# 前言

在学习unity3d编程的时候，大家肯定会遇到各种各样的玄学问题，而这些问题的解决也是需要花费大量而又漫长的时间的。这里为了提高开发效率，特编写此文档来记录一些自己遇到的坑，既可以防止自己以后重蹈覆辙，也可以让其他人有所参照，省下大量时间。

# 格式说明&注意事项

其实没啥特别注意的格式，只要说明是什么样的问题，以及解决方法即可。有能力的话可以写一下分析。然后注意一下样式，每一条都使用一个2级标题（如下图）



但是请注意，这里需要写的是一些不是语法上或者简单逻辑上的错误，而是一些通常情况下难以发现问题的，具有代表性的问题。“语句少了分号”、“变量使用之前需要初始化”这些问题还是算了吧

# 正文

## GetComponent函数必须紧跟要改变的属性。

如图所示：





这两种写法是正确的，其中第二张图两句代码尽量连起来，导致多次使用GetComponent函数也没关系。

否则可能会出现这种类型的错误：

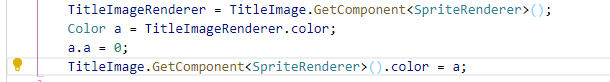
Object reference not set to an instance of an object



但具体原因至今也没能搞清楚，有待进一步探索。

## SpriteRenderer.color的注意事项

这个属性不能直接进行改变，需要这样修改：

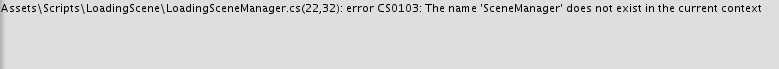


（这里是将color.a的属性，即不透明度，置零）

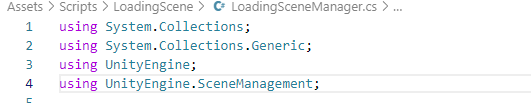
## 函数调用的时候报错的情况：

比方说这种情况：

The name 'SceneManager' does not exist in the current context



首先确认函数名没有错误，然后看文件最上方的调用：



类似于import语句（JavaScript，Java，python）和#include语句（c，c艹），c#的函数调用也需要引入这些包，所以当函数调用出现报错的情况的时候需要注意是不是引入了相应的包。

