



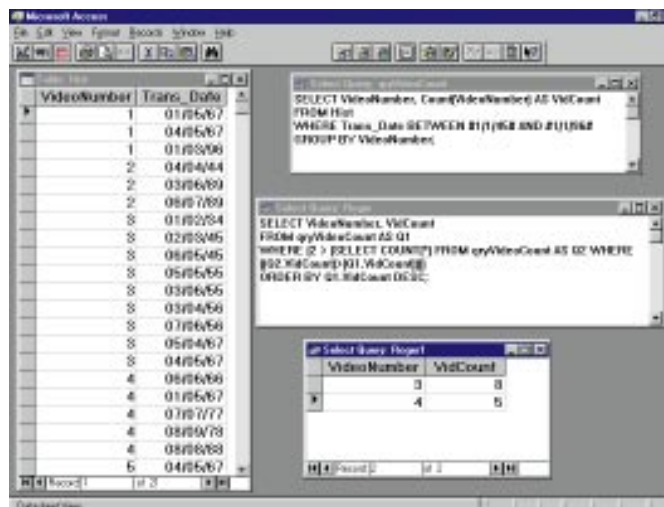
The video **top ten**, again...

A recent query regarding video rentals has prompted some keen response from readers. Mark Whitehorn presents some of your solutions, plus some general tips and teasers.

In the October issue I published a couple of questions from readers. The first concerned a video rental company which wanted to find out the most popular videos rented out over a specific period, essentially the "Top Ten" videos for that period.

Several people sent in further suggestions, including the following from Roger Moran: "By 'standard' SQL, I assume you are referring to the SQL/92 standard, and explicitly excluding all the various vendor-specific extensions such as the 'TOP' command in Microsoft Access SQL."

No, but the misunderstanding is my fault for not being accurate in my choice of



Here is Roger's solution. The only change I have made is to include the dates into the first SQL statement so that the query runs without putting up dialogue boxes

Dear Santa...

Last year, I asked you to make our current crop of RDBMSs a better match to the relational model, and also for more and better database design tools. I don't know about the rest of the kids, but frankly, I was just a little disappointed.

None of the RDBMSs has shown a marked improvement, and the tools situation is still dire.

Sigh. Perhaps I'm asking too much. After all, the vendors don't seem to be interested in what I asked for either, so you must face an uphill struggle in trying to persuade them. This year, I'll try for really achievable requests, but first, a bit of background. Many database professionals are having their serious work disrupted by trivial requests from the management about putting data onto this "web" thing. Now, we all know that the web is just a passing fad. Just like CB radio, once the fuss has died down, I'm sure most people will go back to the telephone. But while it is fashionable, we have to look keen. So, what we need are sensible tools for handling data from databases on the web.

Borland has already made a good start with IntraBuilder, and Microsoft is following with the improved web-publishing abilities of the Office products, but there is still a huge way to go. Neither of the products addresses the problems of signalling between the browser and the database to allow proper transaction control. This is not to say that transaction control is impossible, just that it is currently all left to the programmer. There are other problems as well. With present technology, if the browser sends a query which returns, say, 140 records, those are usually buffered at the web server and sent to the browser in chunks of, say, ten. This is crazy! Every time the user wants to see another ten, there is a pause as the server is contacted and the next ten are sent. This problem isn't going to be solved by a single product; it needs to go right to the standards that these technologies use. If you could wangle your way onto the appropriate standards committee to address this one, we would all be eternally grateful.

By the way, I'd still really like a mouse mat with a penguin on it. I mentioned this last year, but it must have got lost in the sleigh. Perhaps you could check under the seats?



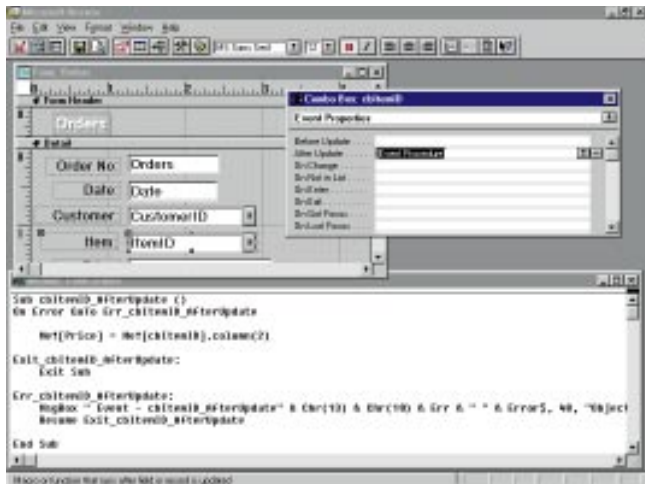
words. Eamonn Mulvihill's original request was for a solution which worked with Borland's Local-SQL which he said didn't support nested SELECT statements. I should have said something like "lowest-common denominator SQL" rather than "standard". Roger Moran continues: "Since MS Access seems to be your first choice when describing problems in your column, I will use MS Access SQL syntax in the following, although the SQL is easily ported to any ANSI-compliant SQL/92 dialect."

