

CS 701-Transaction Processing Using Microsoft Access

Purpose

The purpose of this micro seminar is to discuss the process of developing transaction processing using Microsoft Access.

Problem

The XYZ corporation needs a simple database system in which to track department, employee, and dependent history. To support the corporation's business objectives, the user needs to be able to perform the following transactions:

- Add a department
- Delete a department
- Modify a department
- Add an employee
- Delete an employee
- Modify an employee
- Add a dependent
- Delete a dependent
- Modify a dependent

Database Schema

Department

<u>DNUMBER</u>	DNAME	DLOCATION	DMGRSSN
----------------	-------	-----------	---------

Employee

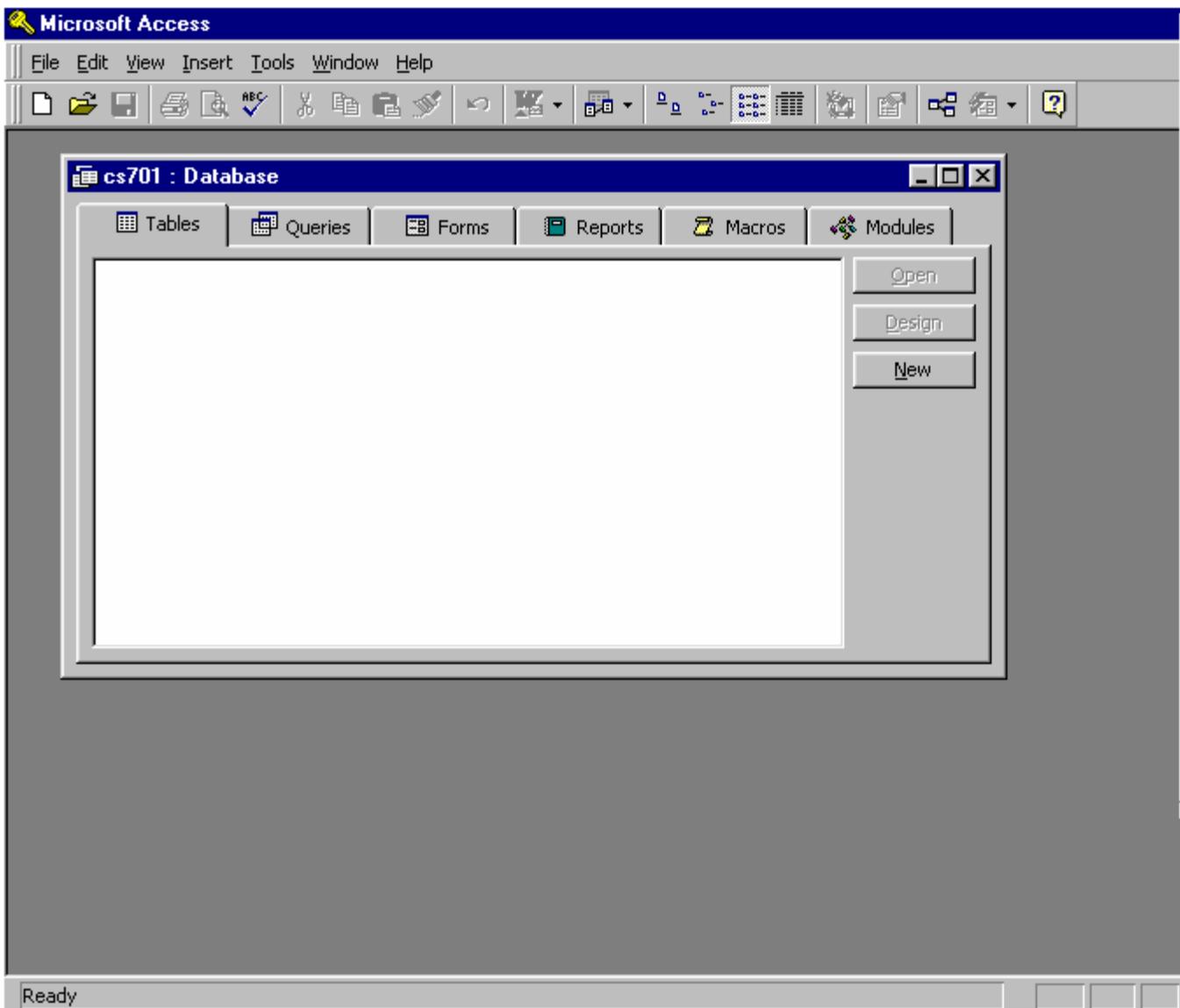
<u>SSN</u>	FNAME	MI	LNAME	BDATE	DNO	SALARY	SEX
------------	-------	----	-------	-------	-----	--------	-----

