

# Create a Scrollable History Field

WILLIAM HALPERN

Some applications benefit by having historical data from multiple records in a single record. Setting up your database to perform the feat can be tricky, but the rewards make it worth the effort.

**SUPPOSE** while viewing a customer's record you'd like to see his or her order history. You can always change to the orders database, retrieve the records, and look them over in a screen report or custom table view. This takes time and keystrokes, though, and it can be particularly inconvenient when the customer is on the phone and you need to see a summary of the orders right now. Using Q&A 4.0's expanded field capabilities, posting, and a few formatting tricks, you can make that history available in the customer's record (see **Figure 1**). The technique is adaptable to accommodate other, similar application needs as well.

In the multiline History field in **Figure 1**, the most recent order is added to the top, and the older orders are moved down a line. A field can hold up to 32K, so you'll be able to store hundreds of lines of history before performing any maintenance.

## Redesigning the database

To add this feature to your database, type the column headings as fixed text on the line just above where you'll start the History field. Add the multiline, labelless History field to your form using the "<" and ">" angle brackets to make it rectangular, and size it to as many lines as you'd like to see without having to press F6 to scroll it. In **Figure 1**, the field has six lines.

You'll also need to add several fields to receive the information posted from the orders database. You can place these out of the way on a back

*Continues on page 4*

— Customer Information —			
Cust No: 123456	Company: Acme Widget Company		
Contact: James Smith			
Phone: (215) 555-2345	Ext: 111		
Addr1: 100 Commercial Way			
Addr2: Bldg 101 Mail Stop 3			
City: Philadelphia	State: PA	Zip: 19111-0123	
— Sales History — (Press F6 to Expand)			
Order No	Description	Qty	Date
1234	Gold Watch	1000	01/07/1996
1214	Brass Watch	800	11/09/1995
1101	Diam Bracelet	250	10/21/1995
1034	Gold Watch	500	06/01/1995
1001	Gold Ring	700	01/25/1995
961	Gold Ring	200	12/01/1994

**Figure 1.** A customer database with a multiline, scrollable History field.

# The Quick Answer™

The Independent Guide to Q&A Expertise

April 1996

Volume 7, Number 4

- 1 Create a Scrollable History Field  
*William Halpern*
- 2 README.1ST  
*Tom Marcellus*
- 3 The Quick Answer 1995 Monthly Index by Title
- 4 QuickTip: Don't Run DEFRAG via @SHELL  
*Peter Venuto*
- 5 Complex Report Calculations? Let the Database Do the Work  
*Laurie Slatin*
- 6 QuickTip: When One Record Won't Merge Print  
*Tom Marcellus*
- 7 @Help  
*Dave Reid*
- 8 QuickTip: Single-Key Pick-List Control  
*Tom Marcellus*
- 8 QuickTip: Print Ordinal Dates  
*Alec Mulvey*
- 9 Easy-to-Manage, Easy-to-Use Pick-Lists  
*Michael Bell*
- 10 QuickTip: Conditionally Suppress Report Calculations  
*Peter Haskett*
- 11 And the Winner Is ...
- 13 Manage Custom Pick-Lists on a Network  
*Tom Marcellus*

# README.1ST

TOM MARCELLUS, EDITOR

**A**s we approach our sixth anniversary, we're giving *The Quick Answer* a face-lift. For us, it represents an affirmation not only of our continuing commitment to explore all that Q&A has to offer, but also because the software industry has had ample time to come up with a product that does a better job of meeting our data management needs with less work, and hasn't. We hope you like the new look.

Remember the contest we announced in the February issue? Plenty of entries came in during March, and we have a winner. See page 11.

Pick-lists. You can't overemphasize their value. Not only do they help you control and standardize the information in your databases, they turn data entry into a simple point-and-shoot affair. The problem, though, is that you can't update your lists as you go along. You have to stop what you're doing and shut down the database while you add, delete, or edit selections. Or do you? We show you two powerful techniques that keep you out of the Program Spec no matter how often or drastically your pick-lists change. **Mike Bell** stores an entire application's pick-lists in a single external record and, with a few lines of code, lets you update any list on the fly. My approach takes more work to implement, but it lets you grant pick-list editing privileges on a field-by-field and user-by-user basis.

The usefulness of some applications increases markedly when they're designed to store historical data from multiple external records. Adding the capability to an existing database takes some doing, but the rewards make it well worth the effort. **Bill Halpern** guides you through the process.

I like to program, but I can't say I enjoy creating elaborate derived report columns. When you're confronted with a report requiring painfully complex calculations, you might be able to simplify the report by pre-processing them in the database. **Laurie Slatin** shows you how she does it.

Have you tried merge printing in Q&A 5.0 only to have your system crash? In this month's @Help column, **Dave Reid** describes the bug that causes it, and the workaround. **QA**

## Floppy Confusion Driving You Batty?

If you're tearing your hair out shuffling floppies, get the **ProCataloger** database for Q&A 5.0.

Keeping track of floppies couldn't be easier. Pop one into the A: or B: drive, hit Enter, and the **ProCataloger** copies its file details into the database record. Add a title, a category—search words and comments, if you like—then you can quickly search them by filename, or by partial filename, and the **ProCataloger** will tell you which disk you need. The customizable **ProCataloger** can also auto-number your disks, print labels, and perform other actions you define.

End hard disk clutter. Move rarely-used documents, databases, and other files onto floppies and into your **ProCataloger**. Whether you can dance or not, you'll never get caught doing the floppy shuffle again.

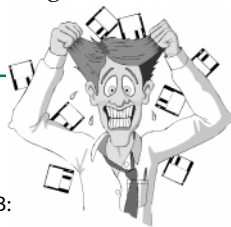
### **ProCataloger Floppy Disk Tracker for Q&A 5.0**

\$49 (\$29 for *The Quick Answer* subscribers—save 40%) + \$2 shipping

Checks, major credit cards welcomed

Marble Publications • PO Box 9034 • Gaithersburg, MD 20898

800-780-5474 • Fax 301-424-1658



# The Quick Answer™

The Independent Guide to Q&A™ Expertise

Editor Tom Marcellus  
Publisher Michael Bell  
Copy Editor Laurie Moloney  
Production Editor Paul Gould

*The Quick Answer* (ISSN 1052-3820) is published monthly (12 times per year) by Marble Publications, Inc., 9717 Delamere Ct., Rockville, MD 20850.

Cost of domestic subscriptions: 12 issues, \$79; 24 issues, \$142. Outside the U.S.: 12 issues, \$99; 24 issues, \$172. Single copy price: \$10; outside the U.S., \$12.50. All funds must be in U.S. currency. Back issues are available upon request, for the same price as a single copy.

Second-class postage paid at Rockville, MD.  
POSTMASTER: Send address changes to  
*The Quick Answer*, PO Box 9034, Gaithersburg,  
MD 20898-9034.

Copyright © 1996 by Marble Publications, Inc. All rights reserved. No part of this periodical may be used or reproduced in any fashion whatsoever (except in the case of brief quotations embodied in critical articles and reviews) without the prior written consent of Marble Publications, Inc.

Address editorial correspondence, @HELP questions, or requests for special permission to: Marble Publications, Inc., *The Quick Answer*, PO Box 9034, Gaithersburg, MD 20898-9034. Phone 800-780-5474 or 301-424-1658. Fax 301-424-1658. CompuServe 73370,1575. Prodigy NEPY97A.

For Q&A technical support, call Symantec: 503-465-8600.

Q&A is a trademark of Symantec Corp. Other brand and product names are trademarks or registered trademarks of their respective holders.

This publication is intended as a general guide. It covers a highly technical and complex subject and should not be used for making decisions concerning specific products or applications. This publication is sold as is, without warranty of any kind, either express or implied, respecting the contents of this publication, including but not limited to implied warranties for the publication, quality, performance, merchantability, or fitness for any particular purpose. Marble Publications, Inc., shall not be liable to the purchaser or any other person or entity with respect to any liability, loss, or damage caused or alleged to be caused directly or indirectly by this publication. Articles published in *The Quick Answer* do not necessarily reflect the viewpoint of Marble Publications, Inc.

