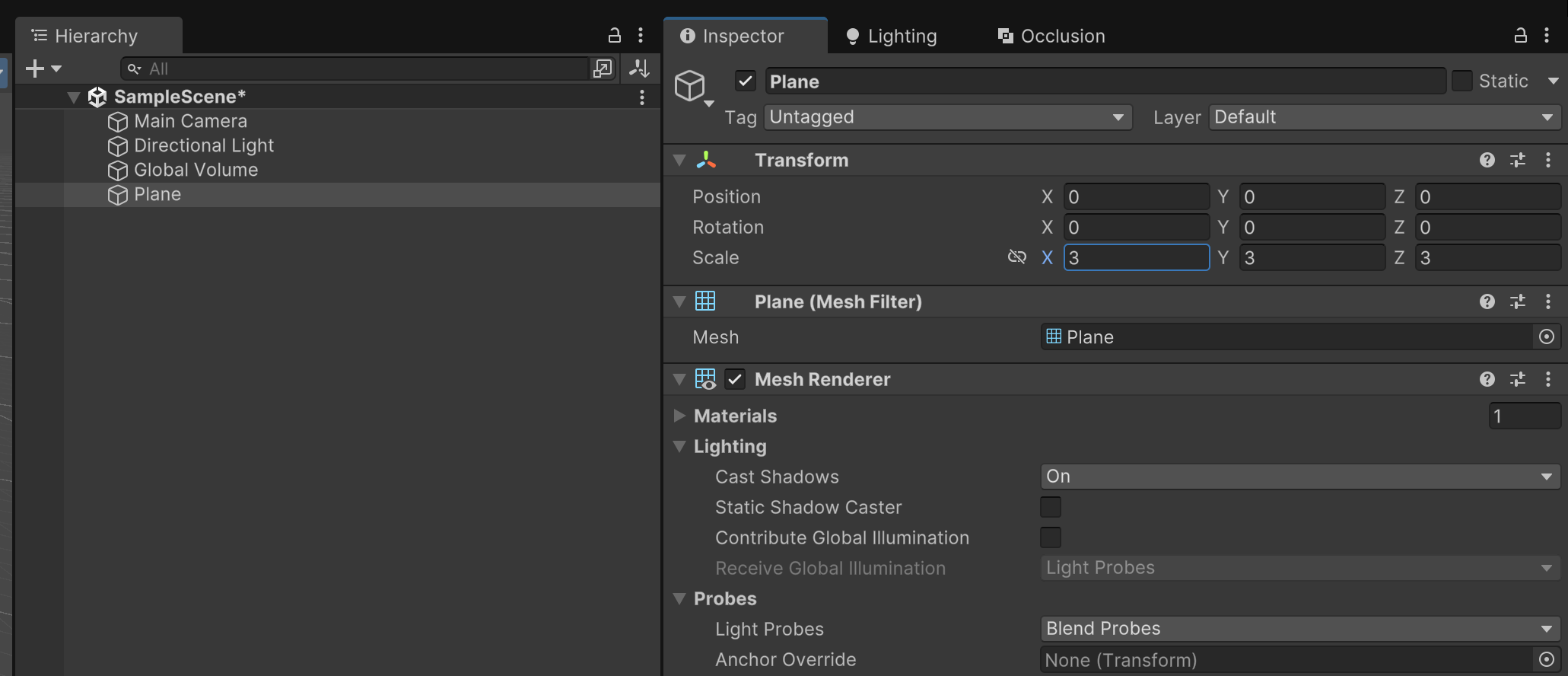
폴더명 Models 로 변경



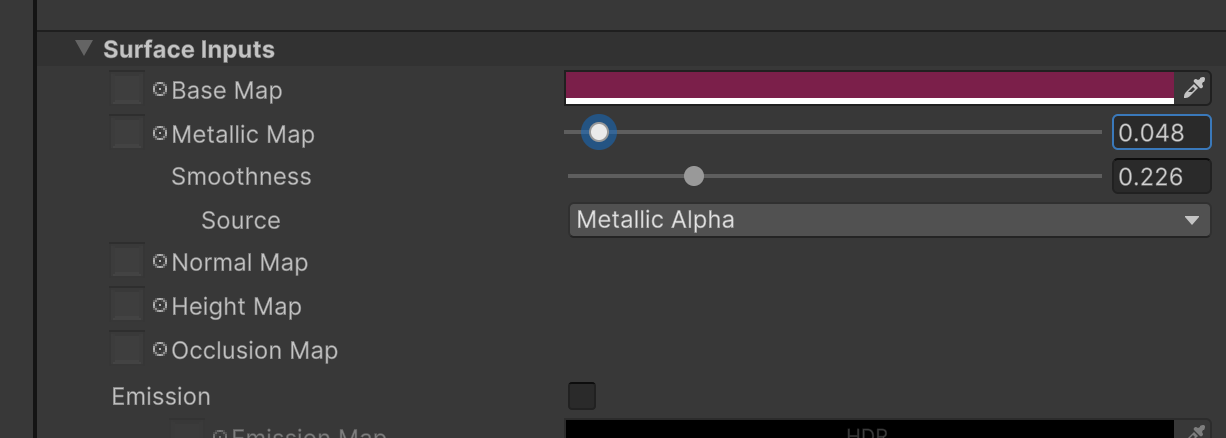
Mummy\_Mon 찾아서 materials



