TECH-TALK

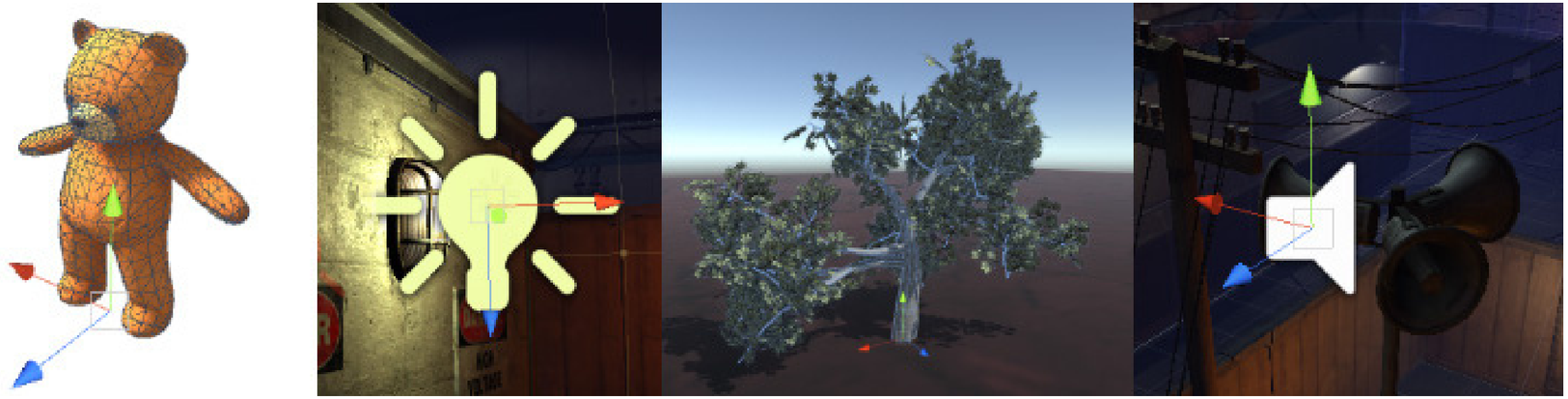
오브젝트 풀링이란?

컴퓨터공학과
박세웅

목차 LIST

01 오브젝트의 정의

Unity의 오브젝트는 게임(씬)에 존재하는 모든 것
→ 게임 오브젝트(Game Object)를 의미



01 오브젝트 생성 방법

생성: Instantiate()
기존에 존재하는 게임오브젝트 or 프리펩를 복제하여 새로운 인스턴스를 씬에 생성
시스템 콜을 통해 동적으로 힙 메모리에 공간을 할당받고 저장

