AsyncOperation 异步加载场景的

关于AsyncOperation：     AsyncOperation的progress（0-1）属性在isDone为false时，最大加载到0.9就会暂停，直到isDone为true时才会继续加载0.9-1.0的这10%，而isDone为true或fasle的标志为是： 当allowSceneActivation = false，isDone为false ，allowSceneActivation = false 的作用是让场景不会在加载完成后自动跳转到下一个场景，   当allowSceneActivation = true，isDone才可以为true，直到progress = 1.0时 isDone = true，这点官方API也有提到

————————————————

版权声明：本文为CSDN博主「倾恋你的美」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/qq_42462109/article/details/83096135>

**Build Report Tool 插件使用教程**

<http://www.narkii.com/club/thread-411482-1.html>

一款辅助开发的插件，能显示log，资源管理。。Build Report Tool(BRT)工具提供了一个很好的前端用于查看Unity编译信息。BRT显示了编译时包括的每个资源占用的存储空间。DLL列表：查看有什么Mono DLLs被编译到项目中，并且显示每个DLL占用的空间。

未使用资产列表：或许你想看有什么文件你不在使用了？使用未使用资产列表来查看有什么资源在编译时没有使用。保存到XML：保存编译报告到XML文件。可以存储多个编译报告提供参考或者发送给你的团队其他人。项目设置：查看编译时项目的项目设置。当自动编译时可以检查设置。适用于个人版和移动版。

Reporter 一款unity中移动端查看资源情况的插件

BaseController 总控制器manager

BaseModel

BaseView 总图像manager

GameManager 版本更新，网络初始化，模块初始化等等

AssetsUpdate 网络资源更新加载等

SetTimeout 有点像日期控制

SoundProcess 音效管理器

SixqinSDKManager sdk管理器

SQGPSLoader GPS定位

SQToolHelper ip地址相关，域名

ProtoIdMap 协议

NetProcess 网络消息应用

ConfigManager 配置文件manager

MyGrid 有关列表格

GameFieldController 游戏下级界面（游戏级别选择）

GameUtils 游戏里面的杂功能（像逻辑功能就可以使用静态）

继承自BaseController的子类都要构造自己，传入参数name和path。

能否创建一个类统一管理每个Controller的构造参数值。在每个的controller中声明static变量，在一个类里定义。这个类一定要在前面执行。

------------------------------------------------------------

Unity新的网络请求方式UnityWebRequest

UnityWebRequests处理与Web服务器的HTTP通信流。 其他对象 - 特别是DownloadHandler和UploadHandler - 分别管理下载和上传数据。

OnApplicationPause 游戏进入后台

= () =>{} Lambda表达式

Lambda表达式有以下形式：

（参数） => 表达式或语句块

使用：一般多用于委托中，包括按钮的委托（按钮的委托有两种形式）

Delegate(){};

()=>{};

Screen.sleepTimeout=SleepTimeOut.NeverSleep 禁止屏幕锁屏

Unity\_帧率设置Application.targetFrameRate

scrollRect.onValueChanged. ScrollRect滚动时会产生回调

activeHierarchy可以理解为场景中的可见状态。如果某个对象在场景中不可见，对应的activeHierarchy和activeSelf属性一定为false，而该对象的子对象的activeHierarchy也为false,其activeSelf属性却取决于自身的状态，不依赖于父对象。

原文链接：<https://blog.csdn.net/coffeecato/article/details/70185596>

Stack.peek()和Stack.pop(）的相同点是 获取栈顶的值，不同点 则是 Stack.peek()只是获取栈顶的值，而Stack.pop()是获取栈顶的值然后删除。

String.ToLower() 将string变大小写

int.TryParse(string s,out int i) 的参数： s是要转换的字符串，i 是转换的结果

列表元素移动，其坐标不变。

子类继承父类：继承父类，子类中继承父类的属性算是重新开辟内存空间。

枚举可以用ToString（）将里面的类型转成string类型。

------------------------------------------------------------------------

问题：

在一个统一类里面对应用类进行实例化与将应用类进行单例化，两者有何优缺点

玩法围绕着money还是根据玩法来调整money

在做循环列表时，列表中的元素的坐标在显示面板是灰的，这是不可读取？

答：当在scroll view中的content上使用grid layout group就会造成列表中的元素坐标固定（localposition,position）;

