概述

2014-2-14

游戏中常用的菜单如图3-17所示，其中菜单项可以是图片、系统字，或者自定义的字体。

CCMenu是一个菜单项的容器，用来装载各种菜单项。代码清单3-23就是一个定义CCMenu类实例的过程，是tests项目中MenuTest.cpp的MenuLayer2的构造函数。

MenuLayer2::MenuLayer2()

{

for( int i=0;i < 2;i++ )

{

CCMenuItemImage\* item1 = CCMenuItemImage::create(s\_PlayNormal, s\_PlaySelect, this, menu\_selector(MenuLayer2::menuCallback));

CCMenuItemImage\* item2 = CCMenuItemImage::create(s\_HighNormal, s\_HighSelect, this, menu\_selector(MenuLayer2::menuCallbackOpacity) );

CCMenuItemImage\* item3 = CCMenuItemImage::create(s\_AboutNormal, s\_AboutSelect, this, menu\_selector(MenuLayer2::menuCallbackAlign) );

item1->setScaleX( 1.5f );

item2->setScaleX( 0.5f );

item3->setScaleX( 0.5f );

CCMenu\* menu = CCMenu::create(item1, item2, item3, NULL);

CCSize s = CCDirector::sharedDirector()->getWinSize();

menu->setPosition(ccp(s.width/2, s.height/2));

menu->setTag( kTagMenu );

addChild(menu, 0, 100+i);

m\_centeredMenu = menu->getPosition();

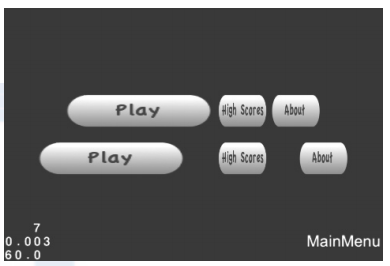
}

m\_alignedH = true;

alignMenusH();

}

首先定义菜单项（关于菜单项，本书会在后面的章节中做讲解），然后用它们定义初始化菜单CCMenu实例，最后将CCMenu实例加入CCLayer中显示出来，效果如图3-18所示。



菜单类还提供了alignItemsVertically和align-ItemsHorizontally等函数。如代码清单3-24所示，tests项目中MenuTest.cpp的MenuLayer2的构造函数alignMenusH就是alignItems-Horizontally水平对齐两种方法对比，一种是alignItems-Horizontally水平对齐，底下是alignItemsHorizontallyWithPadding留空间水平对齐，效果对比

请见之前的图3-18。

void MenuLayer2::alignMenusH()

{

for(int i=0;i<2;i++)

{

CCMenu \*menu = (CCMenu\*)getChildByTag(100+i);

menu->setPosition( m\_centeredMenu );

if(i==0)

{

// TIP: if no padding, padding = 5

menu->alignItemsHorizontally();

CCPoint p = menu->getPosition();

menu->setPosition( ccpAdd(p, ccp(0,30)) );

}

else

{

// TIP: but padding is configurable

/// 留¢?空?间?水?平?对?齐?

menu->alignItemsHorizontallyWithPadding(40);

CCPoint p = menu->getPosition();

menu->setPosition( ccpSub(p, ccp(0,30)) );

}

}

}