

chapter03

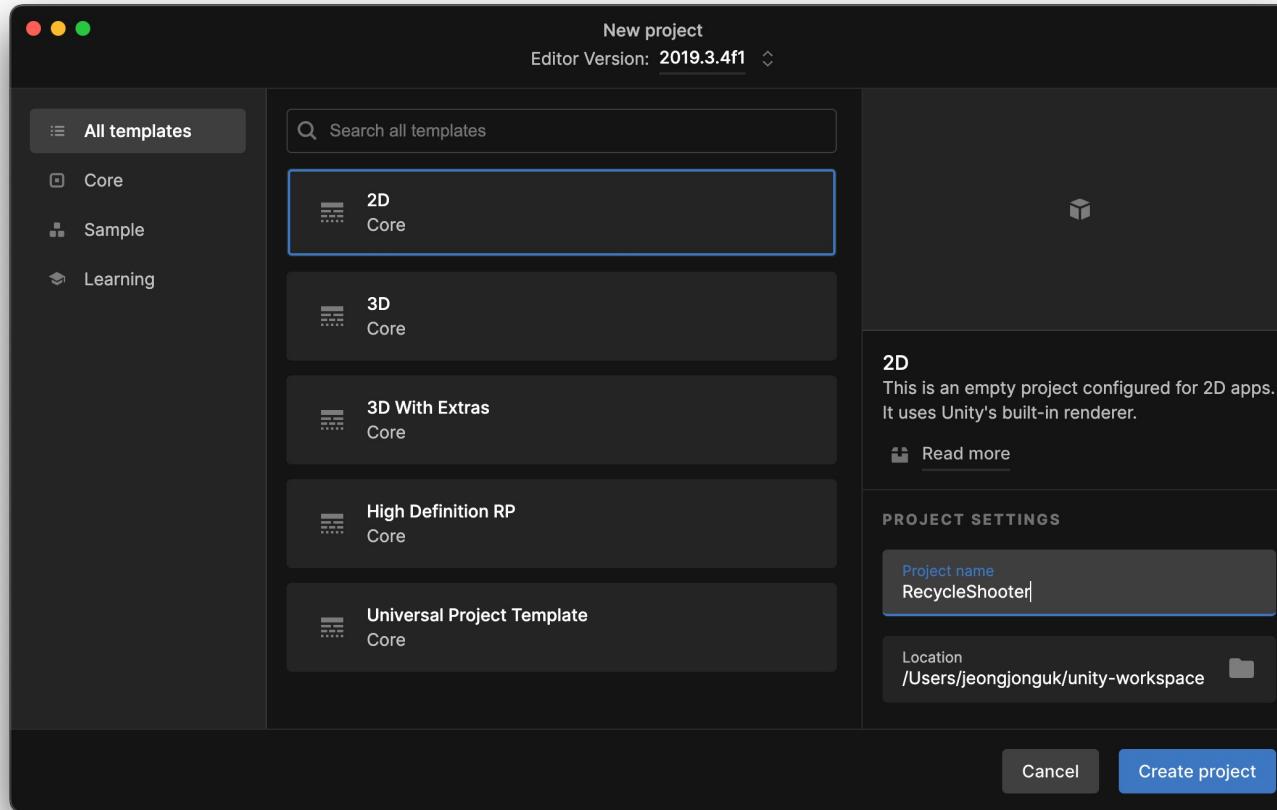
게임 만들기

1. 프로젝트 생성
2. 게임 이미지 구하기
3. 배경 만들기
4. 움직이는 배경 만들기

<https://github.com/jongukjeong/unity>