Dependent

<u>ESSN</u>	SSN	FNAME	MI	LNAME	BDATE	BDATE
-------------	-----	-------	----	-------	-------	-------

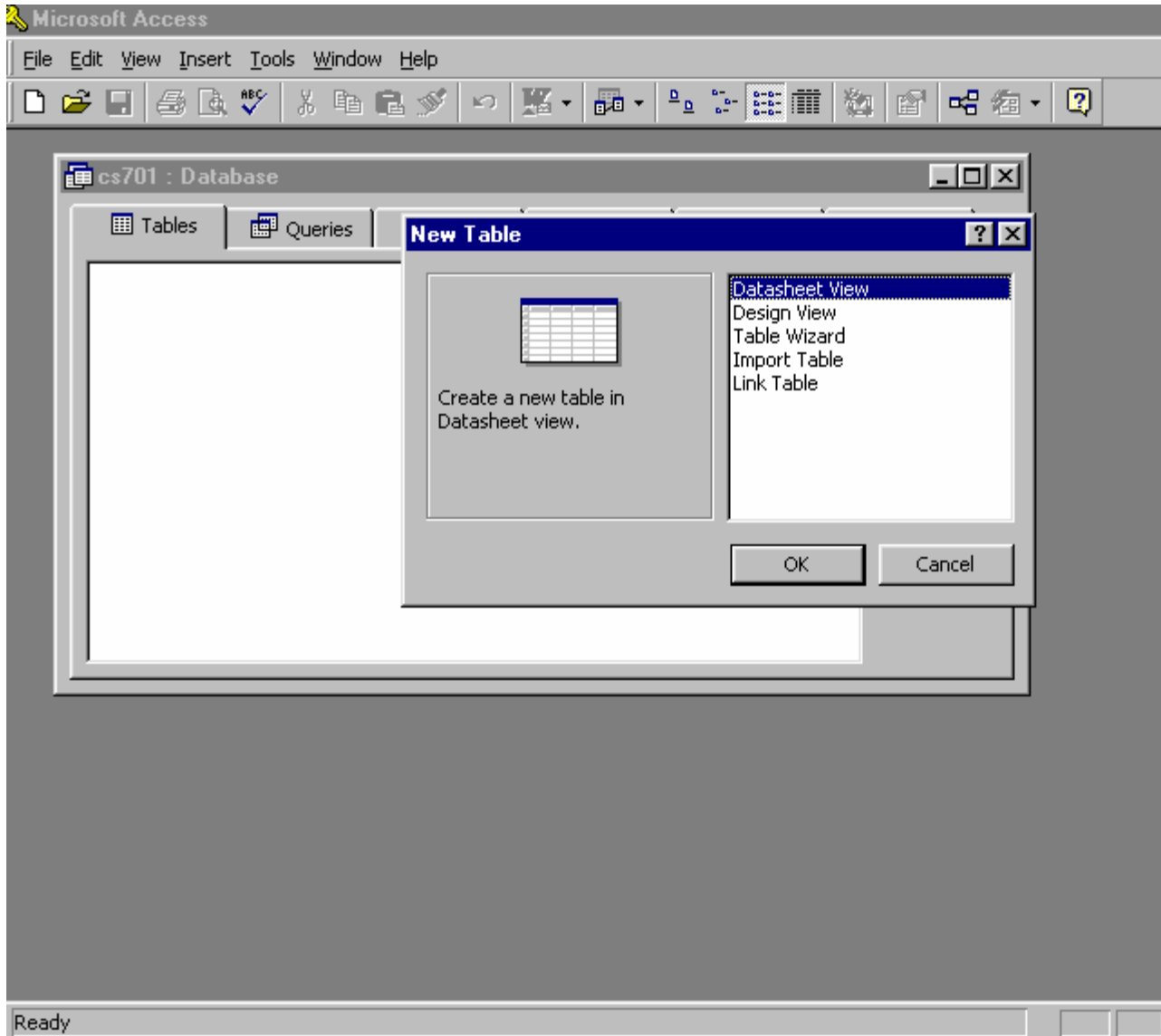
Create The Table

From the Access main screen, click on the Blank Database command button to request the creation of a new database.