# The Quick Answer 1995 Monthly Index by Title

(Includes articles, tips, and @Help questions)

## January 1995

Any Comments? (Storing free-form comments in database), 9  
Automate Your Check Writing and Printing, 1  
Calculate a Discount, 10  
Calculate Future Dates, 7  
Dedicated Key Field Not Needed for XLookups, 13  
Handle QABACKUP Restore Failures, 16  
How to Restrict a Field Entry's Length, 12  
Make Custom Menus More Reliable, 10  
Matching Values Between Databases with @Num, 9  
Search for Irregular Values, 14  
Use Caution with @Add Context Function, 16  
Use Real Tabs to Align Merge Document Columns, 11  
Who Has a Birthday this Month?, 11  
Work Around Field Naming Restrictions, 10

## February 1995

Create a Database to Automate Address Printing, 7  
Master Data-Entry Form Layout, 11  
More on Physically Sorting a Database, 8  
My Fonts Aren't Printing!, 7  
Position Fields Precisely, 13  
Print Labels One at a Time on Your Laser Printer, 1  
Repeat Duplicate Values in Reports, 8  
Retrieve Records Where Two Address Fields Differ, 6  
Search and Replace Fonts, 7  
Simplify and Reduce Programming with Indirection, 14  
Tweak Macro Files to Run "Impossible" Procedures, 9

## March 1995

Create a Custom Investment Tracker, 3  
Date- and Time-Stamp Your Reports, 10  
Design a Time Performance Report, 11  
Find Records with Two Matching Values, 10  
Get More Font Power, 9  
No Password Protection for Documents, 10  
Perform Conditional XLookups, 10  
Protect Fields from Changes, 14  
Q&A for DOS Font Solutions and Your Laser Printer, 1

## April 1995

Add a "Secret" Field to Your Form, 16  
Automate Cursor Navigation, 8  
Converting Documents to WordPerfect, 8  
Create an Employee Scheduling System, 10  
Cut and Paste with the Mouse, 15  
Drawing Straight Lines, 6  
Fix Blank Lines in Address Blocks, 8  
Follow the Dotted Line to Improved Report Layouts, 1  
Get Better Screen Prints, 15  
Lengthen, Shorten, or Expand a Field, 15  
Print a Q&A for DOS Screen in Windows, 13  
Print One Page of a Lengthy Report, 7  
Save Time When Updating Related Records, 6  
Search for Non-Dates in Date Fields, 7  
See Your Forms for a Change, 9  
Solve XLookup Problem, 8  
Suppress Report Columns, 7

## May 1995

Change Text Attributes, 14  
Create and Print Invoices with Unlimited Line Items, 1  
Determine Future Dates, 15  
Double-Sided Printing, 8  
Fix Broken Line Draw Boxes, 8  
Fix Label Wrap Problem, 7  
Improve Search Performance, 7  
Mail Merge Program Not Executing, 8  
Moving to Q&A for Windows, 9  
Multiple Top Margins, 8  
Run a Report and Return to the Same Record, 14  
Use If-Then Logic with XLookups, 7  
Use Up Your Label Sheets, 16

## June 1995

Avoid Risk to Your Macro File, 10  
Better Envelope Printing, 10  
Change Your Default Font in Q&A Write, 10  
Combine Fields on a New Line, 8  
Control Your Alternate Programs, 9

Easy-to-Use Utilities Make Your Mail Look Better and Travel Faster, 11  
Loops, Arrays, and Bubble Sorting, 1  
Sort Reports by Derived Values, 8  
Subtotal Time Values, 9

## July 1995

A Clipboard for Q&A for DOS, 9  
Add Comments to Your Macro Files, 16  
Check for Overdue Accounts, 5  
Create a Documents Catalog, 1  
Create a Price Analysis Calculator, 14  
Delete Q&A Write Files, 5  
Fix Mysterious Corruption, 5  
Maintain Accurate History Records, 6  
Print Your Data in Professional-Looking Forms, 7  
Quick Delete During Data Entry, 15  
Separate Date Components, 13  
Update Imported Records, 5  
Use Posting to Maintain Transaction Histories, 10  
Using XLookups with Multiple Search Criteria, 11

## August 1995

Creeping Envelope Solutions, 16  
Mark Modified Records, 15  
Network Printing Problems, 15  
Organize Your Report and Macro Lists, 14  
Q&A 5.0 for DOS—A Review, 1  
Save Merge Specs, 15  
Use the Same Database in DOS and Windows, 15

## September 1995

Conditionally Calculate a Running Total, 12  
Highlight Negative Numbers, 13  
Make Macros Work Universally, 11  
Mark Records for Previous or Following Month, 10  
Posting in Q&A for Windows?, 5  
Print Merge Documents from Multiple Databases, 14  
Secrets of Running Q&A for DOS in Windows, 6  
Simple Programs Speed Data Entry, 11  
Speedy Retrieval Paradox, 12  
Spell Check Database Records, 13  
Swapping Macro Files, 12  
Useful Tricks with Posted Text, 1  
Using Documents in Databases, 5

## October 1995

Alternate F11 and F12 Clipboard Keys, 13  
Building Multiple Database Applications, 7  
Convert Money Values to Text, 14  
Create a Field Navigation Menu, 6  
Database Copying Made Easier, 1  
Enhance Posting and Mass Updates with Programming Options, 5  
Honey, Where's My Underline?, 13  
How to Check for Lengthy Names (mail labels), 13  
Manage Documents with a Database, 13  
Storing Pictures in Your Database: What'll It Cost You?, 15  
Troubleshoot and Modify Your Q&A Database Faster with DTFSPESCS, 11  
Use the Navigation Spec to Boost Program Functionality, 4

## November 1995

Auto-Jump to a Document, 11  
Create Versatile Pick-Lists That Respond to User Preferences, 12  
Ditto Your Data During Search/Updates, 1  
Division Error Unearthed in Q&A 4.0, 9  
Easier Multiple Date Searches, 9  
How to Lookup and Display Information from Multiple Records, 5  
How to Print 1099 and W2 Forms, 11  
Taking Yes and No Beyond "Yes" or "No," 10  
(Too) Many Unhappy RETURNS, 9  
When Speedy-Unique Fields Get In the Way, 11

## December 1995

@Mod in Action, 11  
Create and Update Your Pick-Lists in Write, 5  
Create Reports with Graphic Lines and Boxes, 13  
Faster Program Spec Saves in DOS, 8  
Force a Selected Entry, 9  
Handier Than the Average Macro Hot Key, 12  
How to Get Faster, More Accurate Data Entry, 1  
Invalid Programming Messages, 9  
Make @Macro the Last Command, 12  
Print Group Subtotals in a Separate Column, 6  
Quickly Find Files in DOS, 10  
Removing Leftover Files, 9  
Simpler Birthdate Program, 7  
The Case of the Disappearing Data, 9  
The Case of the Missing @Msg, 8

QA

# Scrollable History Field ...

Continued from page 1

page, and make them read-only:

Field Name	Format	Field ID
Order No	Text	#200
Description	Text	#400
Qty	Number	#300
Order Date	Date	#115

## Adding the program

The trick is to make the data in the multiline History field align with the fixed column headings, and place the latest order information at the top. If you do this correctly, you'll also be able to print the History field's data or include it in reports in its native format. The secret is in the History field's (#205) program, shown in **Listing 1**.

