**《网络编程课程项目分析说明书》**

|  |  |
| --- | --- |
| **项目名：** | 商品商城 |
| **专业：** | 计算机科学与技术 |
| **班级：** | 3班 |
| **学号：** | 1708010348 |
| **姓名：** | 戴锦波 |

**编写日期：**

目录

[**1 摘要** 1](#_Toc25829746)

[**2 系统运行环境** 1](#_Toc25829747)

[**3 系统功能分析** 1](#_Toc25829748)

[3.1使用内容： 1](#_Toc25829749)

[3.1.1 注册个人用户账号和密码，登陆个人商城进行操作处理 1](#_Toc25829750)

[账号/密码的主要代码： 1](#_Toc25829751)

[3.2 功能内容：包括个人信息查询，商品管理，商品信息，商品添加以及商品的删除 2](#_Toc25829752)

[3.2.1 个人信息模块主要是进行用户信息的账号，密码，身份的修改和认证 2](#_Toc25829753)

[密码修改的主要代码： 2](#_Toc25829754)

[4 项目运行结果 9](#_Toc25829755)

**正文部分要求：**

**1 摘要**

（这部分，大概说明项目的背景，主要说明项目功能和采用的技术等）

项目的背景：模拟用户商品的处理机制，功能/技术：是通过C++/C语言技术实现用户登录注册后进行商品的增删改查处理

**2 系统运行环境**

（说明系统开放用到的开发工具、采用的编程语言、程序运行的电脑要求等）

　开发工具：Microsoft Visual Studio,编程语言：c语言/c++语言，电脑要求：window7以上,处理器：Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz 3.10GHZ,安装内存：2GB以上，安装类型：32位操作系统

**3 系统功能分析**

（此部分为文档的主体。主要描述项目的功能，主要采用了哪些技术实现这些功能。以功能模块为单位进行分析说明，需要有功能模块对应的主要代码（代码尽可能有详细的注释），可以结合运行结果图和流程图等来辅助说明）

3.1使用内容：

3.1.1 注册个人用户账号和密码，登陆个人商城进行操作处理

账号/密码的主要代码：

void CLoginDlg::OnBnClickedButton1()

{

// TODO: 在此添加控件通知处理程序代码

//登陆按钮点击事件

UpdateData(TRUE); //将编辑框里面的值，同步到变量里面

if(m\_name.IsEmpty() || m\_pwd.IsEmpty())

{

MessageBox(TEXT("用户名或密码不能为空"));

return;

}

//获取正确的值

CInfoFile file;

CString name, pwd;

file.ReadLogin(name,pwd);

if(name == m\_name) //用户名一制

{

if(pwd == m\_pwd) //密码一致

{

//关闭当前窗口

CDialog::OnCancel();

}

else

{

MessageBox(TEXT("用户名或者密码错误"));

return;

}

}

else

{

MessageBox(TEXT("用户名或者密码错误"));

return;

}

}

3.2 功能内容：包括个人信息查询，商品管理，商品信息，商品添加以及商品的删除

3.2.1 个人信息模块主要是进行用户信息的账号，密码，身份的修改和认证

密码修改的主要代码：

int CMainFrame::OnCreate(LPCREATESTRUCT lpCreateStruct)

{

if (CFrameWnd::OnCreate(lpCreateStruct) == -1)

return -1;

if (!m\_wndStatusBar.Create(this))

{

TRACE0("未能创建状态栏\n");

return -1; // 未能创建

}

m\_wndStatusBar.SetIndicators(indicators, sizeof(indicators)/sizeof(UINT));

//设置图标

SetClassLong(m\_hWnd, GCL\_HICON,(long)AfxGetApp()->LoadIconW(IDI\_ICON\_WIN));

//设置右侧标题

SetTitle(TEXT("哈t哈t哈t"));

//设置窗口大小

MoveWindow(0,0,800,500);

//设置窗口居中

CenterWindow();

//测试CInfoFile类

/\*CInfoFile file;

CString name, pwd;

file.WritePwd("哈哈哈", "123456");

file.ReadLogin(name, pwd);

file.ReadDocline();

MessageBox(name);

MessageBox(pwd);

\*/

return 0;

}

BOOL CMainFrame::PreCreateWindow(CREATESTRUCT& cs)

{

if( !CFrameWnd::PreCreateWindow(cs) )

return FALSE;

// TODO: 在此处通过修改

// CREATESTRUCT cs 来修改窗口类或样式

return TRUE;

}

3.2.2 商品管理主要是进行商品订单的购买以及商品订单信息的查询

购买的商品的主要代码：

void CSellDlg::OnBnClickedButton1()

{

UpdateData(TRUE);

// 购买功能的实现?

if(m\_num<=0)

{

MessageBox(TEXT("购买个数不能小于等于0"));

return;

}

if(m\_num>m\_left)

{

MessageBox(TEXT("购买个数不能大于库存?"));

return;

}

//购买

//获取到具体的要购买的商品名称?

