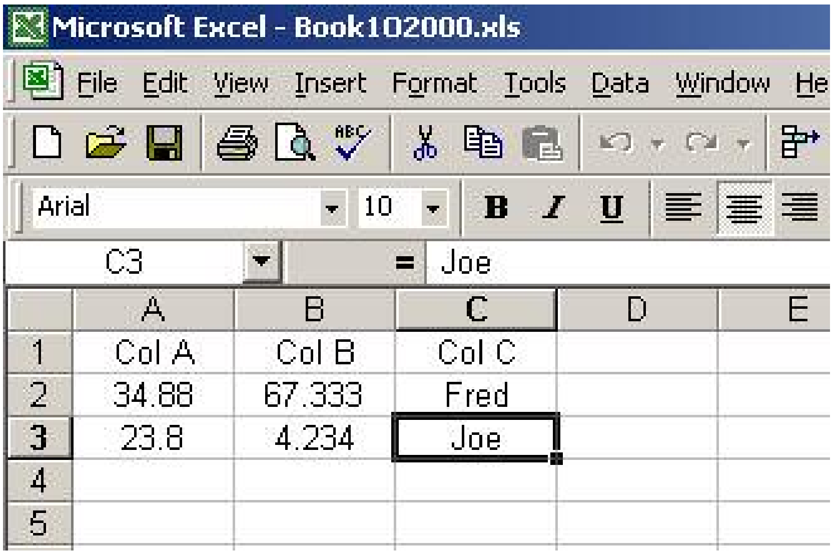
Connecting an Excel Spreadsheet to Delphi using ADO

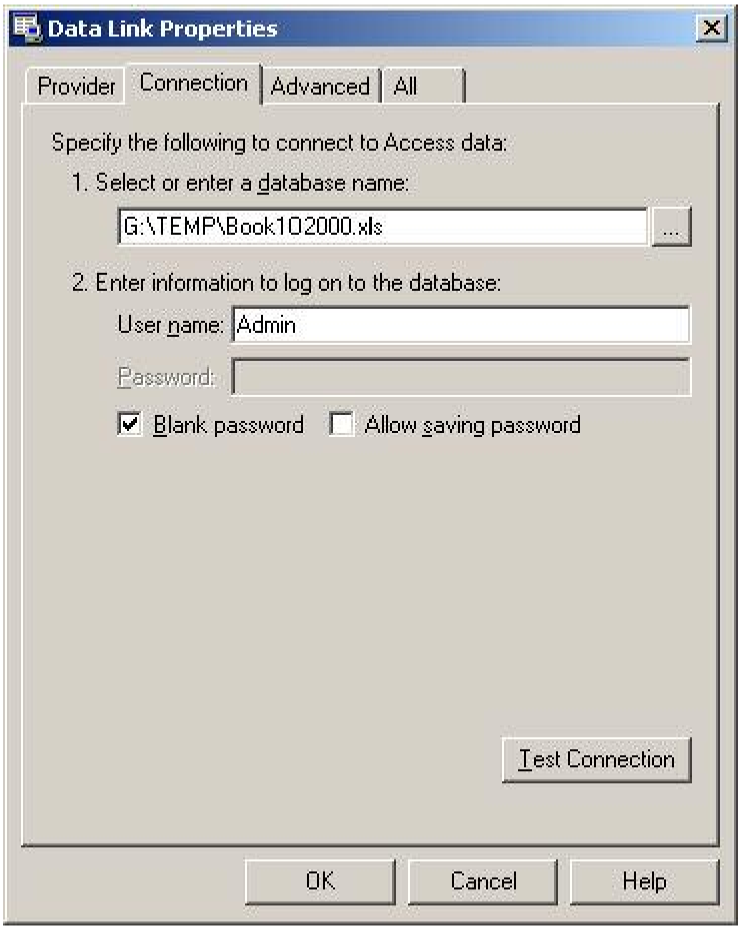
Just as Delphi can connect to MS-Access tables using ADO components, it can use the same method to connect to an Excel Spreadsheet. The following example uses an Excel workbook created in Office 2000; the method has been confirmed with Office2003 also:



C3 = Joe

	A	B	C	D	E
1	Col A	Col B	Col C		
2	34.88	67.333	Fred		
3	23.8	4.234	Joe		
4					
5					

Set up a Delphi Project, in its own folder, and add an ADOTable component. Build its ConnectionString by choosing Microsoft Jet 4.0 OLE DB Provider. Click Next, and browse to your spreadsheet. You will need to change the file filter to "All files" otherwise only .mdb files will be shown. If you test the connection at this stage, it will fail (it should have succeeded for an Access database at this point). You also need to enter an Extended Property to the ConnectionString. To do this, click on the "All" tab:



Specify the following to connect to Access data:

