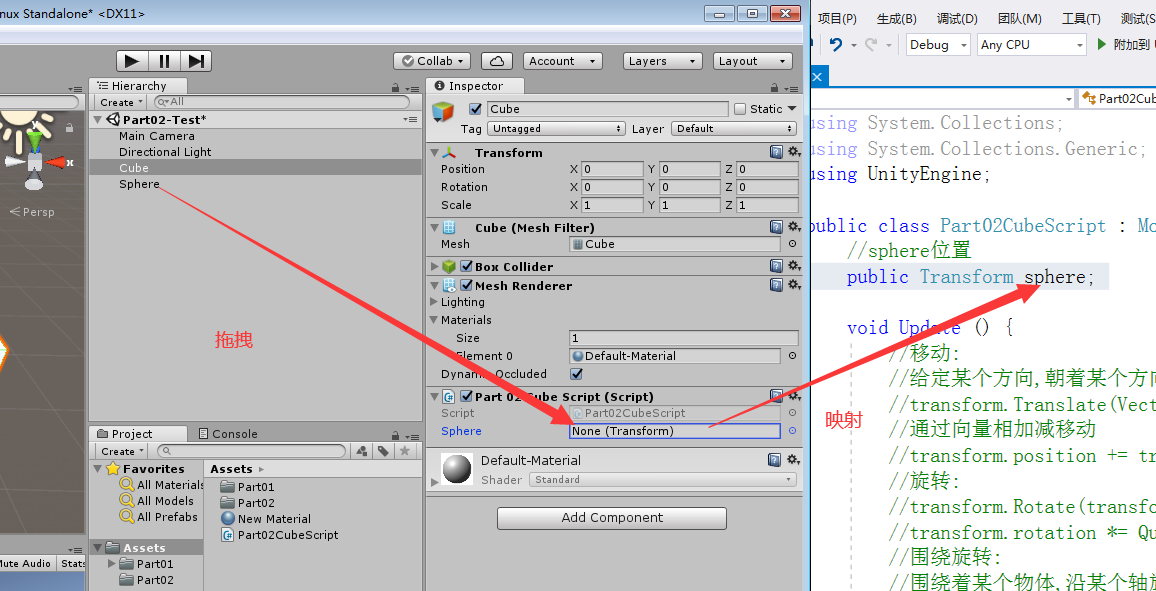
**物体位置**

找到sphere

（1）**首先写上**

**（2）在公开字段,拖拽的方式,引用物体**



The variable Sphere of Part2cubescript has not been assigned.

You probably need to assign the Sphere variable of the Part2cubescript script in the inspector.

**代码：**

//sphere位置

public Transform sphere;

void Update()

{

//移动

//1.给定某个方向，朝着某个方向移动

//transform.Translate(Vector3.forward \* Time.deltaTime \* 5);

//2.通过向量相加减移动

//transform.position += transform.forward \* Time.deltaTime \* 6f;

//旋转：

//transform.Rotate(transform.up \* 45f);

//transform.rotation \*= Quaternion.Euler(0,45,0);

//2.围绕旋转：围绕着某个物体，沿某个轴旋转多少度

//参数1：旋转的物体位置，参数2：绕着旋转的轴，参数3：旋转的度数。

transform.RotateAround(sphere.position, sphere.up, 45\*Time.deltaTime);

//LookAt：看向某个位置

//z轴朝向该位置

transform.LookAt(sphere.position);

}

**摄像机跟随：在摄像机物体上的脚本。**

1:摄像机和人物的距离永远保持一致



**（1）确定人物和摄像机之间的向量**  向量:有大小有方向的量

人物到摄像机的向量 = 摄像机的位置-人物的位置

**方法：**Find, 拖拽

**代码：**

Transform player;

Vector3 offset;

void Start()

