

**代码：**

public class CubeRay : MonoBehaviour

{

/\*1.

//碰撞器可以拦截射线

RaycastHit hit;

void Update()

{

//Ray:射线

//创建射线

//参数1：射线产生位置

//参数2：射线的方向

Ray ray = new Ray(transform.position, transform.forward \* 10);

//发射射线： Physics.Raycast

//参数1：要发射的射线

//参数2：射线碰撞物体信息（位置，名字，标签等等）

bool iosk = Physics.Raycast(ray, out hit);

if (iosk)

{

hit.transform.gameObject.GetComponent<MeshRenderer>().material.color = Color.red;

//打印射线

Debug.DrawLine(transform.position, transform.forward \* 10, Color.red);

}

}

\*/

public Transform sphere;

RaycastHit hit;

private void Update()

{

if(Input.GetMouseButton(0))

{

//Physics.Raycast

//参数1：射线产生的位置

//参数2：射线发射的方向

//参数3：代表射线碰撞物体信息

//maxDistance：最大距离，可以不写，默认无限长

if (Physics.Raycast(transform.position, sphere.position - transform.position, out hit))

{

hit.collider.GetComponent<MeshRenderer>().material.color = Color.red;

}

}

}

}

public class CameraRay : MonoBehaviour

{

public Transform cube;

RaycastHit hit;

void Start()

{

}

void Update()

{

if (Input.GetMouseButton(0))

{

Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);

if (Physics.Raycast(ray, out hit))

{

//位置赋值

//将射线碰撞物体的位置赋值给cube

cube.position = new Vector3(hit.point.x,0.5f, hit.point.z);

}

}

}

}