

/ \*

## 기본 세팅

Player는 원형 오브젝트, 장애물은 사각형 오브젝트를 사용했습니다.

또한 empty 오브젝트와 Collider를 이용해 플레이 화면의 사방을 막았습니다.

Player에 Rigidbody를 적용하였으나, 이는 충돌 처리를 위해서이기 때문에 중력은 0으로 설정했습니다.

\*/

```

Assets > Scripts > Player.cs > ...
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
0 references
5  public class Player : MonoBehaviour
6  {
7      [SerializeField]
8      private float moveSpeed = 10f;
9      // Start is called before the first frame update
0 references
10     void Start()
11     {
12
13     }
14
15     // Update is called once per frame
0 references
16     void Update()
17     {
18         if(Input.GetKey(KeyCode.LeftArrow)) {
19             transform.position += Vector3.left * moveSpeed * Time.deltaTime;
20         } else if(Input.GetKey(KeyCode.RightArrow)) {
21             transform.position += Vector3.right * moveSpeed * Time.deltaTime;
22         } else if(Input.GetKey(KeyCode.UpArrow)) {
23             transform.position += Vector3.up * moveSpeed * Time.deltaTime;
24         } else if(Input.GetKey(KeyCode.DownArrow)) {
25             transform.position += Vector3.down * moveSpeed * Time.deltaTime;
26         }
27     }
0 references
28     private void OnCollisionEnter2D(Collision2D collision) {
29         Destroy(gameObject);
30     }
31 }

```

/\*

### Player(원형 오브젝트)에 적용한 스크립트

input action을 사용해보려 했으나 아직 기능 사용이 미숙하여(...)

이전의 인풋 시스템으로 구현하였습니다.

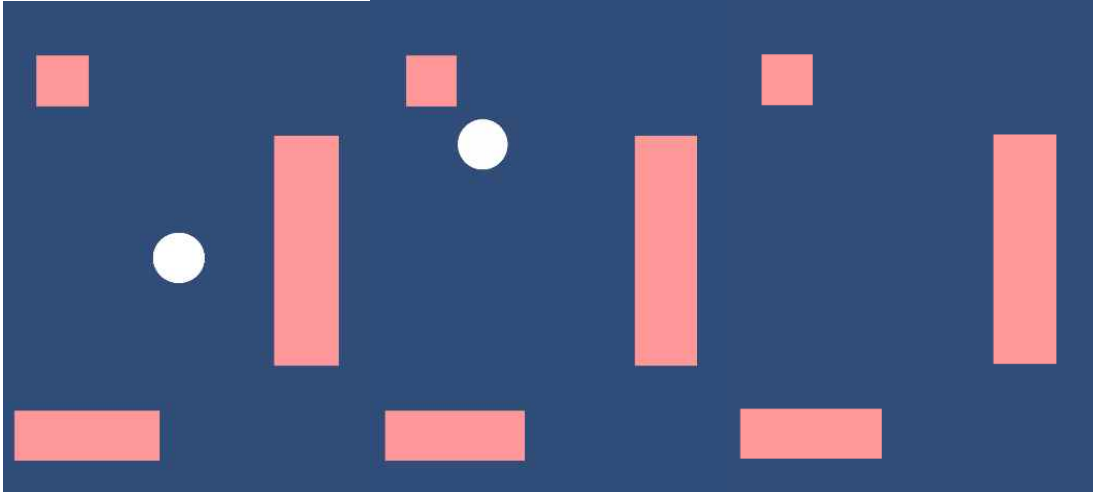
기본 속도는 10이나 유니티 내부에서 즉시 조절이 가능하도록 SerializeField를 사용했습니다.

화살표 Key의 입력을 받으면 해당 방향에 따라 움직입니다.

기기 성능에 따른 속도 차이를 방지하기 위해 Time.deltaTime을 이용했습니다.

장애물에 부딪히면 플레이어 오브젝트가 삭제됩니다.

\*/



/\*

### 결과

상하좌우 예상대로 움직이며, 장애물 혹은 벽에 부딪혔을 때 Player가 삭제되는 것을 확인할 수 있었습니다.

\*/