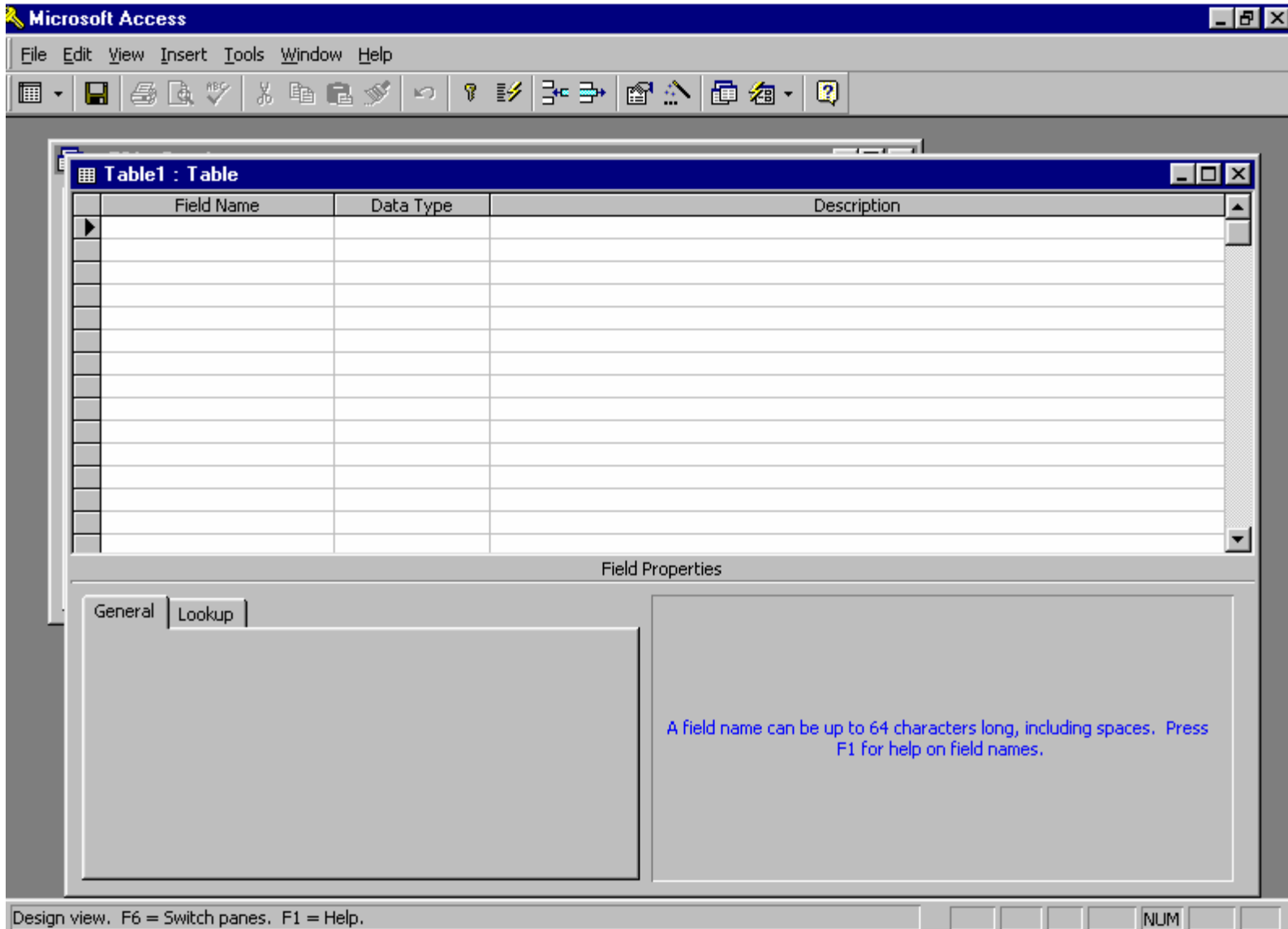


Click on the Table command button to request the creation of a new table.

When the New Table Dialog appears you can choose either the Table Wizards command button or the Design View command button. I recommend the Table Wizards button if you are new to this process.