Materials 찾아넣어서 어플라이



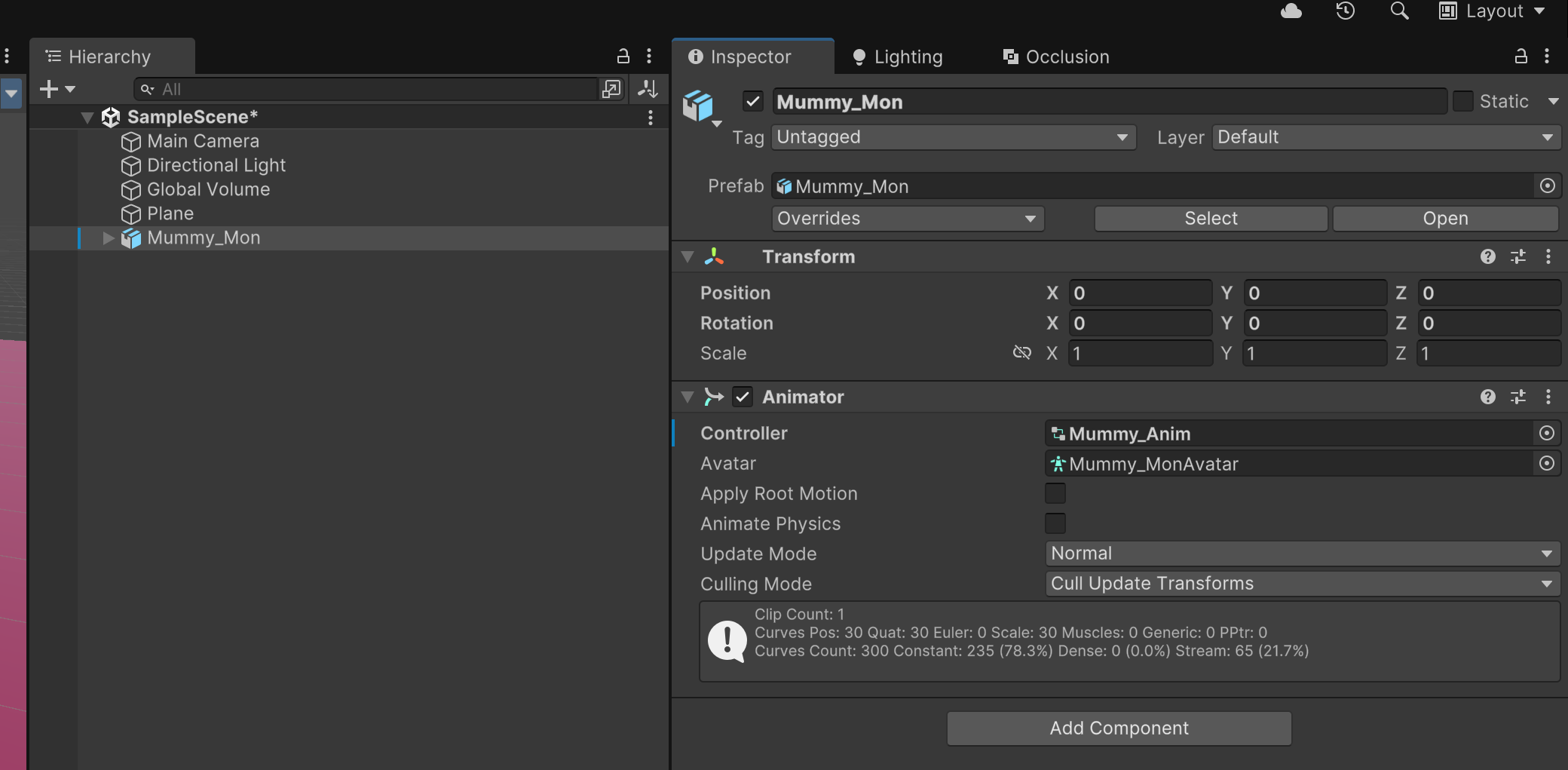
Plane 추가 후 scale 3, Materials 폴더에서 Ground 찾아서 입혀주기