CString name;

int index = m\_combo.GetCurSel();

m\_combo.GetLBText(index, name);

CInfoFile file;

file.ReadDocline();

for(list<msg>::iterator it=file.ls.begin(); it != file.ls.end(); it++)

{

if((CString)it->name.c\_str() == name)

{

//同步到库存控件上?

it->num = m\_left - m\_num;

m\_left = it->num; //控件的库存量?

//告诉用户具体的购买信息

CString str;

str.Format(\_T("商品名%s \r\n购买数量?：o%d \r\n单价?：o%d \r\n总价?：o%d"),name, m\_num, m\_price, m\_price\*m\_num);

m\_sellInfo = str;

UpdateData(FALSE);

MessageBox(\_T("购买成功?"));

}

}

file.WriteDocline();

//清空数据

m\_num = 0;

m\_sellInfo.Empty();

UpdateData(FALSE);

}

3.2.3商品信息，主要是把添加的和删除商品时，查询商品的增删处理，主要包括商品的类型，单价，库存个数，以及添加个数的要求

主要代码：

void CSelectView::OnInitialUpdate()

{

CTreeView::OnInitialUpdate();

// TODO: 在此添加专用代码和或调用基类

//图标资源的加载?CWinApp::LoadIcon

//IDI\_ICON\_RE为图标资源的ID

HICON icon = AfxGetApp()->LoadIconW(IDI\_ICON\_RE);

//图片列表的创建CImageList::Create

//30,30 指定图标的宽度和高度¨¨

//ILC\_COLOR32 图标格式?

//1，ê?1 有多少´图标就写多少个?

m\_imageList.Create(30,30,ILC\_COLOR32,1,1);

//图片列D表添加图标CImageList::Add

m\_imageList.Add(icon);

//获取视图的树控件CImageList::GetTreeCtrl

m\_treeCtrl = &GetTreeCtrl();

//树控件设置图片列D表CTreeCtrl::SetImageList

m\_treeCtrl->SetImageList(&m\_imageList,TVSIL\_NORMAL);

//树控件设置节点CTreeCtrl::InsertItem

m\_treeCtrl->InsertItem(TEXT("个人信息"),0,0,NULL);

m\_treeCtrl->InsertItem(TEXT("商品管理"),0,0,NULL);

m\_treeCtrl->InsertItem(TEXT("商品信息"),0,0,NULL);

m\_treeCtrl->InsertItem(TEXT("商品添加"),0,0,NULL);

m\_treeCtrl->InsertItem(TEXT("商品删除"),0,0,NULL);

}

//指示选定项已从项转为另项?

void CSelectView::OnTvnSelchanged(NMHDR \*pNMHDR, LRESULT \*pResult)

{

LPNMTREEVIEW pNMTreeView = reinterpret\_cast<LPNMTREEVIEW>(pNMHDR);

// TODO: 在此添加控件通知处理程序代码?

\*pResult = 0;

//获取当前选中项?

HTREEITEM item = m\_treeCtrl->GetSelectedItem();

// 通ª过选中的项获取选中项的内容

CString str = m\_treeCtrl->GetItemText(item);

////MessageBox(str);

if(str==TEXT("个人信息"))

{

//需要包含框架类头文件include "MainFrm.h"

//CWnd::PostMessage 将个消息放入到窗口的消息队列

//AfxGetMainWnd(); 框架窗对象的指针

//AfxGetMainWnd().GetSafeHwnd();获取返回窗口的句柄，CWnd::GetSafeHwnd

//WM\_A 自定义消息

//(WPARAM)WU\_A 指定了附加的消息信息

//(LPARAM)0 指定了附加的消息信息，此参数这里没有意义

::PostMessage(AfxGetMainWnd()->GetSafeHwnd(),WU\_A, (WPARAM)WU\_A, (LPARAM)0);