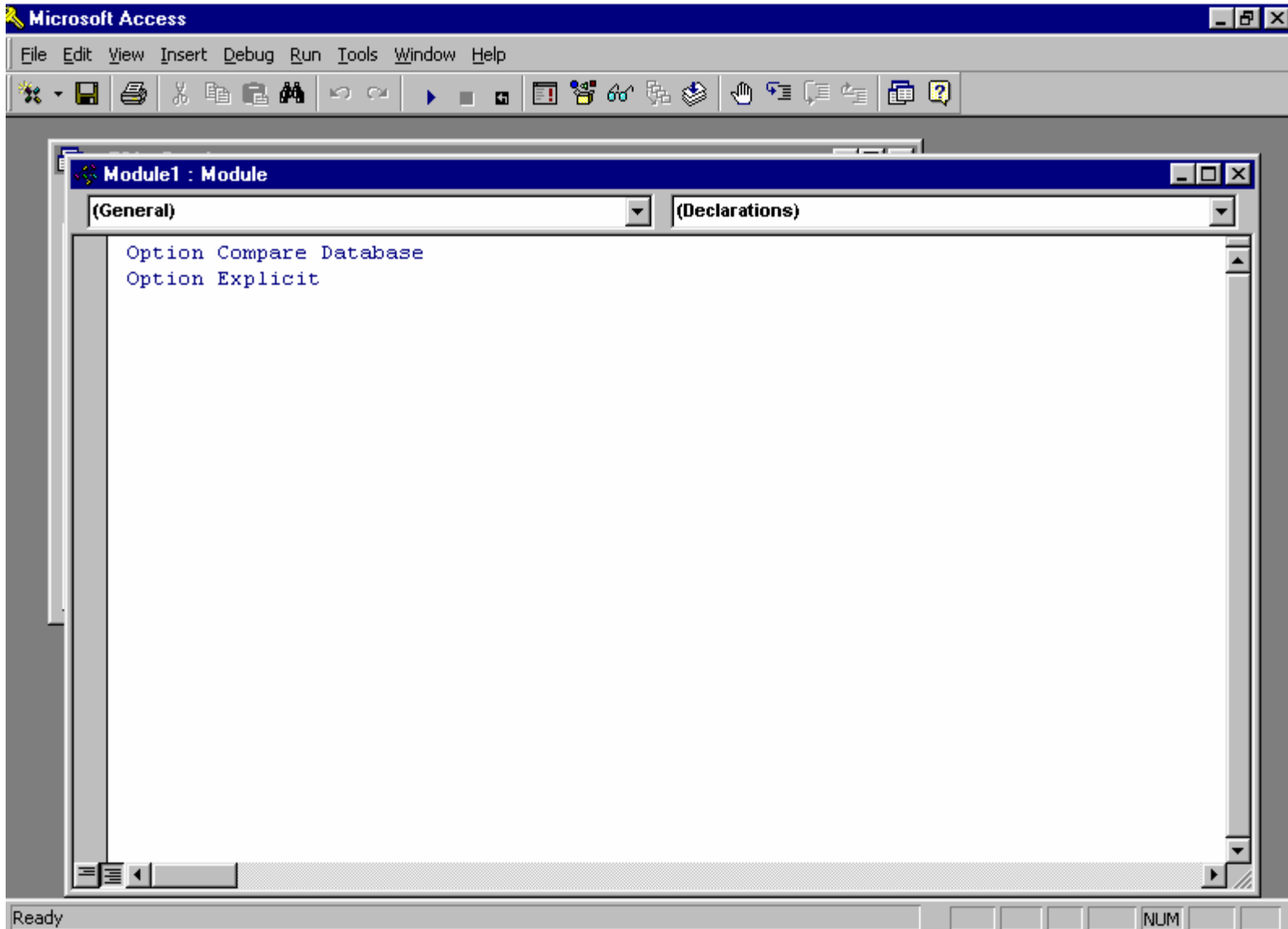


For this demonstration I chose the Design View command button. This causes the empty table template to appear. The table definitions then need to be entered as required.

