

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

public class Player : MonoBehaviour
{
    public float Speed; //動く速さ
    [SerializeField]private GameManager gm;
    private Rigidbody rb; //Rigidbody

    void Start()
    {
        //Rigidbodyを取得
        rb = GetComponent<Rigidbody>();
    }

    void Update()
    {
        //カーソルキーの入力を取得
        float moveHorizontal = Input.GetAxis("Horizontal");
        float moveVertical = Input.GetAxis("Vertical");

        //カーソルキーの入力に合わせて移動方向を設定
        Vector3 movement = new Vector3(0, 0, 0);

        //Rigidbody に力を与えて玉を動かす
        rb.AddForce(movement * Speed);
    }

    void OnTriggerEnter(Collider other) {
        //衝突したタグがItemかどうか
        if (other.gameObject.tag == "Item")
        {
            //衝突した相手のGameObjectを消す
            Destroy(other.gameObject);

            //カウント更新
            gm.UpdateCount();
        }
    }
}
```

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

public class Player : MonoBehaviour
{
    public float Speed; //動く速さ
    [SerializeField]private GameManager gm;
    private Rigidbody rb; //Rigidbody

    void Start()
    {
        //Rigidbodyを取得
        rb = GetComponent<Rigidbody>();
    }

    void Update()
    {
        //カーソルキーの入力を取得
        float moveHorizontal = Input.GetAxis("Horizontal");
        float moveVertical = Input.GetAxis("Vertical");

        //カーソルキーの入力に合わせて移動方向を設定
        Vector3 movement = new Vector3(moveHorizontal, 0, moveVertical);

        //Rigidbody に力を与えて玉を動かす
        rb.AddForce(movement * Speed);
    }

    void OnTriggerEnter(Collider other) {
        //衝突したタグがItemかどうか
        if (other.gameObject.tag == "Item")
        {
            //衝突した相手のGameObjectを消す
            Destroy(other.gameObject);

            //カウント更新
            gm.UpdateCount();
        }
    }
}
```

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

public class Camera : MonoBehaviour
{
    public GameObject player; //玉のオブジェクト

    private Vector3 offset; //玉からカメラまでの距離

    void Start()
    {
        //2点の位置(カメラとボール)の距離を求める
        offset = transform.position - player.transform.position;
    }

    void LateUpdate()
    {
        //ボールを基準に距離分を足す
        transform.position = player.transform.position + offset;
    }
}
```

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

public class ItemRotate : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        //Time.deltaTime(1秒間)に回る角度を指定
        transform.Rotate(new Vector3(0, 90, 0) * Time.deltaTime);
    }
}
```

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

public class GameManager : MonoBehaviour
{
    [SerializeField]private TextMeshProUGUI CountText;
    private int totalCount;
    private int nowCount;
    private GameObject[] item;

    // Start is called before the first frame update
    void Start()
    {
        //獲得したボールの数
        nowCount = 0;

        //配置したアイテムのGameObjectを取得
        item = GameObject.FindGameObjectsWithTag("Item");
        totalCount = item.Length;
    }

    // Update is called once per frame
    void Update()
    {
    }

    public void UpdateCount()
    {
        //獲得したボールの数をカウント
        nowCount++;

        //テキスト更新
        CountText.text = "Count : " + nowCount.ToString();

        //全て獲得したか判定
        if(nowCount >= totalCount)
        {
            //シーン再読み込み
            SceneManager.LoadScene(SceneManager.GetActiveScene().name);
        }
    }
}
```