# 3.22

## NGUI按钮添加事件方法

## 加载物体

****

## **[C#索引器](http://www.cnblogs.com/ArmyShen/archive/2012/08/27/2659405.html)**

索引器允许类或者结构的实例按照与数组相同的方式进行索引取值，索引器与属性类似，不同的是索引器的访问是带参的。

索引器和数组比较：

(1)索引器的索引值(Index)类型不受限制

(2)索引器允许重载

(3)索引器不是一个变量

索引器和属性的不同点

(1)属性以名称来标识，索引器以函数形式标识

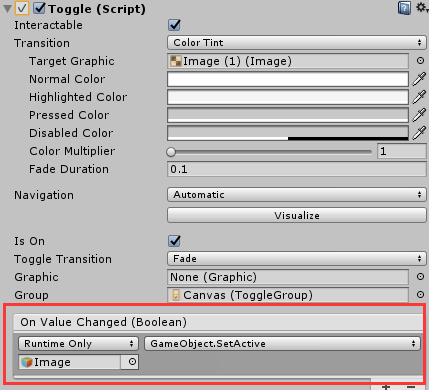
(2)索引器可以被重载，属性不可以

(3)索引器不能声明为static，属性可以

# 3.24

## 利用Toggle制作选项卡

设置一个Toggle组，将需要显示的物体拖放到



### 分页滑动，关卡选择

实现OnDragEndHandler，获取结束时Scroll的位置，计算出每页的位置，然后循环遍历判断当前位置和哪页位置更近，然后设置Scroll的位置即可

可设置Toggle的图标，实现显影

# 3.25

## 任务列表，垂直滑动窗

滑动范围受滑动窗口的大小约束

### UGUI显示排序

可根据摆放位置顺序排序，还可以为图片添加Canvas控件勾选Override Sort属性的值，越大越前

### NGUI显示排序

根据Depth属性的值显示

### 获取当前PopupList的值

如果选择失败，可以先去除空格之后在判断

****

### ITween

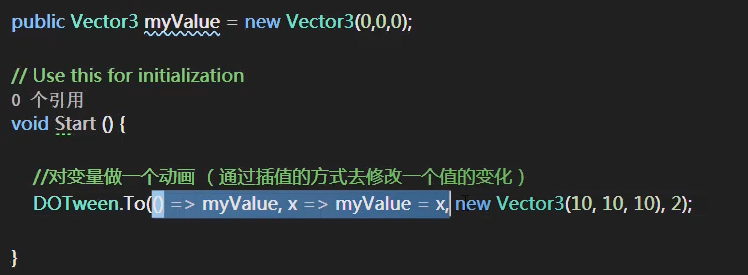
Pixelplacement

****

# 3.26

## ReSharPer代码生成器工具

## DoTween

****

### 按钮控制往返动画播放

****

### 从目标位置移动到当前位置

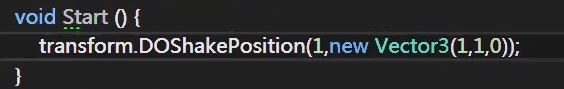