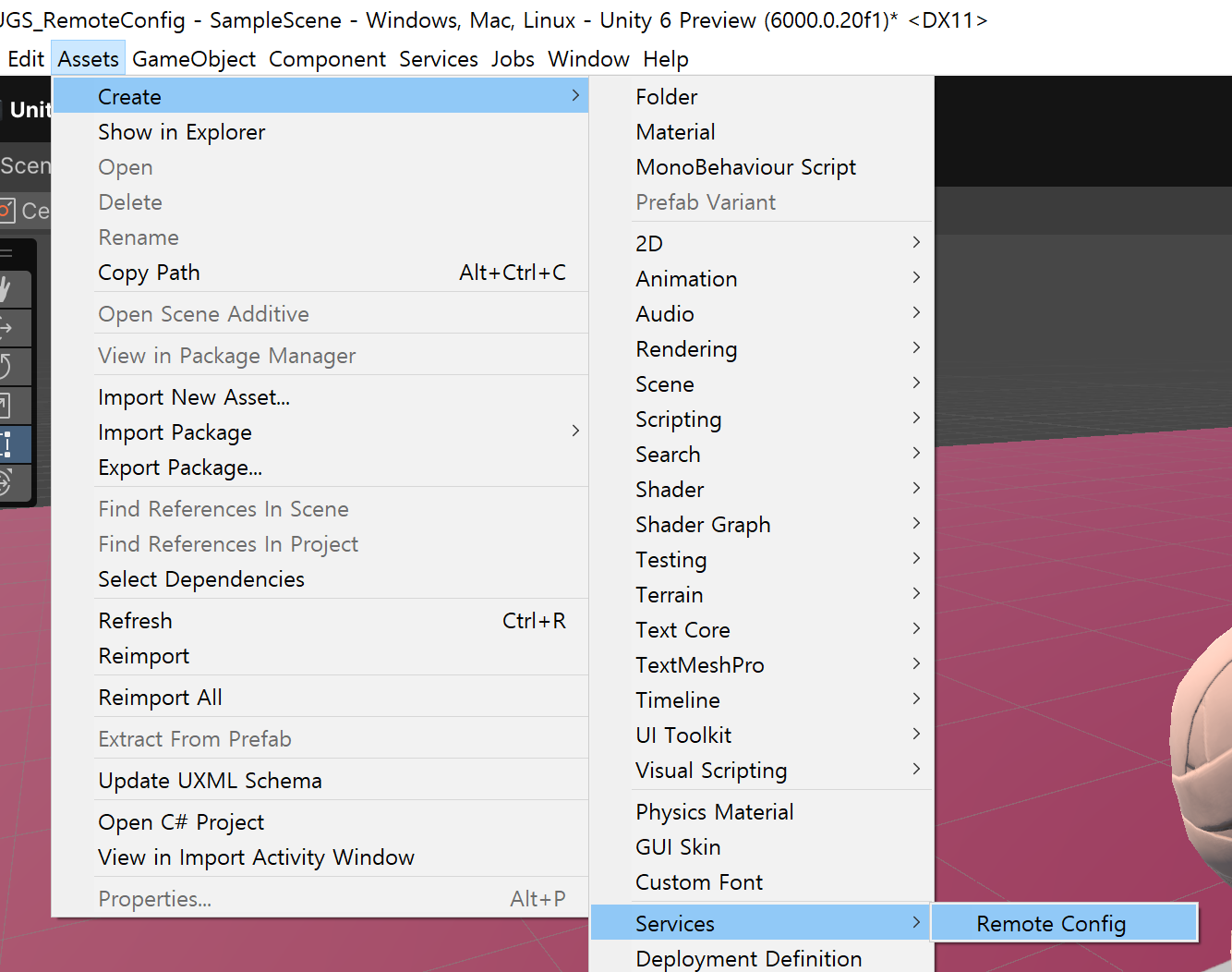
Metallic , smoothness 조절하면 바닥 재질 느낌 바꿀 수 있음



