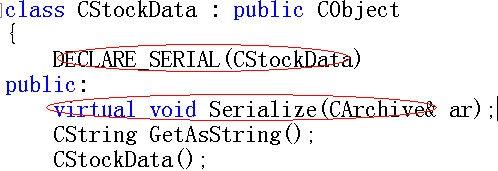
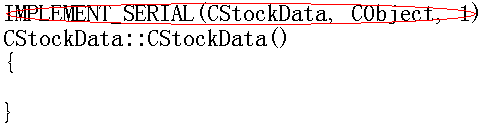
股票数据预览程序 第五部分 实现序列化

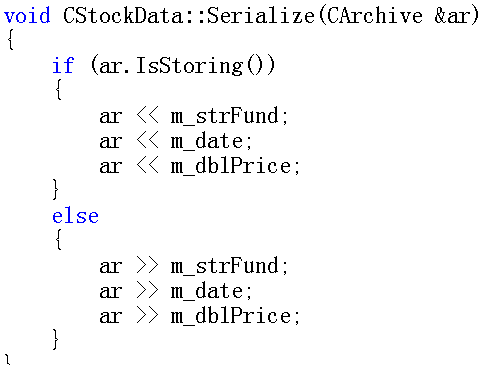
1. CStockData类的序列化操作
   1. 在StockData.h文件中添加序列化操作声明宏：DECLARE\_SERIAL(CStockData)以及序列化函数声明virtual void Serialize(CArchive& ar)



* 1. 在StockData.cpp文件中添加序列化操作实现宏：IMPLEMENT\_SERIAL(CStockData, CObject, 1)



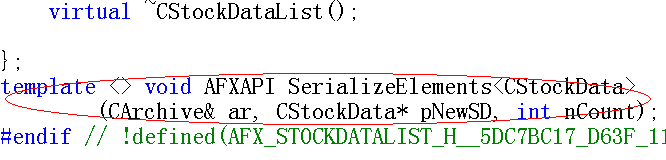
* 1. 用CStockData\_Serialize.cpp文件中的代码实现Serialize函数：



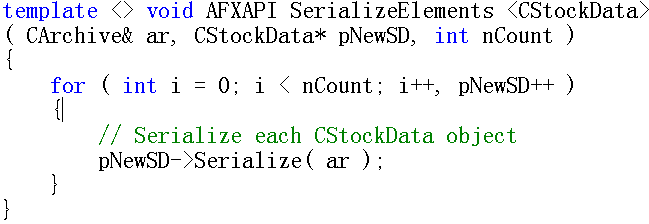
1. CStockDataList类的序列化操作
   1. 在StockDataList.h类定义后,#endif之前加入模板序列化函数声明：

template <> void AFXAPI SerializeElements<CStockData>

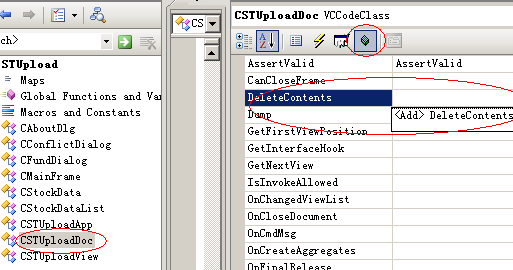
(CArchive& ar, CStockData\* pNewSD, int nCount);



* 1. 在StockDataList.cpp文件末尾，用加入SerializeElements函数的实现：



1. CSTUploadDoc类的序列化操作
   1. 用CSTUploadDoc\_Serialize.cpp中的代码实现CSTUploadDoc::Serialize函数
   2. 向CSTUploadDoc类添加虚函数DeleteContents



* 1. 使用DeleteContents.cpp中的代码实现DeleteContents

1. 在用户修改了数据之后提示用户保存文件
   1. 将CSTUploadDoc::OnDataImport函数的最后一句改为：

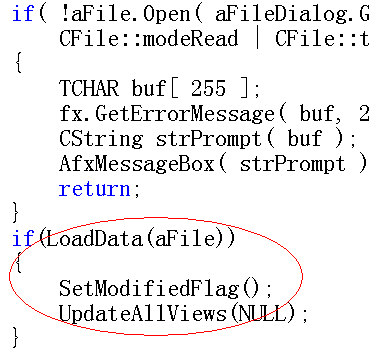
if(LoadData(aFile))

{

SetModifiedFlag();

UpdateAllViews(NULL);

}



* 1. 在CFundDialog::OnLbnSelchangeFundlist函数最后添加pDoc->SetModifiedFlag(true);

1. 编译并运行程序，此时可以点击【保存】菜单把当前内容存到磁盘上，下次可以通过【打开】菜单打开该文件