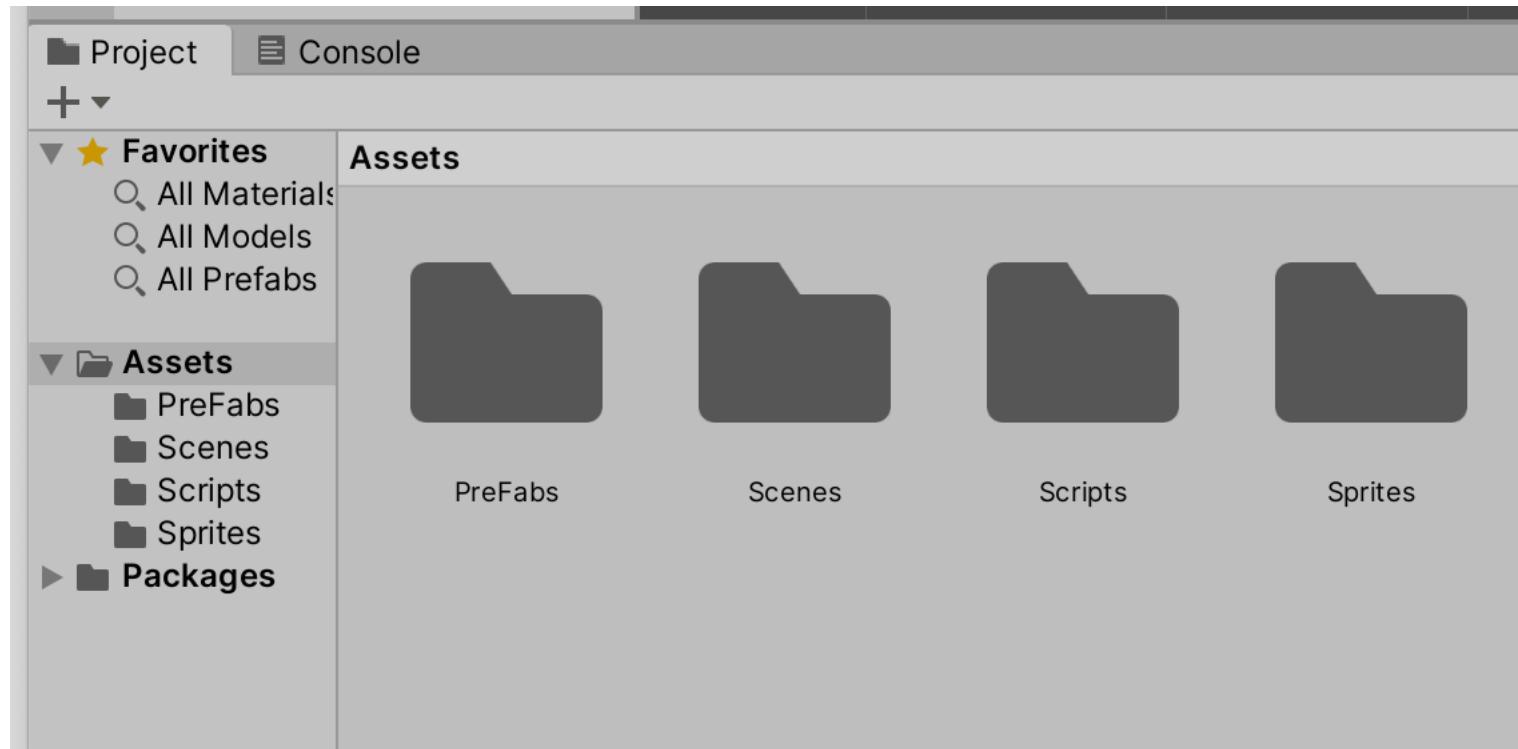
01 프로젝트 생성

■ RecycleShooter 프로젝트 생성



01 프로젝트 생성

■ Asset 하위 폴더 구성하기



02 게임 이미지 구하기

■ Game Resource 용 이미지 검색