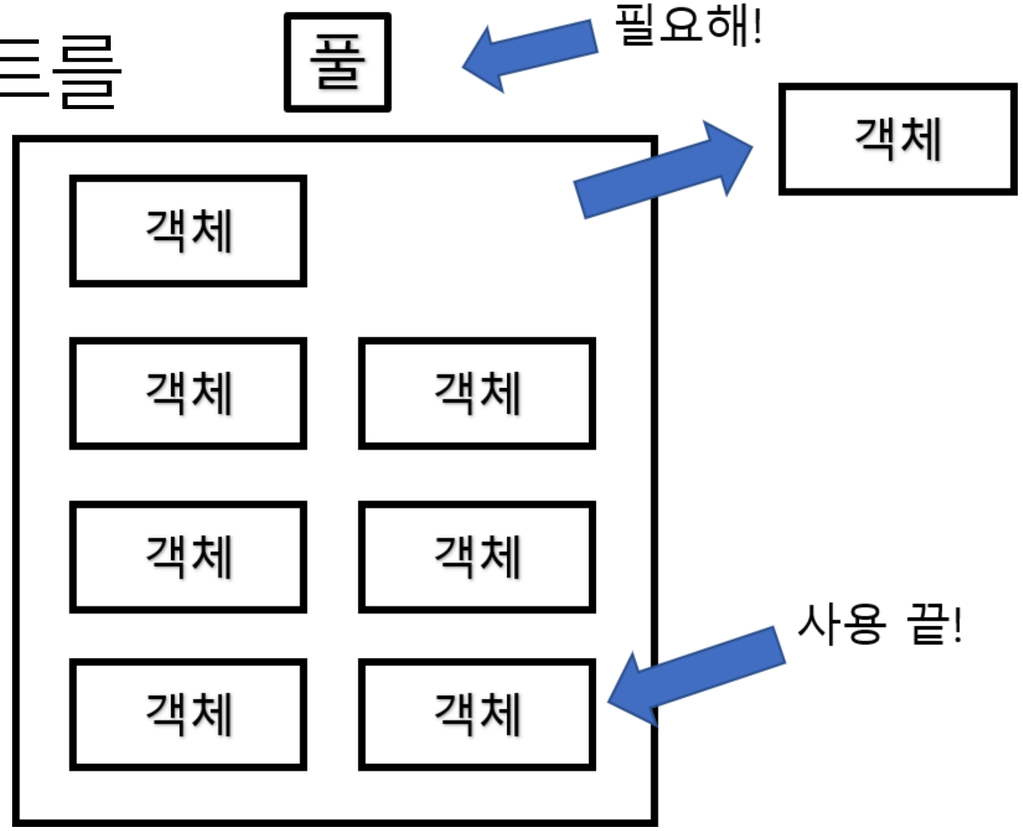
01 오브젝트 생성 방법

파괴: Destroy()
씬에 존재하는 게임오브젝트를 메모리에서 제거
가비지 콜렉터(GC)가 오브젝트를 마크스윕(Mark&Sweep) 방식으로 소멸

1. 프로젝트의 접근 가능한 모든 객체를 탐색하면서 마킹
2. GC가 힙 메모리에서 마킹되지 않은 객체를 모두 제거

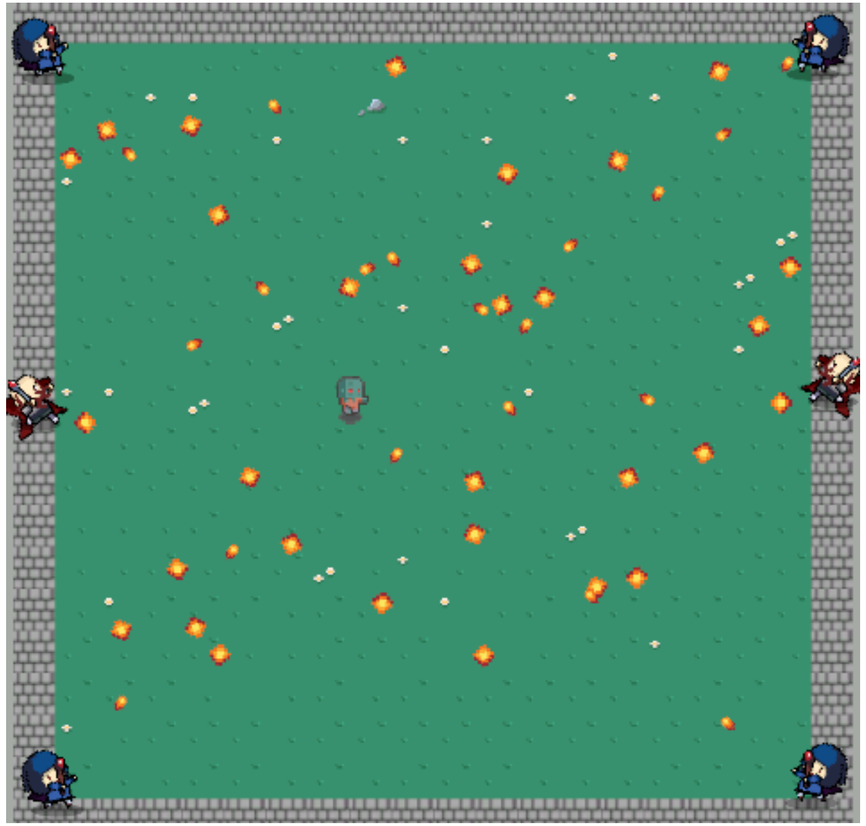
02 오브젝트 풀링이란

1. 게임에서 자주 생성,삭제하는 오브젝트를 미리 일정량 생성하여 풀(pool)에 저장
2. 필요할 때 꺼내 쓰고
3. 사용이 끝나면 비활성화하여 풀에 반환하는 방식



