

【目標】ステージ上に配置した敵キャラを、画面表示範囲に入るまで待機させる。

ここでは、画面外に配置された敵を、プログラム開始時には待機させておき、画面表示範囲内に入った時点で動き出す処理について学習します。

ステージが画面外右側に拡張された状態と仮定します。

ここでの重要キーワード： OnWillRenderObject、

1. 待機処理

配置されたエネミー自身が、表示が面内にあるかどうか判断し、待機状態から動作状態に切り替えます。

画面表示範囲に入ると言うことは、特定のカメラに移っているかどうか、今回は MainCamera の表示範囲かどうかを判断すればいいことになります。

判定するためにはレンダラーを用い、OnWillRenderObject()はそのレンダラーがカメラに映るときに呼び続けられるので、これを利用します。

Enemy1.cs

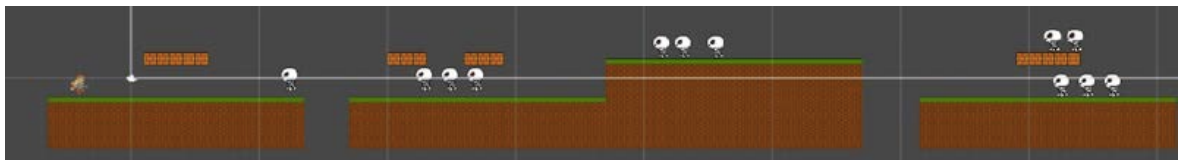
```
1  using UnityEngine;
2  using System.Collections;
3
4  public class Enemy1: MonoBehaviour {
5
6      Rigidbody2D rigidbody2D;
7      public int speed = -3;
8      public GameObject explosion;
9      public int attackPoint = 10;
10     private Life life;
11
12     //メインカメラのタグ名の設定
13     private const string MAIN_CAMERA_TAG_NAME = "MainCamera";
14     //カメラに映っているかの判定フラグ
15     private bool _isRendered = false;
16
17     void Start () {
18         rigidbody2D = GetComponent<Rigidbody2D>();
19         // ライフゲージの取得
20         rigidbody2D = GetComponent<Rigidbody2D>();
21         lifeScript =
22             GameObject.FindGameObjectWithTag("HP").GetComponent<LifeScript>();
23     }
24
25     void Update () {
26         // 表示範囲に入ると動作する。
27         if ( ) {
28             rigidbody2D.velocity = new Vector2 (speed, rigidbody2D.velocity.y);
29         }
30     }
31
32 }
```

```
33 void OnTriggerEnter2D (Collider2D col)
34 {
35     if (col.tag == "Bullet") {
36         Destroy (gameObject);
37         Instantiate (explosion, transform.position, transform.rotation);
38         // 1/4 の確率でアイテム生成
39         if (Random.Range (0, 4) == 0) {
40             Instantiate (item, transform.position, transform.rotation);
41         }
42     }
43 }
44 void OnCollisionEnter2D (Collision2D col)
45 {
46     //UnityChan とぶつかった時
47     if (col.gameObject.tag == "UnityChan") {
48         //Life の LifeDown メソッドを実行
49         life.LifeDown(attackPoint);
50     }
51 }
52
53 //Renderer がカメラに映ってる間に呼ばれ続ける
54 void OnWillRenderObject()
55 {
56     //メインカメラに映った時だけ_isRendered を true
57     if(Camera.current.tag == MAIN_CAMERA_TAG_NAME){
58         _isRendered = true;
59     }
60 }
61 }
```

一番下に追加された OnWillRenderObject メソッドは、カメラに映っている間繰り返し呼ばれるメソッドです。そのカメラのタグが MainCamera であった時、つまりメインカメラに映ったら、_isRendered を true にします。

2. エネミーの配置

ステージ上に、下記のようにエネミーを配置しておきます。



これで、画面に映るまでエネミーが待機しますので、任意のタイミングでエネミーを動かすことができます。しかしこのままでは、不具合が発生します。その不具合の発見と、その対処法考え実装して下さい。

※弾を打ちながら、右にスクロールして下さい。エネミーがちゃんと出てきますか？