**Listing 1.** Program for the multiline History field.

```
< #205:
If #200 <> "" then {
#205 = @Lt(@Str(#200) + @Txt(10," "),10) +
@Lt(@Str(#400) + @Txt(21," "),21) +
@Rt(@Txt(11," ") + @Str(#300),11) + "  "+
@Str(@Mid(#115,6,3) +
@Rt(#115,2) + "/" + @Lt(#115,4)) + "
" + #205; Clear(#200,#115,#300,#400)}
```

The If statement in conjunction with the final Clear command makes sure that a line of order summary information is added to the History field just once. The next part of the program allows a one- to 10-digit order number and ensures that the next portion of the line, the description, begins at position 11.

The programming for the description reserves the next 21 characters of the field and left-aligns the data.

To align the quantity (a number) under its column heading, the program uses the @Right function and spaces.

Note the use of the @Str function to create one line of text from different data types (in this case numbers, dates, and text).

The last column contains order dates. Because Q&A can't copy date values to text fields, @Str is used to

convert them to text. Q&A stores dates in the YYYY/MM/DD format no matter how they appear on-screen. The program changes them to a more readable MM/DD/YYYY format.

Now, here's a very important part of the program. Notice that the second to last line ends with a quote mark, and the last line begins with one. While typing the program, be sure to press Enter immediately after you type the first (opening) quote mark, then type the second (closing) quote mark at the beginning of the next line. This makes the program execute a hard carriage return, creating a blank line so the most recent data can be slipped in "on top of" the old.

Because the old data also includes a hard carriage return at the end of each line, the columnar formatting is retained even when you're scrolling the field with the editor open.

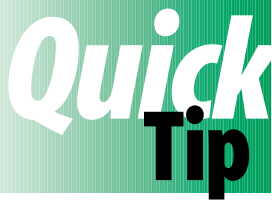
When you've finished typing the program, close the field, press F8, and type 205 (don't include the "#") on the *On Exit Field ID* line. This way, you can execute it during posting.

## Posting

The final step is to create the Batch Posting Spec to post the data from the order records to the matching customer records' hidden fields. When you've filled in your Batch Posting Spec, press F8 for the Auto Program Recalc screen, highlight *Yes* on the *On Record Exit Statements* line, then press F9 to return to the Posting spec. Save the Spec by pressing Shift-F8 and assigning a name to it. The programming calculation options are saved with the Spec, so when you run the post, Q&A will execute field #205's program on the record, updating the History field.

That's all there is to it. If after posting you find that the fixed column headings don't align with the columns of data in the History field below them, simply redesign the form and re-space the headings as necessary. **QA**

Bill Halpern is executive vice president of Professional Computer Technology Associates in Newtown, Pennsylvania. Bill has been designing and installing Q&A business applications for the past eight years. 215-598-8440, CompuServe 71023,356.



## Don't Run DEFRAG via @Shell

To take advantage of Q&A's new @Shell command, I designed a small database and programmed it with @Shell to invoke a variety of DOS commands, including DEFRAG, which defragments and

reorganizes a drive's files to optimize disk performance. I found out the hard way that you should never run this utility from a database that's stored on the drive you're defragmenting. I lost

data, field formats, and programming. **QA**

Peter Venuto, Santa Clara, California

*[Lesson learned. Never run a defragmentation utility over an open file. Also, experience has taught me to run Recover on my Q&A databases before defragmenting the disk where they're stored, and to back up a database before recovering it. —Ed.]*



# Complex Report Calculations? Let the Database Do the Work

LAURIE SLATIN

When prospects for completing that complex report design are discouraging, look at "pre-processing" the calculations in the database.



A client, a local trucking firm, uses Q&A 4.0 to track the number of hours employees have driven. By law, the firm is required to prepare a monthly report of the hours each driver spends on the road. For each driver, and each day of the month, the report must show the number of hours worked that day, along with the cumulative hours worked that day plus the previous seven calendar days.

Rather than risk a nervous breakdown trying to create a complex derived column report to calculate this information, I decided to let the database do the work. Depending on the kind of report calculations you need, you might be able to do it, too.

Figure 1 shows the form design for the TRUCKLOG.DTF database along with the formatting codes for the fields. It's a simple database that I encourage you to create so you can get some hands-on experience with this technique.

When you've designed the database and formatted the fields, make the Key field speedy and unique. From the File menu, select Customize File / Speed up Searches. At the Speed Up Spec, type SU in the Key field, and press F10 to save the Spec.

Next, set all the fields in the programming use only section (Key, NDays, Last8Days, OnExit, and Sub) to read-only. From the File menu, select Design a File / Program a File / Read-only Fields. At the Read-Only Spec, type an "R" in each of these five fields, and press F10 to save the Spec.

## Programming the database

The database requires a small amount of programming in the Key, OnExit, and Sub fields. From the Programming menu, select Program Form. Enter the programming for the three fields as shown in Listing 1. When you've finished, press F8, set the On Exit field ID to 20, and press F10 to save the Spec.

Listing 1. Program for TRUCKLOG.DTF.

```
Key: #10 = @Str(Truck) + ":" + @Str(Date)

OnExit: < #20: NDays = 1;
        Last8Days = Hours;
        GOSUB #30

Sub: < #30: Last8Days = Last8Days +
      @XLookup(@Filename, @Str(Truck) + ":" +
      @Str(Date - NDays), "Key", "Hours");
      NDays = NDays + 1
      If NDays < 8 Then GoSub #30 Else STOP
      RETURN
```

Open the database in Add Data mode, then press Shift-F8 to display the Calculation Options box. Set the Calculation Mode to Automatic, and the Main Program Before Exit option to No.

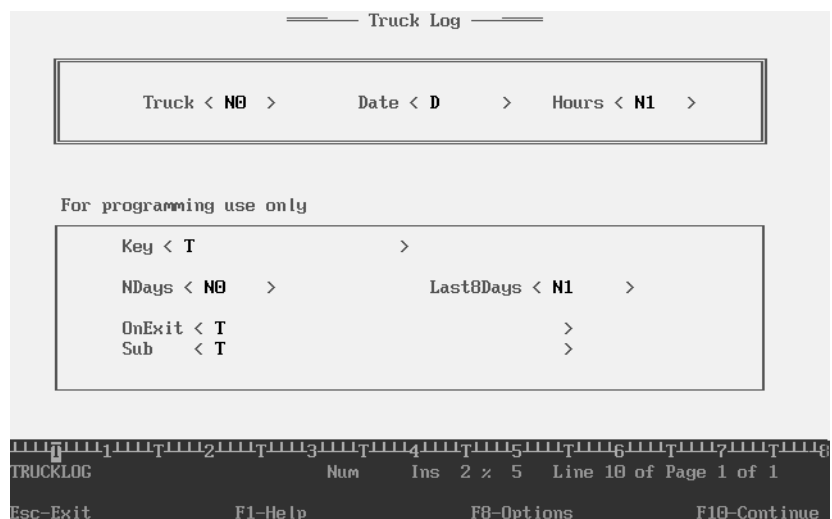


Figure 1. TRUCKLOG.DTF's form design and field formatting codes. Drivers are identified their truck number.

## How the program works

The operator enters the truck number (which identifies the driver), the date, and the hours driven that day. The program sets the Key field to the truck number plus a colon and the date. For example, if the truck number is 100 and the date is 2/10/96, the Key field is set to 100:1996/02/10. The Key field's value is automatically calculated when the Truck and Date fields are filled.

After entering the hours worked and pressing F10 to save, the OnExit field's program is triggered, initializing the Ndays counter field by setting it to 1, and setting the Last8Days field to the current number of hours worked. For example, if the driver worked five hours on 2/10/96, the fields would be set as follows:

```
Ndays: 1
Last8Days: 5
```

Once the Ndays field is initialized, the Sub field's subroutine is called, looking up the number of hours worked for the previous day. It does this by checking for a record for the same truck number, one day earlier. In this example, it would look for a Key value of 100:1996/02/09. The program then adds the number of hours worked that day (let's say it was six) to the number of hours already in Last8Days, then increments Ndays by one. In this example, if you were to freeze the subroutine at this point, the fields would look like this:

```
Ndays: 2
Last8Days: 11
```

The subroutine then checks that the counter field, Ndays, is less than 8. If it is, the subroutine is recalled, adding the hours from two days ago to the Last8Days field. The loop continues until Ndays reaches 8 and the subroutine is stopped.