SixQIn代码里面的事件存在事件类型？

子类继承父类，哪些可以改变父类中的属性，哪些先执行后执行？

生命周期可以执行代码。各种触发。逻辑进行处理，在其他地方进行调用（最终会在生命周期中进行调用？）对逻辑方面的类生成实例，在应用类里面调用

在scroll view中Scroll下面的物体坐标会受到scroll长宽的影响，

答：因为Viewport改变了pivot的大小。改变toggle的center和pivot，可以修改物体上面的pivot。

ScrollView 的UI适配一直不对/////

加载后的预设体的recttransfrom的left，top，right，bottom不对。

列表切换，想合并或者修改下，现在这个对屏幕适应效果很差。

动态增加content的长度。

---------------------------------------------------------------





<http://dingxiaowei.cn/>

<https://github.com/login?return_to=%2Fqiankanglai%2FLoopScrollRect>

<https://www.cnblogs.com/madinglin/p/8476830.html>

# [UGUI组件之ScrollRect 组件简单笔记(Scroll View)](https://www.cnblogs.com/madinglin/p/8476830.html)

# <https://blog.csdn.net/SerenaHaven/article/details/78826851>

# 关于Anchor与Point等的讲解

# <https://blog.csdn.net/techfield/article/details/81664061>

# Unity RectTransform 的 anchoredPosition 与 position 属性辨析

# MVC

由model，view，control组成，三大模块可由一个做manager，下分各个具体的功能模块，其中可以穿插其他的设计模式，单例，观察者。主要目的就是解耦。多用delegate委托。可创建一个委托manager，在里面进行委托的一些类型，添加委托，删除委托，委托的执行等功能封装到一起。委托可以当函数参数传递。也可以单独创建委托然后对委托方法进行监听。委托用于不同函数之间调用方法。

构造函数：

构造函数会比awake先执行，并且挂载脚本到game object上时就会被执行一次，运行会有三次执行（两次在awake前最先执行，销毁时会执行一次）<https://blog.csdn.net/z_dmsd/article/details/77433943> Unity中的start()和构造函数的关系

继承：

子类继承基类的公有成员变量，在子类里面直接修改可以改变变量，但要注意脚本执行顺序，在其它类里生成基类实例可以修改变量。

子类继承父类构造顺序：子类静态构造->父类静态构造->父类构造->子类构造

Bounds 包围盒

项目流程：

问题分析->总结问题->技术分析->技术流程->测试。（循环）

技术流程：

框架搭建->界面搭建->玩法逻辑->打包。

存储数据：五种方法

手机存储路径：



Uniyt的JsonUtility.ToJson转list数据时，list的所有数据都要序列化=》*Serializable。*

*其他的就不知道了。*

修改xlsx里面的数据可以判定是否已经干过什么事。

在一个单例里面数据出现两个不同值。在单例里面new了一个，是不是地址被改变了。

\_instance=new xx;被新建了

# UGUI与模型混合显示

这是官方Ugui demo中的方法，就是利用相机，创建Render Texture，实现混排

1：首先在你的project视图中右键点击create Render Texture

2：然后在新建一个摄像机，对准你的模型

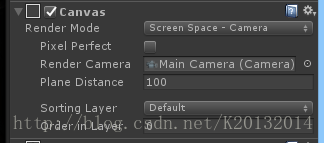
3：给摄像机target texuure赋予Render Texture

此时你可以查看你的render texture 看看是否清晰，如果不符合你的要求，你可以修改render texture

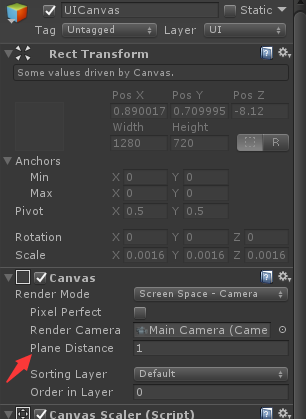


第二种方法实际上是和ngui差不多，我们知道ngui中在UI中有个UICamera对ui进行渲染，所以........

