Spreadsheet Case 10

Rego Medical Supply

Problem: Develop a human resources reporting system.

Management skills: Controlling
Deciding

PC skills: Macro building
Database query and extract

File: Rego_q.xls

Lorraine Alberts is Human Resources director at Rego Medical Supply, a small medical supply firm located near Columbus, Ohio. She is responsible for recruiting and training employees, developing compensation programs, and monitoring whether the firm complies with state and federal laws prohibiting job discrimination based on race, sex, religion, or national origin.

Load the file Rego_q.xls from your data diskette. It shows the employee database maintained by Rego Medical Supply. Most of the fields are self-explanatory. ED LEVEL designates the number of years of formal education an employee has. PERF CODE refers to the employee's job performance rating code. Employees are ranked on a four-point scale. A code of 1 designates excellent; 2, good; 3, average; and 4, below average. BASE PAY refers to the employee's annual salary.

Lorraine has been asked to find a way to identify minority or female employees for EEO reporting. Senior management has also asked her to identify high-performing employees, especially female or minority employees who have high performance ratings. Rego is a small company, so Lorraine was sifting through records one by one to obtain this information.

What Lorraine needs is an automated system to provide this information instantaneously. When she accesses the database, she would like to see the records of the most highly paid employees first or the records of all female employees first. Lorraine would like to enter a performance code and have the system generate a report of all employees with that performance code. She would also like a listing of all female employees.

Tasks

There are two tasks in this problem:

1. Use the macro-building facility of your spreadsheet software to build an easy-to-use system that produces two reports. The first sorts the database by base pay in descending order. The second sorts the database to produce a report of female employees with base pay in descending order.

2. Develop a macro that automatically searches for and extracts records according to any selected criteria from the worksheet entered by the user. For example, the macro should be able to list all female employees, or all female employees with a specified performance code. The macro should be executed after the user enters the specified criteria.

Time Estimates

Expert: 1 hour
Intermediate: 2 hours
Novice: 3 hours

Excel Tutorial For Spreadsheet Case 10

This case requires that you use Excel to build a *macro* to extract records from a database to produce a report automatically, and to perform a sort on a list. A macro is, in essence, a collection of commands. The commands are contained in a module sheet which can be stored in a workbook. The macros can be executed through the **Tools/Macro** menu item or through a custom Toolbar button.

For this exercise you will need Course.xls with some of the changes you made during the Spreadsheet Case 8 tutorial session.

Current versions of Excel have an extensive macro language called Visual Basic. This language is intended to be used by several Microsoft applications. Excel allows macros to be recorded by transcribing a series of operations acted out by the user to the equivalent Visual Basic commands.. The commands are stored in a module.

A macro can be recorded by accessing the command **Tools/Macro/Record New Macro** from the menu and naming the macro.

You can create a macro, for example, to automate the sorting of your student roster database by number of days overdue in ascending order, the same task you performed during the tutorial for Spreadsheet Case 8.

To record an Excel macro, select the **Macro** command from the Tools menu. This opens a sub-menu with more choices. From this menu, select **Record New Macro**. This opens a dialog box with the same name. The Macro Name text-entry box is highlighted, with the default name "Macro" followed by a number. Type in a name for the macro. The highlighted default name for the macro is automatically deleted and replaced with the new name you assigned. Note that macro names cannot have any spaces or punctuation marks, and must begin with a letter. Otherwise, they can contain letters, numbers and underscores. Below the name box is a documentation box

where you describe what your macro does. Excel automatically includes the name of the spreadsheet's author and the date in this area.

To start recording, click the OK button. Any action you perform with the mouse or keyboard will be recorded by Excel as a macro command that will be stored as part of your worksheet. To turn macro recording off, select **Tools/Macro/Stop Recording** from the menu. Excel won't record you selecting it as part of the macro.

The actions to record for automating the sorting of your student roster database by number of days overdue in ascending order would be as follows:

Select the range A14..G18.
Select **Data/Sort** from the menu.
Select the DAYS OVERDUE field as the primary sort key.
Select Ascending Order for the sort.
Makes sure that Header Row is selected.
Click the OK button to execute the sort.

To finish, select Tools/Macro/Stop Recording. Notice that the macro has stopped recording.

To test the macro, undo the changes you made while creating the macro. To run the macro, open the Tools menu, click on the Macro command and select Macros from the sub-menu. The Macro dialog box opens. Beneath the Macro/Name text box is a larger box listing all available macros. Click on your macro's name in this box. The name is displayed in the text box above and the macro's description appears at the bottom of the dialog box. Then click the Run button.