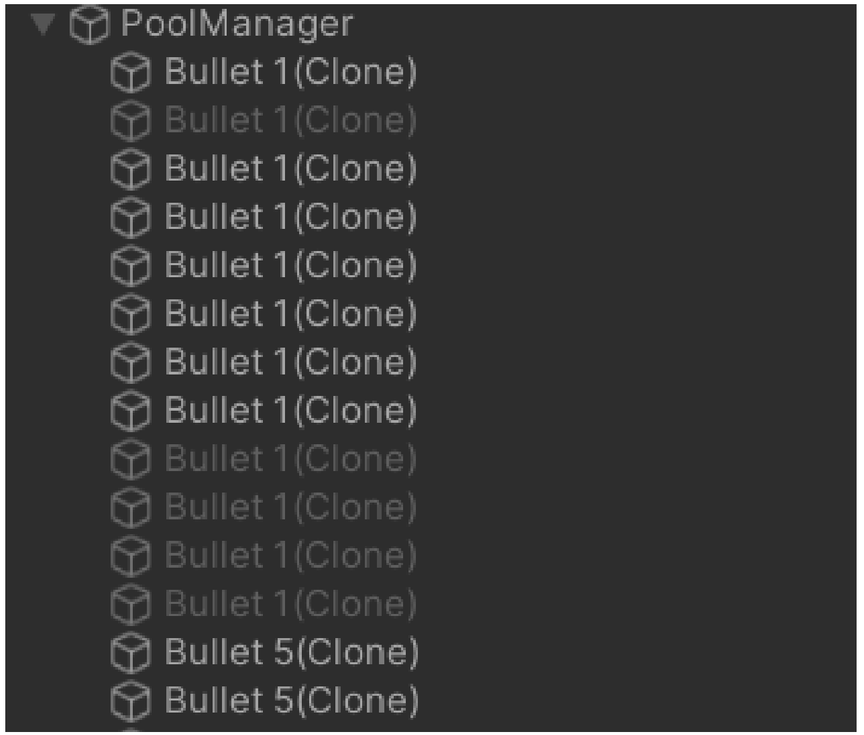
03 오브젝트 풀링을 왜 써야 하는가

Instantiate과 Destroy이 빈번하게
호출되면 메모리의 할당과 해제가 증가
→ 시스템 콜과 GC가 발생
→ 수많은 오버헤드와 부하
→ 프레임 드랍과 렉 발생



03 오브젝트 풀링을 왜 써야 하는가

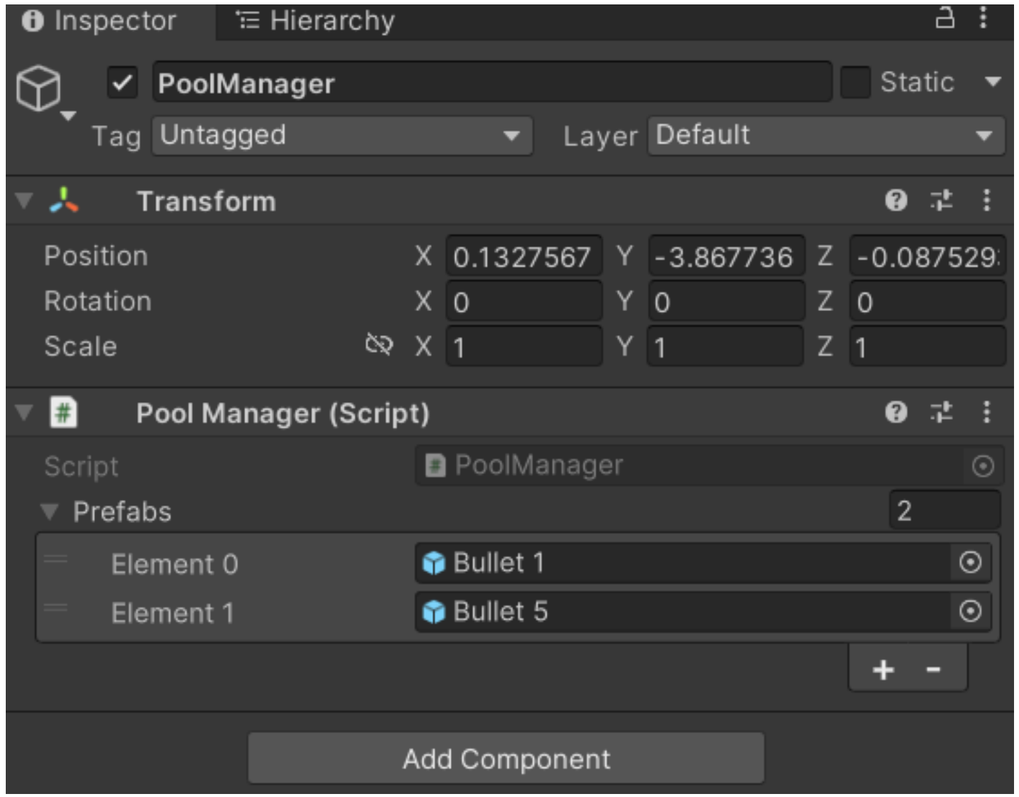
미리 일정량 생성할 때 오버헤드 발생
필요할 때 사용하고
사용이 끝나면 비활성화
→ GC 발생 빈도 감소
→ 일정한 성능 유지