****

如果值为TRUE则从当前位置累加5

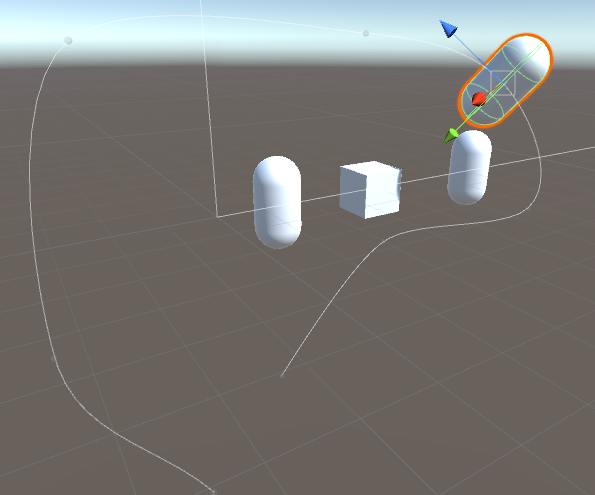
****

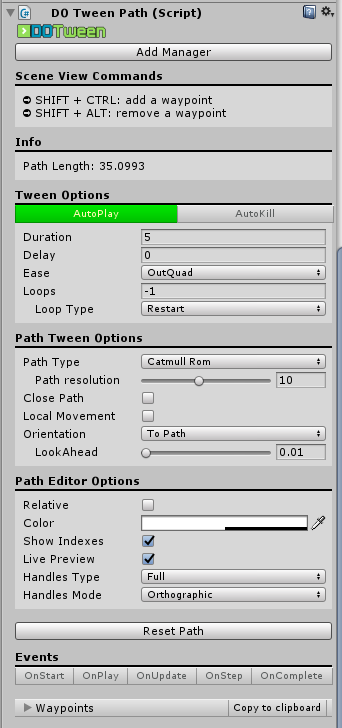
可以做文字打字机的效果，如果有文字则，挨个替换

### 震屏效果

****

### 路径动画

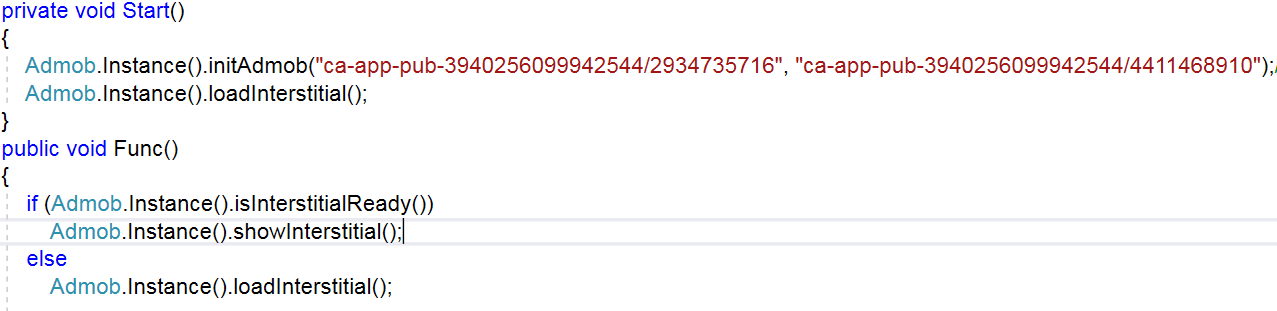
****

****

## Admob谷歌广告接入

谷歌广告官网：<http://www.goole.com/admob/>

插件下载：https://github.com/unity-plugins/Unity-Admob

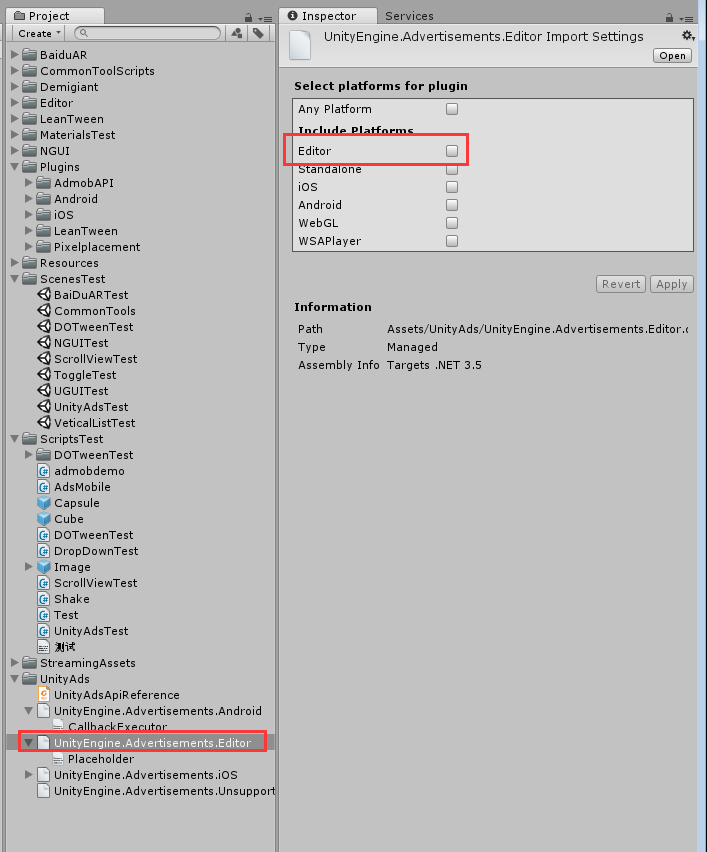


知乎教程：<https://www.zhihu.com/question/27726172/answer/97169740>

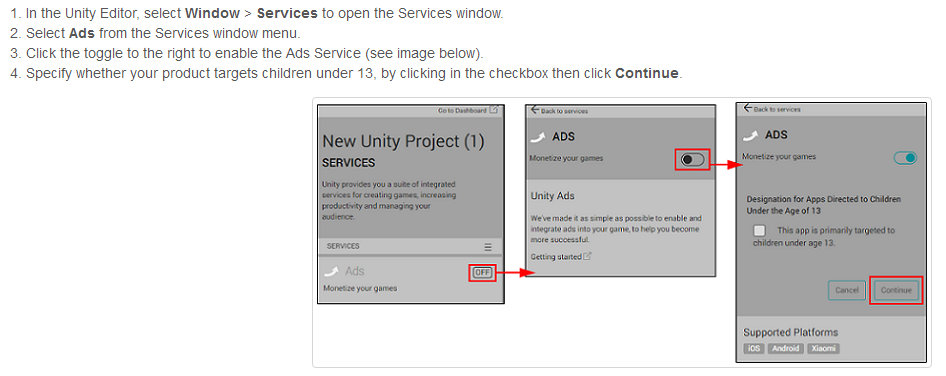
## UnityAds

UnityAds：<https://unity3d.com/cn/unity/features/ads>

如果报错需要取消勾选



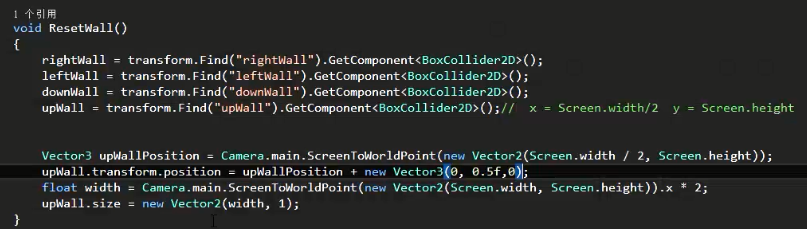
需要打开ADS服务

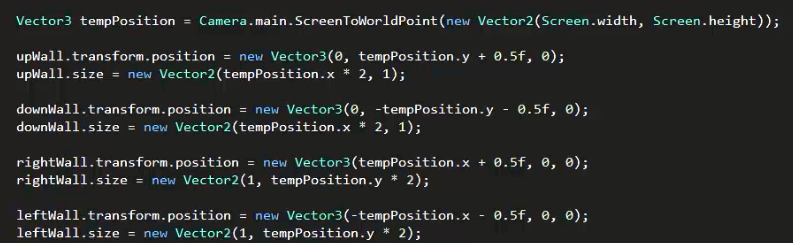


## 翻墙工具

# 3.27

## 根据屏幕的高宽设置位置





## 小球碰撞，乒乓对打游戏

## 方法重载

参数相同，但不可以有不同的返回值

# 3.28

## Timeline

预渲染过场动画

实时渲染过场动画

影视制作

## Cinemachine虚拟相机

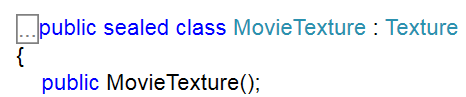
Unity官方直播视频

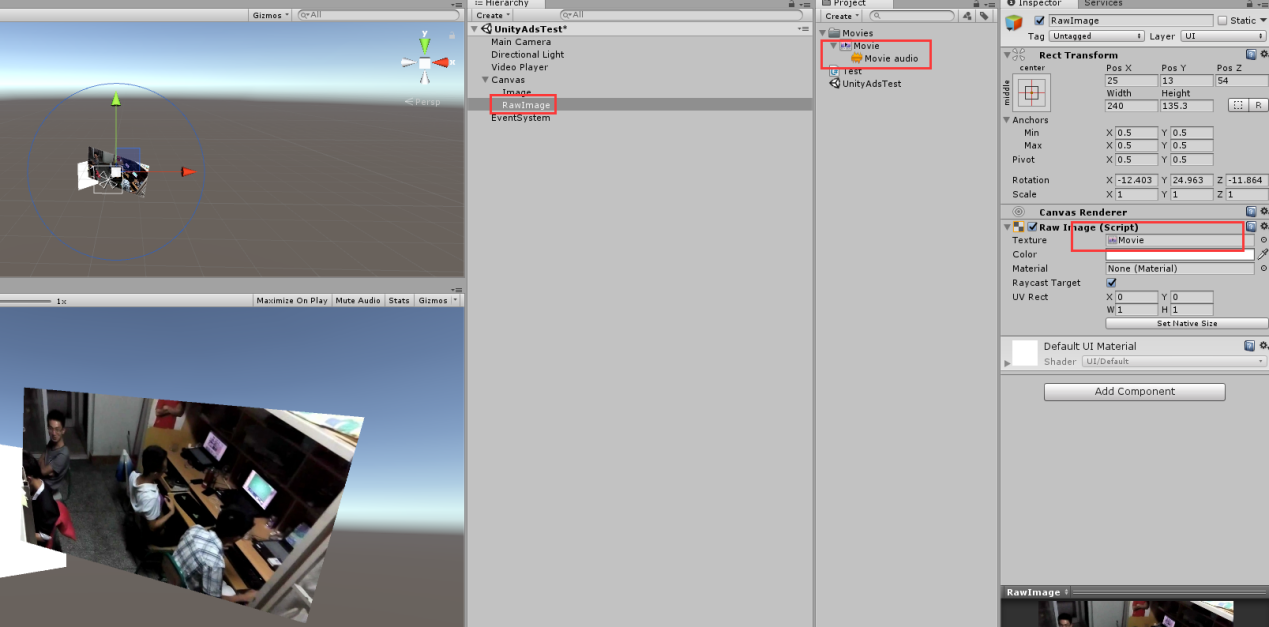
第一部分：https://v.qq.com/x/page/n0527wokwu8.html