## The report

Because the calculations are performed in the database and the number of hours worked during the last eight calendar days is already in the Last8Days field of each record, the report is simplicity itself. From the Report menu, select Design/Redesign a Report, and assign a name to the report. Leave the Retrieve Spec blank, and fill out the Column/Sort Spec as shown in [Table 1](#).

**Table 1.** Column/Sort Spec codes for the Truck Log report.

Field	Codes
Truck	10,AS
Date	20,AS,CS
Hours	30,ST,T
Last8Days	40,H(Last 8!Days)

Save the report design. After entering each month's data, you can print the report using Make Temporary Changes, and change your Retrieve Spec to whatever range of records you want to include. [Figure 2](#) shows a sample report for two trucks over a small date range.

Trucking Log Report				Page 1
Printed on 2/18/96				
Truck	Date	Hours	Last 8 Days	
24	2/8/96	5.5	5.5	
	2/9/96	5.0	10.5	
	2/10/96	0.0	10.5	
	2/11/96	7.0	17.5	
	2/12/96	10.0	27.5	
	2/13/96	13.0	40.5	
	2/14/96	4.5	45.0	
	2/15/96	5.9	50.9	
	2/17/96	10.0	50.4	
	2/18/96	6.0	56.4	
Total:		66.9		
31	2/8/96	3.2	3.2	
	2/9/96	7.1	10.3	
	2/10/96	4.3	14.6	
	2/11/96	0.0	14.6	
	2/12/96	6.2	20.8	
	2/13/96	6.6	27.4	
	2/14/96	3.3	30.7	
	2/15/96	5.9	36.6	
	2/17/96	7.0	40.4	
	2/18/96	6.5	39.8	
Total:		50.1		
====	====	====	====	
Total:		117.0		

**Figure 2.** The Trucking Log report for two trucks within a sample date range. The first records in the database were added on 2/8/96.

When a key application element is a complex report, you might find, as I have, that programming the database to perform the calculations in advance is easier than struggling with derived columns. [QA](#)

Laurie Slatin owns PC Database Solutions, a custom programming firm specializing in Q&A and Microsoft Access. 206-283-4155, PO Box 19593, Seattle, WA 98109, CompuServe 72713,1067.



## When One Record Won't Merge Print



If a record (in a multiuser database, particularly) locks up your PC when you attempt to merge print it, chances are that even though the record looks fine, it may be corrupted. If recovering the database or copying the record doesn't help, try

exporting the record's data to a standard ASCII file, then importing it back into the database to create a new record. Delete the original record, and the replacement record should now merge print without a hitch. [QA](#)

Tom Marcellus



DAVE REID

### 5.0 Mail Merge Bug Uncovered

I recently started using Q&A 5.0 for DOS, and I love it. However, I'm having a problem with mail merge. When I use the Retrieve Spec to select the records to merge, my PC locks up. It doesn't happen all the time, and it never happens when I print regular documents. Am I doing something wrong?



Hazel Hill, Scotts Valley, California

There's a problem with the way Q&A 5.0 handles default font assignments during a mail merge. Once you've saved a default Font Assignments screen, the system will crash when you try to print a mail merge document and press F6 at the Retrieve Spec to expand a field. Accordingly, if you do a mail merge and need to expand a Retrieve Spec field, you must delete any existing default Font Assignments. To do this, select Write / Type/Edit, then press Ctrl-F9. Press Shift-F4 to clear the name of the font file, then step through each line that contains an assigned font, and press the spacebar to clear it. When the entire screen is blank, press F8 to make the now blank screen the default for all new documents.

*[As long as you delete the font assignments, you can leave the font file path and name at the top of the screen. It's one less step when you need to assign fonts. Note that the bug doesn't prevent you from using the Font Assignment screen—just don't make one the default for new documents if you plan to create any merge documents.]*

*Lacking a default Font Assignments screen, you can create a macro that assigns the fonts you normally use. Have the macro "press" Ctrl-F9, fill in the Font Assignments screen, then "press" F10 to return to the document.*

*Or, you can create a blank document template with the font assignments plugged in. Use this template for all new documents, saving them to different filenames. To protect the template, you can make it read-only using the DOS ATTRIB command, like this:*

```
ATTRIB +r C:\QA5\DOCS\TEMPLATE.DOC
```

—Ed.]

### External Programs or Alternate Programs

I've noticed a quirk when I run an Alternate Program from the Main menu, then from the External Programs menu in Q&A 5.0. For example, I'd like to be able to run COMMAND.COM from both places, but when I run it from the External Programs menu, it leaves me in the QA5 directory. How can I get the External Programs feature to work like the Alternate Programs feature?



Chris Pillette, Portland, Oregon

You've discovered one of the differences between Main menu Alternate programs and the new External Programs menu. Some programs must be started from their own directory. Q&A is such a program. Even with the Q&A directory in your path statement, you'll receive a *QA1.EXE not found!* error message if you try to start it from a different directory. Q&A starts Alternate programs from the directory containing them, but starts External programs from the QA5 directory. To start an External program from its own directory, use a batch file that changes to the right directory before starting the program. A sample batch file that runs a DOS prompt from the root directory looks like this:

```
cd\  
command
```

You can specify the batch filename on the Command Line of the External Programs screen.

### Where's the Rest of My Screen?

I use Q&A for DOS on an IBM Thinkpad with an active matrix color display. All my Windows applications use the entire screen, but Q&A doesn't. It uses the full width, but leaves about a half inch vertical gap. Since it's a notebook computer, I can't adjust the screen to compensate. How can I get Q&A to use the entire screen?



Bob Shamo, Chicago, Illinois

First, here's a little background. Q&A for DOS runs in character mode rather than graphical mode, and different monitors display information in different ways. Windows

can run character-based DOS programs in a window or in full-screen mode. I prefer full-screen mode, and your notebook will probably yield the best results in that mode. (You can press Alt-Enter to switch between full-screen and windowed modes. Full-screen mode doesn't have the Windows title bar across the top.) It might be that your notebook always displays full-screen DOS programs this way. Try running another DOS program in full screen mode to see. Also, boot without Windows and start Q&A from the DOS prompt. If Q&A uses the full screen, then you're probably not using full-screen mode in Windows. If the gap is still there, then it could be a hardware limitation. **QA**

Dave Reid is a Symantec senior support analyst providing second-level assistance to the technical support representatives. He's also the coauthor of *The Q&A 4.0 Wiley Command Reference*, published by John Wiley and Sons, and works as an independent Q&A consultant. PO Box 12083, Eugene, OR 97440.

Have a nagging question? Send it to **@Help**, *The Quick Answer*, Marble Publications, Inc., PO Box 9034, Gaithersburg, MD 20898-9034 or fax to 301-424-1658. When writing, please include your name, address, and phone number, along with your Q&A version number (and whether DOS or Windows) and a detailed description of the problem. We'll publish questions we feel are of general reader interest; individual responses are not possible.

## Quick Tip

# Single-Key Pick-List Control

5.0

Suppose you want to conditionally display a Userselect pick-list for an optional field. To leave the field empty, you'd like to press the usual Tab or Enter to bypass it. To fill it, you'd like to press

one key to display the list. Here's how. Record a macro that turns the single left quote mark into a hot key, like this:

```
<begdef> ` <name> " ` <vidon> ` <enter> <enddef>
```

This character is seldom used, and on most keyboards it's the leftmost key on the top row, so it's easy to find. (You can use another key or key combination.) When you press it, the macro "types" the single left quote mark then "presses" Enter to leave the field.

