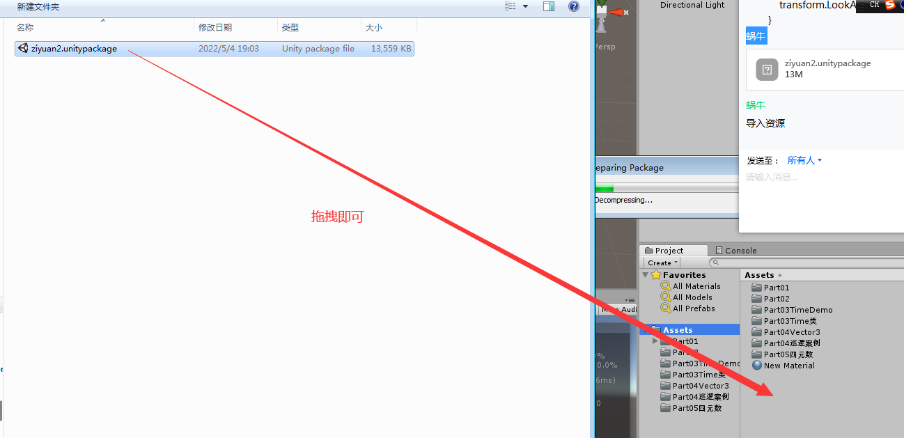
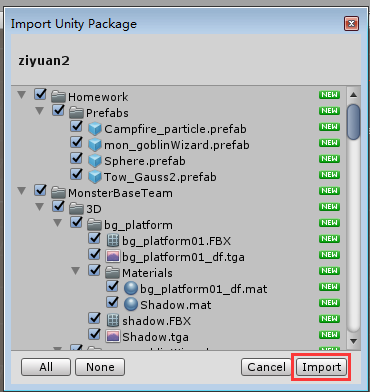
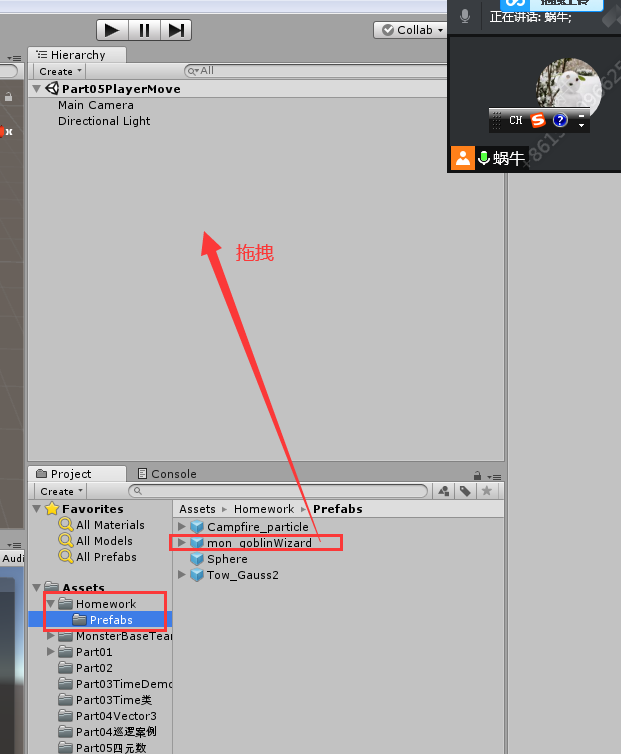
**1.导入资源**

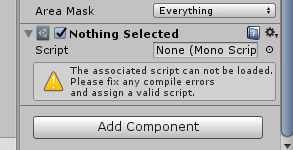


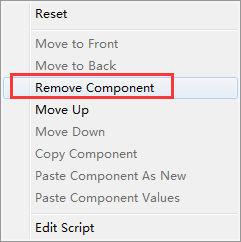
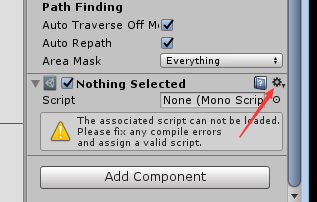




**拖拽之后坐标归零 ，在创建一地板，坐标归零。**

**2. 意味着脚本丢失**





**3.自创脚本**

**（1）代码：**

void Update()

{

#region 键盘移动

////Horizontal:水平，代表的是键盘的a,d。和->，<-。

////取值范围：-1~1，松开手值为0

//float hor =Input.GetAxis("Horizontal");

////Vertical:垂直，代表的是键盘的w,s。和上箭头，下箭头。

////取值范围：-1~1，松开手值为0

//float ver = Input.GetAxis("Vertical");

//transform.position += transform.forward \* ver \* Time.deltaTime \* moveSpeed;

//transform.Rotate(transform.up \* hor \* rotateSpeed);

#endregion

}

**(2)** void Update()

{

#region 键盘移动2

float hor = Input.GetAxis("Horizontal");

float ver = Input.GetAxis("Vertical");

//按下按键才会执行

//输入，w向前旋转，s后旋转，A左旋转，D右旋转

if (hor != 0 || ver != 0)

{

MovementRotate(hor, ver);

}

#endregion

}

private void MovementRotate(float hor,float ver)

{

//旋转

Quaternion qua=Quaternion.LookRotation(new Vector3(hor,0,ver));

//由快到慢旋转:其实就是将快速旋转的过程分成了分帧旋转

//transform.rotation = Quaternion.Lerp(transform.rotation, qua,Time.deltaTime \* rotateSpeed);

//匀速旋转：

transform.rotation = Quaternion.RotateTowards(transform.rotation, qua, rotateSpeed);

//移动

transform.Translate(new Vector3(0, 0, Time.deltaTime \* moveSpeed));

}