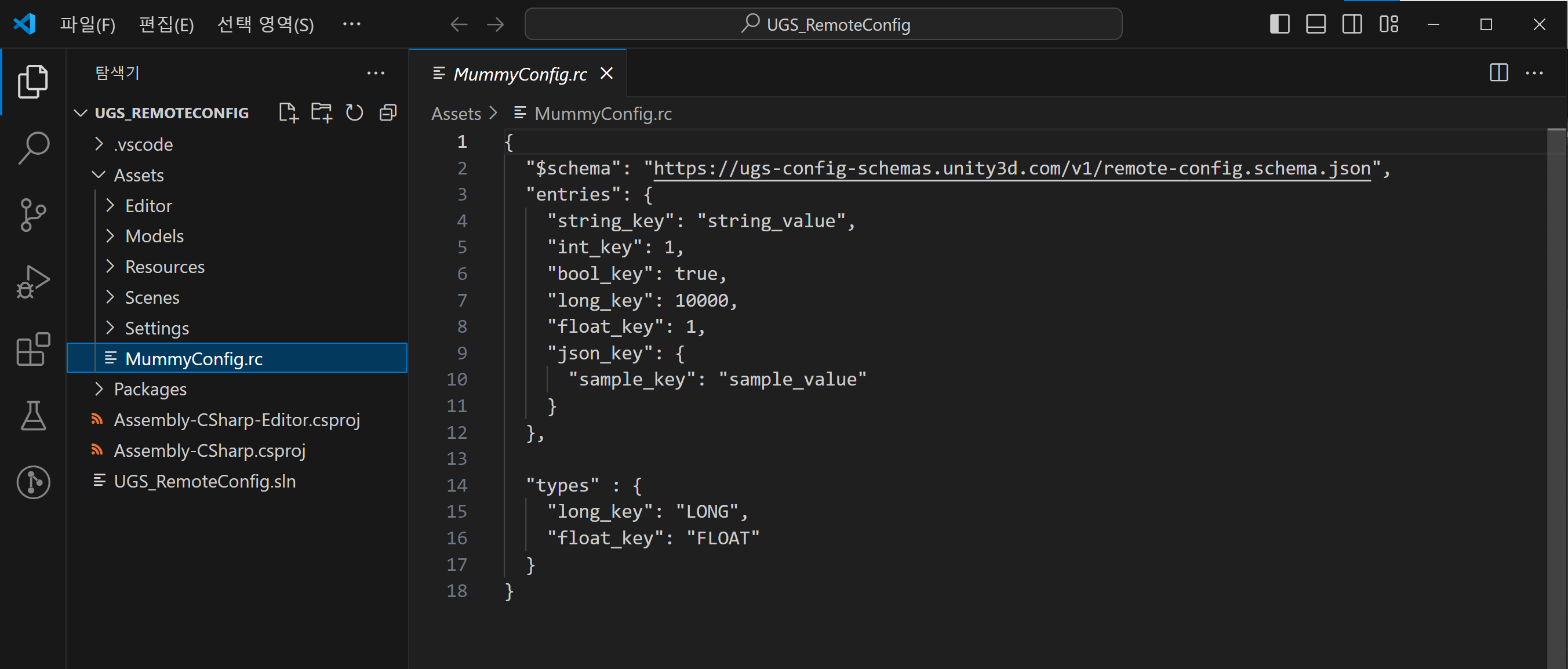
Hierarchy에 mummy 추가



Mummy anim 연결해주기

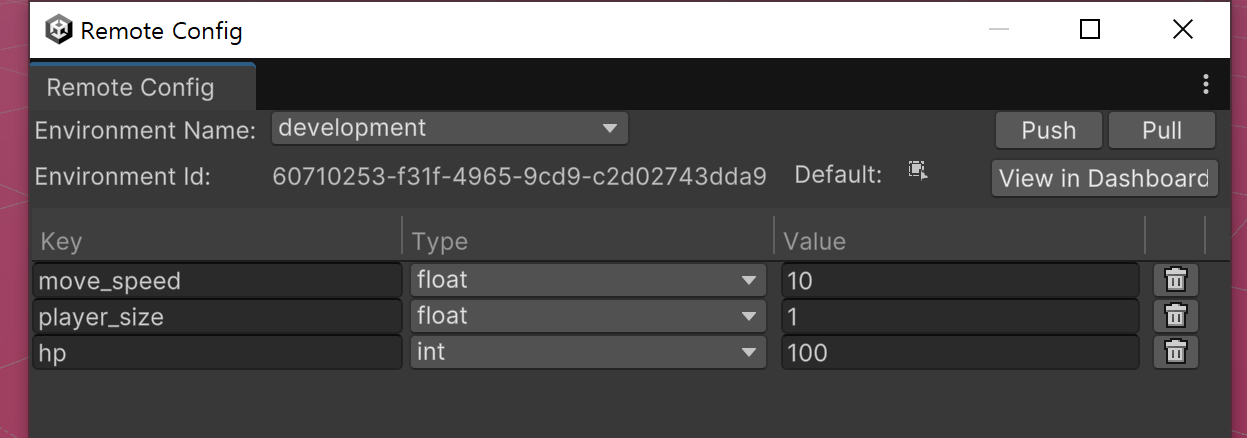


MummyConfig로 이름 설정 후에 생성

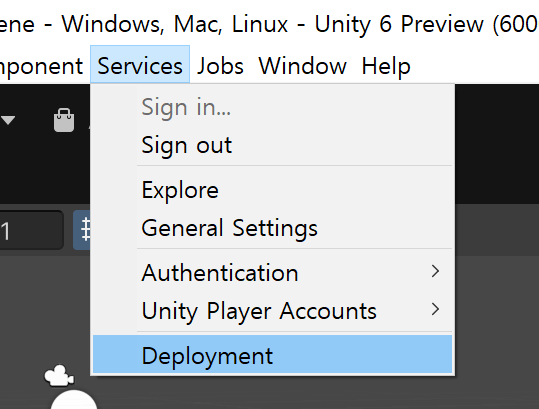


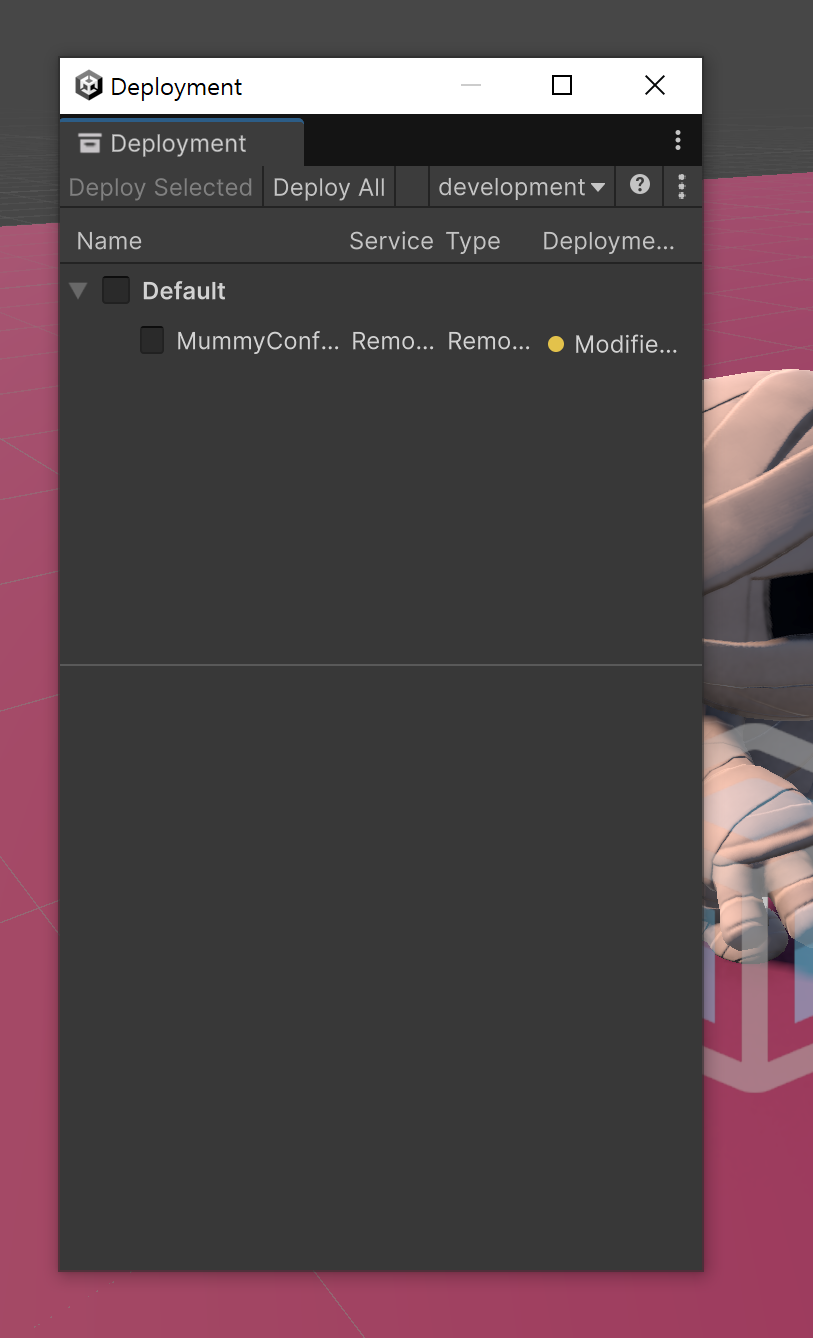


위 화면처럼 변경

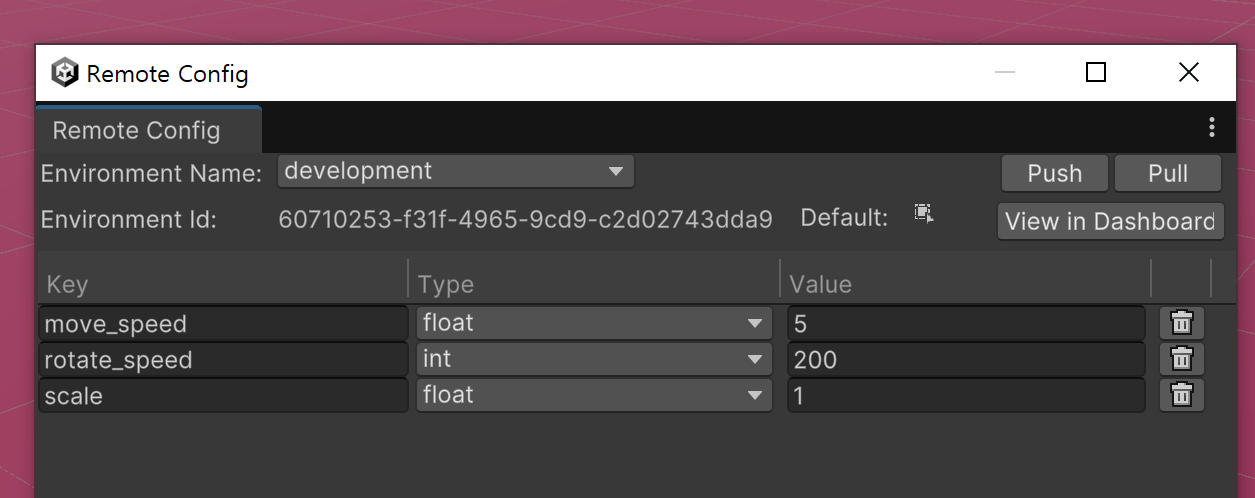