Next, place a program like this one in the field:

```
> If @Left(Field name, 1) = "`" Then
  Field name = @Userselect("Item1,Item2,Item3, etc.")
```

You can now press Enter or Tab to bypass the field, or "`" to display the pick-list. If you display the list, then decide to leave the field blank anyway, simply press Esc (or click outside the list). Note that you can also display the pick-list in a filled field (to update it) by making "`" its first character. To type the single left quote mark in a document or program, simply press the key then press Backspace. **QA**

Tom Marcellus

## Quick Tip

# Print Ordinal Dates in Mail Merge Documents

DOS

Suppose your dates print in a 5 April 1996 or April 5, 1996, format, but you'd prefer ordinal dates with the day of the month

represented as 1st, 2nd, 3rd, and so forth. First, add the following information to the merge database's Lookup Table. (The table shown is abbreviated. You must include all 31 rows.):

Key column	Column 1
1	st
2	nd
3	rd
4	th
5 through 20	th
21	st

22	nd
23	rd
24 through 30	th
31	st

Next, insert this expression in your merge document:

```
*Pg { @Str (@Dom (@Da)) + @Lu (@Dom (@Da), 1) + " "
  + @Mt$ (@Da) + " " + @Str (@Yr (@Da)) } *
```

The expression can't exceed 80 characters, but you can modify it for the date format you need. **QA**

Alec Mulvey, Keyword Training & Consultancy, Ascot, England



# Easy-to-Manage, Easy-to-Use Pick-Lists

MICHAEL BELL

You can store all your application's pick-lists in a single external record, and make them editable on-the-fly during data entry.



**I**n the August 1995 *The Quick Answer*, Tom Marcellus reviewed Q&A 5.0's new features and showed how you can greatly enhance an application's functionality by combining them with pre-5.0 commands. Anyone who thinks 5.0 doesn't add much to Q&A hasn't considered the wonderful tricks that can be performed by this merging of old and new.

In this article, I'll show you how to combine 5.0's Userselect command with @XLookup to give you a powerful tool that radically changes the way you use and manage your pick-lists. You'll get the following pick-list capabilities:

- Pick-lists that display only when the target field is empty.
- Pick-lists that let you press Enter to choose "No Selection" and continue, leaving the target field left empty.
- Pick-lists that can be updated on-the-fly.
- All pick-list information for an entire application can be stored in one external database record.
- You can create different or conditional pick-lists within the same application and password protect pick-list usage and management separate from the databases in which the pick-lists are displayed.
- All on-the-fly pick-list updating and management can be

performed without programming or design changes—a big plus in multiuser environments where you can't afford downtime.

To begin, you'll need to create a global list database to store your pick-list selections. This database can store pick-list selections for all the databases in the application.

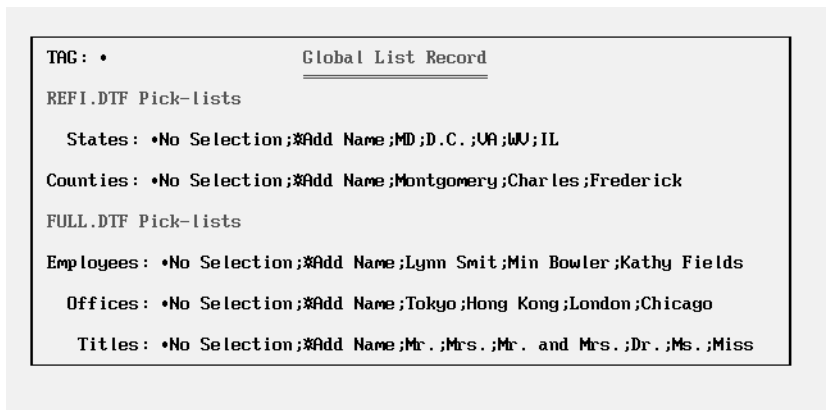
**Figure 1** shows a sample record from such a database containing pick-list information for two databases and five fields. The resulting States field pick-list will look like the one shown in **Figure 2**.

## Adding the program

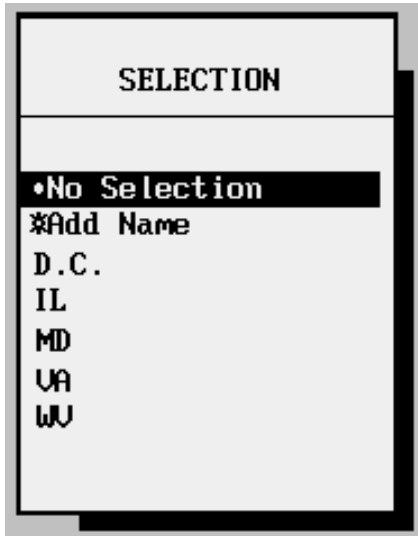
You place the program for each pick-list in the field in the data entry database where you want the list to pop up. Here's a sample program for the States field, followed by an explanation:

```
<#10: If #10 = "" then { #10 = "•";
UserSelect(@XLu("GLOBAL.DTF",#10,"TAG","States"),#10)};
If #10 = "•" or #10 = "•No Selection" then Clear(#10);
If #10 = "•Add Name" then
{#10 = @Shell("C:\QA5\QA.COM -m1");
Clear(#10); goto #10 }
```

Here's how the program works. If Field #10 is empty, a marker (the ASCII 7 bell character) is placed in it and Userselect displays the pick-list selections from the States field in GLOBAL.DTF. If the user doesn't want to make a



**Figure 1.** A database record containing pick-list selections for two databases (REFI.DTF and FULL.DTF) and five fields.



**Figure 2.** Resulting pick-list for the States field.

command temporarily swaps out to a second copy of Q&A and runs the following Alt-1 macro:

selection, he or she simply presses Enter to select the first choice, •No Selection, and the statement on the third line clears the field.

The program's fourth and fifth lines allow the GLOBAL.DTF list to be updated on-the-fly. When the user selects Add Name, the @Shell

```
<begdef><alt1><name>"<caps,>alt1<caps.>"<vidoff><esc>
fs<capsf4>c\qa5files\global.dtf<enter><f10><wait><f10>
<capsf10><esc><esc><esc>x<enddef>
```

The Alt-1 macro retrieves the GLOBAL.DTF record, pauses while the user updates the pertinent list, then resumes when the user presses F10, saving the record, exiting the second copy of Q&A, and returning the user to the original screen.

The final line of the program clears the target field and returns to it, where the newly updated list is displayed.

You can use conditional programming to allow selections from different lists to be displayed within the same field, or to display branching sub-lists that depend on which selections the user makes. And you can do all this in a single field.

The same field in GLOBAL.DTF can be used for multiple pick-lists in the same or a different database. For example, you could use the Titles pick-list in any database containing peoples' names. To do this, just make Titles the external lookup field in the @XLookup statement for each field where you need to make a Title selection. **QA**

Mike Bell is the publisher of *The Quick Answer*. His Program Pack for Q&A 4.0 was the inspiration for many of Q&A 5.0's new features.

# Quick Tip

## Conditionally Suppress Report Calculations



A client wanted a report showing the age of stock balances based on the date the stock was received at the warehouse. I made Warehouse In Date (a date field) column 10 in the report, and created the following derived column to subtract that date from the current date:

```
Heading: Days Old
Formula: @Date - #10
Column: 15, F(N0)
```

This worked unless Warehouse In Date was blank, in which case Q&A returned an absurdly vast number of days. To eliminate the lengthy figures, I needed to suppress the calculation when there was no value in column 10, so I created these two additional derived columns:

```
Heading: Invisible Check
Formula: @Len ( @Left ( #10,1 ) )
Column : 16,I
```

```
Heading: Days Old
Formula: #15 * #16
Column: 20, F(N0)
```

Column 16 is invisible. Its formula takes the length of the leftmost character in column 10, returning a 1 if there's a value in it, or 0 (zero) if it's blank. Column 20 then multiplies the original column 15 (Days Old) by column 16. If column 16 is 0, then column 20 returns 0, effectively suppressing the calculation. The final step was to make the original column 15 invisible. **QA**

Peter Heskett, United Kingdom. 0181-392-1172, CompuServe 100117,467.