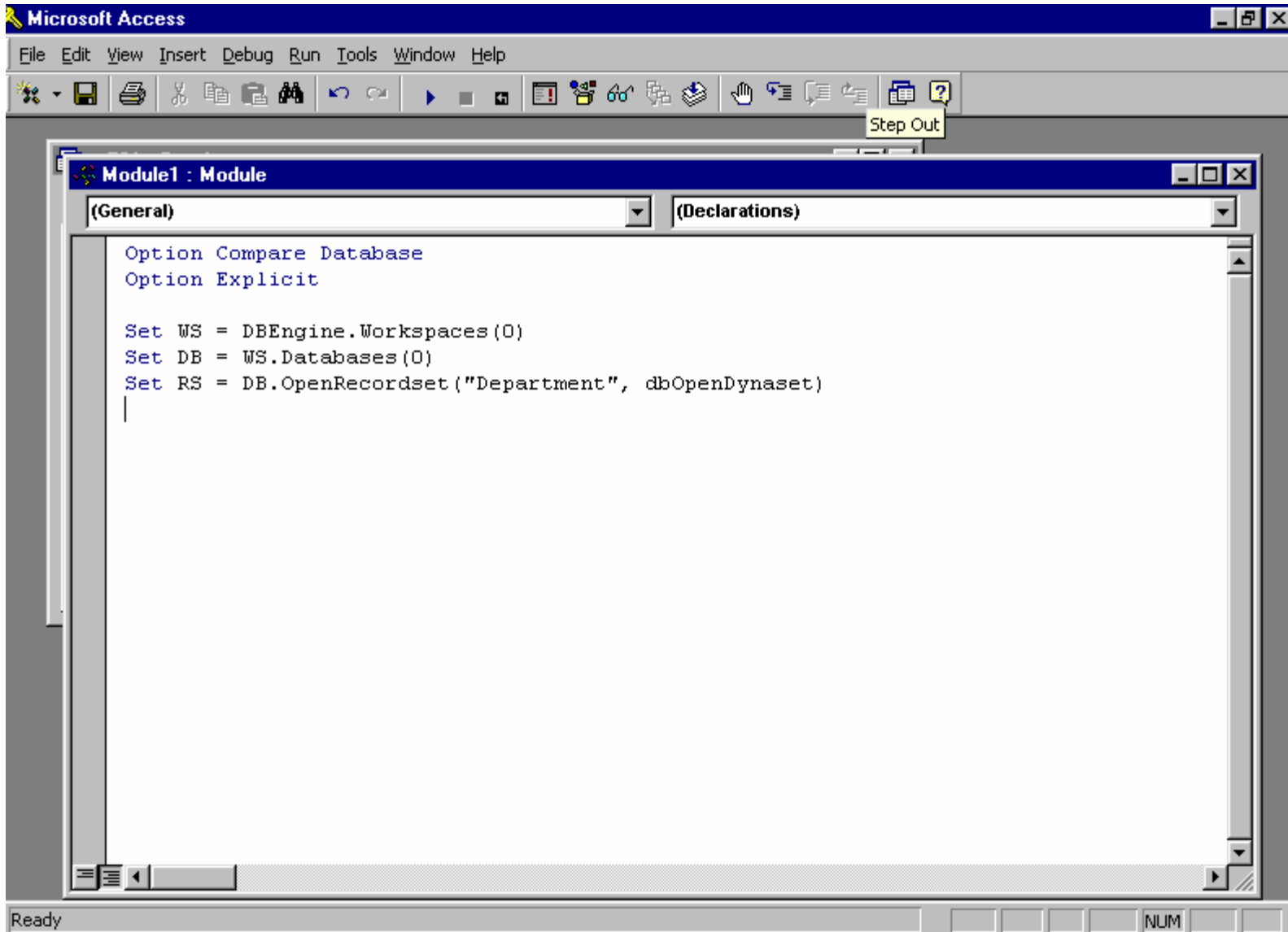


Create The Transactions

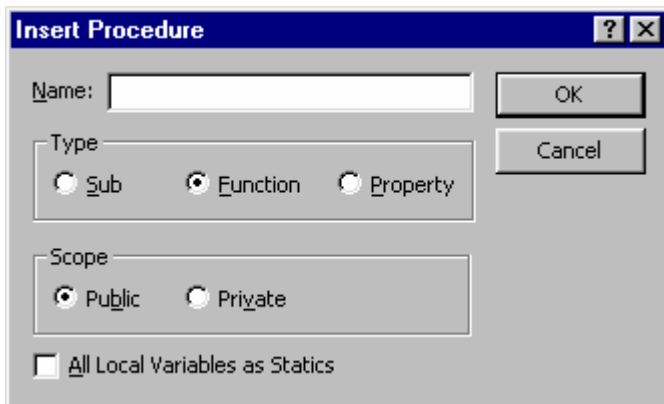
From the Access main screen, click on the Modules Tab and then the New command button to request the creation of a new ABC (Access Basic Code) module. This module will contain the transaction queries.



