

기술 보고서

게임학부 게임소프트웨어 전공 B877033 정주희

게임프로그래밍 패턴

제출일: 2022. 03. 31.

1. 코드 설계

기존 Enemy, Skeleton, Creeper 스크립트를 확인, Skeleton과 Creeper가 추가 코드 작성없이 하나의 스크립트를 컴포넌트로 추가해 작동할 수 있도록 제작하기로 결정했다.

Skeleton과 Creeper은 똑같이 Enemy에서 선언 및 정의된 함수를 사용하고 있으며 이후 FSM 작동 방식 또한 범위의 차이가 있을 뿐이지 동일하다. 그렇기 때문에 EnemyFSM 스크립트를 작성해 두 Object에서 같은 스크립트를 모두 사용한다.

범위의 경우 BoxCollider를 사용해 지정한다.

BoxCollider

1. HitBox(Creeper, Skeleton)

박스 내부에 Player가 들어올 경우 Attack 상태로 전환된다.

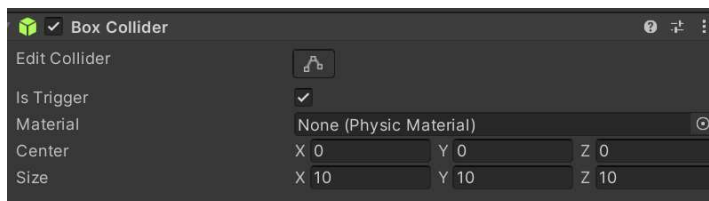
2. ChaseBox(Player)

Enemy object에서 ChaseBox와 닿을 경우 Player를 쫓는다.

2. 구현

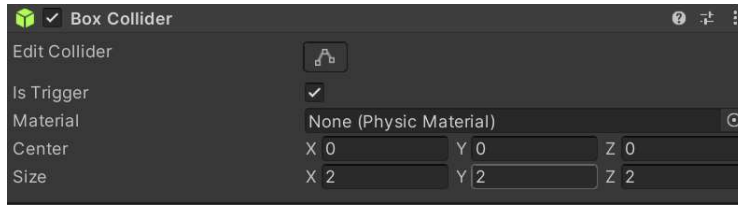
- Skeleton Box Collider

Player와의 거리가 5보다 작을 때 공격이기 때문에 size를 (10,10,10)으로 설정.



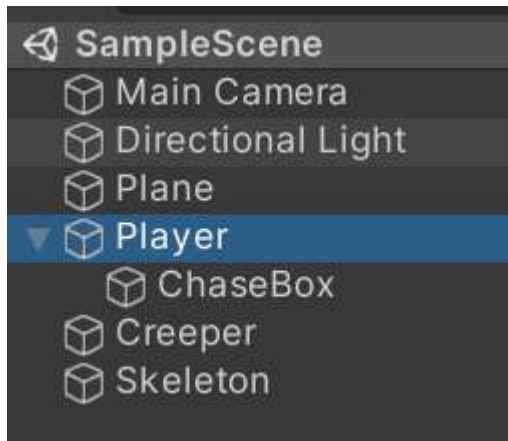
- Creeper Box Collider

Player와의 거리가 1보다 작을 때 공격이기 때문에 size를 (2,2,2)으로 설정.



- Player ChaseBox

Player Object 내부에 ChaseBox를 넣어 Enemy Object 들이 ChaseBox를 Trigger를 통해 인식하면 쫓기 시작하도록 설정.



- 코드 수정

Distance를 삭제하고 collider를 이용하기 위해 대신 bool type 변수를 넣었다. Chase 와 attack 을 통해 Distance 대신 FSM 상태를 변경하도록 조건을 설정한다.

```
//ChaseBox가 감지될 경우 쫓는 상태가 가능하다.
//Player가 감지될 경우 공격이 가능하다.
@ Unity 메시지 | 참조 0개
private void OnTriggerEnter(Collider other)
{
    if(other.name=="ChaseBox")
    {
        chase = true;
    }
    if(other.name=="Player")
    {
        attack = true;
    }
}

//collider 밖으로 벗어나면 상태를 해제해준다.
@ Unity 메시지 | 참조 0개
private void OnTriggerExit(Collider other)
{
    if (other.name == "ChaseBox")
    {
        chase = false;
    }
    if (other.name == "Player")
    {
        attack = false;
    }
}
```

3. 결과

- 원래의 작동과 동일하게 잘 움직이는 것을 확인할 수 있다.

```
[03:19:56] CreeperMoveTowardsPlayer  
UnityEngine.Debug:Log (object)  
[03:20:04] SkeletonMoveTowardsPlayer  
UnityEngine.Debug:Log (object)  
[03:20:08] SkeletonAttack  
UnityEngine.Debug:Log (object)  
[03:20:08] CreeperAttack  
UnityEngine.Debug:Log (object)  
[03:19:56] CreeperFlee  
UnityEngine.Debug:Log (object)  
[03:19:56] CreeperStroll  
UnityEngine.Debug:Log (object)  
[03:20:04] SkeletonFlee  
UnityEngine.Debug:Log (object)  
[03:20:04] SkeletonStroll  
UnityEngine.Debug:Log (object)
```