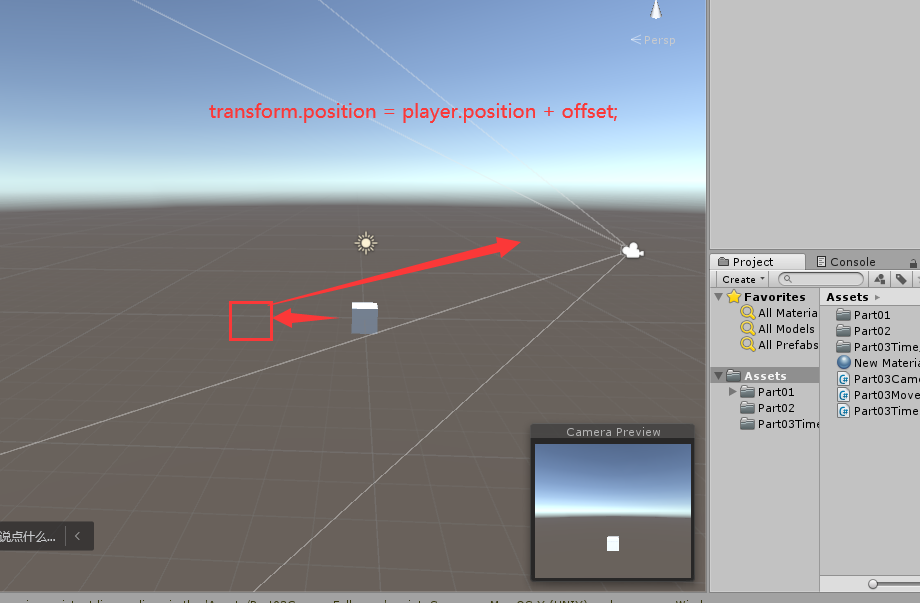
{

player = GameObject.Find("Cube").transform;

//人物到摄像机的向量 = 摄像机的位置-人物的位置

offset = transform.position - player.position;

}

**（2）相机移动位置=人物位置+偏移量**

**代码：**

void Update () {  
        //相机移动的位= 人物位置+偏移量  
        transform.position = player.position + offset;  
}

**暂停和继续**

if(GUI.Button(new Rect(0,0,70,30), "暂停"))

{

Time.timeScale = 0;

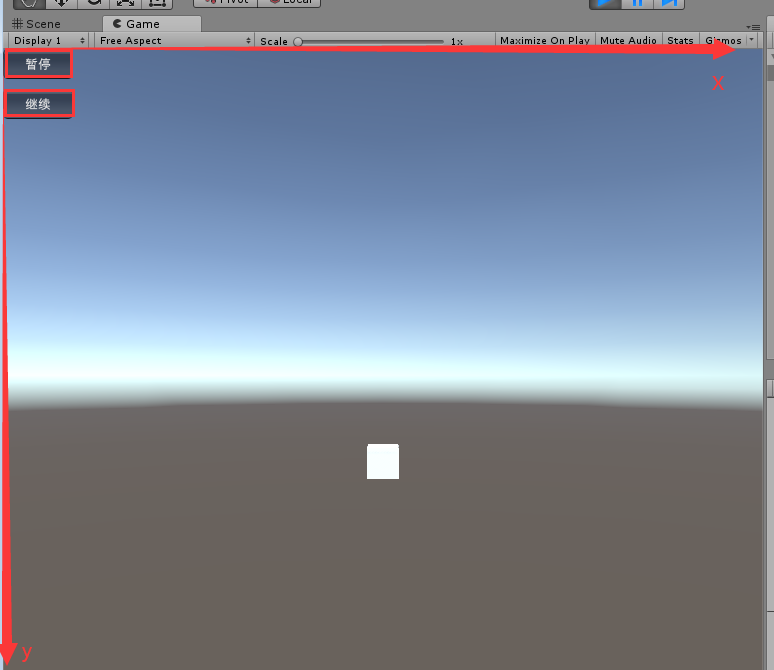
}

if (GUI.Button(new Rect(0, 40, 70, 30), "继续"))

{

Time.timeScale = 1;

}



**在cube中的脚本**

transform.position += transform.forward \* Time.deltaTime \* 6f;

**在摄像机脚本中**

代码：

void Start()

{

//Time类：

//游戏从开始到现在的运行时间

//Debug.Log(Time.time); 放在Update

//时间缩放，可以控制游戏暂停和开始

//值为0：游戏暂停

//值为0：游戏正常

//注意：只能暂停和时间相关的

//Time.timeScale=0;

}

void Update()

{

//Debug.Log(Time.time);

}

private void OnGUI()

{

//手动通过代码，创建UI控件

//Button：按钮

//参数1：Rect（x，y，宽，高）

//参数2：控件的名字

if(GUI.Button(new Rect(0,0,70,30), "暂停"))

{

Time.timeScale = 0;

}

if (GUI.Button(new Rect(0, 40, 70, 30), "继续"))

{

Time.timeScale = 1;

}

}