构造射线的2种方式

```
1.通过构造函数创建射线 new Ray()
2.摄像机创建射线
```

在场景中画一条与射线的起点和方向一样的线

```
  Debug.DrowRay();
```

使用Physics类检测射线和其他碰撞器的碰撞

```
  Physics.Raycast(射线,out 碰撞信息,距离,层)
```

RaycastHit 表示射线碰撞信息,有如下常用属性

Api	解释
point	碰撞点
collider	碰撞器
distance	碰撞点和射线的起点的距离
normal	碰撞点所在面的法线(可以做弹痕效果)

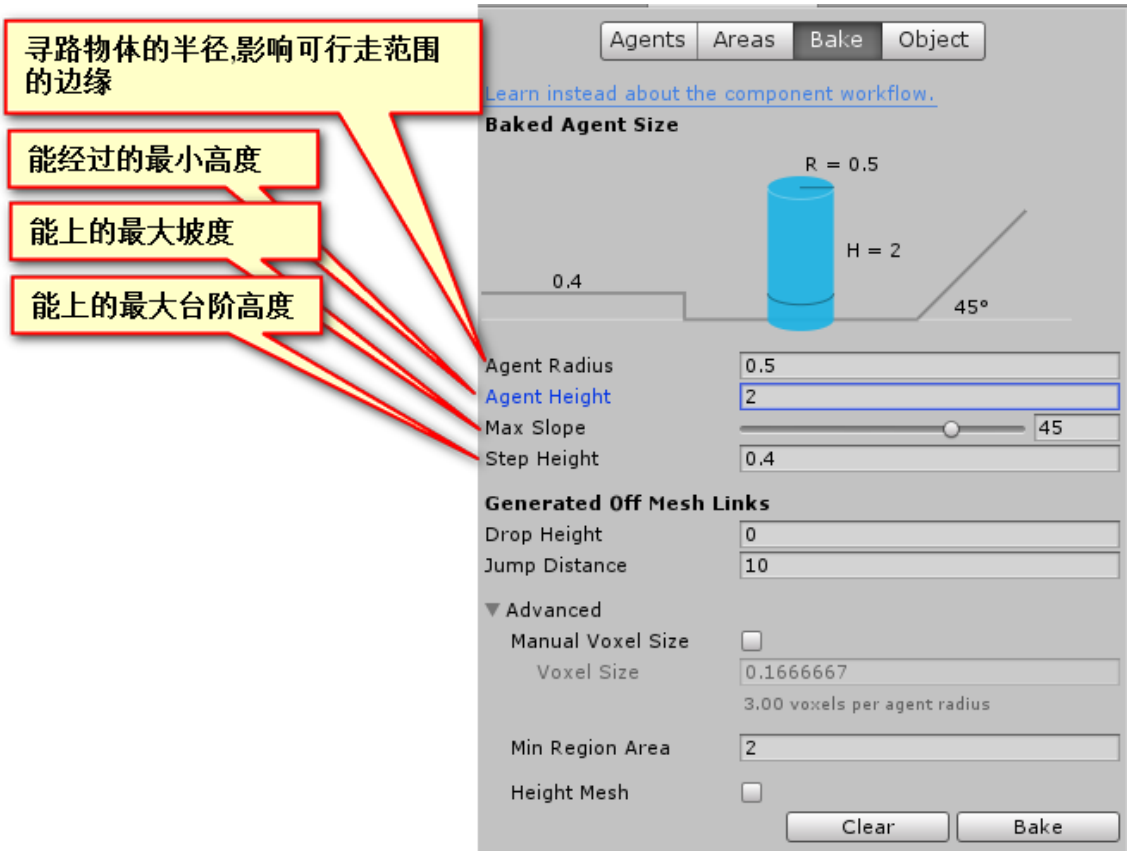
Physics类的其他方法

Api	解释
Physics.RaycastAll()	检测射线穿过的所有物体
Physics.Linecast()	线段检测
Physics.OverlapSphere()	相交球检测(检测球形范围内的碰撞体)