第二部分：<https://v.qq.com/x/page/k0527zk2j05.html>

## 视频播放MovieTexture

视频本身继承自Texture，因此可直接将视频拖放到RawImage上，但是声音需要单独添加





# 3.30

## NGUI导入报错，将报错的地方注释掉

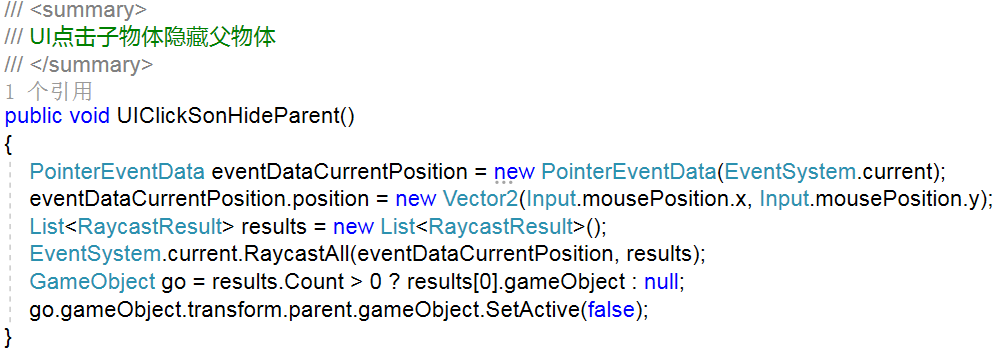
## iTweenEditor升级版

# 3.31

## 方块地图生成插件



## UI点击子物体隐藏父物体

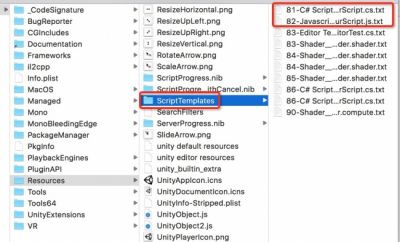


## 更改模板代码

程序员们应该都知道，从项目视图点击右键创建的脚本会自动填充一些代码。其实这些自动填充的代码模板也是可以自定义的。

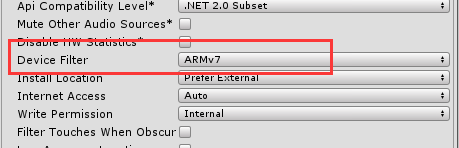
如果使用Mac，找到应用程序中的Unity.app，右键单击显示包内容，找到Resources>ScriptTemplates文件夹，然后选择你要创建模板的脚本类型，复制一份自己命名并编辑后保存，然后退出编辑器之后重新打开，再在项目视图中右键单击创建，就会出现刚刚添加的脚本类型。打开新建的脚本就可以看到自定义的模板代码。

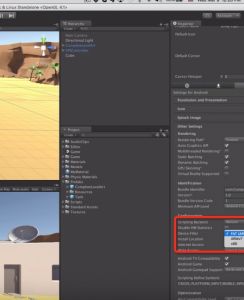
如果使用Windows，则在ProgramFiles(x86)或ProgramFiles目录下的Unity/Data/Resources/ScriptTemplates文件夹下进行同样的操作即可。



## 安卓发布体积减小方法

如果发现Unity打包安卓平台生成的apk体积过大，可以找到PlayerSettings中的安卓平台，默认情况下Device Filter设定为FAT(ARMv7+x86)，表示两种安卓架构平台。如果只需发布到32位或64位平台则单独选择对应的一种即可。这样可以有效减小包体积。

我的手机为



## 物体紧贴地面

如果喜欢物体可以紧贴地面，但倾斜角度不好调整，这时可以在物体上添加Mesh Collider和Rigidbody组件，然后点击运行，借助Unity的物体引擎来计算位置。待物体坠落到地面后复制所有物体，停止运行后删除原先的物体再粘贴运行时复制的内容，并删除所有物体上的Mesh Collider和Rigidbody组件即可。

## 滑动条取值

在float型的字段前添加范围属性声明如[Range(0, 10)]，即可在检视面板中使用滑动条来设置该字段的值。

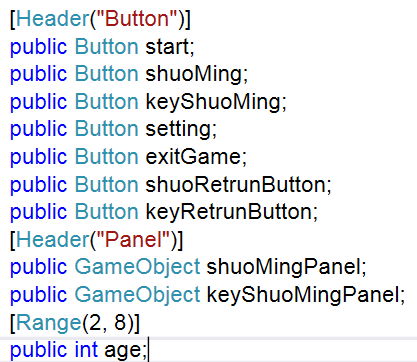




## 字段分组

对一组字段使用属性声明[Header(“XX”)]，可以在检视面板中将字段进行分组。







## 改变运行时界面颜色

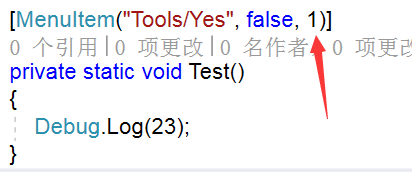
如果常常不小心进入播放模式后编辑场景内容，其实可以依次点击菜单项Edit > Preferences > Colors > Playmode设置编辑器在进入播放模式后的颜色，与正常模式下明显区分开来。

## 改变窗口大小

将鼠标聚焦于编辑器内的某个窗口，按住Shift＋空格键，可以将该窗口最大化至编辑器范围或者还原为原先大小。

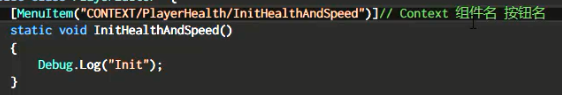
## 编辑器扩展

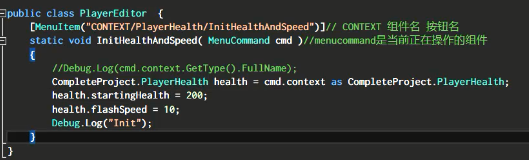
优先级相差大于11则自动添加横线

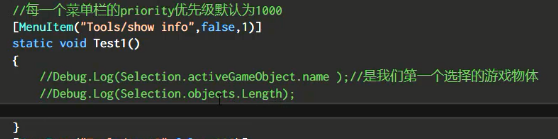


### 给组件添加按钮

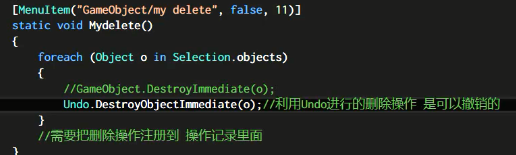
批量修改，初始化组件



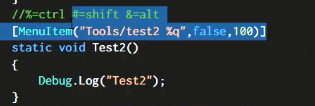
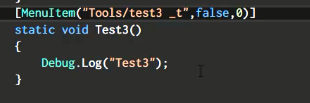