}

if(str==TEXT("商品管理"))

{

//需要包含框架类头文件t#include "MainFrm.h"

//CWnd::PostMessage 将个消息放入到窗口的消息队列D

//AfxGetMainWnd(); 框架窗口对象的指针

//AfxGetMainWnd().GetSafeHwnd();获取返回窗口的句柄，CWnd::GetSafeHwnd

//WM\_A 自定义消息

//(WPARAM)WU\_A 指定了附加的消息信息

//(LPARAM)0 指定了附加的消息信息，此参数这里没有意义

::PostMessage(AfxGetMainWnd()->GetSafeHwnd(),WU\_B, (WPARAM)WU\_B, (LPARAM)0);

}

if(str==TEXT("商品信息"))

{

//需要包含框架类头文件#include "MainFrm.h"

//CWnd::PostMessage 将个消息放入到窗口的消息队列

//AfxGetMainWnd(); 框架窗口对象的指针?

//AfxGetMainWnd().GetSafeHwnd();获取返回窗口的句柄，CWnd::GetSafeHwnd

//WM\_A 自定义消息

//(WPARAM)WU\_A 指定了附加的消息信息

//(LPARAM)0 指定了附加的消息信息，此参数这里没有意义。

::PostMessage(AfxGetMainWnd()->GetSafeHwnd(),WU\_C, (WPARAM)WU\_C, (LPARAM)0);

}

if(str==TEXT("商品添加"))

{

//需要包含框架类头文件#include "MainFrm.h"

//CWnd::PostMessage 将个消息放入到窗口的消息队列

//AfxGetMainWnd(); 框架窗口对象的指针

//AfxGetMainWnd().GetSafeHwnd();获取返回窗口的句柄，CWnd::GetSafeHwnd

//WM\_A 自定义消息

//(WPARAM)WU\_A 指定了附加的消息信息

//(LPARAM)0 指定了附加的消息信息，此参数这里没有意义

::PostMessage(AfxGetMainWnd()->GetSafeHwnd(),WU\_D, (WPARAM)WU\_D, (LPARAM)0);

}

if(str==TEXT("商品删除"))

{

//需要包含框架类头文件#include "MainFrm.h"

//CWnd::PostMessage 将个消息放入到窗口的消息队列

//AfxGetMainWnd(); 框架窗口对象的指针

//AfxGetMainWnd().GetSafeHwnd();获取返回窗口的句柄，CWnd::GetSafeHwnd

//WM\_A 自定义消息

//(WPARAM)WU\_A 指定了附加的消息信息

//(LPARAM)0 指定了附加的消息信息，此参数这里没有意义

::PostMessage(AfxGetMainWnd()->GetSafeHwnd(),WU\_E, (WPARAM)WU\_E, (LPARAM)0);

}

}

3.2.4商品添加：主要是添加商品的个数，以及添加商品新商品内容,主要包括商品的类型，单价，库存个数，以及添加个数的要求

添加的主要代码：

void CAddDlg::OnBnClickedButton3()

{

// TODO: 在¨²此添加控件通知处理程序代码?

UpdateData(TRUE); //取出控件中的值到变量里面去

// 购买功|能的实现

if(m\_num2<=0)

{

MessageBox(TEXT("添加个数不能小等0"));

return;

}

if(m\_price2<=0)

{

MessageBox(TEXT("添加商品单价不能小于等于0"));

return;

}

if(m\_name2.IsEmpty() || m\_name2.GetLength()>8)

{

MessageBox(TEXT("添加商品名为空或者长度大于个字符?"));

return;

}

CInfoFile file;

file.ReadDocline();

file.Addline(m\_name2, m\_num2, m\_price2);

file.WriteDocline();

MessageBox(TEXT("添加成功?"));

//同步到左¨侧的商品名称下拉列表中

m\_combo.InsertString(0,m\_name2);

m\_combo.SetCurSel(0);

OnCbnSelchangeCombo1();

//清空输入的内容

m\_name2.Empty();

m\_price2 = 0;

m\_num2 = 0;

UpdateData(FALSE);

}

3.2.5 商品的删除：主要是删除商品的名称，单价，库存个数以及删除个数

删除的主要代码：

void CDelDlg::OnBnClickedButton1()

{

UpdateData();

// 删除功能的实Ì现?

if(m\_num<=0)

{

MessageBox(TEXT("删除个数不能小于等于0"));

return;

}

if(m\_num>=m\_left)

{

MessageBox(TEXT("删¦除个数不能大于等于库存数量?"));

return;

}

//删除

//获取到具体的要购买的商品名称?

