

Random

```
Random.Range(); //随机
Random.value; //随机取0-1之间的浮点数
Random.ColorHSV(); //获得一个随机颜色
Random.rotation; //获得一个随机朝向(旋转)
Random.insideUnitCircle; //返回半径为1 的圆内的点
Random.insideUnitSphere; //返回半径为1 球内的点
Random.onUnitSphere; //返半径为1 球面的点
```

OnGUI

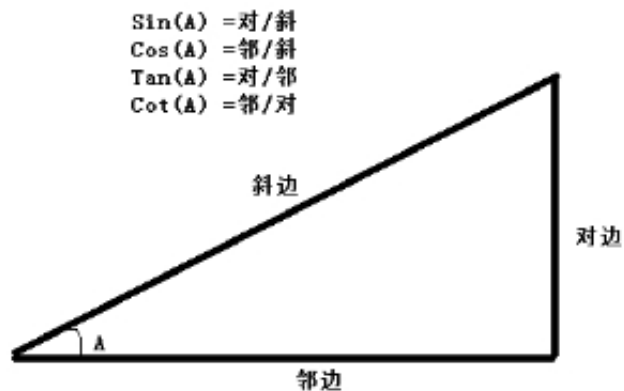
```
1 void OnGUI()
2 {
3     if(GUILayout.Button("测试"))
4     {
5         print("Hello");
6     }
7 }
```

Mathf

```

Mathf.Abs();//绝对值
Mathf.Max();//最大值
Mathf.Clamp();//限制 <*****>
Mathf.Clamp01();//限制在0和1之间
Mathf.Lerp();//插值 根据比例求值 <*****>
Mathf.InverseLerp();//反插值 根据值求比例
Mathf.Deg2Rad;//度到弧度的转化量 360度等于2Pi弧度 2 to
<*****>
Mathf.Rad2Deg//弧度到度的转化量 <*****>
Mathf.Sin();//求正弦值 参数是弧度
Mathf.Cos();//求余弦 参数是弧度
Mathf.Tan();//求正切 参数是弧度
Mathf.Round();//四舍五入
Mathf.Repeat();//重复 (有点像小数取余数)
Mathf.PingPong();//
Mathf.Pow();//求几次方
Mathf.ClosestPowerOfTwo();//返回最近的2的N次方
Mathf.NextPowerOfTwo();//返回下一个2的n次方
Mathf.MoveTowards();//朝某个值变化 <*****>

```



Resources 资源加载

资源必须放在 `Resources` 文件夹下 (`Resources`文件夹可以有很多个,尽量避免资源重名)

```
Resources.Load();//加载
```

```
Resources.LoadAll();
```

```
Resources.UnloadAsset();//卸载 不能卸载预制体
```

```
Resources.UnloadUnusedAssets();
```

Unity中常见的资源类型

预制体:`GameObject`

图片:`Texture`

精灵:`Sprite`

声音片段:`AudioClip`

视频:`VideoClip` (命名空间:`UnityEngine.Video`)

txt文本:`TextAsset`

材质球:`Material`

PlayerPrefs 玩家偏好(保存简单数据)

`PlayerPrefs`的基本特征如下:

1. 存储方式:以键值对存储,类似于字典
2. 读取也是以键值对来读取,若无则返回默认值
3. `PlayerPrefs`类可保存三种类型,浮点型,整型,字符串型

```
//在Windows平台中PlayerPrefs 存放在注册表中
```

```
// 不能使用PlayerPrefs 保存太长的字符串
```

作业

1实现小怪沿路点移动的功能(`Vector3.MoveTowards(起点,终点,速度)`)

2狙击镜增加lerp效果