단, 필요 이상의 생성은 오히려 메모리 낭비

04 오브젝트 풀링 구현 방법

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
    Unity 스크립트(자산 참조 1개)|참조 3개
5   public class PoolManager : MonoBehaviour
6   {
7       // 프리펩을 보관할 함수
8       public GameObject[] prefabs;
9       //같은 종류의 프리펩을 담당할 리스트들, 1대1 대칭
10      List<GameObject>[] pools;
11
        Unity 메시지|참조 0개
12      private void Awake()
13      {
14          //프리펩 초기화
15          pools = new List<GameObject>[prefabs.Length];
16          for (int index = 0; index < pools.Length; index++)
17          {
18              pools[index] = new List<GameObject>();
19          }
20          Debug.Log(pools.Length);
21      }
```



04 오브젝트 풀링 구현 방법

```
public GameObject Get(int index)
{
    GameObject select = null;

    //선택한 풀의 놀고 있는 (비활성화 된) 게임 오브젝트 접근
    //발견하면 select변수에 할당
    foreach (GameObject item in pools[index])
    {
        if (!item.activeSelf)
        {
            select = item;
            select.SetActive(true);
            Debug.Log($"기존 총알 재사용: {select.name}");
            //select.GetComponent<Bullet>().ResetBullet();  //총알 초기화
            break;
        }
    }
```

04 오브젝트 풀링 구현 방법

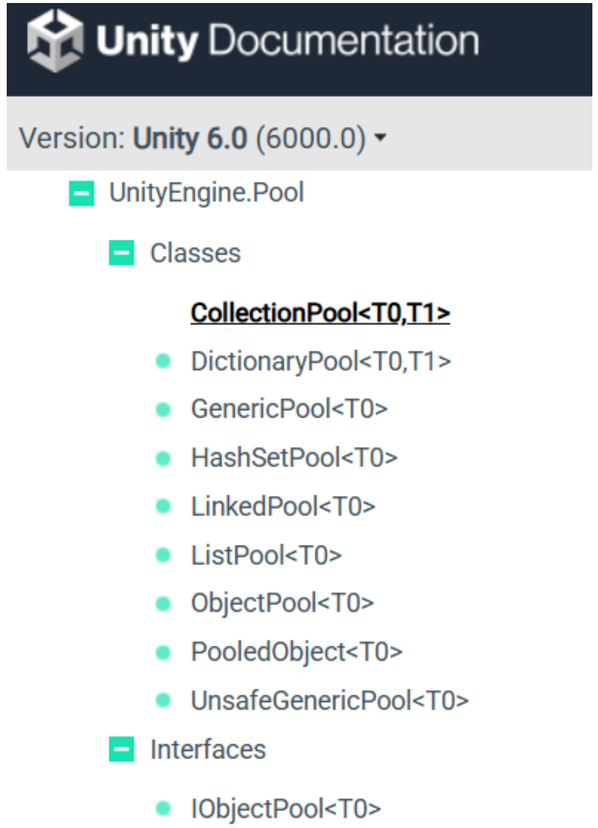
```
//못 찾으면?
//새롭게 생성해서 select 변수에 할당
if (!select)//null은 거짓이니까
{
    select = Instantiate(prefabs[index], transform);
    pools[index].Add(select);
    Debug.Log($"새로운 총알 생성: {select.name}");
}
return select;
```

04 오브젝트 풀링 구현 방법

다양한 자료 구조(queue 등등)를 통해 구현 가능

유니티에서 공식으로 제공하는 형태도 있음

https://docs.unity3d.com/6000.0/Documentation/ScriptReference/Pool.CollectionPool_2.html



04 오브젝트 풀링 구현 방법

프리펩의 레퍼런스를 키로 이용하여 활용한 예시 링크

https://prickly-piccolo-ba8.notion.site/4-2-Object-Pooling-GameManager-1c3fa41fba9f80f9a849f8bbf95d5d30

THANK YOU

감사합니다