다 지우고 푸시

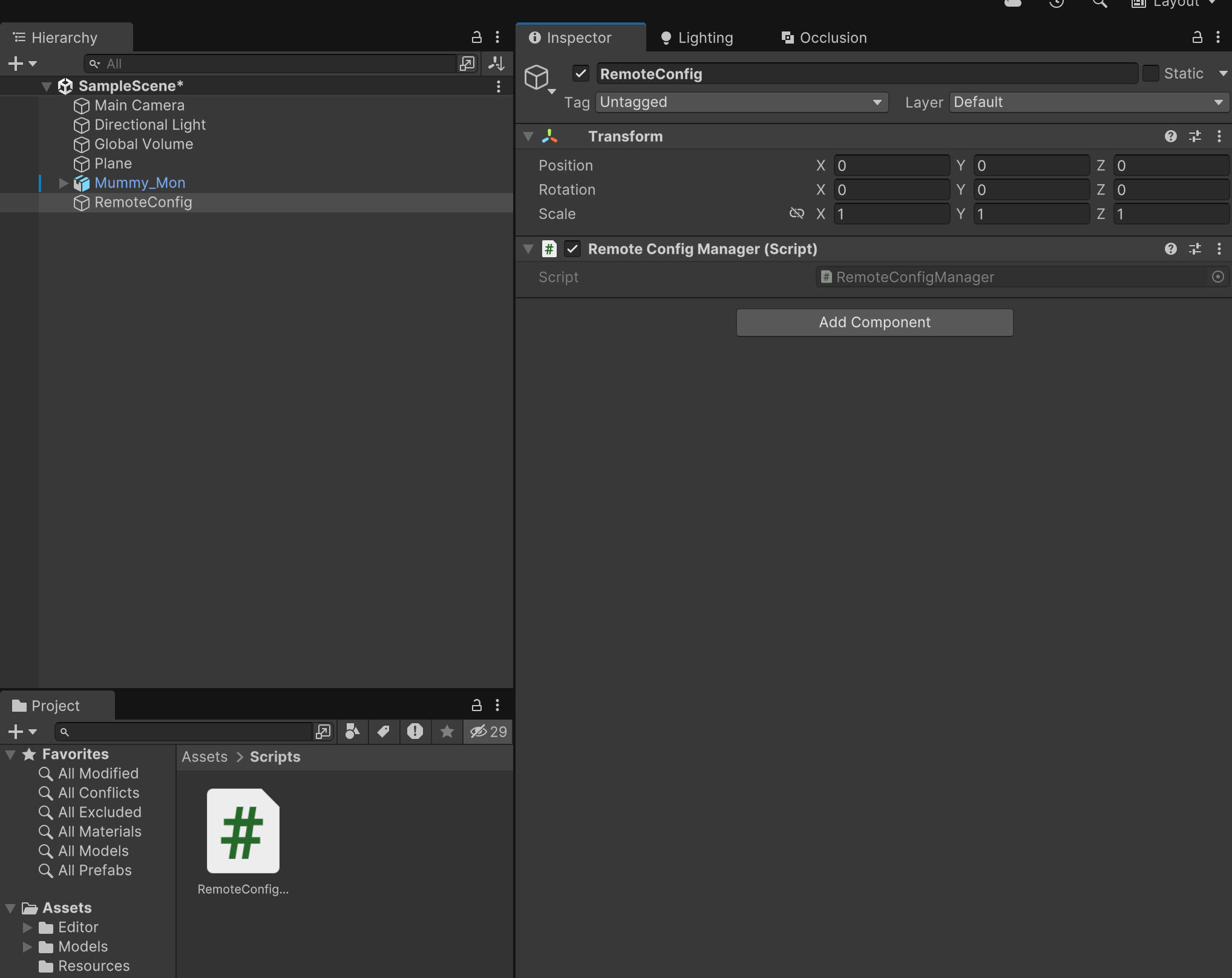




Mummy 파일 푸시하겠다~



잘 들어와있다



Scripts 생성 및 코드 작성

using Unity.Services.Authentication;

using Unity.Services.Core;

using UnityEngine;

public class RemoteConfigManager : MonoBehaviour

{

    private async void Awake()

