All About Field Names

ALEC MULVEY

Field names and field labels aren't the same thing. How you manage the differences can make or break your programming and mail-merges, and explain confounding error messages.

DOS WIN

Q&A is a very tolerant program—that's one of its great strengths. It allows you to design a database in a few minutes, and a mail-merge document in a few more, all without any specialized knowledge of database design and programming. But as your skills increase, you eventually run into areas where you need more knowledge and you may try things that don't work. One such area is field names.

When it comes to field names, there are actually two issues. One is the distinction between field *names* and field *labels*. The other is that Q&A allows you to use *certain* field names in your database, but then doesn't recognize them in programming and mail-merge.

Field names vs. field labels

The distinctions between field names and their associated field labels can be a source of confusion. Q&A users aren't helped by the fact that the two terms are often used interchangeably—and often mistakenly—even in official publications. If you've worked with Microsoft Access, you might have noticed that the corresponding terms are *Field Name* and *Caption*.

Let's explore what's what here with a simple demonstration.

Create a new database named Fldnames.dtf. Add two fields—each on a separate line—**First Name** and **Last Name**, and save the design. Now go to the Field Names Spec (File / Design / Program / Set Field Names), and you'll see that the database form you created has two fields and their labels and names are identical. This is what Q&A does. Every field *must* have a field name. By default, this is the same as the field label you typed when designing the form. Exit this Spec, go to Design / Redesign, and add the following text fields:

```
Address:
Address:
Address:

<   >  (a field with no label)
```

Save the design, return to the Field Names Spec, and you'll see what's shown in Figure 1.

Figure 1. After the initial Address field, Q&A assigns names to additional Address fields with a sequential number. A field without a label will be named by Q&A starting with F0001.

There are two areas of concern here. First, notice that the three fields with the same label (Address) have been assigned unique field names by Q&A. (Field names *must* be unique.) But what Q&A has done here isn't ideal. It would be better if these fields were named Address1 through Address3 (see the sidebar, "Consecutive Field Naming") but you have to do this yourself.
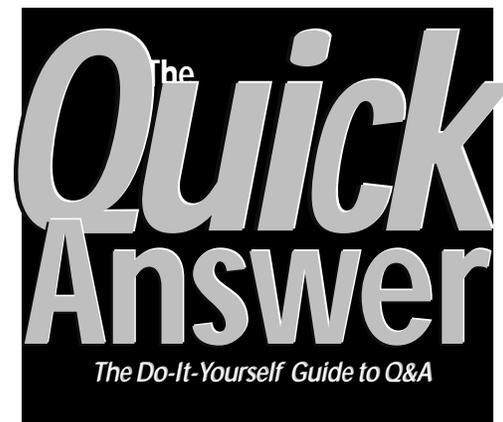
The second point to note is the unlabelled field. As I mentioned, all fields must have a unique name. Q&A assigns a sequential number to unlabelled fields: F0001, F0002, and so on. You should *never* use these field names. One reason is that if you add or remove an unlabelled field, Q&A will then *renumber all the unlabelled fields*, so a field originally named F0004 could become F0006. Q&A will update the field names in the Program Spec, *but only for fields in the current database*. References to external lookup fields (in XLookup

They Come Back to Q&A

TIME FOR ANOTHER PEP TALK. When I hear from *former* Q&A users asking how to return to "Old Faithful," or reports of Q&A being used to plug gaping holes in million-dollar proprietary systems, I yawn.

Pity those poor souls who, from bad advice, on a whim, or under pressure from the know-it-alls in their organizations, go and dump what works in favor of "The Latest."

The people I *don't* have much sympathy for are the morons who've never bothered to subscribe to this newsletter but send us brusque email messages demanding to know (for free, of course) how to "convert" their Q&A applications to Microsoft Access.

If I'm in a good mood, I tell 'em they can export their raw data then start over from scratch. *From scratch.* These are people whose applications have evolved *over years*, yet they have this huge idea that there must be some secret switch (only *we* know where it is) they can throw to make everything—macros, reports, mail-merges, the lot—work just like in Q&A but with cuter graphical screens.

Sheesh!

Even if one day you're forced to abandon Q&A—you still shouldn't. The sidebar on page 12 suggests a few good reasons why not, and scattered throughout this issue are still more.

We have a perfect right to be smug about this.

Quick Tip

Field Not Needed for @Shell

I noticed in the December 1998 issue on page 2 that @Shell was being used in conjunction with a temporary field. There is a way of using @Shell without requiring that extra field and without it making any changes to any fields. If the temporary field isn't needed for other programming reasons, you can simply place @Shell within an @Msg function, like this:

```
@Msg(@Shell("qa -m6"))
```

This command starts another copy of Q&A then runs the Alt-6 macro. Using @Shell this way just displays a "0" on the message line when you return to the original database, and the "0" disappears with the next keystroke. I've been using @Shell this way for a couple of years and it works great.

Carl Underwood

[You can optionally follow the @Shell command with @Msg("") to prevent the "0" return code from appearing at all. —Ed.]

Letters

If you are successfully using an HP DeskJet 722C printer with Q&A DOS, we and this subscriber would like to hear from you.