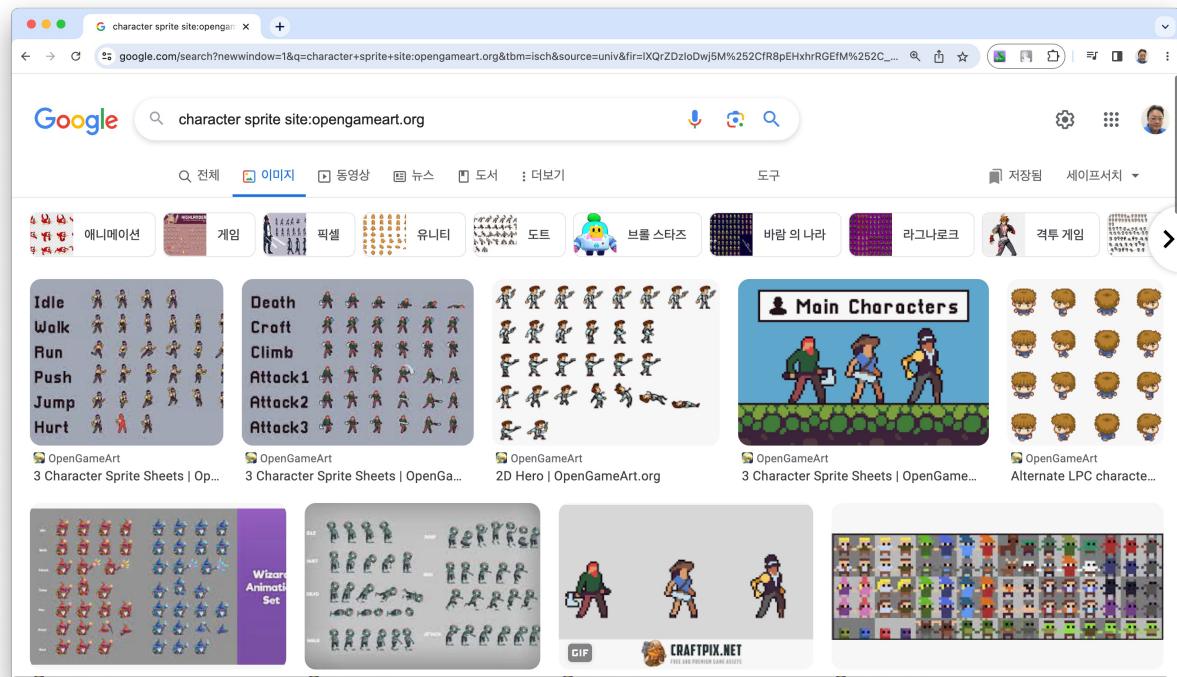
character sprite site:opengameart.org

recycle site:opengameart.org

mana site:opengameart.org