1. Select or enter a database name:

G:\TEMP\Book102000.xls

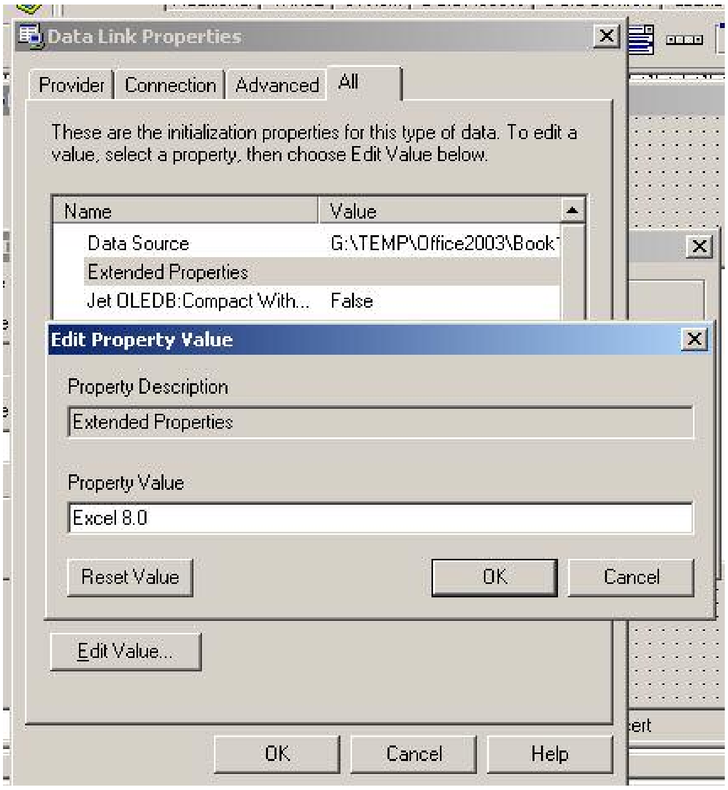
2. Enter information to log on to the database:

User name: Admin

Password:

☑ Blank password ☐ Allow saving password

Select Extended Properties, click Edit Value, and enter "Excel 8.0" (without the quotes).



These are the initialization properties for this type of data. To edit a value, select a property, then choose Edit Value below.

Name	Value
Data Source	G:\TEMP\Office2003\Book`
Extended Properties	
Jet OLEDB:Compact With...	False

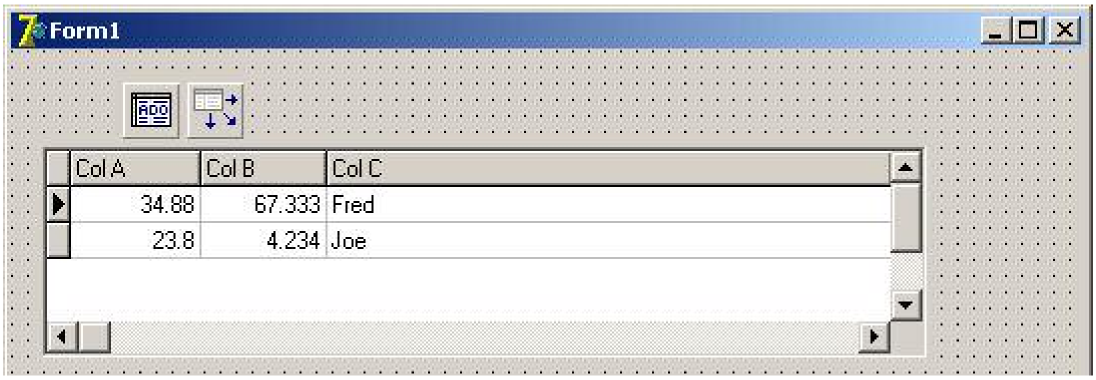
Edit Property Value

Property Description: Extended Properties

Property Value: Excel 8.0

Click OK (twice) to take you back to the Build ConnectionString window again. Click Build again, and this time if you Test the Connection, it should succeed. Click OK (again twice) to close the windows and return to ADOTable1's Object Inspector. If now you select the TableName property, you should find the worksheets listed. Select the appropriate sheet. Then make the TableDirect property True. (The default here is False, which means that Delphi will attempt to access the Table using SQL which it creates in the background. Setting TableDirect to True means that Delphi will access the Table by its name, where the provider is responsible for generating any query code). Then make ADOTable1's Active property True.

Add a DataSource component (DataSet = ADOTable1), and a DBGrid component (DataSource = DataSource1), at which time the spreadsheet should be displayed (in design mode).



Col A	Col B	Col C
34.88	67.333	Fred
23.8	4.234	Joe

If you want to be able to open the spreadsheet at run time, and even to make alterations to it, you may need to set ADOTable1's Active property to False at design time, then set it to True at run time. (This may resolve a possible error whereby Delphi is unable to open the spreadsheet when the program is compiled). You can add a DBNavigator if you wish, and post changes back to the spreadsheet. Note that Delphi may assume that Row 1 of the sheet contains column headings (as in this example), with the data in subsequent rows.