场景与层的创建

3个场景

1、加载场景

h:

```cpp
#pragma once
#include "cocos2d.h"

#include "SceneManager.h"

USING_NS_CC;
class LoadLayer:public Layer
{
public:
    CREATE_FUNC(LoadLayer);
    virtual bool init();

    //方便管理
    SceneManager*tsm;

};
```

cpp:

```cpp
#include "LoadLayer.h"

bool LoadLayer::init()
{
    return true;
}
```

2、开始场景

h:

```
HelloWorld
1        #pragma once
2      ⊟#include "cocos2d.h"
3       #include "SceneManager.h"
4       USING_NS_CC;
5      ⊟class OpenLayer :public Layer
6       {
7       public:
8           CREATE_FUNC(OpenLayer);
9           virtual bool init();
10
11          SceneManager*tsm;
12      };
13
```

cpp：

```
1        #include "OpenLayer.h"
2
3      ⊟bool OpenLayer::init()
4       {
5           return true;
6       }
```

3、时钟场景

h:

```
HelloWorld
1        #pragma once
2
3      ⊟#include "cocos2d.h"
4       #include "SceneManager.h"
5       USING_NS_CC;
6      ⊟class ClockLayer :public Layer
7       {
8       public:
9           CREATE_FUNC(ClockLayer);
10          virtual bool init();
11
12          SceneManager*tsm;
13
14      };
```

cpp:

```cpp
#include "ClockLayer.h"

bool ClockLayer::init()
{
    return true;
}
```

场景管理器类

h：

```cpp
#define __SCENEMANAGER_H

#include "cocos2d.h"

using namespace cocos2d;

class SceneManager
{
public:
    /*
    场景一、加载场景
    场景二、开始场景
    场景三、时钟场景
    */
    Scene*loadScene;
    Scene*openScene;
    Scene*clockScene;

    void createLoadScene();
    void goOpenScene();
    void goClockScene();

private:

};

#endif //__SCENEMANAGER_H
```

cpp：

```cpp
#include "SceneManager.h"

 #include "LoadLayer.h"
 #include "OpenLayer.h"
 #include "ClockLayer.h"
void SceneManager::createLoadScene()
{
    loadScene = Scene::create();
    LoadLayer*layer = LoadLayer::create();
    layer->tsm = this;
    //添加图层到该场景下
    loadScene->addChild(layer);
}
void SceneManager::goOpenScene()
{
    openScene = Scene::create();
    OpenLayer *openLayer = OpenLayer::create();
    openLayer->tsm = this;
    //添加图层到该场景下
    openScene->addChild(openLayer);
    //不是第一个场景，不能使用runWithScene
    Director::getInstance()->replaceScene(openScene);
}
void SceneManager::goClockScene()
{
    clockScene = Scene::create();
    ClockLayer*clockLayer = ClockLayer::create();
    clockLayer->tsm = this;
    //添加图层到该场景下
    clockScene->addChild(clockLayer);
    //切换场景
    Director::getInstance()->replaceScene(clockScene);

}
```