## 弹幕发射不出去的情况：

发射弹幕的时候有时候会出现弹幕直接停在发射点的情况。

出现这种情况的原因是弹幕和角色都作为2D碰撞器，弹幕的初始位置和主角的体积判定位置重合使得弹幕静止。

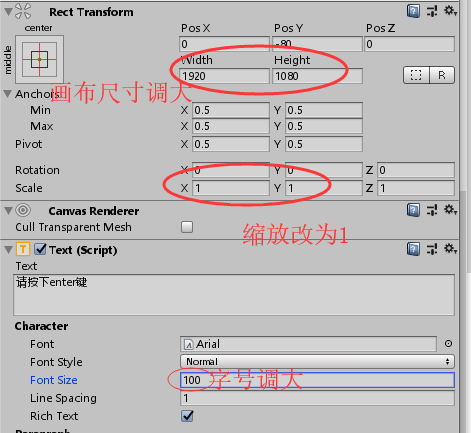
解决方法是将弹幕的初始位置（发射点）修改到与主角的体积判定不重合的位置。

## Git的上传问题：

实现功能完毕准备提交git项目的时候注意unity要出于关闭的状态，否则会报各种各样的神奇错误

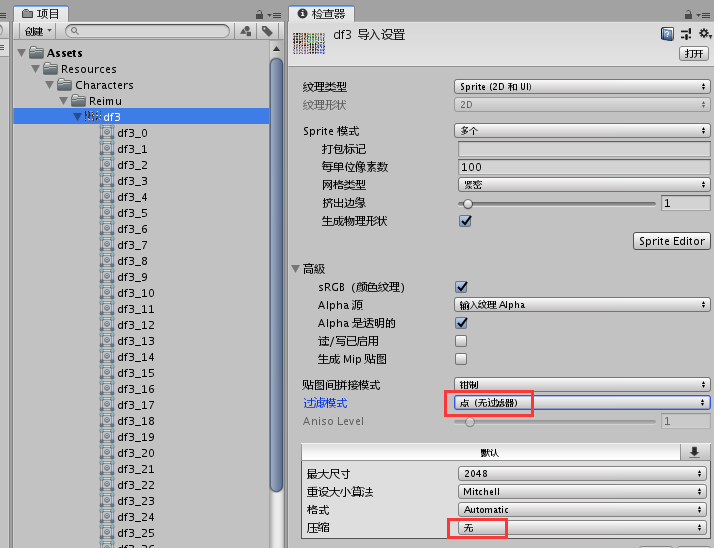
## 避免Text模糊的方法

避免字体模糊的窍门：画布尺寸调大，缩放调小，字号调大

[](https://camo.githubusercontent.com/bd87dc26be7be0e74e988b2ed723e3c8b602ea12/68747470733a2f2f692e6c6f6c692e6e65742f323031392f30392f32332f39694c507a73366b35524e4b7447542e706e67)

## 避免Sprite模糊的方法

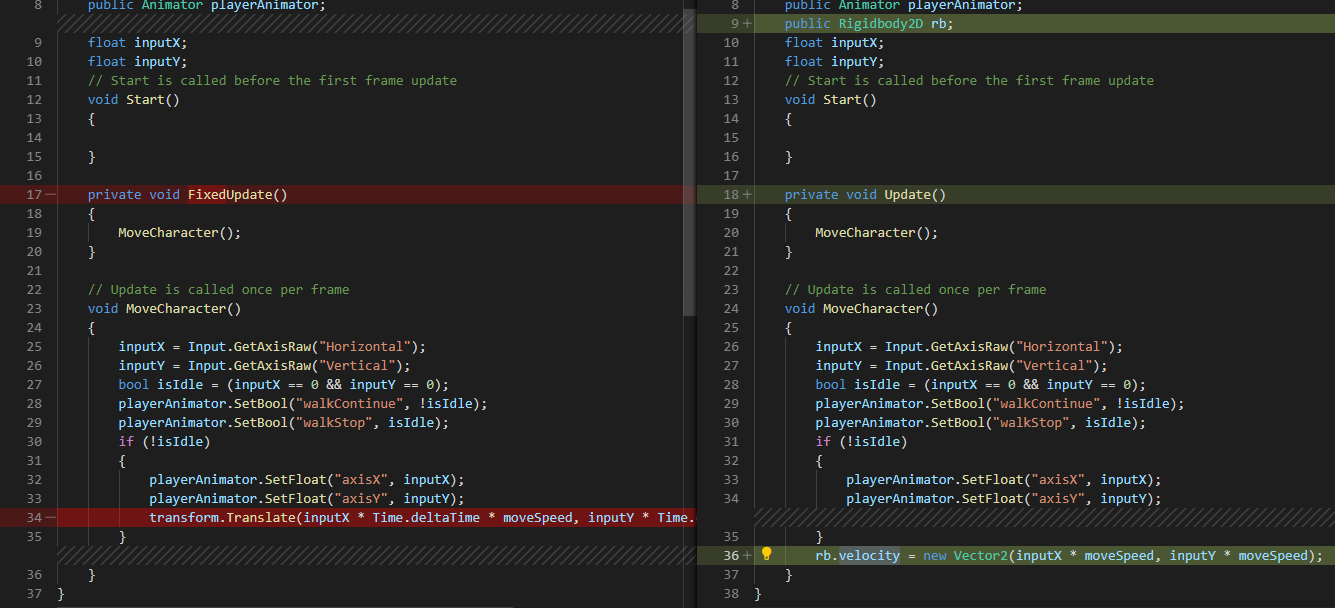
像素风格的sprite在默认情况下放大后会产生模糊，解决方法：



如图，将“过滤模式”设为“点（无过滤器）”；“压缩”设为“无”。

## 让人物移动平滑且碰撞不鬼畜的方法

如图



将移动逻辑写在Update里保证平滑和快速响应。

移动应通过设置2D刚体的速度属性，而不是直接操作transform来实现，这样物理引擎会正确地处理碰撞，不会出现往障碍物里移动时人物鬼畜的现象。

\*2D刚体组件“插值”属性不要设为“外推”，否则依然会鬼畜。