I have Q&A 4.0 and just got an HP 722c printer that won't print from Q&A. I have called both HP and Symantec tech support with no success. I use Q&A's HP 550c driver. Sometimes a letter won't print for five to 10 minutes. Sometimes the printer won't respond at all. It prints envelopes upside down after five to 10 minutes when it prints them at all, which is rare. I have used Q&A with an HP 550c printer for more than two years with no problems whatsoever. Please help as I'm a realtor with a database I can't use. I'm virtually out of business when it comes to follow-up and prospecting.

Dick Fletcher, New Homes Realty, 703-709-8288

Field Names...

commands) and merge field codes in Write documents, to mention two instances, are *not* automatically updated. I strongly advise you to give such fields appropriate names, unless you're certain you'll never need to refer to them in programming, mail-merges, export specs or the like.

When to use field labels?

When do you refer to field labels in Q&A? *Never*. You always refer to a field by it's *name*, and the field *name* is what you see in Posting Spec field lists, report column headings, Alt-F7 field lists in merge documents, and so forth. You might think you're using field labels, but you aren't—it's just that they begin life being the same as the field names.

Using field names in programming

Field names can be used in programming instead of Field ID numbers, and it's actually better to do so. It makes your programming easier to read and debug and has other advantages as well. (See the sidebar, "When You Can't Use Field Names in Programs.") Using field names in programming is said to slow down Q&A, but the only thing that's slower is when Q&A compiles the program when you press F10 to save the Program Spec. (This is where Q&A has to convert your program to a lower-level language.) Once this has been done, program execution takes the same amount of time whether the program uses field names, field ID numbers, or both. And only a very heavily-programmed database on a slow PC will take a long time to compile.

Changing field names and labels

Changing a field name won't change its corresponding field label. Similarly, when you edit a field label, Q&A won't change the field name unless you've just typed a label for a previously unlabelled field.

Changed field names are automatically updated by Q&A in programming, saved Specs, report derived columns, and so forth, in the current database. Just remember that if you change a field name in Database A, any references to that field in Database B programming, reports, and specs *won't* be updated.

Restricted field names in programs and merges

As I mentioned earlier, there are traps you can fall into when it comes to field names. Certain names, while acceptable as field names, cause problems for Q&A when you actually try to use them. It could be argued that Q&A should really test for these at the Field Names Spec.

Firstly, as an aside, there are a few characters that Q&A can't accommodate even in field labels. These include the ; : < > characters.

There are two separate areas in Q&A, each with its own list of restricted field names. These are programming and mail-merge.

Consecutive Field Naming

By judicious use of field names, you can save time and simplify your programming. If you have five fields for costs in your database, by naming them Cost1 through Cost5, you can Clear() or them total them, for example, in this way:

```
Total = @Sum(Cost1..Cost5)
```

This is particularly convenient should you later need to add a sixth Cost field.

Restricted fields in programming

Once you stray from using simple letters and numbers in your field names, you run the risk of Q&A rejecting such names at the Program Spec. Rather than list the ones you can't use, here's the (much shorter) list of those you *can*:

```
#
$
@
```

These further restrictions apply:

- # and $ can't be used as the *first character* of a field name.

- You can't use a field name with a number as the first character (even though numbers are generally okay to use in field names).

- You'll run into trouble if you try to program using a field name that corresponds to a Q&A built-in function name (or its abbreviation). There are dozens of these. The *Application Programming Tools Manual* gives most of them. For the most part, they start with the @ sign. So, in practice you can't use @Date or @AB as field names. Some built-in commands aren't preceded by the @ sign.

When You Can't Use Field Names in Programs

You can use field names exclusively in programming, but *only* in fields controlled by on-field-entry or on-field-exit programming, like this:

```
Full Name:
< If Full Name = "" Then
Full Name = First Name + " " + Surname;
```

For on-record-entry, on-record-exit, and calc programs, as well as Mass Update programs, you have to refer to the *execution* field by field ID number (for example, #300).

These include the Lookup and XLookup family, and Clear. In QA 5.0 these additionally include Rem, and the Userselect, XUserselect and XPost family.

Accordingly, if you enter a programming statement such as:

```
#350 = Tel(b)
```

You'll get a *Not a valid Program Spec...* error message when you attempt to leave that field, as parentheses aren't acceptable as part of the field name in programming.

All of these restrictions rule out quite a few field names you might want to use—these, for example:

```
1st Name
Parent(a)
Salary+Bonus
Litres/Second
Print?
Init.
Clear
US
```

I stress that all these restricted characters *can* be used as both field labels and field names; Q&A balks only when you use them in a programming statement.

What about spaces?

Spaces are perfectly acceptable in Q&A field names. There's just a chance, though, that such a field might then produce an ambiguous merge code—*Join Date*, for example. (More on this later.) Some Q&A developers avoid spaces in field names—using *LastName*, for example. These are typically people who work with other databases such as Microsoft Access, where spaces do cause problems. Although they might routinely use spaces in field *labels*, they avoid them in field *names* for the sake of consistency. One problem with this approach is that the result can mean unattractive column headings in reports (and having to manually set such headings in the report design to the more readable field label.) It's a lot of work for no real benefit.