CString name;

int index = m\_combo.GetCurSel();

m\_combo.GetLBText(index, name);

CInfoFile file;

file.ReadDocline();

for(list<msg>::iterator it=file.ls.begin(); it != file.ls.end(); it++)

{

if((CString)it->name.c\_str() == name)

{

//同步到库存控件上?

m\_left = it->num - m\_num; //原来的库存量需要删除的库存量

it->num = m\_left;

//告诉用户具体的删除信息

CString str;

str.Format(\_T("商品名?\r\n删除数量\r\n单价?\r\n \r\n删除成功? "),name, m\_num, m\_price);

file.WriteDocline();

MessageBox(str);

}

}

//清空数据

m\_num = 0;

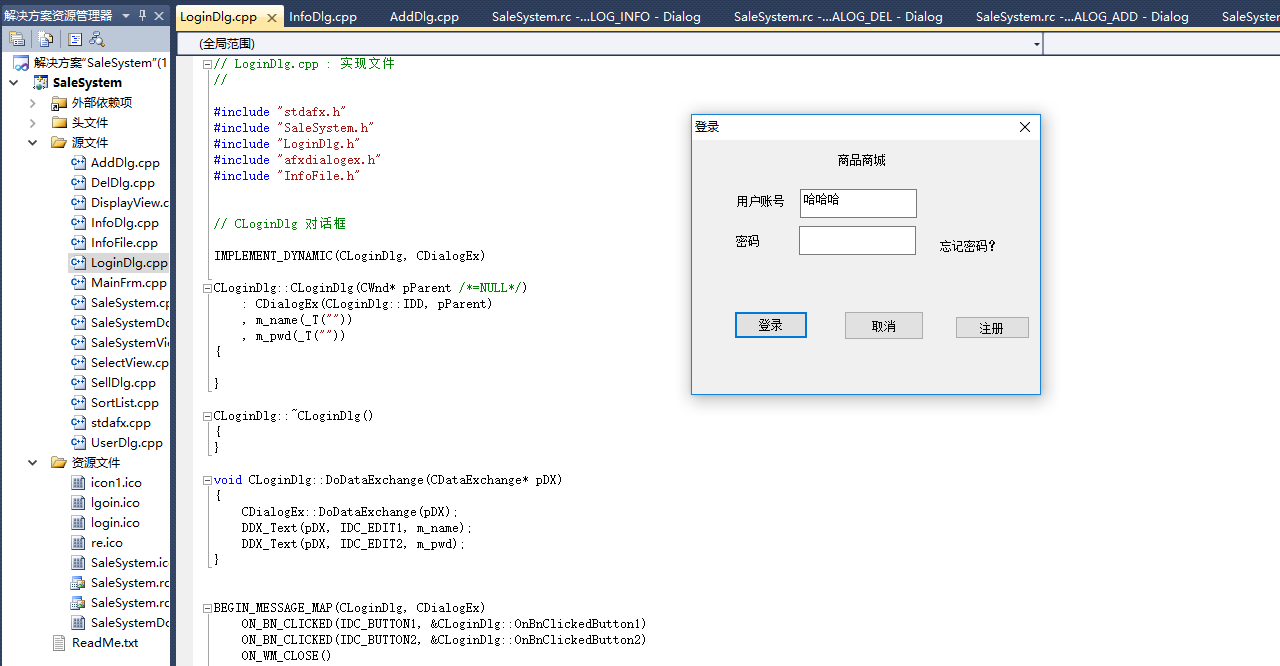
UpdateData(FALSE);