### 编写可以撤销的删除操作



### 添加快捷键



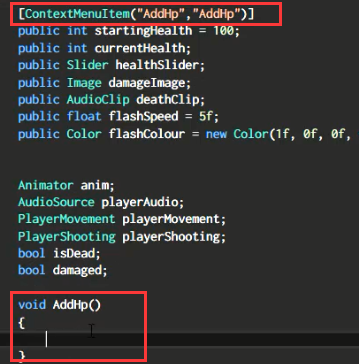
### 判断是否选中决定菜单是否启用



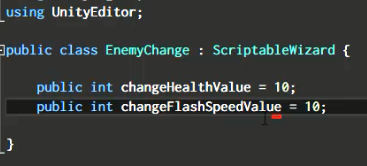
# 4.1

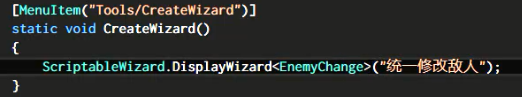
## 给脚本组件字段添加右键菜单

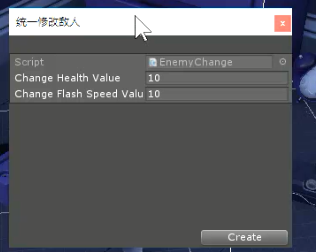
只需在脚本类添加就行



## 弹出对话框



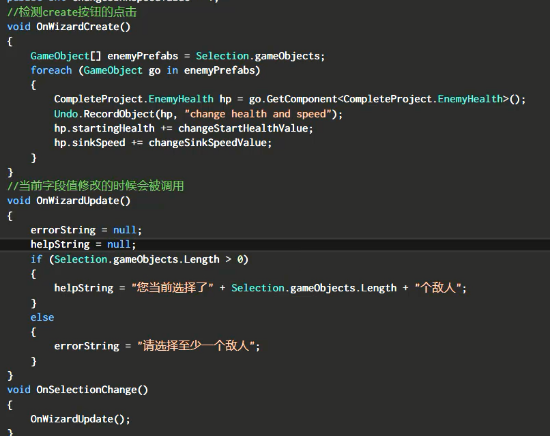




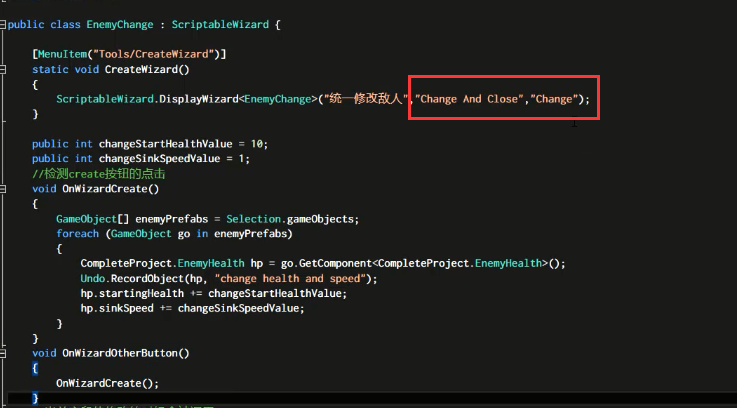
检测按钮点击



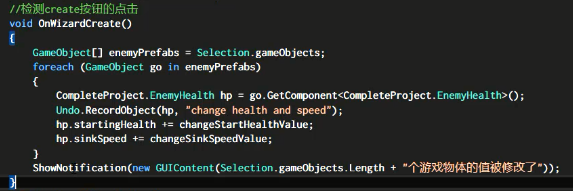
### 选择发生变化



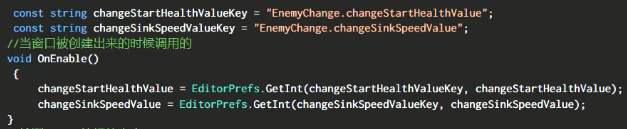
### 改变值不关闭窗口



### 显示提示信息



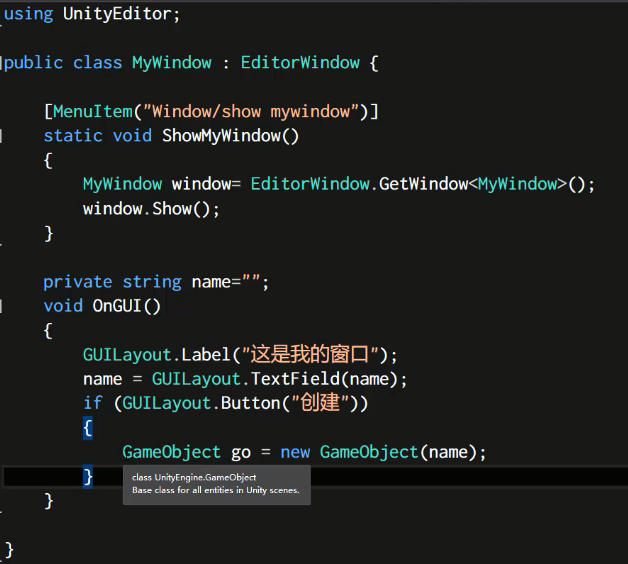




### 显示进度条

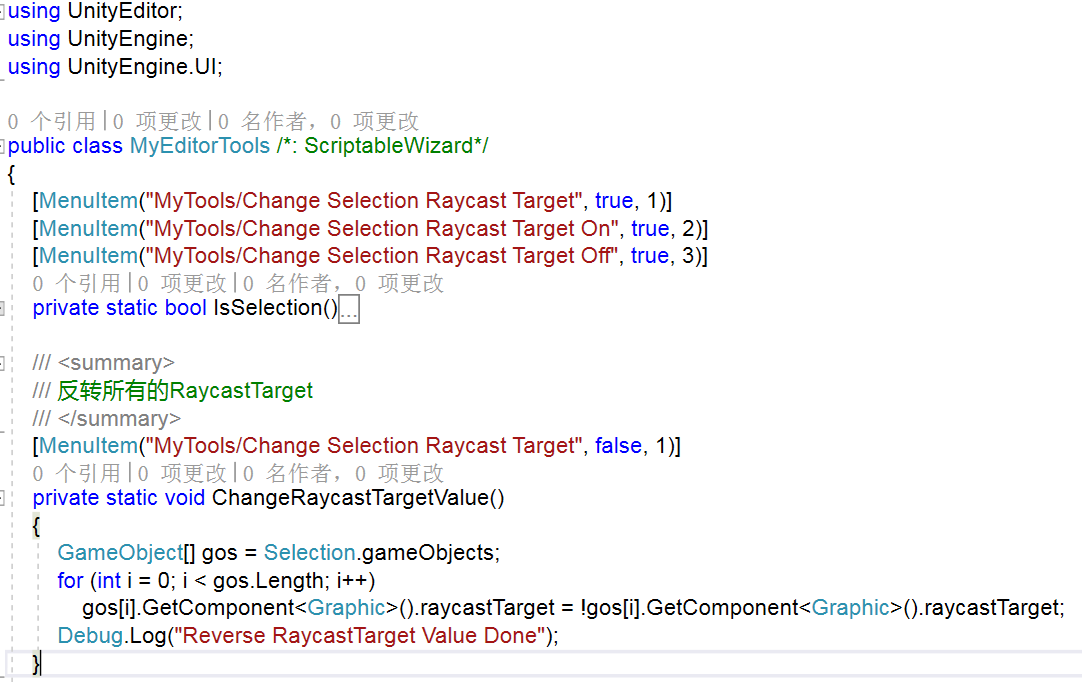


## 创建自定义窗口



## 对选中的RaycastTarget进行操作

Editor，Mytools，递归，自定义工具菜单

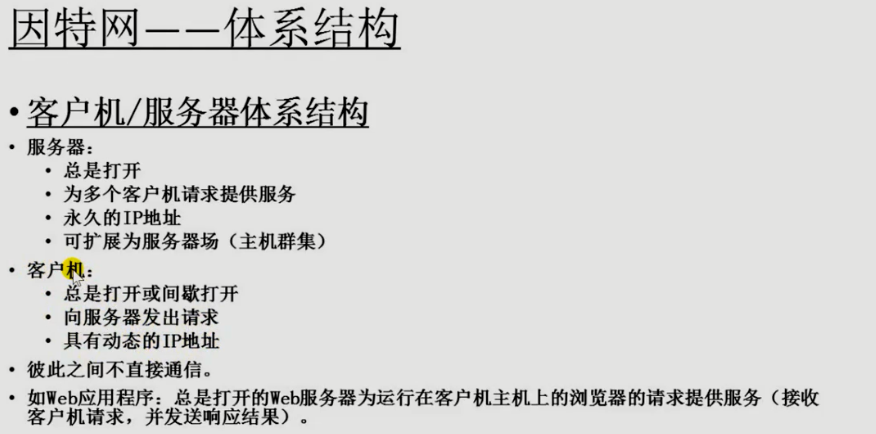


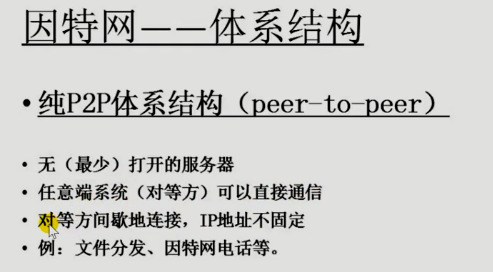


## 查找周边物体，获取物体身上的Transform组件

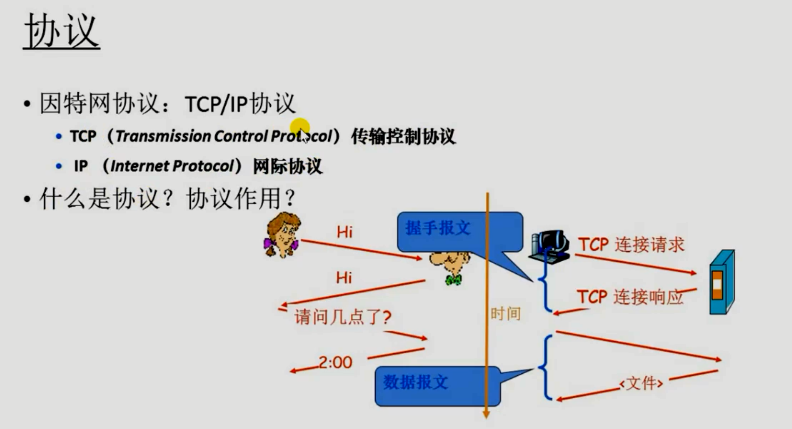


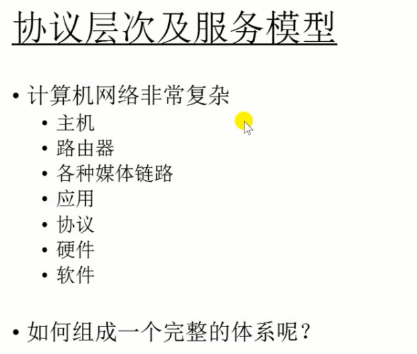
## 因特网

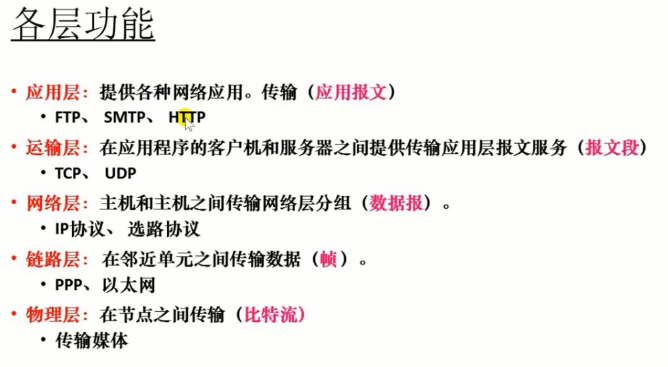




