图片包含 日历

描述已自动生成





**原理：***实例化一个已经编写好的Prefab，这个Prefab包含写好的逻辑且有两种模式可以选择：胜利彩虹动画过后直接返回主菜单、胜利彩虹动画过后显示自定义的UI界面；这个文档主要讲述的是如何制作第二种模式，第一种模式在文档最下面有介绍。*

**为什么要用Prefab实现胜利：***第一原因是，懒；第二原因是，不同关卡的胜利界面可能不一样，但这些界面一定都会有彩虹动画和返回主菜单、重玩的按钮。*

**为什么要用到GameManager：***为了使该功能能够在任何场景以一条代码的方式实现，必须用到被设为DontDestroyOnLoad的GameManager。*

**教程速览：***创建PRB\_Template\_MissionWin的子Prefab，设置好子Prefab中的参数和UI部分，将其加入GameManager。*



第一步：准备好自己的胜利界面所需要的美术资源。

将美术资源导入至自己的关卡资源目录下，如图二。

*如果不需要透明效果，建议使用jpg而不是png格式保存*

电脑萤幕画面

描述已自动生成

第二步：创建PRB\_Temple\_MissionWin的子Prefab并将其放入自己的关卡资源目录内

PRB\_Temple\_MissionWin Prefab位于Assets/CommonAsset/Prefabs文件夹内，如图一是创建其子Prefab，图二是将其移动到自己的关卡资源目录内；这里为了便于测试，使用的是SE\_TestScene关卡。

*子Prefab的图标与Prefab不同*

图形用户界面, 应用程序

描述已自动生成

图形用户界面, 网站

描述已自动生成

第三步：编辑子Prefab下的胜利UI

双击子Prefab进行编辑。

Hierarchy下的PaypageCanvas是我们需要编辑的部分，右键PaypageCanvas，依次点击“UI-Image”以创建图片UI控件(控件是指包含了Image组件的GameObject)，将该控件重命名为Background以便于区分(图一)，然后为其设置第一步导入进来的美术资源。(图二)

设置好后，视口区域不会发生任何变化，因为此时Image实在是太小了，所以要给它放大。

选中刚刚的Background，然后按照图三三步设置。

设置好后，图片就会填充屏幕框了。

*缩放放大会使UI变模糊，请尽量使用RectTransform组件提供的Left、Top、Right、Bottom属性进行放大*

截图里有图片

描述已自动生成图形用户界面

描述已自动生成

图形用户界面

描述已自动生成

第四步：添加返回主菜单按钮功能

右键PaypageCanvas，依次点击“UI-Button”以创建按钮UI控件。

删除Button下的Text，因为我们想要的是点击任意处返回主菜单，不需要按钮上的文本。

按照图一设置Button属性，以铺满全屏；按照图二设置Button下Image组件的Color属性以使按钮透明。

设置后，在Button最下方的OnClick()处点击“+”以创建点击事件，然后按照图三将Prefab根组件拖入“None”处，然后按照图四设置点击事件。

*CONTROLL\_CompleteLevelAndLeave函数的作用是解锁下一关并返回主菜单*

图形用户界面

描述已自动生成图形用户界面, 应用程序

描述已自动生成

图片包含 图形用户界面

描述已自动生成

图形用户界面

描述已自动生成

第五步：在SE\_TestScene测试其可行性

将做好的Prefab放进关卡里，然后把该Prefab的XYZ坐标归零*(重要)*。

开始游戏，当彩虹动画播放完成后，点击任意处返回主菜单。

截图里有图片

描述已自动生成

图形用户界面

中度可信度描述已自动生成

第六步***(u did it!)***：将该Prefab设置到GameManager中以实现“一码调用”

打开SE\_LoginPage关卡，点击Hierarchy中的GameManager，把做好的Prefab拖拽赋值MissionWinPrefab，如图所示。

从此以后，使用一串代码即可使游戏胜利。

GameManager.Instance.DoCelebrate();

*图形用户界面, 文本, 应用程序

描述已自动生成*

等等…如果我不想要自定义UI，我只想要彩虹效果过后就直接回到主菜单，应该怎么办？

第二步结束后，双击编辑子Prefab，将Inspector内的“NoPaypage”勾选，然后从第五步开始继续做即可。