"We start with an aggregate query to provide us with a count of transactions per video within a given date range. This is trivial:

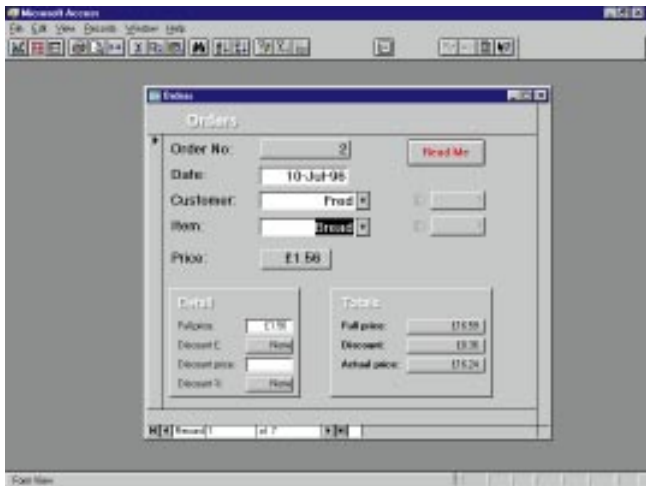
```
SELECT VideoNumber,
       Count(VideoNumber) AS VidCount
FROM Hist
GROUPBY VideoNumber
WHERE Trans_Date BETWEEN
[StartDate] AND [EndDate];"
```

I know it will sound picky, but although the sentiment behind this is quite clear, I think the syntax for Access actually needs

(right) Tony Wall's solution, showing the form and the code behind the combo boxes



(below) Geoff's form, which also totals the orders



number of records in qryVideoCount (Q2) whose VidCount is greater than that of the 'current' record in Q1. If that number is less than ten, we know the 'current' record should be included in our 'Top Ten' list. We apply the test to each record in Q1 and the result will be the 10 desired records.

```
SELECT VideoNumber,
Count(VideoNumber) AS VidCount
FROM Hist
WHERE Trans_Date BETWEEN StartDate
AND EndDate
GROUP BY VideoNumber;
```

"In MS Access, we can use this query directly as the datasource for the second stage. In other products which do not support daisy-chaining queries in this fashion, it is a simple matter to convert the above query into an append query which populates a temporary working table, and then use that as the datasource instead.

"Let's assume we have saved the above query with the name [qryVideoCount]. The 'Top Ten' query will look as follows:

```
SELECT VideoNumber, VidCount
FROM qryVideoCount AS Q1
WHERE (10 > (SELECT COUNT(*) FROM
qryVideoCount AS Q2 WHERE
((Q2.VidCount)>(Q1.VidCount))))
ORDER BY Q1.VidCount DESC;
```

"Here's a basic description of what's going on in the above query. For each record in qryVideoCount (Q1), we find the

"The aforementioned query works, and is written in standard SQL, but it cannot be considered very efficient, since for each record in qryVideoCount (between 1 and 12,000 in Eamonn Mulvihill's situation), we must execute the sub-query which determines whether the record is to be included in the 'Top Ten' list. In a real world situation, it would probably be quicker to write some procedural code to pull out the ten records from qryVideoCount.

"Hope this information is useful." Yes, it certainly is. It won't do as a solution for Eamonn's problem because of the nested SELECT, and I agree that it is likely to have speed problems on a large data set. However, Roger is quite right that his is a solution comfortably within SQL-92 and I have great admiration for its elegance!

You can clearly tweak the second SQL statement very easily to alter the number of records that it returns, simply by changing the numeric value in the line:

```
WHERE (10 > (SELECT COUNT(*) FROM
qryVideoCount AS Q2 WHERE
```

I have changed it to two in the sample database (ROGER.MDB), because the HIST

table has so little data in it.

Alistair Logie sent in this solution for FoxPro users faced with a similar problem: "I read with interest your PCW Databases column about SQL selections of the top ten rented videos. One solution in FoxPro would be to use two SQL statements and a Cursor:

```
SELECT VideoNumber,
COUNT(VideoNumber) AS VidCount ;
FROM hist ;
INTO CURSOR mytable ;
GROUP BY VideoNumber ;
ORDER BY VidCount DESC

SELECT VideoNumber, VidCount ;
FROM mytable ;
WHERE RECNO() < 11
```

"I don't know if this is usable on other databases. It does rely on being able to use the absolute record number, RECNO(), as a selection field. The attraction of using a Cursor as a temporary file is that FoxPro creates and erases it itself."