}

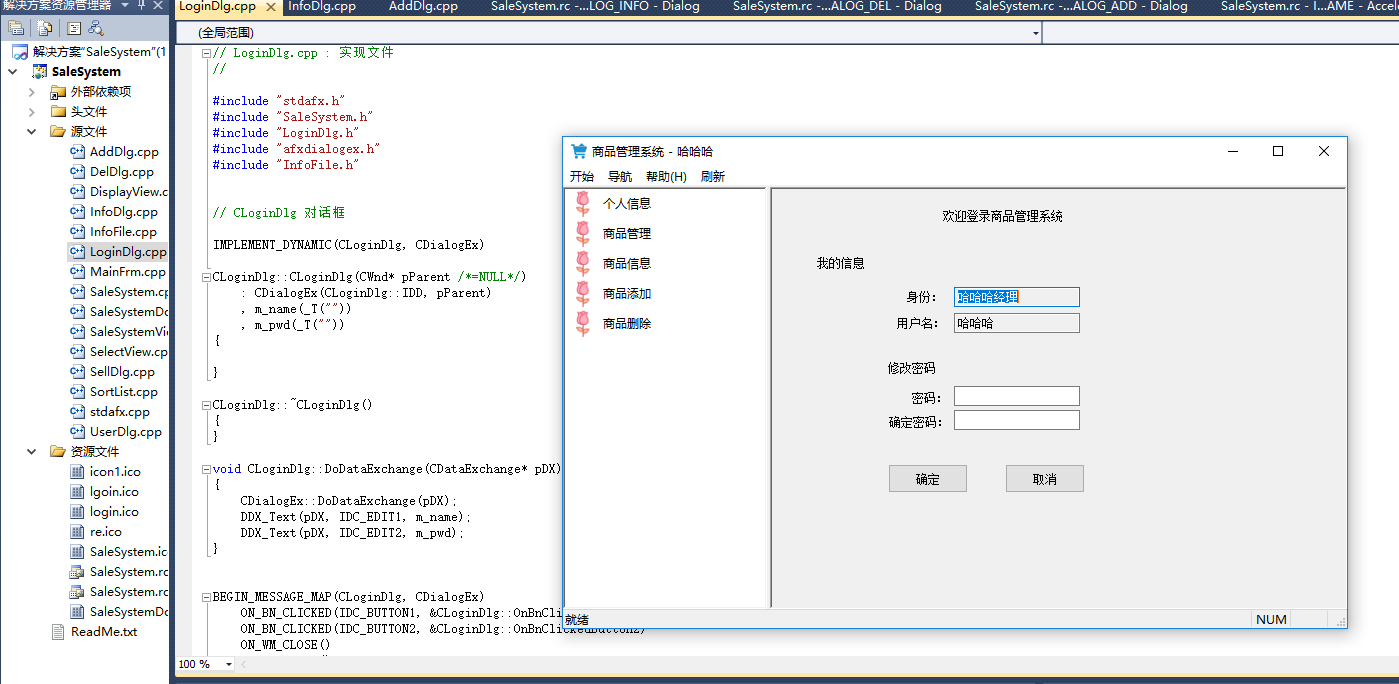
4 项目运行结果

（项目截图展示，并且每个图片有对应的功能简要说明）

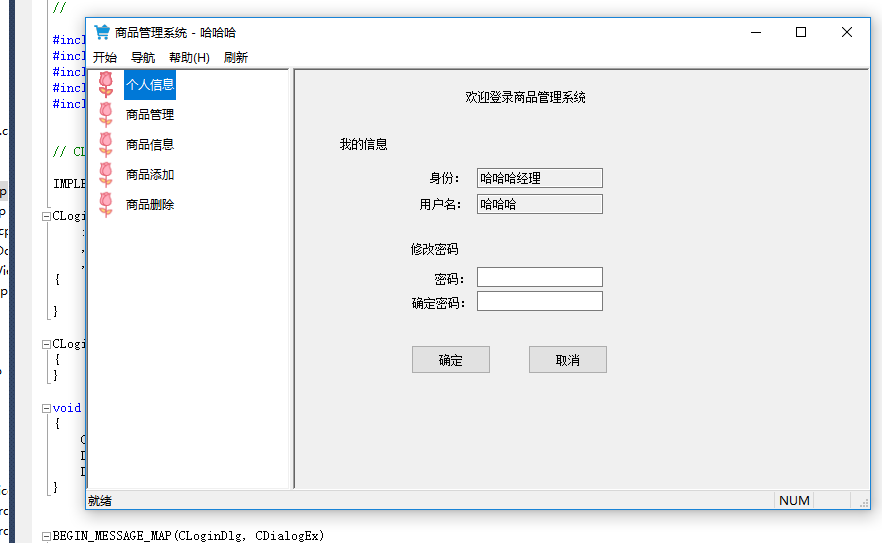
首页：



