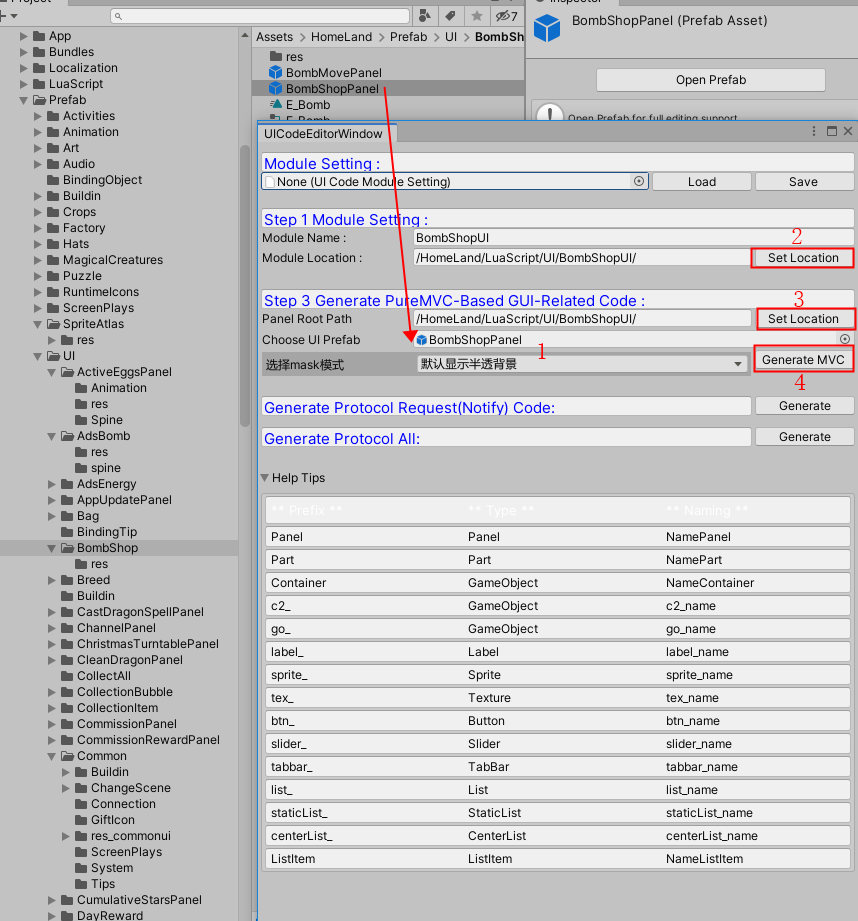
一：根据功能名在Assets\HomeLand\Prefab\UI\ 下创建一个文件夹

如 Assets\HomeLand\Prefab\UI\BombShop

二：制作UI面板预设，以 Panel结尾，BombShopPanelPanel制作完毕后打开Unity 菜单栏 Build Tools-> Code Editor

弹出如下窗口



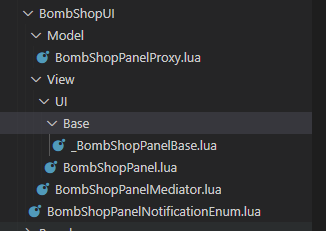
(2.1)在 /HomeLand/LuaScript/UI/ 目录下创建一个文件夹 BombShopUI，用来存放生成的 lua 代码

(2.1)将BombShopPanel 拖拽至 Choose UI Prefab

(2.2)在 Module Location 点击 Set Location 选择生成的 lua 脚本目录/HomeLand/LuaScript/UI/BombShopUI

(2.3)在 Panel Root Path 点击 Set Location 选择生成的 lua 脚本目录/HomeLand/LuaScript/UI/BombShopUI

(2.4) 点击 Generate MVC 按钮，lua 代码将生成到 /HomeLand/LuaScript/UI/BombShopUI 目录



Assets\HomeLand\LuaScript\UI\PanelRegisters 目录下 生成 PanelRegister\_TestUIMakePanel.lua

内定义了 GlobalPanelEnum.TestUIMakePanel

PanelManager.showPanel 打开界面

PanelManager.closePanel 关闭界面

三：生成的 lua 代码作用

(3.1) \_BombShopPanelBase.lua 继承自 Base Panel

当前面板基类，获取UI控件、注册监听等

\_BombShopPanelBase.bindView()

预设加载结束，绑定Button、Text、等UI组件，

        local function OnClick\_btn\_close(go)

            sendNotification(TestUIMakePanelNotificationEnum.Click\_btn\_close)

            local a = 0

        end

    --insertDeclareBtn

        Util.UGUI\_AddButtonListener(self.btn\_close.gameObject, OnClick\_btn\_close)

这么着添加按钮监听，需要在 TestUIMakePanelMediator:listNotificationInterests()

方法中添加

function TestUIMakePanelMediator:listNotificationInterests()

    return

    {

        --insertNotificationNames

        TestUIMakePanelNotificationEnum.Click\_btn\_close

    }

end

最终在TestUIMakePanelMediator:handleNotification(notification)方法中处理，需要每个单独判断

function TestUIMakePanelMediator:handleNotification(notification)

    local name = notification:getName()

    -- local type = notification:getType() -- uncomment if need by yourself

    -- local body = notification:getBody() --message data  uncomment if need by yourself

    --insertHandleNotificationNames

    if(name ==TestUIMakePanelNotificationEnum.Click\_btn\_close) then

        local a = 0

    end

end

TestUIMakePanelNotificationEnum.lua 也是自动生成的

TestUIMakePanelNotificationEnum =

{

    Click\_btn\_close = "TestUIMakePanelNotificationEnum\_Click\_btn\_close"

}

(3.2) BombShopPanel.lua 继承自 BombShopPanelBase

UI 逻辑

(3.3)BombShopPanelMediator.lua

面板打开、关闭、动画播放完成等等事件的通知

(3.4)BombShopPanelProxy.lua 目前没怎么使用

四：BasePanel.lua

所有面板超类

五：BaseMediator.lua

界面打开、关闭、销毁 控制中心

六：PanelCallbacks.lua

七：PanelManager.lua

PanelManager.closePanel(GlobalPanelEnum.DragonAttrPanel)

PanelManager.showPanel(GlobalPanelEnum.CommonRewardPanel,{

rewards = rewards

}, PanelCallbacks:Create(finishCallback))