

Unity 教學 小朋友下樓梯

 morningfungame.com/2018/10/unity.html



<https://youtu.be/lpd3Ku9Fryg>

CameraManager

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CameraManager : MonoBehaviour {

    public float downSpeed;
    void Start () {

    }

    //預設每秒執行50次
    void FixedUpdate () {
        transform.Translate(0, -downSpeed * Time.deltaTime, 0);
    }
}
```

DeadZone

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class DeadZone : MonoBehaviour {

    private void OnCollisionEnter2D(Collision2D other)
    {
        if (other.gameObject.CompareTag("玩家"))
        {
            Player.isDead = true;
            //Debug.Break();
        }
    }
}
```

GameManager

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;

public class GameManager : MonoBehaviour {

    public Button restartButton;
    public GameObject player;
    void Start () {
        restartButton.gameObject.SetActive(false);
    }

    void Update () {
        if (Player.isDead)
        {
            player.SetActive(false);
            restartButton.gameObject.SetActive(true);
        }
    }

    public void ReloadScene() {
        SceneManager.LoadScene(SceneManager.GetActiveScene().name);
    }
}
```

GroundManager

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class GroundManager : MonoBehaviour
{
    readonly float leftBorder = -3;//左邊界
    readonly float rightBorder = 3;//右邊界
    readonly float initPosotionY = 0;
    readonly int MAX_GROUND_COUNT = 10;//最大地板數量
    readonly int MIN_GROUND_COUNT_UNDER_PLAYER = 3;//玩家下方最少地板數量
    static int groundNumber = -1;
    [Range(2, 6)] public float spacingY;
    [Range(1, 20)] public float singleFloorHeight;
    public List<Transform> grounds;
    public Transform player;
    public Text displayCountFloor;

    void Start()
    {
        grounds = new List<Transform>();
        for (int i = 0; i < MAX_GROUND_COUNT; i++)
        {
            SpawnGround();
        }
    }
    public void ControlSpawnGround()//控制產生地板
    {
        int groundsCountUnderPlayer = 0;//玩家下方的地板數量
        foreach (Transform ground in grounds)
        {
            if (ground.position.y < player.transform.position.y)
            {
                groundsCountUnderPlayer++;
            }
        }

        if (groundsCountUnderPlayer < MIN_GROUND_COUNT_UNDER_PLAYER)
        {
            SpawnGround();
            ControlGroundsCount();
        }
    }

    void ControlGroundsCount()//控制地板數量
    {
        if (grounds.Count > MAX_GROUND_COUNT)
        {

```

```

        Destroy(grounds[0].gameObject);
        grounds.RemoveAt(0);
    }
}
float NewGroundPositionX()
{
    if (grounds.Count == 0)
    {
        return 0;
    }
    return Random.Range(leftBorder, rightBorder);
}

//計算新地板的Y座標
float NewGroundPositionY()
{
    if (grounds.Count == 0)
    {
        return initPosotionY;
    }
    int lowerIndex = grounds.Count - 1;
    return grounds[lowerIndex].transform.position.y - spacingY;
}

//產生單一地板
void SpawnGround()
{
    GameObject newGround = Instantiate(Resources.Load<GameObject>("地板"));
    newGround.transform.position = new Vector3(NewGroundPositionX(), NewGroundPositionY(), 0);
    grounds.Add(newGround.transform);
    groundNumber++; //地板編號+1
    newGround.name = "地板" + groundNumber; //修改物件名稱為地板+流水編號
}
float CountLowerGroundFloor() {
    float playerPositionY = player.transform.position.y;
    float deep = Mathf.Abs(initPosotionY-playerPositionY);
    return (deep / singleFloorHeight)+1;
}
void DisplayCountFloor() {
    displayCountFloor.text = "地下" + CountLowerGroundFloor().ToString("0000") + "樓";
}
void Update()
{
    ControlSpawnGround();
    DisplayCountFloor();
}
}

```

Player

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Player : MonoBehaviour {

    public float forceX;//水平推力
    public static bool isDead;
    Rigidbody2D playerRigidBody2D;
    readonly float toLeft = -1;
    readonly float toRight = 1;
    readonly float stop = 0;
    float directionX;
    void Start () {
        isDead = false;
        playerRigidBody2D = GetComponent<Rigidbody2D>();
    }

    void Update () {
        if (Input.GetKey(KeyCode.LeftArrow))
        {
            directionX = toLeft;
        }
        else if (Input.GetKey(KeyCode.RightArrow))
        {
            directionX = toRight;
        }
        else
        {
            directionX = stop;
        }
        Vector2 newDirection = new Vector2(directionX,0);
        playerRigidBody2D.AddForce(newDirection*forceX);
    }
}

```