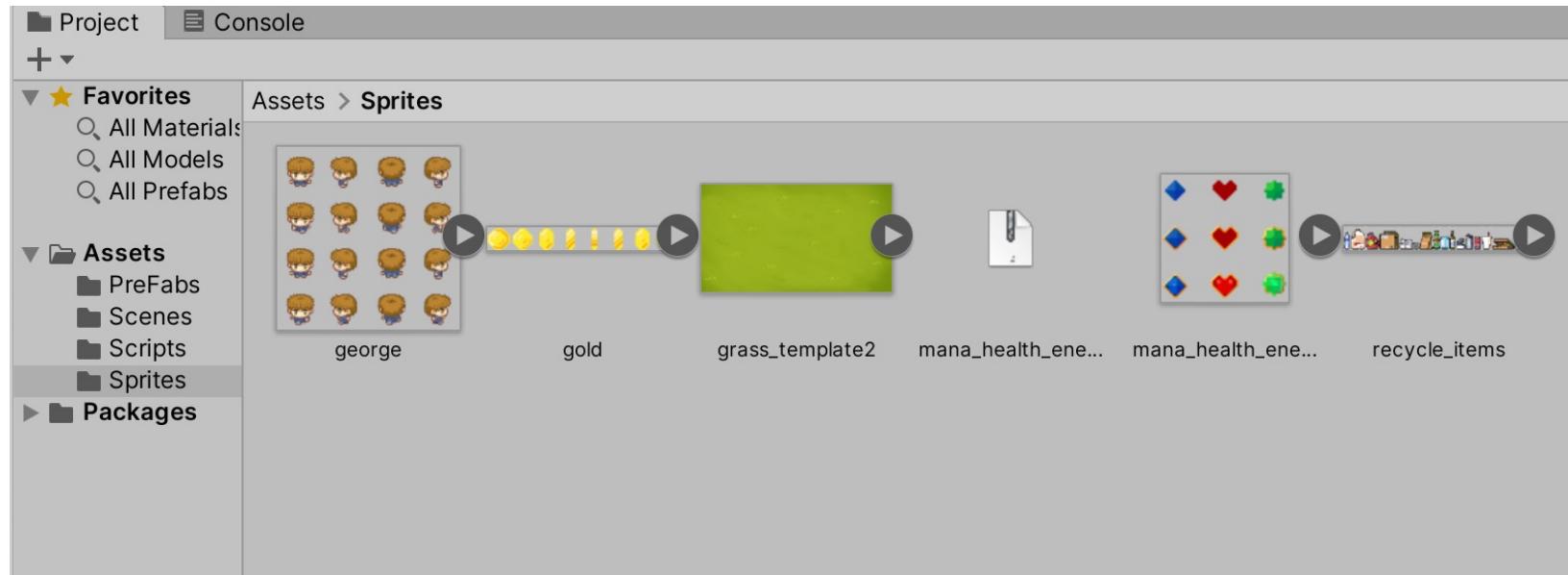
grass background site:opengameart.org

coin site:opengameart.org



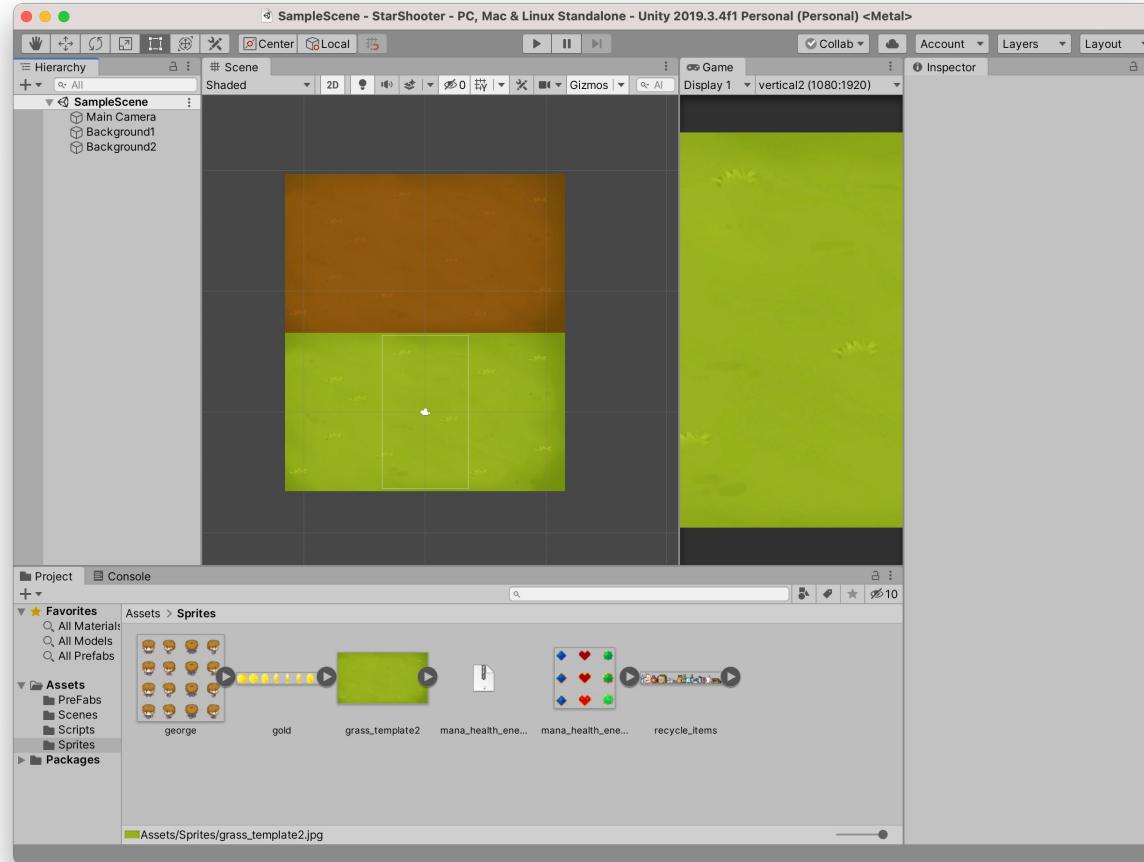
02 게임 이미지 구하기