Using restricted field names in programming

You can use restricted field names in programming, but you have to modify the normal syntax. For example, you can force Q&A to accept such a name by using the "pound-quote" syntax, like this:

```
#350 = #"Tel(b)"
```

(Note: What in the U.S. is normally called the "pound" sign— the "#" symbol—we in the U.K. call the "hash" symbol. In the U.K., we think of "pound" as our currency symbol.)

These restricted field names can also be used in XLookups this way:

```
Tel = @XLu("Customer",Ref,"ref","Tel(b)")
```

Notice that both usages involve placing the "bad" field names inside quotes.

Restricted field names in mail-merge

Restricted field names are less of a problem in mail-merge than in programming. In a merge document, you can insert codes to tell Q&A to insert a file, apply justification, and the like. These are similar to field codes, and they all have abbreviations (such as *S* for *STOP*). (See the *Q&A Quick Reference Guide*, page 6.) This is not as restrictive as it might seem at first glance, because these commands are likely to require a switch—one that specifies the file to insert, for example. If used without a switch, they're interpreted as fields.

Accordingly, you can use *J* or *Join* as a field name in a merge document with the code *J*, just as long as your field name is not *J C*. In this latter case, Q&A would attempt to *join* a file named *C* to the document. Ignoring the long commands, the problem abbreviations are as follows. (In each case, the *x* indicates additional characters, and *n* a number.):

```
JY x
P x
J x
Q x
QP x
G x
SS x
PF x
PS x
PG x
LS n
```

My tests have revealed just two definite problem field names in mail-merge—*Stop* and its abbreviation, *S*. You can't use them as field names in a merge document

Conclusion

It's important to understand the distinctions between field labels and field names, and the overwhelming importance of the latter. Don't use the default field names Q&A assigns to unlabelled fields. It's probably better to avoid learning what characters you can and can't use in field names altogether by just avoiding everything in field names except letters and numbers—and ensuring that the first character in any field name is a letter.

Alec Mulvey owns Keyword Training & Consultancy based in Ascot, near London, England. Alec has been building Q&A applications and training on Q&A for nine years, and is the current president of the (Inter)National Q&A User Group. Keyword Training is also the UK distributor for the International English edition of Q&A (the "Definitive Edition" CD-ROM) and also John Dow's Utilities. Fax +44-1344-884-111, alec@keywordtraining.com, http://www.keywordtraining.com.

Case Study

Sorting Dates Spanning the Turn of the Century

SHELDON J. EINHORN

Y2K

Date fields can contain non-date values and even invalid dates. But how do you find and fix them?

DOS
WIN

SUCCESSFUL residential real estate agents keep in touch with their previous customers for many years. In our company, agents can enroll their customers in a "Club" that delivers to them a homeowners' magazine over a five year period. Each issue identifies the agent who provided the subscription.

We want to know when these subscriptions expire so we can keep in touch in other ways. To do this, we maintain a Q&A database with information on all past customers. At present, we have more than 6,000 records in this database which contains a date field named **Sub Til** for the date the magazine subscription expires.

Sometimes we need to sort selected records in Sub Til date order. This proved to be a problem. When the database was designed, date format number 3 had been selected. This meant that an expiration date of March 19, 2000 appeared as 3/19/00. When we sorted the records, *Q&A treated them as though they were dates in the year 1900.*

As a first step in solving this problem, we retrieved the records in Sub Til date order, moved the cursor to the Sub Til field, and switched to Table View by pressing Alt-F6. The records appearing at the top of the table contained nonstandard Sub Til entries such as "Check" or "Ask Tim." These had been entered in lieu of actual expiration dates. Also, and to my initial surprise, there appeared to be about 50 dates randomly scattered among these nonstandard text values.

On closer examination, all of these "dates" were "2/29/00." The year 2000 will be a leap year, and there will be a 29th day in February. Many subscriptions expired on this date (All subscriptions expire on the last day of the month). When this date was entered into a record, it was entered as "2/29/00." Q&A interpreted it as 2/29/1900, but 1900 wasn't a leap year, so the entry generated an error message. The person entering the data had overridden that message by pressing Enter again, forcing Q&A to accept the value, *but not as a date!* That's why these "dates" were intermingled with other nonstandard values.

The first step I took to clean up the Sub Til field was to perform a Mass Update to get rid of the erroneous dates. Here's the Retrieve Spec I used:

```
Sub Til: ]2/29/00
```

The right square bracket told Q&A to include just the records containing the nonstandard 2/29/00 value.

Here's the Update Spec I used:

```
Sub Til: #1 = 2/28/00
```

I answered "Yes" when Q&A asked, *Do you want to confirm each update individually?* After I had confirmed enough updates so that I was confident the process was okay, I pressed Ctrl-Fl0 to complete the rest of the updates automatically.

The next step was to redesign the database. I stepped through the process until I got to the screen permitting a choice of date formats. This time, I selected format 5 (3/19/1968). I went back and scanned some records to make sure all date fields had enough room for this longer style.