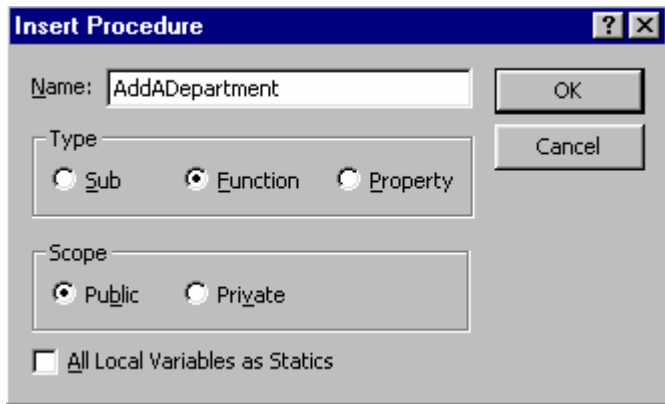
This will then bring you into the declarations section of the new module. This is where you declare variables visible within all functions in the Access Basic Module. Here is where we will declare variables for Workspace, Database, and Recordset instances.



The user then needs to create a new function procedure by clicking the Insert procedure command button.



And enters the name of the transaction to be created. For this example AddADepartment.



The user then needs to click the OK command button to open the function declaration area.


```

WS.BeginTrans
Sqlcontent = "SELECT COUNT(*) FROM DEPARTMENT WHERE DNUMBER =" &
DNUMBER
Set RS = DB.OpenRecordset(Sqlcontent, dbOpenDynaset)

If RS.Fields(0).Value = 0 Then
RS.Close

    If DMGRSSN = " " Then

        If DNUMBER <> " " And DNAME <> " " And DLOCATION <> " " Then
            Sqlinsert = " insert into Department(DNUMBER, DNAME, DLOCATION) "
            Sqlvalues1 = " values (" & DNUMBER & ", " & DNAME & ", " & DLOCATION
& " )"
            Sqlcontent = Sqlinsert & Sqlvalues1
            MsgBox (Sqlcontent)
            DB.Execute Sqlcontent
            WS.CommitTrans
            MsgBox ("Department Added!")

        ElseIf DNUMBER <> " " And DNAME <> " " And DLOCATION = " " Then
            Sqlinsert = " insert into Department(DNUMBER, DNAME) "
            Sqlvalues1 = " values (" & DNUMBER & ", " & DNAME & " )"
            Sqlcontent = Sqlinsert & Sqlvalues1
            MsgBox (Sqlcontent)
            DB.Execute Sqlcontent
            WS.CommitTrans
            MsgBox ("Department Added!")

        ElseIf DNUMBER <> " " And DNAME = " " And DLOCATION = " " Then
            Sqlinsert = " insert into Department(DNUMBER) "
            Sqlvalues1 = " values (" & DNUMBER & " )"
            Sqlcontent = Sqlinsert & Sqlvalues1
            MsgBox (Sqlcontent)
            DB.Execute Sqlcontent
            WS.CommitTrans
            MsgBox ("Department Added!")

    End If

Else

    Sqlcontent = "SELECT COUNT(*) FROM Employee WHERE SSN =' " & DMGRSSN &
" , "
    Set RS = DB.OpenRecordset(Sqlcontent, dbOpenDynaset)

    If RS.Fields(0).Value = 0 Then
        WS.Rollback
        MsgBox ("Department not Added!")

    Else

