#13 Importing and Exporting Comma Delimited Text Files

Written by Ron Dell'Aquila October, 1987

Importing and Exporting comma delimited text files with 4th Dimension.

When working with 4th Dimension, it is sometimes necessary to import previously entered information from other data base systems. Even though 4th Dimension has extensive import and export facilities, importing information from an MS/DOS database can be a trying experience due to the additional line feeds and quote delimiters.

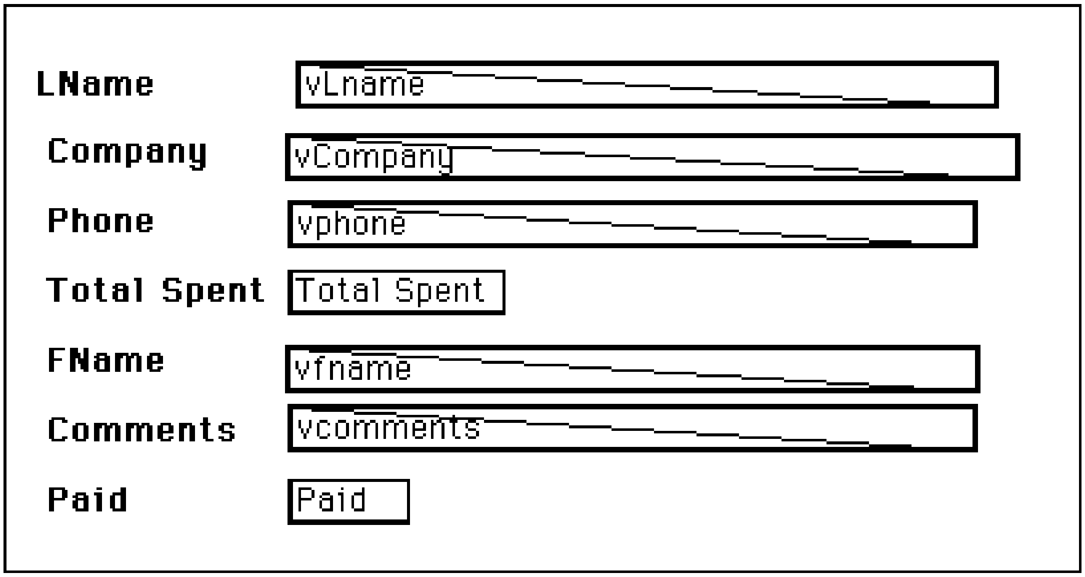
Importing Comma Delimited text files

4th Dimension uses the current input layout when importing data. Any input procedures present will be executed automatically for each record coming through the layout during import.

In this example, we will import a text file where each record has been created in the following manner:

"LastName","Company","Phone",TotalSpent,"FirstName",PaidAmt,"Comments"<CR><LF>

This is a typical comma / carriage return file structure. Note the TotalSpent and the PaidAmt numeric fields do not have the surrounding quote delimiters, this is the way numerics ae represented in thei kind of a text file.. Each record ends with a Carriage Return followed by a Line Feed. Our job is to eliminate the quotes, the Carriage Return and Line Feed. Since the numeric fields do not have any surrounding quotes, quote filtering will not be necessary on these fields.



LName	vLname
Company	vCompany
Phone	vphone
Total Spent	Total Spent
FName	vfname
Comments	vcomments
Paid	Paid

Input layout used for importing and exporting text

In the above input layout, please note the non numeric fields are represented by variables and the numeric fields are the actual fields from the 4d data file, because the surrounding quotes on the string fields count as characters upon import, we need to insure the actual data will not be cut off during import. For example, if we wish to import a 10 character field into a 10 character alpha field, the initial import will loose the last 2 characters (the last actual character and the last quote).

Yes, it's ok to import information using a layout with *both* fields *and* variables accepting the data.

Enter a layout procedure to strip the surrounding quotes from the text fields:

```
        `layout Procedure
LName:=Strip (vLName)                'call Strip function and pass the contents of the variable
Company:=Strip (vCompany)
Phone:=Strip (vPhone)
FName:=Strip (vFName)
Comments:=Strip (Comments)
```

By passing the contents of the variable to the function, we are able to strip any quotes with a simple two step procedure. Create a Function or global procedure called "Strip":

```
        `global proc: Strip
$0:=Substring($1;2;32000)                              `data after 1st quote
$0:=Substring($0;1;Position(Char(34);$0)-1)    `data up to last quote-1
```

Next, in the User environment, set the Field Delimiter to 44 (comma) and the Record Delimiter to 10 (line feed).

Import the data. Couldn't be simpler.

Exporting Comma Delimited text files

Sometimes it is necessary to export your data file into a comma / carriage return / line feed text file. The following layout and global procedure demonstrate the basic methodology.

4th Dimension exports data through an Output layout. The output layout's procedure will govern the conversion of the actual field data into variables on the output layout. The layout variables will contain the necessary quote delimiters surrounding all strings. Numerics do not need the surrounding quotes. You must include the variables containing the numeric fields, in the above example, vTotalSpent and vPaid are the numeric variables

Create a copy of the above layout, and enter the following lines as it's layout procedure:

```
DQ:=Char(34)
vLName:=DQ+LName+DQ
vCompany:=DQ+Company+DQ
vPhone:=DQ+Phone+DQ
vFName:=DQ+FName+DQ
vComments:=DQ+Comments+DQ
vPaid:=String(Paid)+Char(13)
```

Note the last field (Paid) is a numeric type, and does not need any surrounding quotes, however, as the last field, it must be followed by a carriage return (Char(13)), thus the conversion to string because of the concatenation.

In the user environment, under the File menu, select the Export Data... item. Change the delimiters to: End of field:

44 (comma) and End of Record:10 (line feed). Export the data.