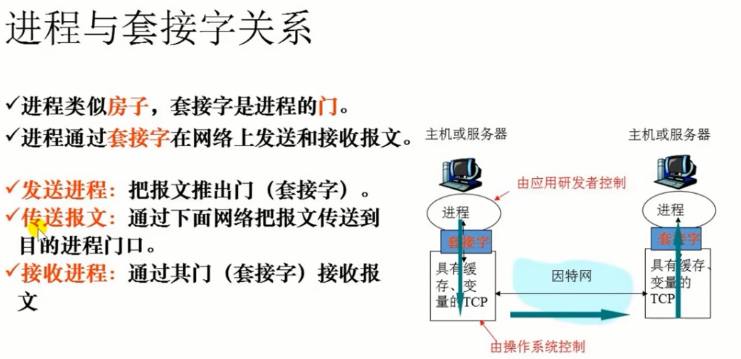
协议

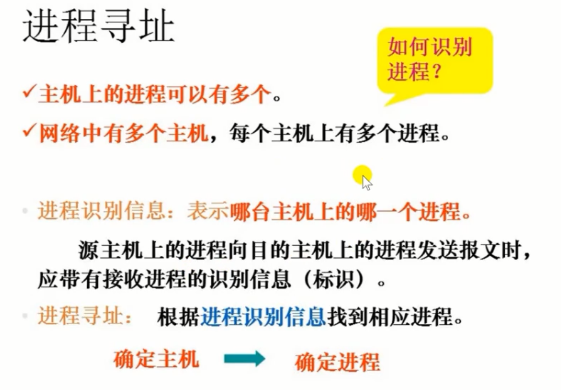


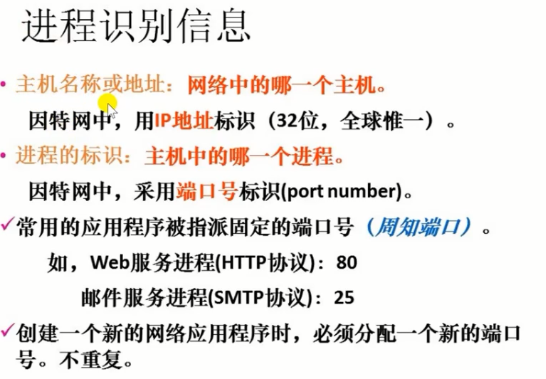


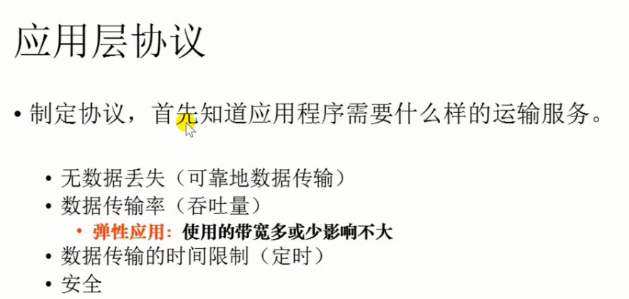


## 进程与套接字关系

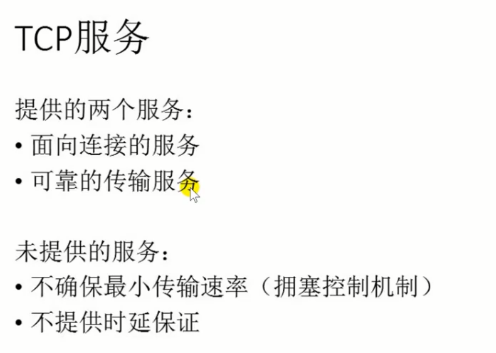


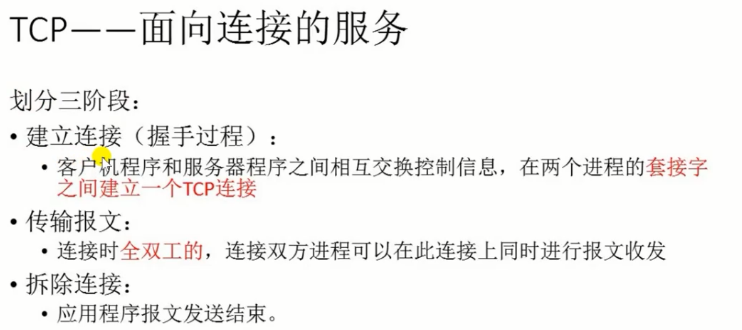


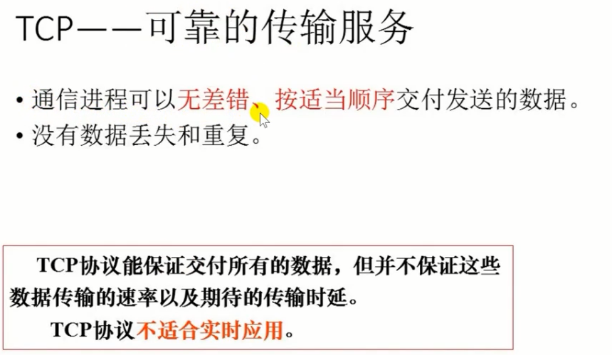




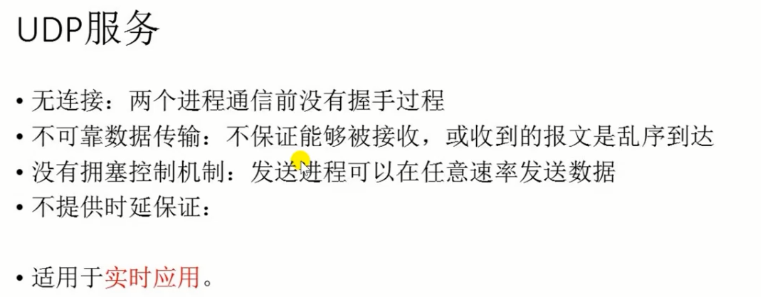
## TCP服务







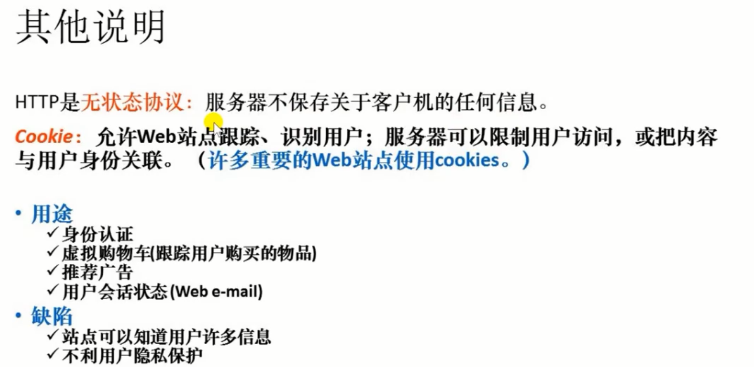
## UDP服务



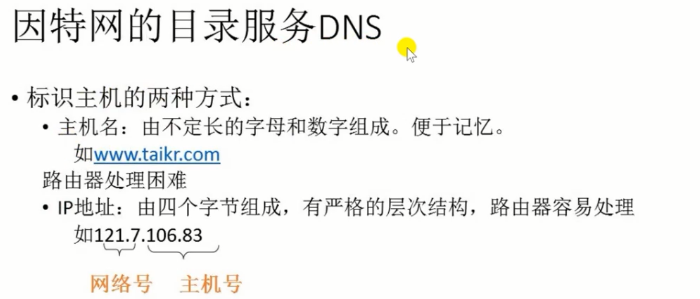
## Web应用和HTTP协议

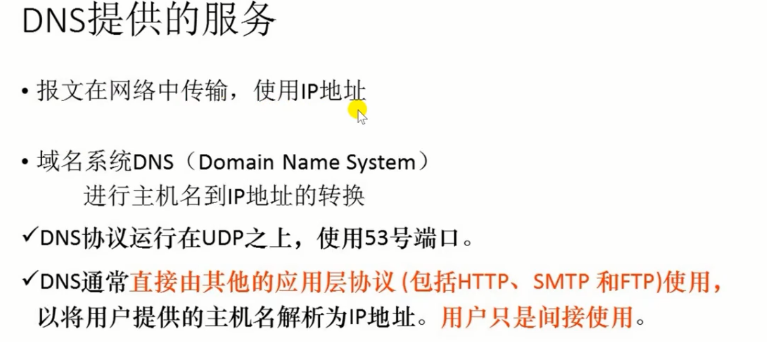
### URL

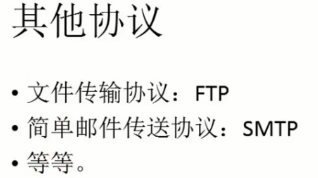




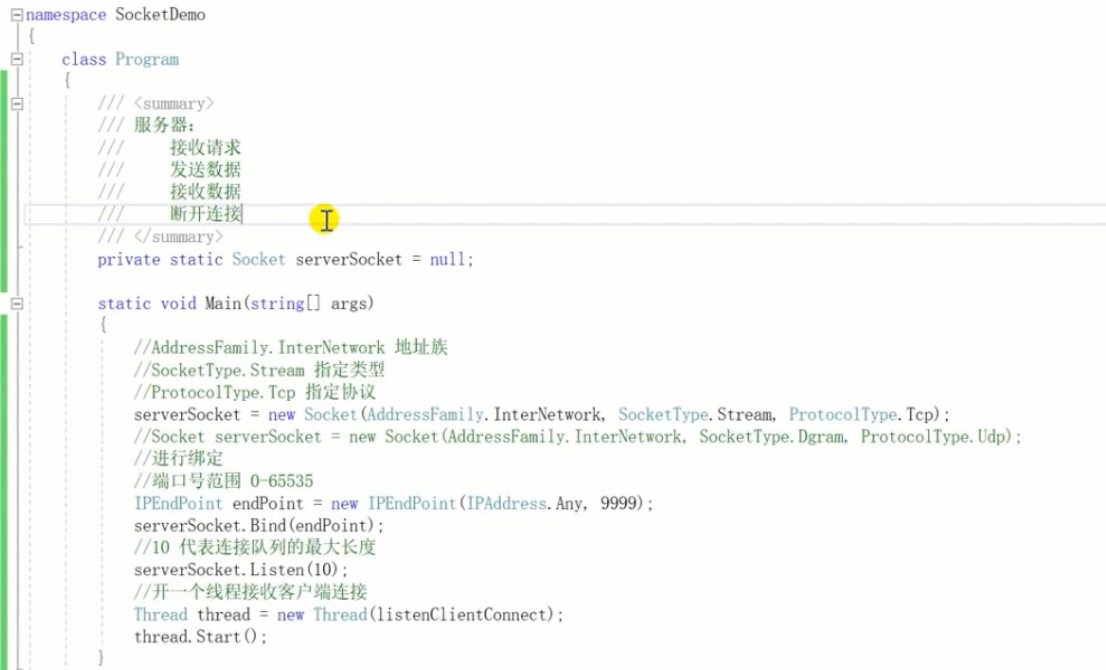
## DNS

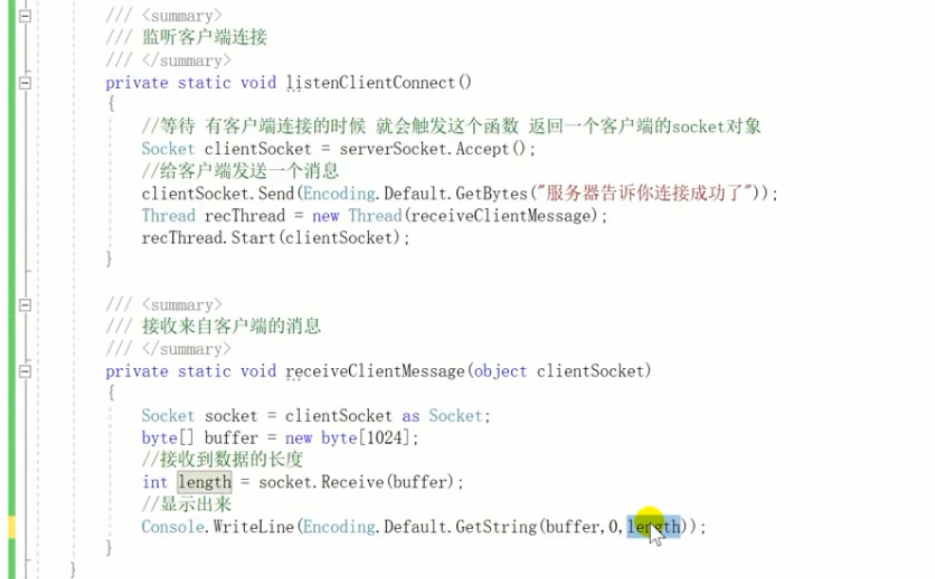






## SocketDemo





# 4.2

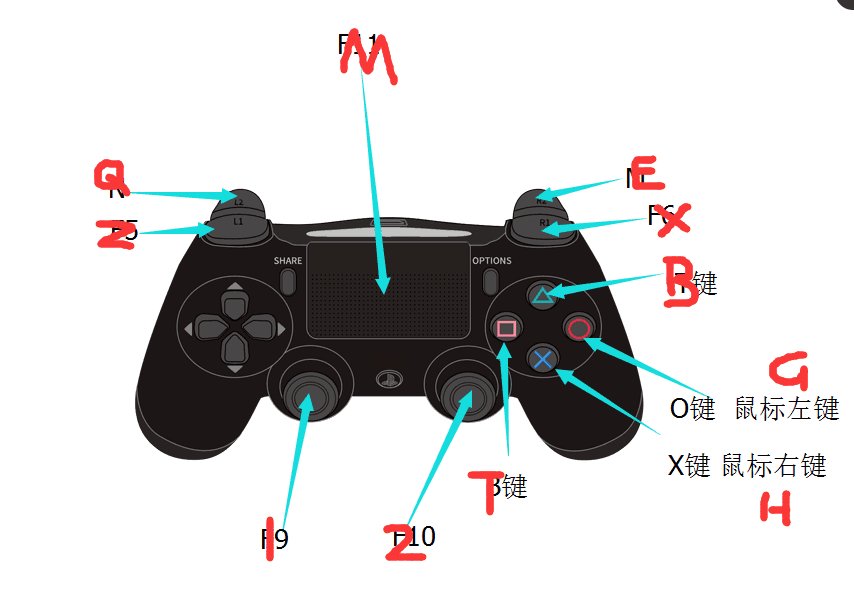
## 账号和密码

http://192.168.0.129   redmine帐号:lizijun   密码：12345678

修改密码：YouLe1213  
 http://192.168.0.129:70/   GIT帐号:lizijun 密码:123456