    {

        await UnityServices.InitializeAsync();

        AuthenticationService.Instance.SignedIn += () =>

        {

            Debug.Log($"로그인 완료 : {AuthenticationService.Instance.PlayerId}");

        };

        await AuthenticationService.Instance.SignInAnonymouslyAsync();

    }

}

인증 까지 가능한지 확인

public class RemoteConfigManager : MonoBehaviour

{

    [SerializeField] private float moveSpeed;

    [SerializeField] private float rotateSpeed;

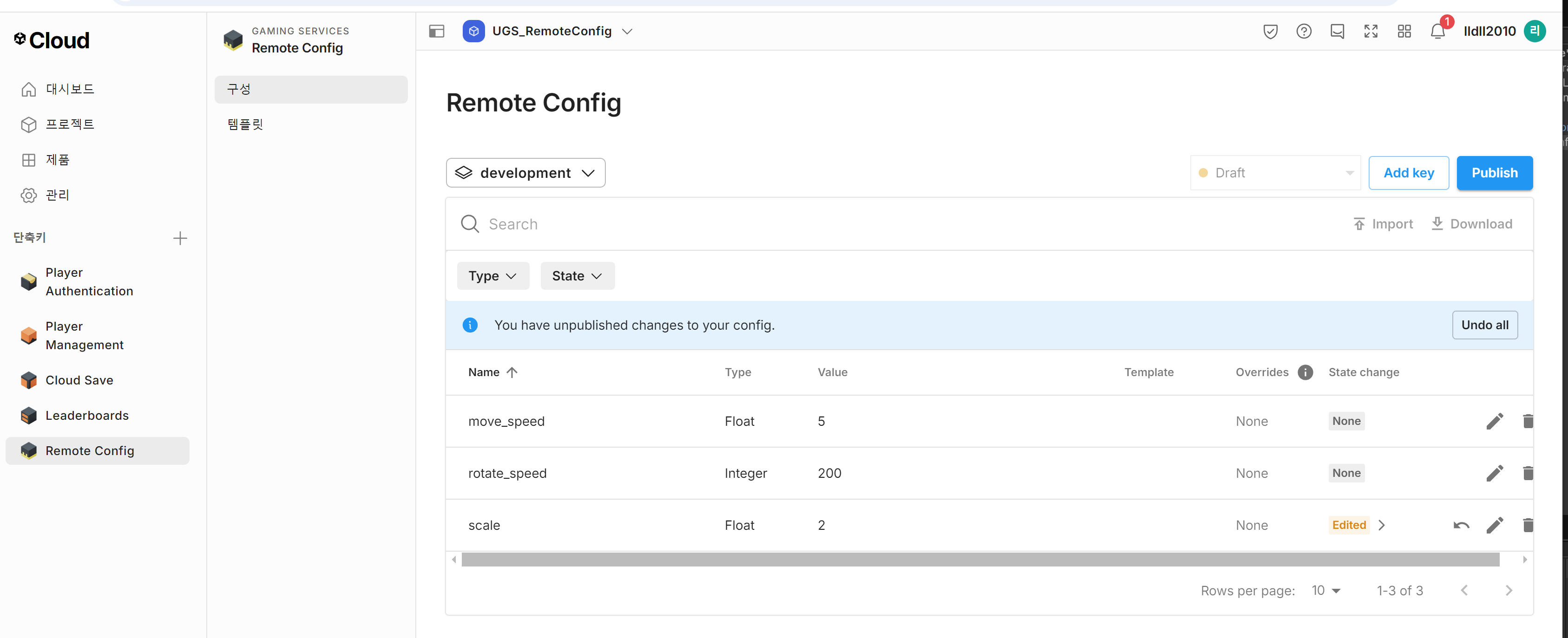
    [SerializeField] private int scale;

