鼠标转世界坐标

//Vector3 screenPos = Camera.main.WorldToScreenPoint(m\_obj.transform.position);//获取需要移动物体的世界转屏幕坐标

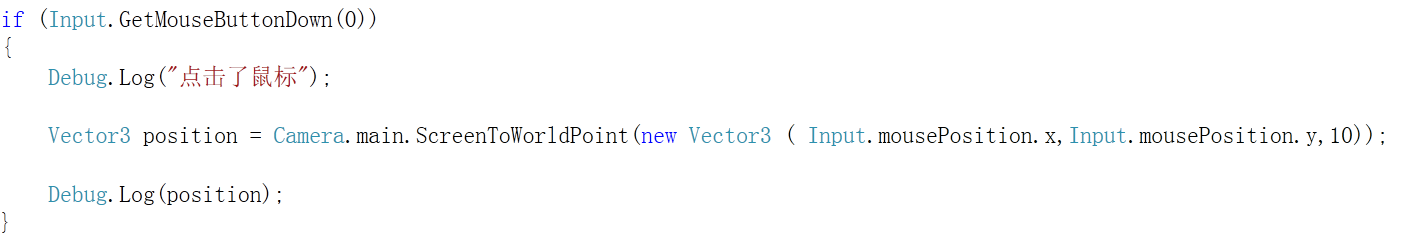
//Vector3 mousePos = Input.mousePosition;//获取鼠标位置

//mousePos.z = screenPos.z;//因为鼠标只有X，Y轴，所以要赋予给鼠标Z轴

//Vector3 worldPos = Camera.main.ScreenToWorldPoint(mousePos);//把鼠标的屏幕坐标转换成世界坐标

//m\_obj.position = worldPos;//控制物体移动

}



血条名字跟随

//ui对象

public RectTransform UIObj;

//需要跟随的对象

public Transform FollowedObj;

1. UIObj.position = RectTransformUtility.WorldToScreenPoint(Camera.main,FollowedObj.position)
2. UIObj.position = Camera.main.WorldToScreenPoint(FollowedObj.position);
3. 世界坐标到屏幕坐标，再到制定的canvas的相对位置的世界坐标，用于ui。因为是在canvas上所在的世界坐标，值很大，不是在摄像机里面的位置

//物体世界坐标在世界相机所形成的屏幕里面所在的位置：屏幕坐标，这个坐标再对应到UI相机上，然后求出来这个UI相机上这个位置的视口坐标在3D世界里的位置。

screenPoint = Camera.main.WorldToScreenPoint(FollowedObj.position);//世界到屏幕坐标

var canvasTran = GameObject.Find("Canvas").GetComponent<RectTransform>();

Vector3 worldPos;

RectTransformUtility.ScreenPointToWorldPointInRectangle(canvasTran, new Vector2(screenPoint.x, screenPoint.y), null, out worldPos);//屏幕到世界坐标

UIObj.transform.position = worldPos;