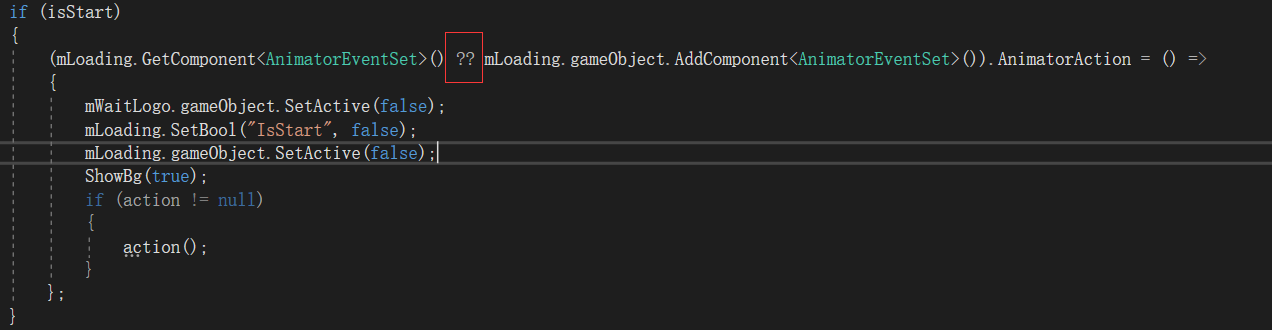
http://192.168.0.163:7373  帐号: lizijun 密码:123456

## 按键示意图



## ？？什么意思

如果此运算符的左操作数不为null，则此运算符将返回左操作数，否则返回右操作数。



# 4.3

## 添加ESC退出鼠标旋转功能



### 代码

private bool isESC = false;

private void Update()

{

#if UNITY\_EDITOR

if (Input.GetKeyDown(KeyCode.Escape))

isESC = true;

if (Input.GetMouseButtonDown(0))

isESC = false;

if (isESC == false)

{

if (axes == RotationAxes.MouseXAndY)

{

float rotationX = transform.localEulerAngles.y + Input.GetAxis("Mouse X") \* sensitivityX;

rotationY += Input.GetAxis("Mouse Y") \* sensitivityY;

rotationY = Mathf.Clamp(rotationY, minimumY, maximumY);

transform.localEulerAngles = new Vector3(-rotationY, rotationX, 0);

}

else if (axes == RotationAxes.MouseX)

{

transform.Rotate(0, Input.GetAxis("Mouse X") \* sensitivityX, 0);

}

else

{

rotationY += Input.GetAxis("Mouse Y") \* sensitivityY;