*[Here's a derived column that suppresses the calculation if the Warehouse In Date field (column 10) is blank:*

```
Heading: Days Old
Formula: @Text(#10 <> "", @Date - #10)
Column: 15
```

—Ed.]

# And the Winner Is . . .

**W**E received a many entries responding to *The Quick Answer* challenge announced in the February 1996 issue. Here, again, is the requirement Q&A 5.0 user Joe Dokes wants satisfied:

*The starting point is the Q&A Main menu. With one keystroke or mouse click, Joe wants a list of search values from his favorite database to appear, so he can conveniently select the record to view.*

Five entries met contest requirements—they came from Joe Bryant, Alec Mulvey, Spencer Wulwick, and Erica Yoxall. (Spencer covered his bases by submitting two entries.)

An entry from Salvatore de Lello displayed a custom Table View instead of a pick-list. Though his technique didn't meet the *list* requirement, it's easy to implement, adaptable, and it works in Q&A 4.0 as well as 5.0.

An entry received from Darrel Hoefling went far beyond contest requirements. Though we suspected a solution might be somewhere in his entry, we were hard-pressed to find it among his 10 pages of closely typed programming code.

## The two-database approach

Three entries used a two-database approach to satisfy Joe Dokes' requirement. Here, more or less, is how they work:

1. A Main menu macro opens a one-record supplementary database programmed to display an XUserselect or XUserselectR pick-list of search values (names or company names) from the target database.
2. A second macro copies the selection to the Clipboard, exits the supplementary database, opens the target database's Retrieve Spec, pastes the search value into the appropriate field, and retrieves the record.

Alec Mulvey's customer database auto-fills a Speedy, read-only FullName field with the last name, a comma, a space, then the first name. In his supplementary database, he uses XUserselectR with a starting range of "a" and a blank ending range to display a FullName pick-list that returns the selection's record ID to the ID field (#2), like this:

```
< #1: XUsr("CUSTOMER", "FullName", "ID", "a", "", ID);
      Cnext
< #2: @Macro("Send Value")
```

Alec's *Send Value* macro copies the record ID to the Clipboard, pastes it into the Retrieve Spec of the CUSTOMER database, then turns it into an expression:

```
<begdef><nokey><name>"Send Value"<vidon><f11><capsf10>
fscUSTOMER<enter><f12><alti>{ID=<end>}<f10><enddef>
```

The macro turns insert mode on with <alti>, inserts {ID= in front of the number, moves to the end of it, and inserts the closing French brace. The Retrieve Spec contains a retrieval expression like the following. (ID is a number field, so quotes around the search value aren't required.):

```
{ID=56}
```

The advantage of this technique over using a literal search value is that any field can contain the expression, and the procedure won't bomb if the form is later redesigned. With large databases, though, a retrieval expression takes longer to search than a literal value.

Erica Yoxall's employee database contains Employee Name and Employee Code fields. She programs the database to fill a read-only Speedy field with the name plus the code. The program in her supplementary database parses the code from the XUserselect pick-list selection, which her macro copies then pastes into the Employee Code field at the Retrieve Spec.

## The one-database approach

Though the two-database approach is nifty, the solution doesn't require it, as Joe's and Spencer's entries proved. Both add a one-character "control" field to a company database, and program it with @XUserselect to display company names.

Joe opens his database in Search/Update mode. Spencer opens his in Add Data mode, so his pick-list displays on a blank form—a nice touch. Joe suggested this option, but required that a blank record be kept in the database.

We found Joe's Search/Update mode a bit faster than Spencer's Add mode because Joe's macro can "press" F7 for a blank Retrieve Spec. (Spencer's macro must exit then reopen the database in Search/Update mode.) However,

Joe's technique displays the pick-list during regular Search/Updates, too, when it isn't needed.

Joe gets away with one macro to Spencer's two. Joe's pauses until the user highlights a pick-list selection and presses Enter to resume. It's fast for those who don't use a mouse, but it prevents those who do from simply clicking on their selection.

It was a close race. Although Spencer's technique performs a bit slower than Joe's, Spencer wins the \$100 prize because his pick-list appears on a blank screen without requiring a blank record in the database. He lets you select the item with or without the mouse, and his pick-list displays only when it's needed. Joe wins a runner-up one-year subscription to *The Quick Answer*, and Alec and Erica receive honorable mention and a certificate of recognition.

Here's how Spencer describes his winning technique.

## How to Satisfy Joe Dokes' Requirement

I use the CLIENTS database that comes with Q&A 5.0. You can copy it from the \OFFICE subdirectory to your default data directory.

The Start menu must be set to No, Automatic Execution must be set to Yes, DOS SHARE must be loaded, and Sharing Mode for the database must be set to *Allow* because you'll be looking up clients in the same database.

Follow these steps to create the procedure:

1. Add a macro to the Q&A Main menu. At the Alternate Programs screen, enter the following on any pair of empty lines:

```
Alternate program: altz
Menu selection....: Clients
```

The Menu selection can be anything, as long as it doesn't conflict with the first letter of any other Main menu selection. You can use another alt-letter macro hotkey combination—just be sure to change the macro accordingly.

2. Add the following macros to your macro file, then save the file and reload it:

```
<begdef><altz><name>"<caps,>altz<caps.>"<vidoff>
<esc><esc><esc><esc><esc>fa<capsf4>clients<enter>
<dn><capstab><enter><enddef>
*
<begdef><nokey><name>"Find"<vidoff><capstab><f11>
<esc>ys<f10><dn><f12><f10><enddef>
```

3. Add a one-character field to the CLIENTS database. Place the field just to the right of the Cust. No. field on the same line, then press F10 to save your changes.

4. Go to the Program Spec, and enter > *Goto #2* in the Cust. No. field. In the new field you added, enter this:

```
> #5 = @Userselect(@Fn,"Company"); @Macro("Find")
```

Finally, enter #2 in the Company field. (The Goto in the Cust. No. field makes Q&A bypass the new field during regular database use.)

Now, press "C" from the Main menu to display the list of company names. Select one, and Q&A will display the record.

You can press Alt-Z from elsewhere in Q&A, and the procedure should work equally well.

Spencer Wulwick is a Symantec Premier Consultant in the New York/New Jersey area specializing in Q&A. 718-507-6647. CompuServe 73232,2410.

## How The Quick Answer Does It

By using a control *character*, you can avoid using a control field. Program the CLIENTS Company field like this:

```
> If @Instr(Company, "^") then
{
Company = @Userselect(@Fn, "Company");
If Company <> "" then @Macro("Get Company")
Else { @Msg("Search abandoned") }
}
```

Record the first macro that follows and install it on the Main menu with the description *C - Company*. Record the second macro starting from the Address field after making a selection from the pick-list.

```
<begdef><altc><name>"<caps,>altc<caps.>"<vidoff>
fa<capsf4>clients<enter><dn>^<enter><enddef>
*
<begdef><nokey><name>"Get<sp>Company"<vidoff><up>
<f11><esc>ys<enter><dn><f12><f10><enddef>
```

The Alt-C macro opens CLIENTS in Add mode, moves to the Company field, "types" the "^" character, and "presses" Enter to exit the field and display the pick-list of companies. The selection fills the Company field, and the cursor moves to the Address field. (If you press Esc or click outside the list, the Company field is cleared, the *Search abandoned* message appears, and the procedure ends.)

The second macro "presses" Up arrow to return to the Company field, copies the selected company to the Clipboard, abandons the record, changes to Search/Update, pastes the value into the Company field, and retrieves the record.

The program lets Joe quickly switch to another record by typing a "^" in the Company field and pressing Enter to redisplay the pick-list. The *Get Company* macro abandons the record, so any changes to it must be saved first. **QA**

# Manage Custom Pick-Lists on a Network

TOM MARCELLUS

When pick-list selections change from day to day and user to user, you can avoid downtime by making those lists conditionally updatable during data entry.