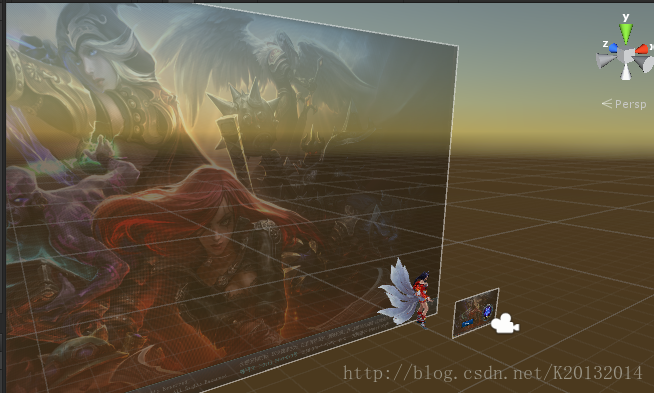
1：首先我们建立两个canvas,一个专门是背景图等等的BgCancas,一个是对UI的UICanvas,然后将两个canvas的渲染模式设置成相机渲染



对于BgCanvas我们的Plane Distance我们可以用设置大一点，然后UIcanvas我们可以设置的小一点，符合你的要求就ok



2：然后将你的模型位置调整最后呈现如下效果



# <https://blog.csdn.net/K20132014/article/details/78117344>

# 球球突围可以打包2018.3.14输入两次密码

# 加入了TopOn插件包,然后各种问题

# Java插件默认实现了两个SourceSet，main和test。每个SourceSet都提供了一系列的属性，通过这些属性，可以定义该SourceSet所包含的源文件。比如，java.srcDirs，resources.srcDirs。Java插件中定义的其他任务，就根据main和test的这两个SourceSet的定义来寻找产品代码和测试代码等。

# Repositories Gradle的代码仓库

# Gradle的学习网址

# <https://www.jianshu.com/p/84ac62747e59>

### allprojects是对所有project的配置，包括Root Project。而subprojects是对所有Child Project的配置

包名冲突：

# Execution failed for task ':app:transformClassesWithMultidexlistForRelease

# Caused by: com.android.build.api.transform.TransformException: Error while generating the main dex list.

# Sdk初始化，广告初始化

# Toutiao 穿山甲

# ps切图软件 像素大厨

# 

# 不同UI间传递数据

# <https://blog.csdn.net/coffeecato/article/details/70185596>

# Unity中GameObject.activeInHierarchy和GameObject.activeSelf的区别

# <https://blog.csdn.net/dengshunhao/article/details/80450937>

# UI适配的很多知识，包括scroll view适配，.UGUI 2D碰撞盒适配

# <https://www.cnblogs.com/Damon-3707/p/11367585.html>

# DoTween的一些Api用法

# <http://www.cocoachina.com/cms/wap.php?action=article&id=34648>

# 同上

# <http://dotween.demigiant.com/documentation.php>

# 官方文档

# <http://www.cocoachina.com/articles/859019?filter=rec>

# 代码质量检测工具

# <https://blog.csdn.net/linshuhe1/article/details/51781749>

# 《Unity 3D游戏客户端基础框架》 protobuf网络框架

public static List<int> AddNum(List<int> num,int addNum)

{

string \_addNum = addNum.ToString();

for (int i = 0; i < \_addNum.*Length*; i++)

{

int \_num = num[i]+int.Parse(\_addNum.*Substring*(i, 0));

int next;

num[i] = AddNum(out next,\_num);

num[i+1] =

}

}

static int AddNum(out int nextNum, int num)

{

nextNum = 0;

if (num/10>0)

{

nextNum = 1;

return num%10;

}

else

{

return num;

}

# }

# **数据增多超出数据类型范围。（未完成）**

# **动画树融合1d，想要控制不同动画，需要把动画设置成循环。**

# **https://indienova.com/indie-game-development/game-programming-for-beginners/**