rotationY = Mathf.Clamp(rotationY, minimumY, maximumY);

transform.localEulerAngles = new Vector3(-rotationY, transform.localEulerAngles.y, 0);

}

mCameraUITrans.localEulerAngles = transform.localEulerAngles;

SetMouseToAnyOfScreenPosition();

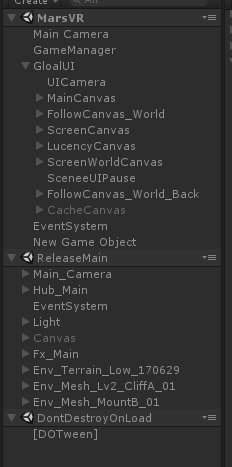
}

#endif

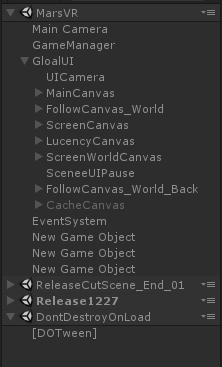
}

## 场景

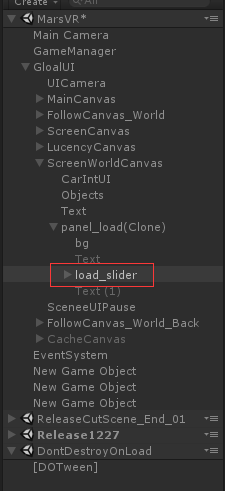
### 启动场景



### 游戏场景



### 场景加载进度条



## C# 堆栈（Stack）

堆栈（Stack）代表了一个**后进先出**的对象集合。当您需要对各项进行后进先出的访问时，则使用堆栈。当您在列表中添加一项，称为**推入**元素，当您从列表中移除一项时，称为**弹出**元素。

### Stack 类的方法和属性

下表列出了 **Stack** 类的一些常用的 **属性**：

|  |  |
| --- | --- |
| **属性** | **描述** |
| Count | 获取 Stack 中包含的元素个数。 |

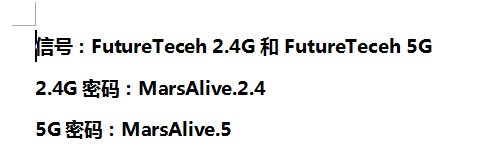
下表列出了 **Stack** 类的一些常用的 **方法**：

|  |  |
| --- | --- |
| **序号** | **方法名 & 描述** |
| 1 | **public virtual void Clear();** 从 Stack 中移除所有的元素。 |
| 2 | **public virtual bool Contains( object obj );** 判断某个元素是否在 Stack 中。 |
