**Unity3D教程：射线碰撞**

Physics.Raycast 光线投射  
  
    1.一个静态的发射射线的方法，在场景中投下可与所有碰撞器碰撞的一条光线。  
  
    参数解释：  
  
    origin:射线起始点  
  
    direction:射线方向  
  
    distance:射线长度  
  
    layerMask:只选定Layermask层内的碰撞器，其它层内碰撞器忽略。  
  
    返回布尔类型，当光线投射与任何碰撞器交叉时为真，否则为假。（注意：如果从一个球型体的内部到外部用光线投射，返回为假。 ）这一点测试不成功！  
  
    2.在场景中投下可与所有碰撞器碰撞的一条光线，并返回碰撞的细节信息。[Unity3D](http://www.unitymanual.com/" \t "_blank)教程手册  
  
    Parameters参数  
  
    origin：在世界坐标，射线的起始点。  
  
    direction：射线的方向。  
  
    distance：射线的长度。  
  
    hitInfo：如果返回true，hitInfo将包含碰到器碰撞的更多信息。  
  
    layerMask：只选定Layermask层内的碰撞器，其它层内碰撞器忽略。  
  
    Returns：当光线投射与任何碰撞器交叉时为真，否则为假。  
  
    使用c＃中out关键字传入一个空的碰撞信息类，然后碰撞后赋值。可以得到碰撞物体的transform,rigidbody,point等信息。  
  
    详细解释一下: LayerMask 层蒙版  
  
    每一个物体都有一个Tag和Layer标签，tag就是一个简单的标签，可以用来区分不同的物体，LayerMask的作用是可以选择性地过滤物体，例如当投射射线时，比如camera.cullingmask和光线投射时候。[Unity3D教程](http://www.unitymanual.com/" \t "_blank)手册  
  
    可以在TagManager中编辑tag和Layer。然后设置物体的Layer层级，在摄像机中设置camera.cullingmask,可以控制摄像机的渲染层级，用在射线上，可以控制射线碰撞什么，不碰撞什么。

//Raycast hit 　　RaycastHit hit; 　　LayerMask mask = 1 << 8; 　　void testRay(){ 　　if(Physics.Raycast(transform.position,Vector3.right,out 　　hit,100, mask.value)){ 　　Debug.DrawLine(transform.position,hit.point,Color.red,1); 　　} 　　}

 上面的例子是从当前脚本所绑定的物体做标点向右发射长度100的射线，射线碰撞层级为8，其他层级忽略。为了测试，我们在碰撞到物体后画一条红色线条，表示射线，停留一秒钟。上面这些是射线的一些基本操作。