GameManager:

Enum GameState：游戏状态

{

Dialog, // when doing dialog(enemy don't attack)

Battle, // when gamer is on battle

RPG, // when gamer is exploring the world

BackPack, // check backpack etc.

Victory, // when gamer has won

Die // when gamer is dead

}

Enum BattleState：战斗状态

{

Battle：战斗中，（不能点击背包？）

Peace：开放模式，可进入战斗状态

Forbid：禁止，对话和查看背包时，暂停/禁止进入战斗状态，史莱姆不能锁定目标 &攻击

}

Enum MoveState：人物移动状态

{

YES，允许移动

NO 不允许移动

}

UpdateState（newState）：更改状态，每个状态的Manager中调用

Event<GameState>OnStateChanged : 游戏状态被改变时，注册了该事件的类会调用对应 函数

DialogManager：

？如何触发对话，对应位置调用UpdateState改为Dialog

UpdateState中调用HandleDialog（）：改变BattleState为Forbid，MoveState为NO

Dialog类：有一个函数封装F，调用即执行完整的一段对话（显示对话框，鼠标点击显示下一句，取消对话框）

A中调用F执行该段对话，然后调用UpdateState（RPG）进入开放模式

BattleManager：

？如何判断进入战斗状态（史莱姆ai？）对应位置调用UpdateState改为Battle

UpdateState中调用HandleBattle，改BattleState为Battle，MoveState改为YES

？如何判断战斗结束，对应位置调用UpdateState改为RPG

RPGManager：

UpdateState中调用HandleRPG，将BattleState改为Peace,MoveState为YES

函数A注册OnStateChanged：状态改变为RPG时调用（我也不知道这儿要干啥）

BackPackManager：

背包点击打开事件，调用GameManager.Instance.UpdateState,将GameState改为BackPack

UpdateState中调用HandleBackPack，将BattleState改为Forbid，MoveState改为NO

函数A注册OnStateChanged：状态改变为BackPack时调用

背包Instance中的关闭背包函数调用UpdateState，改为RPG

DieManager：

人物血量扣为0（战斗，遇水...）时通过在人物脚本调用UpdateStateDie，改为Die

函数A注册OnStateChanged：状态改变为Die时调用，将人物血量恢复，移动到复 活点（？复活点怎么确定）

UpdateState改为RPG

MoveState为NO时人物不能移动

BattleState为Forbid时史莱姆不能攻击