| 3 | **public virtual object Peek();** 返回在 Stack 的顶部的对象，但不移除它。 |
| 4 | **public virtual object Pop();** 移除并返回在 Stack 的顶部的对象。 |
| 5 | **public virtual void Push( object obj );** 向 Stack 的顶部添加一个对象。 |
| 6 | **public virtual object[] ToArray();** 复制 Stack 到一个新的数组中。 |

问题

电梯蠕动、不会自动关门

## Wifi密码



# 4.4

## DOTween声音渐变

Tween tween = DOTween.To(() => source.volume = minVolume, value => source.volume = value, maxVolume, time);

## 静态构造函数

初始化静态字段

静态构造函数只调用一次

不能有访问修饰符

不带参数

只能访问静态字段

自动调用，New，使用静态字段时自动调用

## 析构函数

释放资源，一般不用写

对象由CLR公共语言运行池释放

~Person（）

{ }

## Const 与Readonly区别

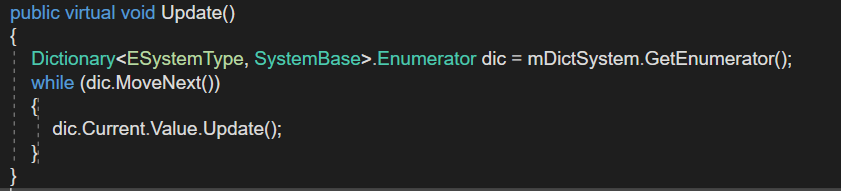
Const申明时必须赋值，默认为static

Readonly申明时不赋值则只能在构造函数中赋值，只能修饰类成员

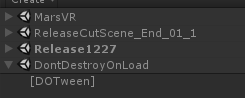
## 单例模式

只有一个类的实例，全局访问

## 迭代器遍历字典



## 界面



## 问题

结束界面语言错乱



控制中心，移除不到位



## 疑问

怎么摄像机看见物体了，并显示提示



怎么做到随任务的位置而转向



怎么检测进入腔体

## UI标签

