sign off 咪啪

对于临时UI需要完成的任务以及对应的接口做一个简单说明

临时UI现在需要相应的操作几乎只有一个：按！按钮！

因为所有的动作都要分两步“作用者”，“被作用者”，所以

第一步：能够做到响应“双按钮事件”

我的意思是：如果我“按了一个按钮”“再按了一个按钮”，程序就能够明白是什么和什么发生了动作，并且把动作的执行者和被执行者给搞清楚。

当然，在你按下第一个按钮时，按钮上的文字应该加粗或者添加文字，强调你知道这个按钮已经被按下了。

另外，如果一个按钮被连续按了两次，代表动作取消，不应该显示任何信息，而且取消原来强调选定的状态。（如果你按了一个再按另一个，动作被执行结束了，也应该取消原来强调选定的状态！）

第二步：能够做到更新游戏信息。

在Main\_Game->GameBase\_UserInfo类里面存储了所有的卡牌信息和随从信息。你应该能够做到将这些信息显示到UI里面。

（建议：使用一个单独的成员函数，例如update，代表从Main\_Game里面读取游戏信息）

在GameBase\_UserInfo->GameBase\_Card/GameBase\_Unit里面存储了足够的信息，而且应该也十分明确。如果不清楚，可以向我询问。

第三步：能够做到将双按钮事件与游戏事件一一对应起来。

双按钮事件分为3种：点击了一个卡牌，再点击一个空的随从位，代表召唤一个随从；点击一个随从，再点击一个随从，代表随从攻击另外一个随从；点击一个卡牌，再点击一个随从，代表使用法术攻击随从。

不过这次暂时只要求完成“随从攻击随从”的操作。（如果能做出这个，以后就能参考着做出其它）

在写代码之前，你需要熟悉Trigger\_Set\_InputInfo类，因为每当游戏信息从UI系统传入Main\_Game系统时，你将需要写一个这个类，以负责将游戏动作如实地传入Trigger中。然后还需要参考Trigger\_Begin\_Base类的头文件，以了解这个类是怎么使用的。（而你并不一定需要了解具体实现细节）

然后你的具体动作细节则应当是这样的：

1.选择合适的Trigger\_Begin类，并且做一个这个的代理类(Trigger\_Begin\_Base\_Proxy)

2.根据要求写好Trigger\_Set\_InputInfo类

3.将这两个类提交到

Main\_Game::execTrigger(Trigger\_Begin\_Base\_Proxy,Trigger\_Set\_InputInfo)

这个函数。

这样就算是完成了。