Now it was time for the final and most significant Mass Update. I used the following Retrieve Spec:

```
Sub Til: >= 1/1/1900..<1/1/1920
```

Any year greater than 1900 would have worked for the upper limit.

My Update Spec was this:

```
Sub Til: #1 = #1 + 36525
```

Why 36525? Q&A does date arithmetic based on the number of days between two dates. 36525 is the number of days in 100 years, including a leap day every four years. (Actually, that's not quite true. Because, as noted above, 1900 was not a leap year, if you start from a date between 1/1/1900 and 2/28/1900, there are only 36524 days in the next 100 years. The difference was of no consequence for our purposes.) Now all subscription expiration dates were the way we wanted them.

If your database contains dates after the turn of the century in a 3/19/00 format, a procedure like the one above could help you sort your records.

Dr. Einhorn is Vice-President of Einhorn-Adler, Inc., located in the Philadelphia suburbs. He has supported the operations of his company for nine years using Q&A. Before starting his real estate career, he worked for 25 years in defense research.

@Help

EDITED BY GORDON MEIGS
AND WILLIAM HALPERN

Programming Space Limitation   DOS WIN

While programming a database, I added some lookup and target fields to an XLookup command, hit F10 to save the Spec, and Q&A displayed an error message saying, *This Spec is too long....* So I printed the Spec to a disk file (by pressing F2, F10) and noted that its file size was 65,903 bytes. I tried saving the Spec once more and received the same message. I had no choice but to Esc out of the Spec, losing my edits.

The Q&A documentation states that a field can hold up to 32,000 characters and a record up to 65,536 characters. Does this (horror of horrors) also apply to programming? I thought it might, so I shortened many field names. In the programming I removed spaces around operators and between paragraphs and used the function name abbreviations. This time I was able to save the Program Spec, and noticed that the corresponding disk file was a little over 58,000 bytes. I then added about seven lines to the XLookup command and the *Spec too long* message reappeared. Is there a limit on something else—the number of XLookups, for instance—that could be causing Q&A to generate the error? Are there some space reductions that *don't* count, such as shortening field names? I've noticed that Q&A has trouble with lengthy XLookup commands. (I've had to break some up into multiple commands). Is there a maximum number of lookup fields you can use in one XLookup statement?

Incidentally, the database in question is an upgraded part of an existing networked Q&A system at a major hospital. All the other hospital departments have migrated either to other Windows databases or to proprietary Windows-based software that can't do a fourth of what Q&A does!

Gloria J. Heidengren via the Internet

A Q&A Program Spec can't exceed 64K. Navigation Specs are limited to 32K. There's another limit—no more than 32K programming in any *field*—but that can be a bit misleading because it can depend on your system configuration, Q&A version, and the amount of available memory. You can safely count on being able to place about 24-28K programming in a field. More than that is flirting with danger since a subsequent system change could corrupt that field. There's also a limit as to how many fields you can include in a single XLookup statement. The upper limit is somewhere in the low 20's. We make it a habit never to have one XLookup statement perform more than 16 lookups and retrievals.

Reducing your programming will help:

- Use valid abbreviations for function names and programming commands. (For example, *XLu* for *XLookup* and *@Usl* instead of *@Userselect*.)

- Keep any comments (REM statements) brief.

- Minimize the use of spaces and carriage returns. Although they make programming easier to follow, they count toward the total programming limit.

- Use external databases and text files to store selections for Userselect lists, rather than storing them in the program itself. (Example, @Usl("C:\Lists\Shipvia.txt") —where Shipvia.txt is text file containing a list of semicolon-separated shipping methods—can be used instead of including the list in the program itself.)

Shortening field names won't help since, as John Dow has pointed out, Q&A uses a four-character field pointer regardless of field name length.

We don't know how you're writing your programs, but 64K is a lot of programming for one database, particularly if you're taking advantage of features designed to lessen the volume of programming you actually need. Our guess is that you're using many repetitive programming statements. If so, you can save lots of programming space by employing GOSUB routines.

For example, here's a "standard" programming technique:

```
>#1: If #1<>"" Then
   XLu("External Database", #1,"X#10","X#11",#2,"X#12",
```

```
#3,"X#13",#4); Goto #5
>#5: If #5<>"" Then
   XLu("External Database",
#5,"X#10","X#11",#6,"X#12",#7,"X#13",#8); Goto #9
>#9: If #9<>"" Then
   XLu("External Database",
#9,"X#10","X#11",#10,"X#12",#11,"X#13",#12); Goto #13
```

Here's a comparable GOSUB technique:

```
>#1: If #1<>"" Then {#99=1;Gosub #99}; Goto #5
>#5: If #5<>"" Then {#99=5;Gosub #99}; Goto #9
>#9: If #9<>"" Then {#99=9;Gosub #99}; Goto #5
<#99: XLu("External  Database", @(#99), "X#10", "X#11",
@(#99+1), "X#12", @(#99+2), "X#13", @(#99+3)); Return
```

This simple example replaces 262 characters with just 217—a 17% savings over just a few lines of code.