```

```

    Sqlinsert = " insert into Department "
    Sqlvalues1 = " values (" & DNUMBER & "," & DNAME
    Sqlvalues2 = "," & DLOCATION & "," & DMGRSSN & ")"
    Sqlcontent = Sqlinsert & Sqlvalues1 & Sqlvalues2
    MsgBox (Sqlcontent)
    DB.Execute Sqlcontent
    WS.CommitTrans
    MsgBox ("Department Added!")
End If
End If
Else
    WS.Rollback
    MsgBox ("Duplicated DNUMBER!")

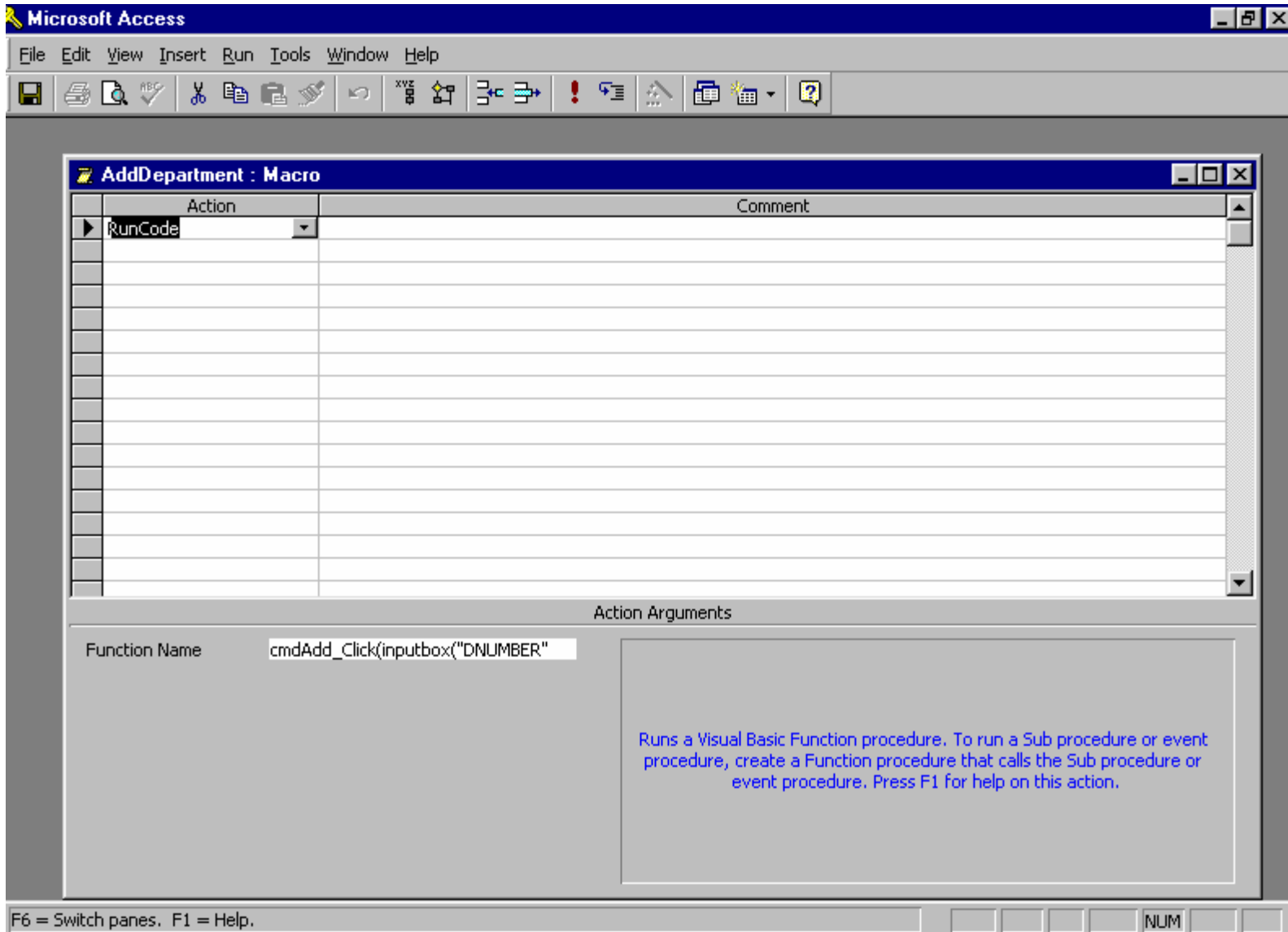
End If

End Function

*****/

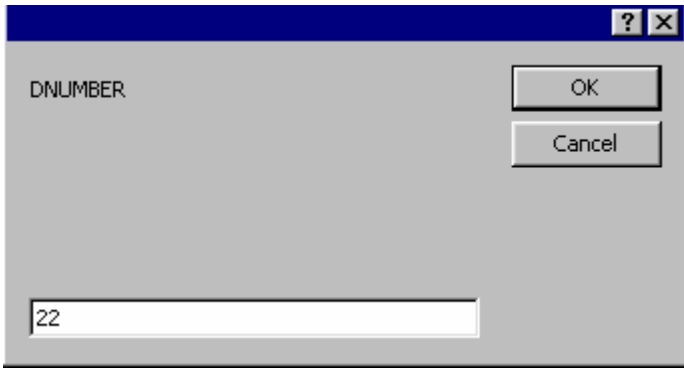
```

One way that this function can be executed is through the use of a macro that executes the function with a sequence of inPutBox\$ builtin methods supplying the input variables. A macro is created similar to creating other object instances in Access by clicking on the Macro tab and then clicking the New command button.