**Q** &A 5.0's Userselect command lets you display a pick-list of selections for any field. One of its drawbacks, though, is that to update the list you have to reprogram the database. This can mean trouble in a multiuser environment because no one can use the database while you're tinkering away in the Program Spec. Another drawback is that a field's pick-list is the same for all users. You might want Sally's list to include only items A through G and Tim's to include only items H through Z, but how do you get that kind of flexibility?

I'll show you a few tricks that not only solve these problems, but let you build these advanced pick-list capabilities into any database:

- You, the database administrator (DBA), can make some or all of the pick-lists in any database editable on-the-fly during data entry.
- You can decide which pick-lists to make editable on a

user-by-user basis. This way, only the users you authorize can create or edit their lists for selected fields.

- You can make any pick-list field required (can't be left empty) or optional (may be left empty).
- All pick-lists are editable by you without having to shut down the database for reprogramming.
- You can standardize the programming for all pick-list fields. You simply copy the program to other fields and databases to give them the same capabilities.

Although the technique I'll describe was designed with multiuser databases in mind, it works on single-user databases as well.

## Create two sample databases

You'll need two databases: one where the pick-lists are displayed and used during regular data entry and another to store the pick-lists' selections and DBA controls.

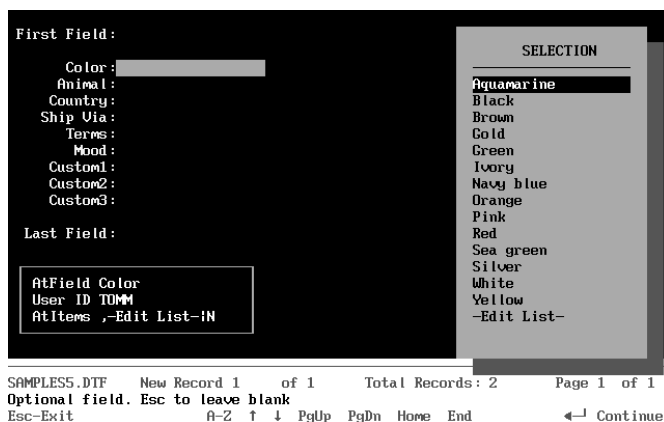
I'll illustrate using two sample databases: SAMPLES5.DTF for data entry, and LIST5.DTF for list storage. A SAMPLES5 record with the pick-list displayed for the Color field is shown in **Figure 1**. **Figure 2** shows the corresponding LIST5 record for the same field.

By taking the time to create the two databases (I'll give you the details as I go along), you'll be able to see the technique in action rather than having to visualize it. This way, you'll find it easier to adapt the technique should you need to. If you'd like a ready-made application to study and work with, the two databases—fully programmed and ready to use—are available on disk. See page 16.

## How the technique works

The SAMPLES5 record in Figure 1 shows the pick-list displayed for the Color field, the first of nine pick-list fields on the form. Your data entry database will be different, of course—SAMPLES5 is strictly for demonstration.

The three fields at the bottom of the form—AtField, User ID, and AtItems—provide the key to advanced pick-list capabilities. In your own databases, you'll place these fields out of reach—on a back screen page, for example.



**Figure 1.** The SAMPLES5 data entry database with the pick-list for the Color field displayed. Note the list's final "-Edit List-" selection.

You can optionally omit their field labels and make them invisible with color.

In Figure 1, AtField is set to the name of the currently highlighted field, User ID is set to user TOMM, and AtItems contains “-Edit List-|N” preceded by a comma. The AtItems code means that the DBA has granted TOMM edit privileges for the Color field’s pick-list and that Color isn’t a required field.

Notice that “-Edit List-” appears as a selection at the bottom of Tom’s Color field pick-list. If Tom wants to, he can select -Edit List- to display, update, and resave the list at any time.

The User ID field is filled by this program in the first field on the form:

```
< User ID = @UserID;
  If User ID = "" or User ID = "Big Cheese"
  Then User ID = "Items"
```

In a single-user database with no User ID, or in a multiuser database being accessed by user “Big Cheese,” the program sets the User ID field to “Items.” (Otherwise, it sets it to the User ID of the person accessing the database.) “Items,” another code, lets the DBA (“Big Cheese,” in this case) display and edit the master list for any pick-list field without having to open the Program Spec and lock other users out of the database. You’ll see how this works later.

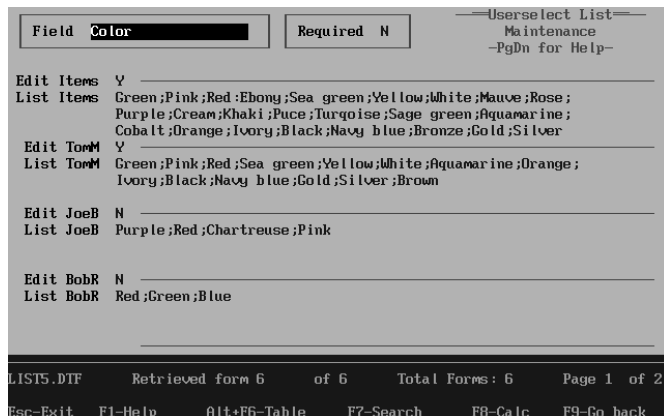
All the fields in SAMPLES5 are formatted for text.

Listing 1 shows the standardized pick-list field program.

**Listing 1.** The program for the Color field. All pick-list fields contain the same program except for the field name reference in the first line.

```
< AtField = "Color";

  If AtItems = "" and @(AtField) = ""
  then Goto AtField;
```



**Figure 2.** The Color record in the LIST5 database corresponds to the Color field in SAMPLES5.

```
If @(AtField) = "" and AtItems <> "|"
then @(AtField) = @Userselect(
@XLu("List5", AtField, "Field", "List " + User ID)
+ @Left(AtItems, @Instr(AtItems, "|") -1));

If @(AtField) = "" and @Right(AtItems, 1) = "Y"
Then { @Msg("Required field");
Clear(AtItems); Goto @(AtField) };

If @(AtField) = "-Edit List-" Then Goto AtField;

@Color(@(AtField), 15, 0);
Clear(AtField, AtItems);
```

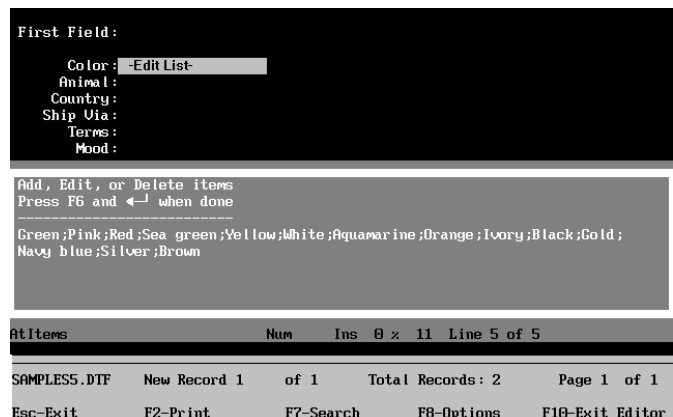
### What the pick-list field program does

When the cursor enters a pick-list field—Color, in this case—that field’s name is assigned to AtField. (AtField keeps track of the current pick-list field.) If AtItems and the field whose name is stored in AtField are empty, the program sends the cursor to AtField. AtField’s program (see Listing 2) does most of the work, then sends the cursor back to the field named in AtField.

Back in the Color field, AtField is no longer blank, so the program resumes, performing a lookup to the LIST5 database to fetch the items for the pick-list. During this phase, the program finds out if the list is editable by the current user and if the field is required or optional.

If the LIST5 Color record indicates that Color is a required field, the program prevents the user from escaping from the list (making no selection). If the same record says Color’s pick-list is editable by the current user, then the program makes -Edit List- the last selection on that user’s Color pick-list. Should the user choose it, control passes to AtField to display the list for editing. (See Figure 3.) (-Edit List- in the program, be sure to use ASCII 196—a long dash. If you use keyboard dashes, -Edit List- will appear at the top of the list rather than at the bottom.)