Back to Q&A from MS Access   DOS WIN

We've been using an older version of Q&A for DOS. We have a sophisticated Q&A database we recently ported to Microsoft Access, where we added some fields. We've discovered that Access isn't as easy to use as Q&A, and we're thinking of switching to a Windows version of Q&A. Can we take the expanded version of the database now in Access and convert it back to Q&A? We're running Windows 95.

A. Peppin, via the Internet

The only Windows-specific version of Q&A is Q&A 4.0 for Windows. But it isn't as functional as the Q&A 4.0 for DOS you've been using since it doesn't support all the features in your current setup. Both Q&A 4.0 and 5.0 for DOS will run very successfully in the Windows environment. All of our clients, and thousands of others (including us) are running Q&A DOS versions in Windows (in a DOS window) on a daily basis. This way you get the full functionality of Q&A (including its ease of use) and the benefits of the Windows environment.

As for your data, you can export it from Access to a comma-delimited ASCII file, then turn around and import it into a Q&A database. Just be sure to export the fields from Access *in the same order* as the fields in the corresponding Q&A database.

Solve Network Printing Problem   DOS

We run Q&A 5.0 under Windows 98. Recently, we installed new computers in our office and, for some reason, only the file server that's physically connected to the printer can print. We have to ask the person using it to print all our jobs. What are we doing wrong? Tension is running high here. Please help.

Roz, via the Internet

Don't panic—and get your co-workers to calm down. The problem is with your printer setup in Windows. For each of the network printers, you have to tell Windows, at each workstation, to "capture" a printer port—LPT2, for example (not LPT1)—and redirect the output. You do this from the Printer Properties general tab by clicking Start /

Settings / Printers. After assigning a capture port, you'll also have to assign the path to where the printer is located. (See Figure 1.) Then, in Q&A, go to the Install Printer screen (Main Menu / Utilities / Install Printer), and set the printer to the LPT2 port. This way, when Windows detects a print job looking for LPT2, it will re-route it over the network to the printer you've set to capture that particular port. It works just fine. We do this in all our network installations.

Q&A and Windows NT   DOS WIN

Our company is migrating to Windows NT. The upgrade people will begin their work next week, and we'll be moving our Q&A applications to the new system by the end of February or early March. What's better—having Q&A and all our Q&A-related files on the server, or just the databases and such on the server and Q&A installed on each workstation?

Rodrigo, via the Internet

For overall program and network performance, it's always more efficient to have the executables on each workstation and the shared data on the server. The flip side is that you lose the personal path switch capabilities, so maintenance and the need for system administration can be higher. It's a tradeoff you have to make on a case-by-case basis. With fast machines and a good network, we would probably lean toward sharing Q&A (using the Network Pack) from the server. Remember, even with NT, the 1.8G maximum hard drive partition limit for Q&A (or any DOS program) still applies.

Fix Damaged Databases   DOS

On our NT network, I'm having problems with a Q&A database locking up and corrupting data. The database won't recover. It blows up at the end of the second Recover phase every time. I even exported and imported the data according to an article I read entitled "Repairing a Damaged Database Using Q&A for DOS." I exported to my local drive to eliminate the network as a problem. I then performed the import and database recovery using the local file. It went fast this way, but the database still blew up at the very end of phase 2. Any suggestions?

Steve, via the Internet

Figure 1. Telling Windows to capture output directed to LPT4 and redirect it to the network printer at \\Gordon\Canon.

Add a Clickable 'Image Map' to Your Database

ERIKA YOXALL

Where do you want to go today? Just click on the map!

DOS

Application in ONLINE Edition

O NE of the most important elements of any database design is that the interface be user-friendly. The easier your forms are to understand, navigate and work with, the happier you or your client will be.

For example, say you have a referral business where clients would be looking for services based on their location. You might want to specify or search only particular states. If the client requests "New England" or "within 500 miles of my home in Pennsylvania," you need to be able to select the appropriate states. In Q&A 5.0 you can use a Userselect list, but then you have to remember which states are in New England or which states border Pennsylvania. In this case, a graphical representation of the country where you can see the states in relation to one another would prove much more useful.

The Internet supports a Web page element called an *image map*. This is a picture with code behind it that performs a function based on *what part* of the picture you click on. With a little imagination, you can create similar graphical elements on your Q&A forms using ASCII characters. I'll show you how to make an image map work in a Q&A 5.0 database. Except for coloring, the procedure is the same in Q&A 4.0.

Creating the map

The map of the United States in this example (see Figure 1) is created by typing the characters that make up the

Figure 1. An image map of the U.S. in a Q&A database form. When you click on a state abbreviation, it's added to the Selected States field. Here, Ohio and adjacent states have been selected.

outlines of the map directly onto your Q&A form as background text. Fields are then defined using the < > characters within the space for each state. In this example, I've added a Selected States field to display the states selected using the image map, and a one-character Home field in the upper left-hand corner of the screen where the cursor can "rest" between selections.

Depending on your layout, Q&A might decide that some of the characters in your map are part of a field name, so it's important that you set your field names (File / Design a File / Program a File / Set Field Names) to something appropriate for each field inside the map.

If you're using Q&A 5.0, you can set colors for background text, fields, and field names that make your image map appear as clearly as possible. (File / Design a File / Customize a File / Change Palette.)

