

VFP5.0 中控制 Excel 的一种方法

代增刚 杨凡德 (总装备部指挥技术学院 101416)

摘要:本文讲述了在 VFP5.0 中,通过 OLE 自动执行对象(Object Linking Embedding Automation)技术直接控制 Excel 的一种方法。利用这一方法,可在 VFP5.0 中设计一系列按钮,直接控制平台中的 Excel 应用系统,将 VFP5.0 中的数据传送到 Excel 中,产生工作簿、统计图、存储、打印等操作。

关键词:OLE 自动执行 Excel VFP5.0

一、引言

在许多数据库应用系统中,往往需要将最终运算数据传送到 Excel 中,生成会计帐目,或利用这些数据产生一些统计图。实现这些功能一般有两种方法:一种是脱离数据库管理系统情况下,再进入 Excel 环境中,直接打开数据库中所要操作的表,利用 Excel 提供的各种命令进行操作,这样做只适合于熟悉 VFP5.0 (Visual FoxPro 5.0) 和 Excel 操作的人员完成,不适合进行程序化操作,更不能用到数据库应用系统中去;另一种方法是直接在数据库应用系统中,控制表记录的生成和产生统计图。过去常用的做法是利用 DDE 动态数据交换进行操作。在 VFP5.0 中,则可以利用 OLE 自动执行对象这一新特性去实现。下面分别将对在 VFP5.0 中如何启动 Excel、增加工作簿、产生统计图、存储、打印工作表、退出 Excel 的具体方法介绍如下。

二、实现方法

1. 基本条件及所用对象、方法和属性

要应用 OLE 自动执行对象技术,还必须对系统的运行环境及所用对象的含义、方法及属性有所了解,现说明如下:

(1) 32 位的操作系统,如 Win95、NT 等。

(2) 确保 Excel. Application 已注册(运行 Windows 的 Regedit. exe 查看)。

(3) 所用对象

对象名称	含义
Application	Excel 应用程序
WorkBooks	Excel 工作簿对象

(4) 所用方法(Method)

对象名称	方法(Method)	执行含义
Application	Cells	设定或传回某单元格的内容
	Range	设定或传回某一范围的单元格
	Charts	设定或传回工作簿的单一统计图表
	Save	激活存储文件对话框
	Quit	结束 Excel Application
WorkBooks	Add	增加一个工作簿
Charts	Add	增加一个统计图

(5) 所用属性(Property)

对象名称	属性(Property)	属性含义
Application	Application	其值为 .T. 时, Excel OLE 对象允许显示
	Value	设定或传回存储文件的内容

2. 启动 Excel

要实现这些功能,首先要启动 Excel,方法是使用 CreateObject() 函数产生一个 OLE Automation 对象,语法如下:

ExcelObject = CreateObject("Excel. Application")

注意:上面语法并不能在屏幕上立即显示 Excel 界面,要显示该界面,必须使该对象的 Visible 属性为 .T.。

语法为:ExcelObject. Visible = .T.

3. 增加工作簿、传送表数据

在 Excel 对象中增加一个工作簿(WorkBook),只要执行 WorkBooks 对象的 Add 方法即可。

语法为:ExcelObject. WorkBooks. Add

要传送数据库中的表记录,首先要打开数据库表(Table),利用 CreateObject() 函数创建一个 OLE 类的 Ex-

cel. Application 对象,再利用 WorkBooks 对象增加一个工作簿,方法是 ExcelObject. WorkBooks. Add,然后利用工作表单元中的方法 Cells(),指定存储单元,定义标题内容(即对应于数据库表中的相应字段),最后将数据库中的表记录逐一传送到 Excel 工作表单元中去。如有一日销售表 Sales,该表共有两个字段,分别是产品名称和销售数量,要实现这一功能,具体程序如下:

```
USE Sales && 打开数据库表 Sales
ExcelObject = CreateObject("Excel. Application")
&& 创建 Excel.Application 对象
ExcelObject.Visible = .T. && 显示 Excel 工作界面
ExcelObject.WorkBooks.Add && 新增一个工作簿
ExcelObject.ActiveWorkbook.Sheets(1).Cells(1,1).
Value;
    = '产品名称' && 在工作表 1 中指定单元格中生成标题字符串
ExcelObject.ActiveWorkbook.Sheets(1).Cells(1,2).
Value;
    = '销售数量'
Select Sales
h=2 && 行变量
l=1 && 列变量
Goto top
* 以下是往指定单元传送数据库表记录
Do While ! eof()
    ExcelObject.ActiveWorkbook.Sheets(1).Cells(h,
1).Value;
        = Sales.产品名称
    ExcelObject.ActiveWorkbook.Sheets(1).Cells(h, l
+ 1).Value;
        = Sales.销售数量
    h = h + 1
Skip
EndDo
```