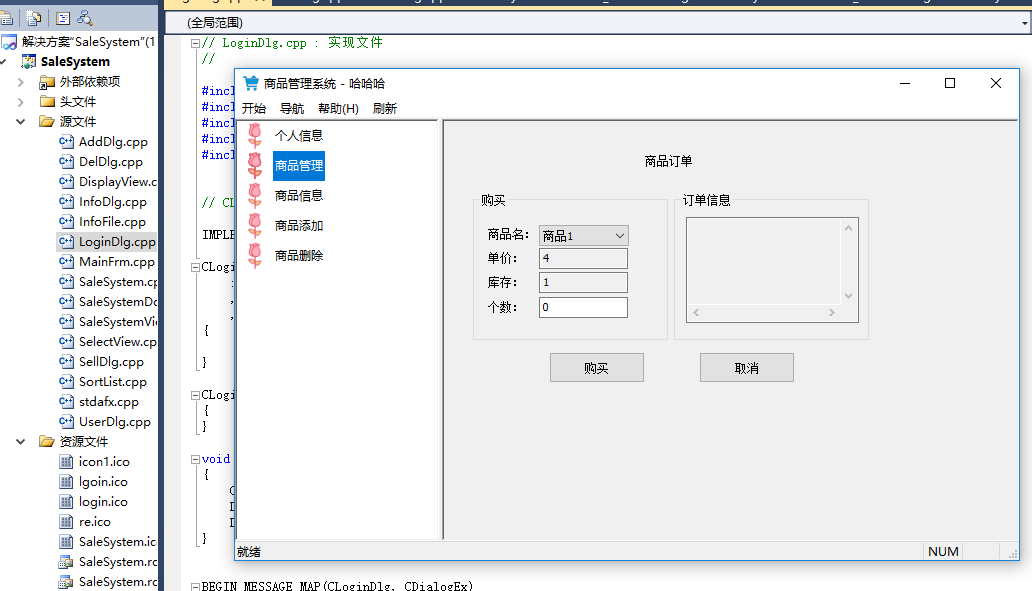
内容区：



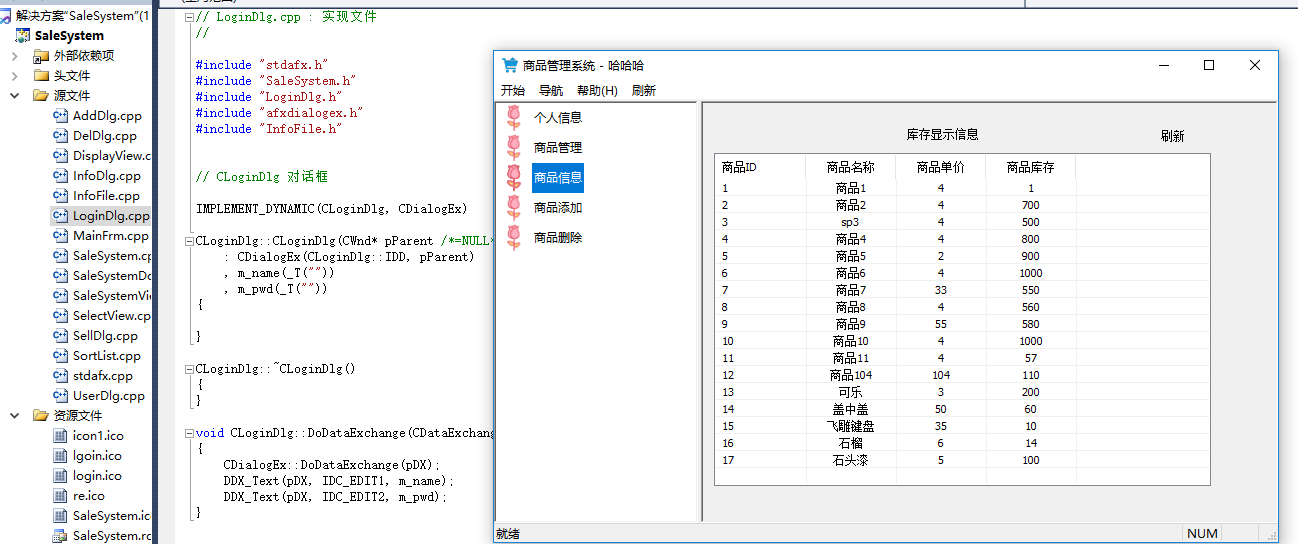
个人信息：



商品管理：



商品添加：



商品添加：



商品删除：