Set Initial Values (File / Design a File / Customize a File / Set Initial Values) for each field inside the image map to display the correct label. This example uses the standard two-letter state abbreviations.

Programming the map

The program for each field inside the image map should be set to execute on-field-entry — when the user clicks in the field. For this application, you want to be able to click on a state, have the state abbreviation added to the Selected States field, and have the cursor go to the Home field to await another selection. Each of the individual state fields has a program like this one for Washington:

```
< WA: If @InStr (SelectedStates, "WA") > 0 Then
{@MsgBox ("You'vealready selected this state.", "", "");
GoTo Home }
Else { SelectedStates = SelectedStates + " WA";
    GoTo Home }
```

If you click in the WA (Washington) field, the program tells you if you've already selected it. If you haven't, it adds WA to the list of selected states. Either way, you're returned to the Home field to await the next selection.

The ASCII character set is in the Q&A *User Guide* Appendix. These characters can be combined to create useful and attractive highlights for your applications.

The ASCII U.S. map is courtesy of Jorn Barger, editor of the Robot Wisdom Weblog, *http://www.mcs.net/~jorn/html/weblogs/weblog.html*.

Erika Yoxall owns Hammer Data Systems in Garrettsville, Ohio, specializing in Q&A and Microsoft Access. Phone/Fax 330-527-4018, hammer@apk.net

How to Lose Text the Easy Way

There's a situation whereby it's easy to lose text without even realizing it. If you have a multiline field (a "Comments" field, for example) that isn't expanded but you can see all the text, it's tempting to move to the field, press the Insert key to switch to insert mode, then start typing the new text.

However, if while doing this you "push" the existing text off the bottom of the field, then that text is gone for good. If you realize you've done this, then the original values for the whole record can be reinstated by pressing Shift-F7.

Alec Mulvey, alec@keywordtraining.com

Ordinal Dates Revisited

In the April 1996 issue of *The Quick Answer*, I showed a technique that prints ordinal dates (the *23rd*, for example) in Q&A mail-merge documents. My technique used the database's Lookup Table to store the dates and suffixes, but it's better to use an external database. Why? Well, for one thing, it's portable. If you want to set up a mail-merge for a different database, you don't have to enter all the data into another Lookup Table—the one external database is all you need. Secondly, the numbers 1 to 31 are then available in the Lookup Table for other uses.

Here's how to do it. Create a new database, Ordinal.dtf, with just two text fields. Name the first one **Dom** (for Day of Month) and the second one **Suffix**. Make DOM an indexed (Speedy) field at the Speed-Up Spec.

Add 31 records to Ordinal.dtf, typing the numbers 1 to 31 in the Dom field. In the suffix field enter *th* for all records except the following:

```
Dom Suffix
1   st
2   nd
3   rd
21  st
22  nd
23  rd
31  st
```

Next, modify the middle part of the merge document's program—the part that adds the suffix (*th* or *nd* for example). The complete program should read:

```
*Pg {@Str(@Dom(@Date))}**Pg
{@XLu("Ordinal",@Dom(@Date),"Dom","Suffix")}* *Pg
{@Mt$(@Date) +" "+@Str(@Yr(@Date))}*
```

Note that there needs to be a space between *Pg* and the *{* (left French brace) character.

The second refinement is to put the suffix in superscript. You can apply superscript using the Text Enhancements menu (Shift-F6). An even better effect can be achieved by making the size of the ordinal suffix three point sizes smaller than the number. So, if you're using 12 pt. body text, apply 9 pt. superscript to the suffix program in the merge document—the code between *Pg {@XLu* and *suffix")}* inclusive.

The finished result should look like this:

```
19th January 1999
```

Alec Mulvey, alec@keywordtraining.com

Resource

The other day I was at a clients and had to perform a Y2K audit. I was faced with several databases, the first of which had about 250 fields. I started by asking myself "which are the date fields?" There is no good way of finding that out, so I used a bad way and found 47 of them. I suggested to John Dow that his DTFSPECS program could be amended to include a Y2K option. Within 48 hours he had it done!

The latest version of DTFSPECS has a Y2K option for the data dictionary (lists just Date fields) and also for the All Specs option. This prints out full info, including programming, for all date fields, and also all non-date fields that reference date fields. He is adding a Y2K reports option that will list the names of reports that have derived columns. This is a real time-saver.

John is selling this latest DTFSPECS upgrade, and it deserves a mention.

—Alec Mulvey

DTFSPECS version 2.5, $150
Upgrade (from version 2.x) $30
Contact sales@johntdow.com to purchase DTFSPECS
More information and a downloadable demo version at www.johntdow.com.

Creating and Using Arrays

MARK MYERS

Make Q&A work more like a relational database with this powerful programmer's tool

DOS WIN

Application in ONLINE Edition

SOMETHING that has been on my wish list almost since the day I started programming in Q&A is the ability to implement an array within a record. Even having the ability to create a one-dimensional array attached to a record would be a vast improvement over Q&A's flat file format. This month and next I'll discuss a few basic implementations of array-like data in Q&A. These implementations will likely be of interest mainly to those working in Q&A 4.0, since Q&A 5.0's XPost function can be coupled with macros to achieve the same result.