The dialog which follows is the result of adding a department entry without already having a record for the department's manager.

Dialog box titled "DNUMBER" with a text input field containing "22" and "OK" and "Cancel" buttons.



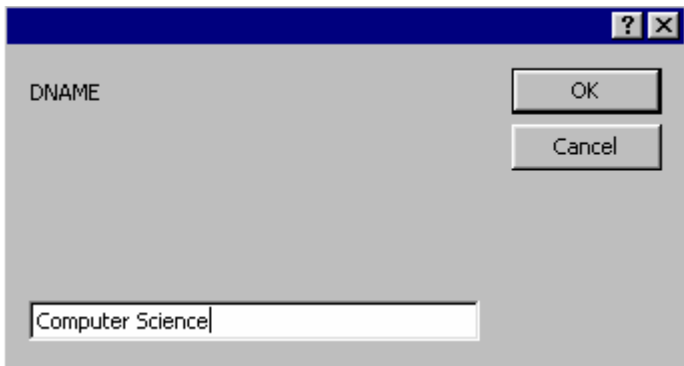
DNUMBER

22

OK

Cancel

Dialog box titled "DNAME" with a text input field containing "Computer Science" and "OK" and "Cancel" buttons.



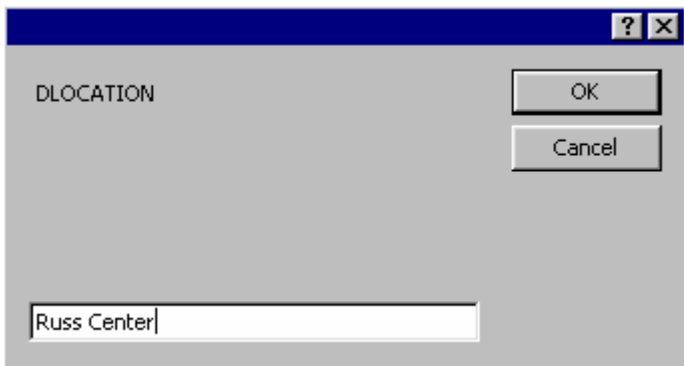
DNAME

Computer Science

OK

Cancel

Dialog box titled "DLOCATION" with a text input field containing "Russ Center" and "OK" and "Cancel" buttons.

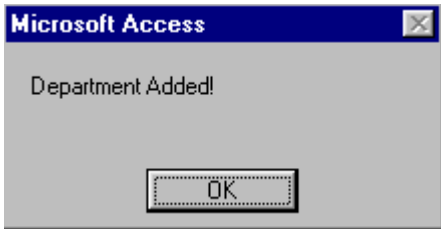
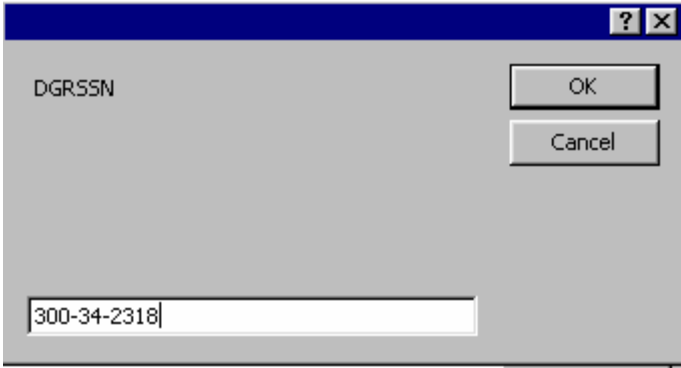


DLOCATION

Russ Center

OK

Cancel



The access user is invited to review the code incorporated to recognize the unites of work described.

Microsoft Access - [Table1 : Table]

File Edit View Insert Tools Window Help

Field Name	Data Type	Description
SSN	Text	
FNAME	Text	
MI	Text	
LNAME	Text	
BDATE	Date/Time	
DNO	Number	
SALARY	Currency	
SEX	Text	

Field Properties

General	Lookup
Field Size	9
Format	
Input Mask	000\-\00\-\0000;;
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	No
Indexed	Yes (No Duplicates)

The data type determines the kind of values that users can store in the field. Press F1 for help on data types.

Design view. F6 = Switch panes. F1 = Help.

NUM