Once a selection is made, the program unhighlights the field, and clears AtField and AtItems in preparation for leaving the field. See the sidebar, “Pick-List Field Options.”



**Figure 3.** The AtItems field opened for editing.

**Listing 2.** The program for the AtField field.

```
< If AtField = "" Then CPrev
  If @(AtField) = "" Then
  {
    @Color(@(AtField), 0, 7);

    AtItems = "," +
    @Xlu("List5", AtField, "Field", "Edit " + User ID);
    If AtItems = ",Y" Then AtItems = ",-Edit List-"
    Else AtItems = "";

    AtItems = AtItems + "|" +
    @Xlu("List5", AtField, "Field", "Required");
    If @Left(AtItems, 2) = ",|" Then
    AtItems =@Del(AtItems, 1, 1);

    If @Error then { @Msgbox(
    "","No list for " + AtField + " field!",
    "You need to add record to LIST5.DTF");
    Goto @(AtField) };

    If @Right(AtItems, 1) = "Y"
    Then @Msg("Required field")
    Else @Msg("Optional field. Esc to leave blank");
    Goto @(AtField)
  };
  If @(AtField) = "-Edit List-" and
  @Instr(AtItems, "-Edit List-") > 0 Then
  {
    AtItems = "Add, Edit, or Delete items" + "
    " + "Press F6 and Enter when done" + "
    " + "-----" + "
    " +
    @Xlu("List5", AtField, "Field", "List " + User ID);
    @Fedit; Goto AtItems
  }
  Else { Clear(AtItems, @(AtField));
  @Msgbox(AtField + " pick-list",
  "not editable in this manner.",
  "See database administrator.");
  Goto @(AtField) }
```

### What the AtField program does

If AtField is empty, its program's first line (see Listing 2) sends the cursor to Last Field, preventing the program from executing from a non-pick-list field.

The second line begins a conditional that executes if the current pick-list field is empty. If it is, that field is set to a contrasting background color so the user can easily recognize the field for which the pick-list is displayed.

The program then checks the corresponding LIST5 record to find out if the current user can edit the pick-list and if the field is required. If there's no record for the field in the LIST5 database, an error occurs and an @Msgbox is displayed.

If the pick-list field contains "-Edit List-" (indicating the user wants to edit the list), the program builds the list in the AtItems field, and opens AtItems' field editor with the list displayed for editing. (See Figure 3.)

The final block of code prevents a user who has no edit privileges for the current field's list from beating the restriction.

Again, when typing "-Edit List-" in the program, use the ASCII 196 dash, not keyboard dashes.

**Listing 3.** The program for the AtItems field.

```
> If not @Instr(AtItems, "-----") Then
  { @Play("Beep");
  @Msgbox("", "Illegal use of field","");
  Clear(AtItems, @(AtField));
```

## Pick-List Field Options

The pick-list field program is standardized. Use the Clipboard to copy (F11) and paste (F12) it to another pick-list field, then change the field name in the first line to the new field's name. Just remember that the database requires the AtField, User ID, and AtItems fields and their programs, and that you need a corresponding record in LIST5 for each pick-list field.

You can optionally add a Cnext or Goto command to the very end of any pick-list field program. This way, once the user makes a selection (or presses Esc from a pick-list for an optional field), the cursor will move to the next specified field.

If you want the user to think twice before leaving an optional pick-list field empty, you can use a Navigation Spec program like this in the pick-list field:

```
> If Color = "" Then if not @Askuser(
  "","Wanna leave this field blank?","")
  Then goto Color
```

```
If AtField <> "" Then
{ @Color(@(AtField), 7,0);
Goto @(AtField) } Else Chome };

AtItems = @Repllas(AtItems, "-----", "|");
AtItems = @Del(AtItems, 1, @Instr(AtItems, "|" ) + 1);
AtItems = @Replace(AtItems, ";;", ",");
AtItems = @Replace(AtItems, ",,", ",");
AtItems = @Replace(AtItems, ";", ",");

If @Left(AtItems, 1) = "|" Then
AtItems = @Del(AtItems, 1, 1);
If @Right(AtItems, 1) = "|" Then
AtItems = @Repllas(AtItems, ";", ",");

If @Askuser("", "Save updated list?","")
Then {
XPost("List5", AtField, "Field", AtItems, "List " +
User ID);
Clear(AtItems, @(AtField)); Goto @(AtField) }
Else { Clear(AtItems, @(AtField)); Goto @(AtField) }
```

### What the AtItems program does

AtItems' program (see Listing 3) executes when the user closes the field editor (after updating the pick-list) and presses Tab or Enter to leave the field. Two initial checks are performed to verify the user didn't enter the field illegally, then the updated list is cleaned up. The instruction and separator lines (the three lines at the top of the field editor in Figure 3) are deleted, along with any superfluous semicolons, commas, and so forth. The program then prompts the user to save the updated list. If the response is Yes, the list is XPosted to the user's List field in the matching LIST5 record, and the user is returned to the original pick-list field where the newly updated list is displayed.

## The LIST5 database

In the LIST5 database in Figure 2, Field (Speedy/Unique) provides the link to the corresponding pick-list field in the

SAMPLES5 database. LIST5's Required field is a Yes/No field.

The Edit Items and List Items fields are the master pick-list fields. In other words, if SAMPLES5 is single-user, or it's multiuser and being used by the DBA ("Big Cheese"), the entries in List Items will appear on the pick-list, and Edit Items will control whether or not the list is editable on-the-fly.

Looking again at Figure 2, you'll see that TomM, JoeB, and BobR are three users, each with his own list and corresponding edit privileges. TomM can edit his Color field pick-list, while JoeB and BobR can't—they're stuck with their pick-lists unless the DBA updates them in the corresponding LIST5 record.

LIST5 can include an Edit [User ID] field and corresponding List [User ID] field for any reasonable number of users. Simply redesign the form and add the required fields.

Format the Edit [User ID] fields for Yes/No values, and the List [User ID] fields for text.

Should BobR leave the department and be replaced by JaneT, simply change the JoeB portion of the two field labels (and field names) to JaneT.

You can set initial values for the Required, Edit Items, and Edit [User ID] fields. This way, you'll already have default values entered when you add a new pick-list record to LIST5.

As long as a LIST5 record exists for the corresponding SAMPLES5 pick-list field, if the user has on-the-fly edit privileges, he can not only edit but create his pick-list. You, the DBA, can leave his LIST5 List [User ID] field

empty, and he can decide what to put on his pick-list at any time.

### Preventing unauthorized access to LIST5

Because all users need to perform XLookups to LIST5, and those with edit privileges must be able to XPost to it, protecting LIST5 can be tricky. One way is to password protect it, create an Access Control Form for each user, grant only Add/Edit privileges, then place a statement like this in its first field:

```
< If @UserID <> "Big Cheese" Then @Exit
```

This simple statement will immediately toss anyone but the DBA (User ID "Big Cheese") out of the database should access in Add Data or Search/Update mode be attempted. **QA**

Tom Marcellus is editor of *The Quick Answer* and author of *PC World Q&A Bible*, published by IDG Books. His QuickClick Calendar Plus—a time- and activity-tracking database for Q&A 5.0—is available from Marble Publications, publisher of *The Quick Answer*.

The two databases featured in this article (SAMPLES5 and LIST5), fully programmed, ready to use, and not password protected, are available on disk for \$29 plus \$2 shipping from:

Marble Publications, Inc.  
PO Box 9034, Gaithersburg, MD 20898-9034  
800-780-5474 / Fax 301-424-1658  
Major credit cards, checks accepted.

PO Box 9034  
Gaithersburg, MD 20898-9034

Second-Class  
Postage Paid at  
Rockville, MD