The second question in the October issue covered a more general problem. Suppose you have a set of Customers and Items for sale. Given that each item has a specific, unique price, it is clear that the price fits into the ITEMS table. If each Customer has a unique price for each item, it is equally clear that the price needs to go into a separate table which lists each customer, item and the unique price for that key value.

What if the situation is somewhere in between, where most prices are standard but there are some exceptions? I produced a solution, but asked for suggestions, improvements or comments. The first comes from Tony Wall, and his database is on the CD as TONYWALL.MDB.

"I have enclosed an alternative solution to the Customer & Prices problem from the October issue. Whether it will work efficiently with the 400 customers and 300+ products is another matter.

"In the first instance I took your Default List and Exception List and used these to create a union query as follows:

```
SELECT *
FROM [PriceList] as PL
WHERE NOT EXISTS
(SELECT *
FROM [ExceptionList] AS EL
WHERE PL.[CustomerID] =
EL.[CustomerID] and PL.[ItemID]
=EL.[ItemID]);
UNION SELECT *
FROM [ExceptionList];
```

"This is used as the basis for the combo box data source on the orders form for retrieving the correct price."

Tony then sent another mail message, saying that the performance was terrible with a larger data set. His suggested improvement is well worth looking at, and is in TONY2.MDB. Geoff Wyss sent in another; see GEOFWYSS.MDB.

"In the October 96 issue of PCW, you posed a problem about merging prices from two sources to make up an Order from a default price list and an exceptions list. The attached database file shows two possible solutions which might be of interest, one using a Union query, the other using an IIF statement. Click on form Orders2. There is a ReadMe button on the form which gives an explanation."

For this month's problem, I have held off producing a solution myself. This is mainly due to cowardice, since this problem brings us once again into the holy wars of "strict relational" vs. expediency. I tend towards a purist view, but in this case I can see reasons for a more heretical solution. Before committing myself to be burned at the stake, what do you think? Let's try not to get too wound up about this one. It is Christmas, after all.

"I am a teacher, and recently used Microsoft Access 2.0 to produce a database for recording and reporting on pupils' performance in tests. To keep the database as flexible as possible in use, I store test results in a single table with four fields: each record in the Scores table includes an identifier for the pupil, an identifier for the test, the percentage score achieved by the pupil in the test, and the position of the pupil out of his/her teaching group. Records are added to the table as data are entered by the class teacher. (The pupil and test identifiers together make up the primary key, as each pupil appears in the table many times, as does each test, but each pupil can only take a given test once.)

"It is useful to have the computer calculate the position of the pupil automatically, once the data for a full teaching group has been entered, but I have not found a function in Access to do this easily. The pupils' teaching groups are recorded in a separate table. It is straightforward to produce a query listing for a particular test, the pupils in one teaching group and their percentage scores, and this query can be sorted appropriately. But then to work out the

Teasers

A couple of teasers from Jeff Jenkins:

1. Strange database design

A database I was working on stored amounts as positive numbers and had a separate field to indicate the sign of the number. This field had values of one or two to indicate positive or negative (don't ask why, I didn't design the database). The problem was how to extract the amounts as signed numbers. The solution I came up with was a mathematical one, selecting "amount*(3-(sign*2))". This works as "(3-(sign*2))" evaluates to 1 or -1. A bit of a rigmarole just to get a signed number out.

2. Summing only the positives

I needed to provide a total of all the positive numbers in a particular field in the database (e.g. all the accounts with a credit balance). I couldn't find an sql command to do this, so I resorted to another mathematical solution: "SUM((amount+ABS(amount)) / 2)"

Adding the absolute (unsigned) value of a number to itself comes out to zero for negative numbers, and dividing by two negates the effect of adding a number to itself, so the sum works. You can also sum up all the negative numbers with:

```
"SUM( amount - ( ABS( amount ) ) / 2 )"
Incidentally, the first one also works as
"SUM( ( ABS( amount ) +amount ) / 2 )"
but the negative version
"SUM( ( ABS( amount ) - amount ) / 2 )"
gives a positive total of all the negative
numbers.
```

position of the pupils within the group and store this in the appropriate field, I have resorted to a complex sequence of operations involving creating and deleting temporary tables.

"Using this method to calculate pupils' positions slows down the program significantly and, because of the use of temporary tables, has led to problems when two teachers using the network attempt to enter test scores at the same time. Is there a direct way to fill in the 'position' field in the Scores table? I have tried different techniques, but have not found a wholly satisfactory solution." Andy

Andy's test database (not containing real data about real children) is on the CD as ANDY.MDB.

•PCW Contacts

Mark Whitehorn welcomes readers' correspondence and ideas for the Databases column. He's on m.whitehorn@dundee.ac.uk