What's an array?

An example of a useful application of a one-dimensional array is a vendor who wants to store all his customer information in one database—including a searchable list of products purchased by each customer in the last twelve months. A normal flat file arrangment would require that records be created at the *product* level and that client information such as ID number, address, telephone number, and so forth be duplicated in each record. Not only does this waste file space, it creates data management problems. Providing another dimension to the data with the use of an array allows those records to be created at the *client* level.

A flat file implementation can be done at the client level with a series of "sibling" fields (Product1, Product2, Product3, and so on), but this is poor design. No matter how many of these fields you have, one day there won't be enough. What's more, you'll probably have many times more sibling fields than needed for the majority of records in the database.

On the other hand, a well-designed array could allow more than 600 entries at an average of 500 characters per entry. And if the average entry were only 50 characters, the number of possible entries jumps to 6,000.

There are a number of ways to do it. Each involves concatenating (stringing together) multiple entries and storing them as text values in one or two fields. One method is to use a single field and assign a fixed length for each entry. Each individual entry would have a header (also fixed-length) that stated the number of valid characters in the actual data entry. Since both the header and the data are of fixed length, the starting point for each header would conform to a logical series. For instance, if the maximum length of an entry were 7, the header could

be just one character long (since all possible lengths are less than 10). The starting points for entries would be 1, 9, 17, 25, 33, and so on. If you're having trouble visualizing what such a field might look like, here's how it would appear after the items *cat*, *bird* and *pig* had been added:

```
3catxxxx4birdxxx3pigxxxx
```

Each individual entry would first be created by the following function, then added to the end of the array:

```
@Left(@Str(@Len(texttoadd)) + texttoadd + "xxxxxxx", 8)
```

New entries could be placed in the array in alphabetical order, or simply added to the end of it. The advantage to alphabetical ordering is that as the array grows, the time needed to search for an element need not grow linearly. (It *does* grow linearly if the array is unordered.) If you maintained the above array in alphabetical order, the entry *bird* would have been added to the front of the array. For ordered arrays, the list is searched to find the proper insertion point, then it's split at the insertion point and the two ends are spliced to the beginning and end of the new value.

If you wanted to retrieve the nth element in the array, the following function would do the job:

```
@Mid(array, n*entrylength + 2, @Tonumber(@Mid(array,n *
entrylength + 1, 1)))
```

In this case, *entrylength* is 8.

If you wanted to remove an element, you could use the @Left and @Right functions to grab the data on either side of the element to be removed, then splice these two ends together. To remove the nth element, set the array equal to this:

```
@Left(array,n*entrylength) + @Right(array,@Len(array) -
(n+1) * entrylength)
```

Only a slight variation to this function could be used to add elements to an array:

```
@Left(array,n*entrylength) + @Left(@Str(@Len(texttoadd))
+ texttoadd + "xxxxxxx", 8) + @Right(array, @Len(array) -
n * entrylength)
```

You could also set up a function to search for a particular element using either a sequential search or, if the array is ordered, a binary search.

The technique I've outlined here is a good starting point for understanding how to implement arrays in

Q&A. It's optimal if entries are fixed length (product ID numbers for example), though it will limit users to entries within the maximum length and might also consume more space than necessary. In practice, if speed and memory conservation are of concern (remember that you can have no more than 64K of data in a database record), you'll need to tailor the dimensions of each array to the specific requirements of your deployment.

I created a working array "toolkit" (ARRAY.DTF) that's included in the download file for Online Edition subscribers. This array implementation supports entries up to 999 characters in length with a constant insertion size of *@Len(validdata) + 8*. I recommend reading the code comments. (Since Q&A 4.0 doesn't support comments, I've placed them in a text field.)

The array in ARRAY.DTF is slightly more complex than the one discussed above. Instead of a header at the front of each entry that tells how much valid data is to follow, ARRAY.DTF uses two fields. One contains the actual data in the list, and the other contains a series of pointers that keep track of where each entry is in the data field and how long it is. The advantage is that the length of each pointer is held constant instead of fixing the length of the entry, and the data entries can be spliced one after the other with no place holders.

For example, if the data used in the array structure I originally described were added this way, the two fields would read as follows:

```
Data:   catbirdpig
Pointer: 0000400400000100300008003
```

(With this data, ARRAY.DTF will actually "waste" more memory than the original implementation because ARRAY.DTF is designed to efficiently hold much larger entries of up to 999 characters. If we used the one-field implementation to support that length of entry, these three entries would already have consumed 3006 characters.)

The first field is simply all entries added together. New entries are always tacked onto the end. The second field is a series of three eight-character pointer values. The first pointer says that the first element in the ordered array begins at the fourth position in the data field and is four characters long (*bird*). The second eight-character entry says that the second element in the ordered list (*cat*) begins at the first position in the data field and is three characters long. Finally, the third pointer shows that the third element in the ordered list (*pig*) begins at the eighth position in the data field and is three characters long.