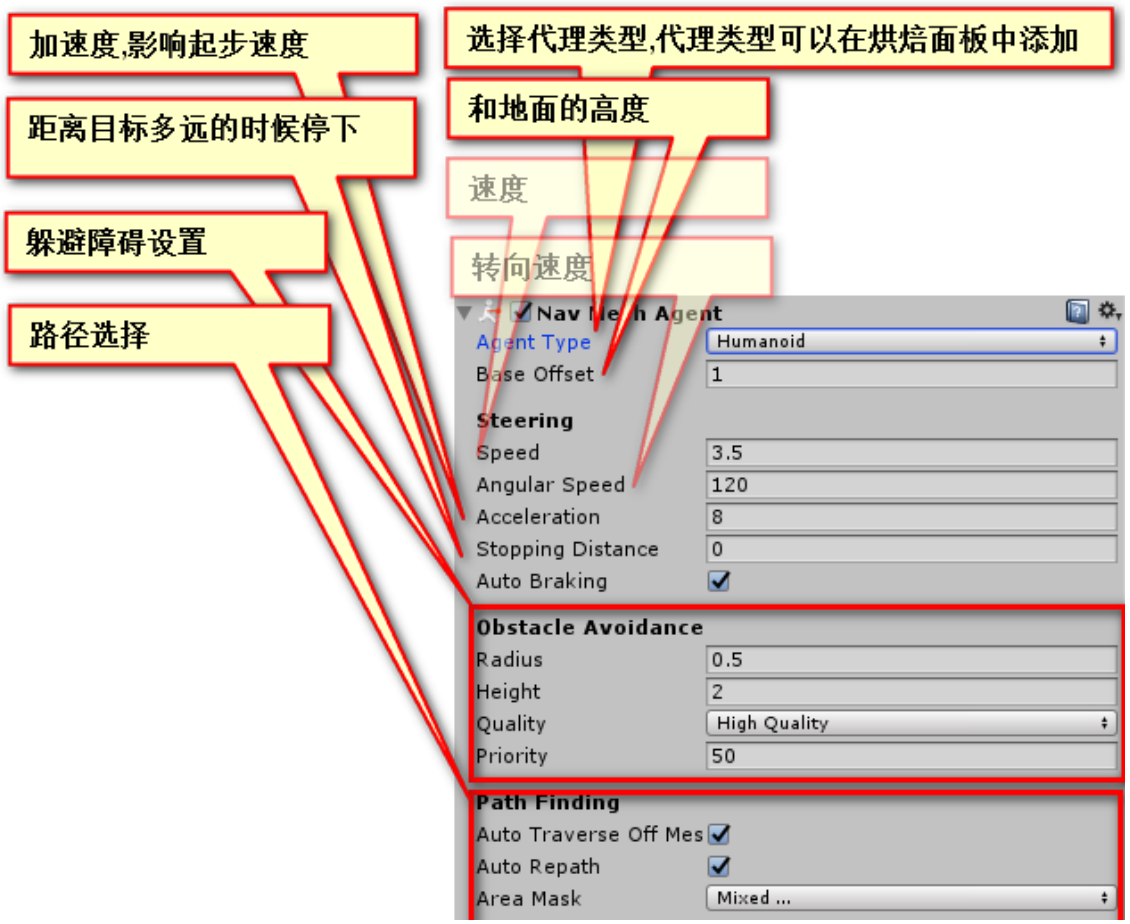
NavMesh寻路系统(导航网格)

构建导航网格4步骤

1.选中场景中的静态物体(运行时不会移动的物体)
2.勾选NavgationStatic
3.设置烘焙面板参数:菜单>Window>AI>Navigation>Bake选项卡
4.点击烘焙按钮



角色寻路组件: NavMeshAgent



通过脚本设置寻路目标

```
public class D07_NavMesh : MonoBehaviour
{
    NavMeshAgent nav;//寻路组件
    private void Start()
    {
        nav = GetComponent<NavMeshAgent>();
    }
    void Update()
    {
        //点击地板寻路到指定位置
        if (Input.GetMouseButtonDown(0))
        {
            Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
            RaycastHit hit;
            if (Physics.Raycast(ray,out hit,1000))
            {
                nav.SetDestination(hit.point);//设置寻路目标
            }
        }
```

```
20      }
21  }
```

作业:

> 点击方块表示选择方块,并把选择的方块变为红色
> 点击地板则让当前选择的方块移动到鼠标点击的位置