■ Unity Editor – Project 뷰 – Sprites 폴더에 이미지 추가



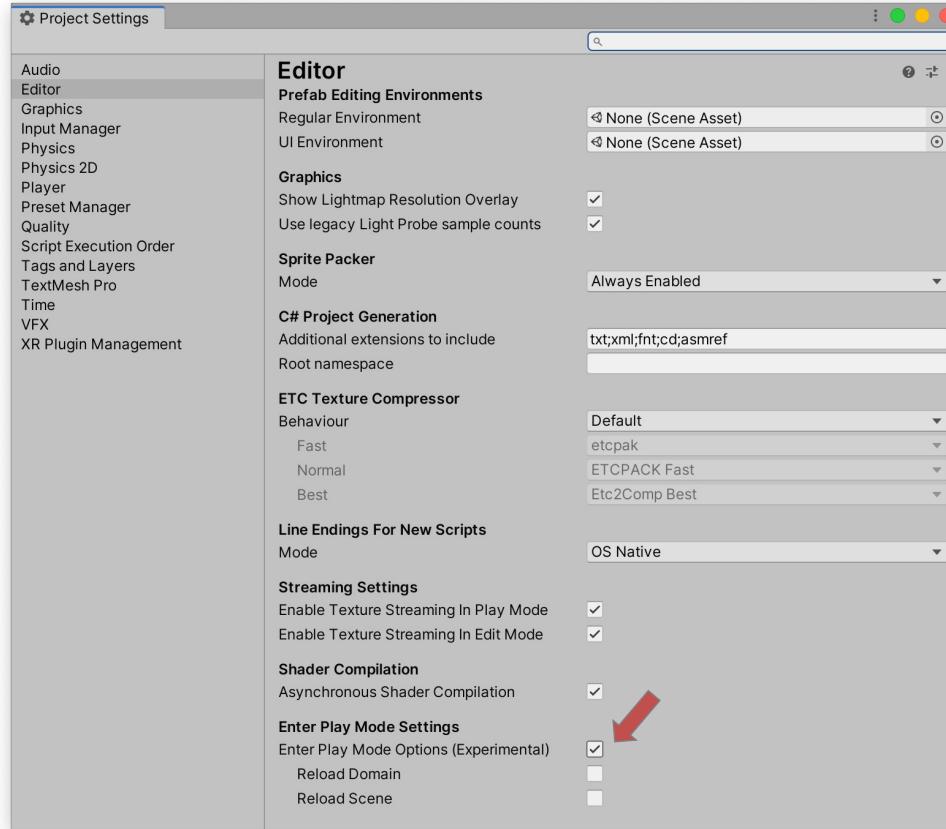
03 배경 만들기

■ Asset 의 이미지를 Object 화



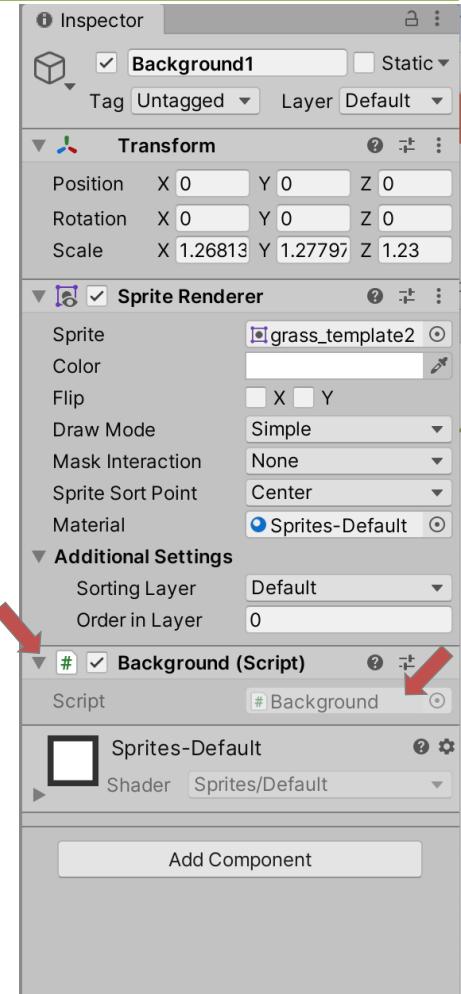