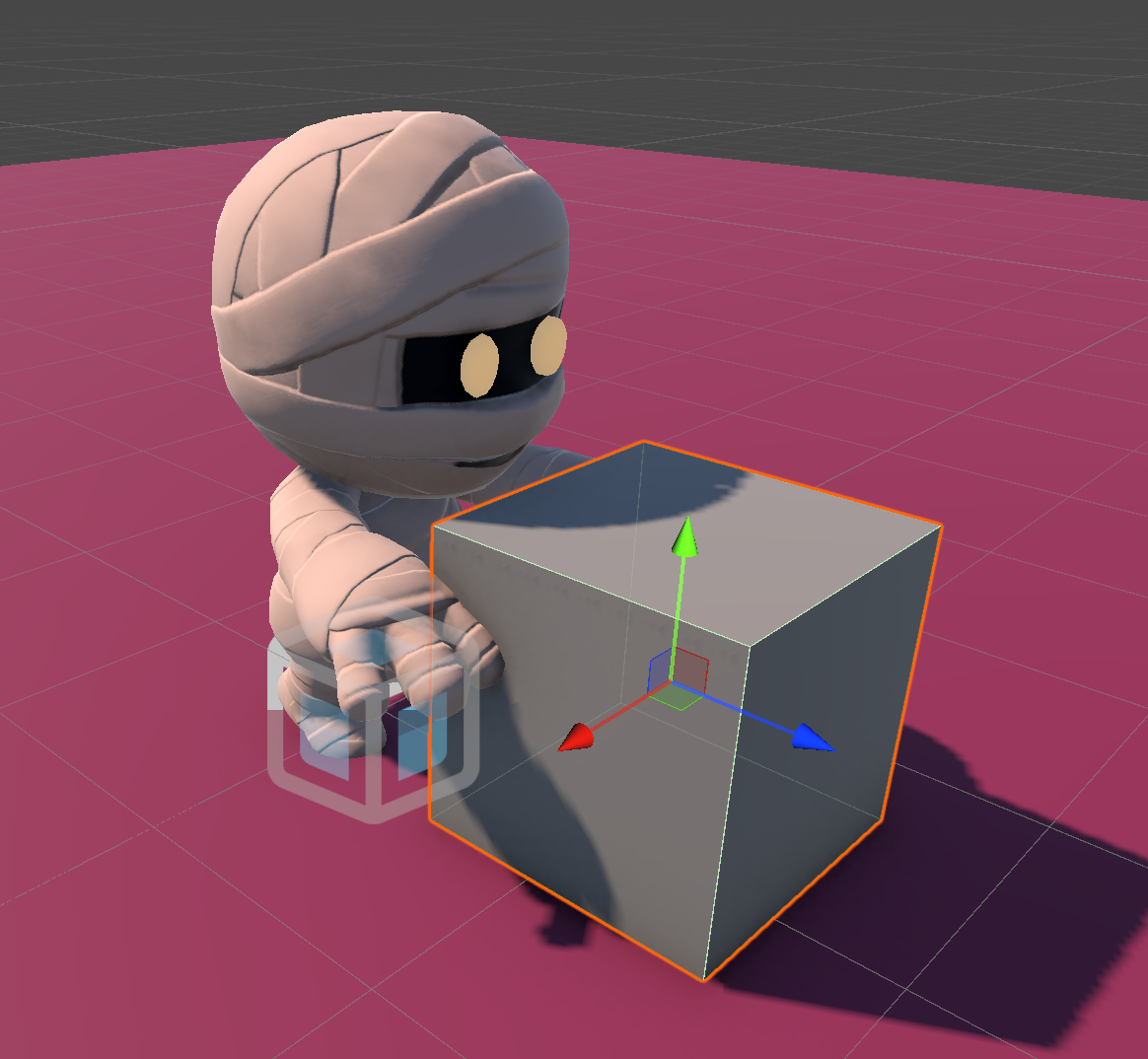
맨 위에 필드추가

코딩 후 RemoteConfig 에 Mummy\_Mon 연결해주기



Mummy scale 코드 추가 후에 dashboard 에서 scale 2배로 변경해주고 서밋, 퍼브리쉬





실행하면 스케일이 2배로 커진다

//