You should be aware of two issues when designing and utilizing arrays. First, in the code given above and in ARRAY.DTF, the array starts with element 0. Accordingly, to get the first element in an array, you need to ask for element 0. (The nth element is *element –1*). If this isn't clear, plug 0 and 1 into some of the functions until it makes sense. Also, a good implementation will use @Len to make sure that adding the entry won't cause the data field to exceed Q&A's maximum 32K field length.

ARRAY.DTF includes the following five functions crucial in any array-handling tool.

- *AddToArray* adds the text in Value to Data and places a value in Pointer that shows where the new element begins in Data and what its length is. The value is added to Pointer such that all entries in Data can be extracted in alphabetical order by moving through Pointer from beginning to end.

- *IsPresent* searches the existing data in the array for an exact match to the data in the Value field. If an exact match is found, Value is set to 1. If an exact match isn't found, Value is set to 0.

- *Search* returns to Value the first valid insertion point found for the text in Value. Note that if the text in Value is an exact match of an entry in Data, the first valid insertion point found may come either before or after the duplicate.

- *Get[n]* returns the nth element in the ordered array, where *n* is equal to @Tonumber(Value).

- *Remove[n]* removes the nth element from the array by removing the actual data and then removing the pointer. This function also adjusts all pointers so that they still point to their original data.

Another function, *AddToArray1*, is actually a binary search function that returns a valid insertion point for the value contained in SearchValue. It can't be called directly but must be called by another function.

New functions can, of course, be built from the ground up, though it might be possible to build any new functions you need by combining functions from the original six. For example, you could combine *IsPresent*, *Search* and *Remove[n]* to remove a particular value. Or, you could use *Search* in combination with *Get[n]* and some of your own code to retrieve a value that's either the next lower or next higher value in the array. One candidate function would write selected entries to a blank text field so that they could be printed in a report.

This article is an overview of how to implement an array in Q&A. Unfortunately, it isn't possible to develop a single array structure that's optimized for all programming situations. If you need an array specifically designed to provide a stack or a queue, you'll need a different implementation than an array that simply stores a sorted list. You might also want to have each entry hold more than one piece of information. Next month, I'll discuss how to accomplish this and how to develop arrays that themselves contain two-dimensional (and even three-dimensional) arrays. For many requirements, though, a simply one-dimensional array can offer an entirely new and sufficient set of options.

Mark Myers is a Programmer/Analyst with Bluegreen Corporation in Boca Raton, Florida.

@Help. . .

When a database locks up at Recover phase 2, it's usually a symptom of one of the following: (1) a badly damaged form design, (2) corrupted record, or (3) severely damaged index file. *Never* attempt to recover a database on the server from a workstation. Always copy it to the local drive, make the repairs, then copy it back. Once the database is damaged as above, recovery will *not* fix it. Fortunately, you have some options. First, extract the data using QEXTRACT (available from the Free Downloads page at *www.quickanswer.com*) or John Dow's DTFDOCTR utility. If you have an old undamaged copy of the database, make it generation one. (See Alec Mulvey's article in the November and December 1998 issues), remove any old records, and import the data into it.

If you don't have an old copy of the database, use QEXTRACT to extract the form design and programming in addition to the data. (DTFDOCTR extracts report designs.) Then design a brand new database. At the blank design screen, press F8 and select Document / Insert. Select the exported form design from QEXTRACT, press F10, format your fields, and you'll have a healthy new database. You can then paste in the reports using DTFDOCTR, import your data, copy and paste any programming into the pertinent fields, and you'll be back in business.

Bill Halpern and Gordon Meigs own Professional Computer Technology Associates (PCTA) in Newtown, Pennsylvania. PCTA specializes in data management and supporting systems and services. 215-598-8440, 71023.356@compuserve.com

Catch us Online at
www.quickanswer.com

Greasing the Wheels of Progress

Since 1985 I've used and programmed in Q&A. Recent employment changes have forced me into other systems such as ERP, MRP, and MRPII. In each case I found Q&A the ideal tool for migrating existing data into these new Super Systems my employers chose to pay for. Q&A can import almost anything—and do it without adding headers to the files—and accept data from almost any source. Then you can export it to Access, Excel, Alliance MRP, Avante—you name it. By adding a $99 barcode scanning wand as an interface, you can scan any field into Q&A, perform magical field computations, XLookups, or whatever on it, and take advantage of barcoding into high-priced systems that don't have these capabilities. Q&A can be locgical a link for a data interchange not possible even with million dollar systems.

　　　　　　—Wayne Lund, from the Delphi Forum on Q&A

I heartily echo your comments. I use a specially-designed relational xbase database system here. Whenever I have to perform imports, reformatting of incoming client data, special file format exports, or the like, Q&A is the tool I use. We export xbase records for our clients on hourly, daily, weekly, and monthly bases in an unattended automated environment. Each such export includes an associated "Audit Report." Guess what we use for the reports? Yep. What an excellent tool for many purposes. I think of Q&A as that multi-purpose tool—a spoon, screwdriver, pliers, knife, and wire-stripper all rolled into one. If I've got a job to do, my first attempt is with Q&A.

　　　　　　—Greg, from the Delphi Forum on Q&A