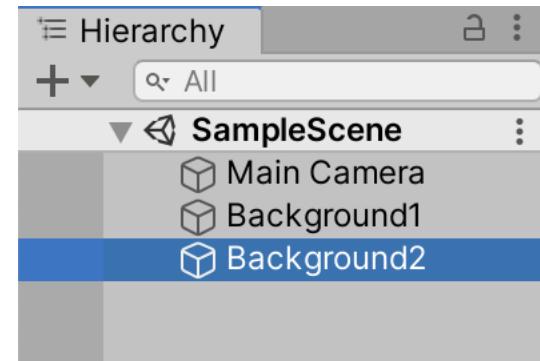
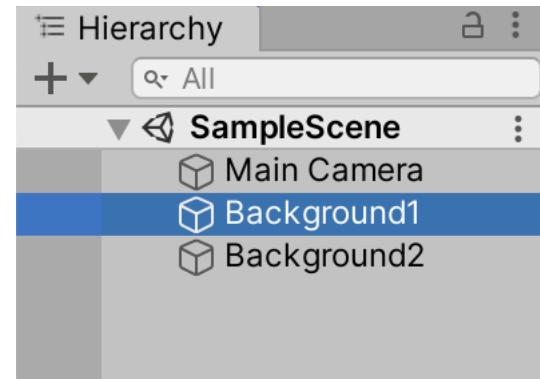
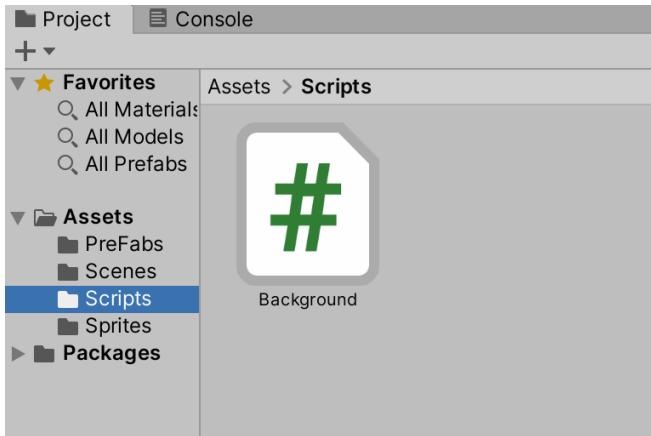
03 배경 만들기