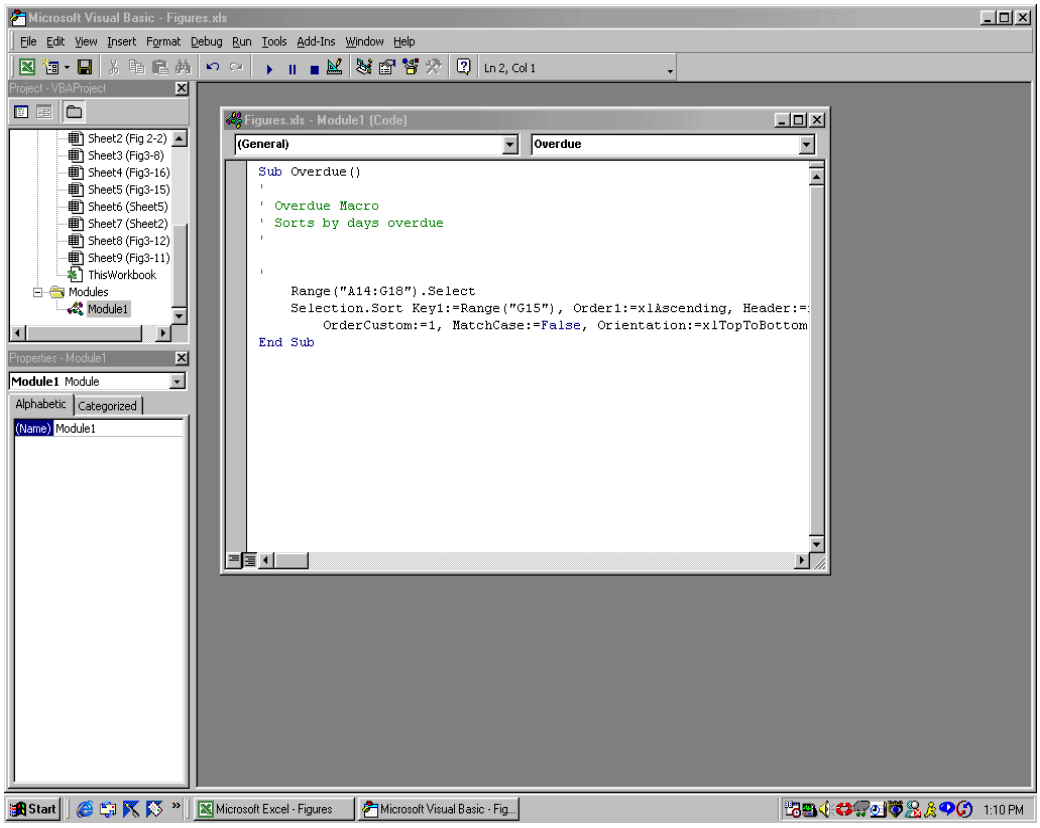
If the macro runs correctly, the database should be sorted in the appropriate order. For most purposes, recording a macro is the best way of creating macros and learning the language.

Viewing the Macro

You can view your macro commands by selecting Tools/Macro/Macros, then selecting the Edit option from the Macro dialog box. The Visual Basic module for the macro appears and can be edited. The module for the macro to sort the student roster database would look like Figure 3-17.

As you can see, modules have a non-tabular format, and lack the rows and columns typical of regular worksheets. Instead, data is displayed as lines of text against a plain, white background.

Figure 3-17


Microsoft Visual Basic - Figures.xls
File Edit View Insert Format Debug Run Tools Add-Ins Window Help
Ln 2, Col 1
Project - VBAProject
Sheet2 (Fig 2-2)
Sheet3 (Fig3-8)
Sheet4 (Fig3-16)
Sheet5 (Fig3-15)
Sheet6 (Sheet5)
Sheet7 (Sheet2)
Sheet8 (Fig3-12)
Sheet9 (Fig3-11)
ThisWorkbook
Modules
Module1
Properties - Module1
Module1 Module
Alphabetic
Categorized
(Name) Module1
Figures.xls - Module1 (Code)
(General)
Overdue
Sub Overdue()
'
' Overdue Macro
' Sorts by days overdue
'
'
Range("A14:G18").Select
Selection.Sort Key1:=Range("G15"), Order1:=xlAscending, Header:=
OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom
End Sub
Start
Microsoft Excel - Figures
Microsoft Visual Basic - Fig...
1:10 PM