

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Player : MonoBehaviour
6 {
7     public GameObject bullet;
8     public GameObject spaw;
9     public LayerMask floor;
10    CharacterController chaControl;
11    Vector3 character;
12    public Transform chackfloor;
13
14
15
16    float detegarvity = -6.5f;
17
18    public int haveJump;
19    public int jumpCount;
20    public float horizontal;
21    public float vertical;
22    public float garvity;
23    public float radius;
24    public int speedMove;
25    public int speedJump;
26    public float speedRotation;
27    // Start is called before the first frame update
28    void Start()
29    {
30        chaControl = GetComponent<CharacterController>();
31    }
32
33    // Update is called once per frame
34    void Update()
35    {
36
37        MovePlayer();
38        Chackface();
39        shot();
40        Garvity();
41    }
42
43    void MovePlayer()
44    {
45        garvity += detegarvity * Time.deltaTime;
46        horizontal = Input.GetAxis("Horizontal");
47        vertical = Input.GetAxis("Vertical");
48        character = new Vector3(horizontal * speedMove *
                                Time.deltaTime, garvity * Time.deltaTime, 0);
49        if (Input.GetKeyDown("w") && haveJump > 0)
50        {
51            haveJump--;
52            character = new Vector3(0, vertical + speedJump *
                                Time.deltaTime, 0);
```

```
        Time.deltaTime, 0);
53         garvity = speedJump;
54
55     }
56     chaControl.Move(character);
57
58 }
59
60 void shot()
61 {
62     if (Input.GetKeyDown("space"))
63     {
64         Instantiate(bullet, spaw.transform.position,
65                     spaw.transform.rotation);
66     }
67
68 void Garvity()
69 {
70     if(Physics.CheckSphere(chackfloor.transform.position, radius,
71                             floor))
72     {
73         haveJump = jumpCount;
74         garvity = 0;
75     }
76
77 void Chackface()
78 {
79     if(horizontal < 0)
80     {
81         transform.rotation = Quaternion.Lerp(transform.rotation,
82         Quaternion.Euler(0, 0, 0), speedRotation * Time.deltaTime);
83     }
84     if (horizontal > 0)
85     {
86         transform.rotation = Quaternion.Lerp(transform.rotation,
87         Quaternion.Euler(0, 180, 0), speedRotation *
88         Time.deltaTime);
89     }
90 }
91 }
```