■ 게임 플레이가 느릴때 – Enter Play Mode Options 체크



04 움직이는 배경 만들기

■ Object에 적용할 Script 준비



스크립트를 적용할 오브젝트를 선택후 해당 스크립트를 인스펙터뷰 쪽으로 드래그&드랍

04 움직이는 배경 만들기

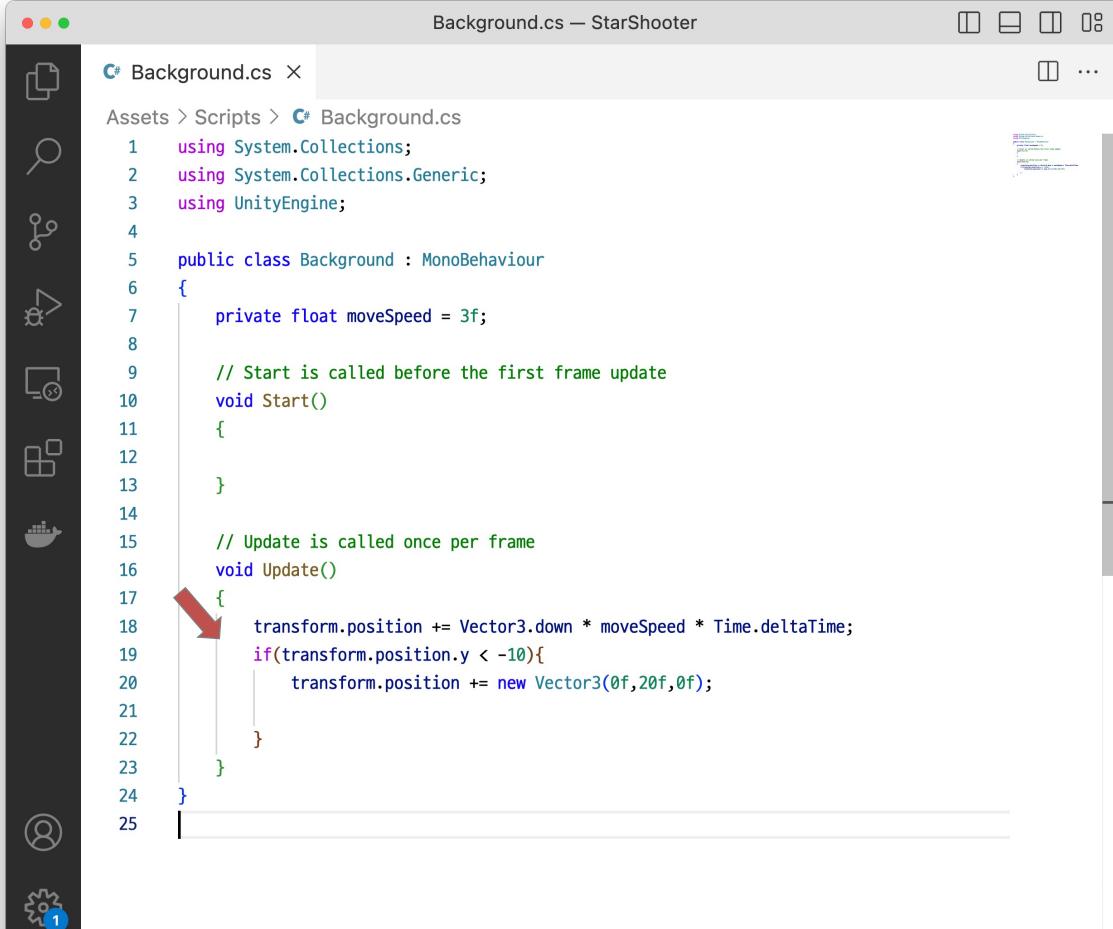
■ Script 작성

- transform.position 변화 == 움직임

transform.position 을 변화 시켜 움직이게 하거나

transform.position.y 값을 체크해서 현재 위치를 확인할 수 있다

스크립트 저장하고 Unity로 돌아가서 Play 해보면서 적당한 값 적용



```

Background.cs — StarShooter

C# Background.cs ×

Assets > Scripts > C# Background.cs

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;

4
5  public class Background : MonoBehaviour
6  {
7      private float moveSpeed = 3f;
8
9      // Start is called before the first frame update
10     void Start()
11     {
12     }
13
14
15     // Update is called once per frame
16     void Update()
17     {
18         transform.position += Vector3.down * moveSpeed * Time.deltaTime;
19         if(transform.position.y < -10){
20             transform.position += new Vector3(0f, 20f, 0f);
21         }
22     }
23
24 }
25

```

04 움직이는 배경 만들기

■ 배경색 원래상태로 되돌리기