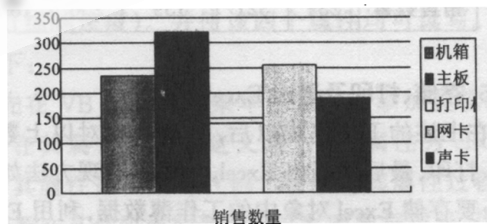
4. 生成统计图表(Chart)

要产生统计图,必须先用方法 Range 定义要生成统计图的单元范围,然后用 Charts 对象执行 Add 就可以生成一份统计图,如下图所示,实现程序如下:

```
USE Sales && 打开数据库表 Sales
ExcelObject = CreateObject("Excel. Application")
```

&& 创建 Excel.Application 对象

```
ExcelObject.Visible = .T. && 显示 Excel 工作界面
ExcelObject.WorkBooks.Add && 新增一个工作簿
ExcelObject.ActiveWorkbook.Sheets(1).Cells(1,1).
Value;
    = '产品名称' && 在工作表 1 中指定单元格中生成标题字符串
ExcelObject.ActiveWorkbook.Sheets(1).Cells(1,2).
Value;
    = '销售数量'
Select Sales
Count to jls && 统计记录数
h=2 && 行变量
l=1 && 列变量
Goto top
* 以下是往指定单元传送数据库表记录
Do While ! eof()
    ExcelObject.ActiveWorkbook.Sheets(1).Cells(h,
1).Value;
        = Sales.产品名称
    ExcelObject.ActiveWorkbook.Sheets(1).Cells(h, l
+ 1).Value;
        = Sales.销售数量
    h = h + 1
Skip
EndDo
```



销售统计图

```
jlsfw = "A1:B" + Alltrim(Str(jls + 1)) && 定义统计的单元范围
```

```
ExcelObject.Range(jlsfw).Select && 选择统计的单元范围
```

```
ExcelObject.Charts.Add && 增加一份统计图
```

日销售表

产品名称	机箱	主板	打印机	网卡	声卡
销售数量	234	321	139	256	189

5. 存储、打印及退出 Excel

在上述的工作完成以后,我们需要对以上数据进行存储、打印,最后将退出 Excel,具体的实现方法如下:

- 要存储 Excel 对象中的工作簿数据,利用 Excel.Application 对象的 Save()方法,即可完成数据的存储。

其语法为: ExcelObject.Save()

它将激活 Excel 的存储对话框供用户选择。

- 打印工作簿是利用 WorkBooks 对象的 PrintOut()方法。

其语法为: ExcelObject.ActiveWorkBooks.PrintOut
(参数 1, 参数 2, 参数 3, 参数 4)

号

参数 1:数值型 - 指定工作表(sheet)的开始顺序编

参数 2:数值型 - 指定工作表(sheet)的结束编号

参数 3:数值型 - 指定打印工作表(sheet)的份数

参数 4:逻辑型 - 值为.T.时进行预览,值为.F.时不进行预览而直接打印

- 要退出 Excel,结束 Excel.Application 对象,使用方法 Quit,返回到数据库应用系统中去。

其语法为:ExcelObject.Quit

参考文献

- [1] Les Pinter, John Pinter 著·Visual FoxPro 编程手册,北京:科学出版社,1997
- [2] 陈宗兴编著·Visual FoxPro 5.0 完全 OLE 应用手册,北京:宇航出版社,1998

(来稿时间:1999 年 1 月)