

By Susan Sales Harkins

When you summarize data, you probably create a report. And most of the time, that's the right way to go. However, if you just need to review data without the grouping and formats typically used in a traditional report, consider SQL's UNION operator. If all you need is no-frills data and a row of summary values—and you need it fast—try these techniques.

Using UNION to summarize data

Normally, you use SQL's UNION operator to combine data from two or more tables or queries. The resulting recordset is read-only and contains unique records from all of the data sources. There's one rule you must keep in mind when using UNION: The number and order of the columns in each data source must match.

But the UNION operator doesn't just combine data. You can also use it to display detail records and summary values. The following are a few terms that you'll see throughout this article:

- **Summary label:** A label that identifies the summarized value. For instance, if you use Sum(), you might use the label "Total" to describe the evaluated results.
- **Summary value:** The evaluated results of a function that summarizes the data in your query, such as Sum(), Avg(), and so on.
- **Summary row:** Both the summary label and the summary value, as a single row within the query's recordset.

When using UNION to summarize data, substitute the SELECT clause's field list element with the summary label and the function that will evaluate the summary value, as follows:

```
SELECT fldlist | * FROM ds
[ALL]
SELECT 'summarylabeltext', aggregatefunction FROM ds
```

Notice that this form uses only one data source (*ds*). You're not combining data from multiple sources; you're evaluating the data in a single source.

Let's look at a simple example using the Products table in Northwind (the sample database that comes with Access). We'll use UNION to display the average product price. To do so, complete these steps:

1. Click the Queries shortcut in the Database window.
2. Click New and then click OK.
3. Close the Show Table dialog box.
4. Choose SQL View from the View menu, shown in **Figure A**.
5. In the resulting SQL window, enter the following SQL statement:

```
SELECT ProductName, UnitPrice
FROM Products
UNION
SELECT 'Average Price', Avg(UnitPrice)
FROM Products
```

How do I... Display summary values in Microsoft Access using UNION?

Notice that the second SELECT statement's field list clause contains the summary label and a function that evaluates data.

Execute the query to display the datasheet shown in **Figure B**. The summary row is in the third record from the top. Using this form, you can't predict where Access sorts the summary row. Most likely, you'll want the summary row at the top or the bottom of the data.

Before we discuss possible solutions, there's one situation where the summary row *will* sort to the bottom of the data. When the column that displays the summary label stores numeric values, the summary row sorts to the end of the recordset if the summary label is text. You can see this by running the following UNION:

```
SELECT EmployeeID, Country
FROM Employees
UNION
SELECT 'Country Count', Count(Country)
FROM Employees
WHERE Country = 'USA'
```

As you can see in **Figure C**, Count() evaluates the number of employees in the United States, and Access sorts the summary row to the bottom of the data. That's ASCII sorting at work. UNION hasn't anything to do with it, but that doesn't mean you can't take advantage of the behavior.

In most situations, however, the summary row is going to get lost among the rows of data. Perhaps the easiest sorting trick is to precede the summary label with a space character and let the ASCII sorting behavior handle the problem. To see how this works, enter the following statement in the SQL window:

```
SELECT ProductName, UnitPrice
FROM Products
UNION
SELECT ' Average Price',
Avg(UnitPrice)
FROM Products
```

Notice that there's a space character before the leading delimiter (') and the first character in the summary label (A). **Figure D** shows the summary row at the top of the detail data.

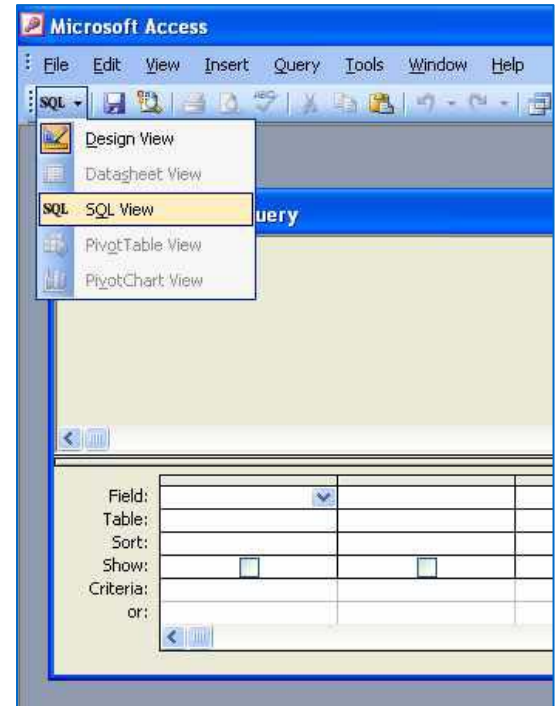


Figure A: Open the SQL window.

A screenshot of a Microsoft Access query window titled "Query1 : Union Query". It displays a table with two columns: "ProductName" and "UnitPrice". The data rows are: Alice Mutton (\$39.00), Aniseed Syrup (\$10.00), Average Price (\$28.87), Boston Crab Meat (\$18.40), Camembert Pierrot (\$34.00), Carnarvon Tigers (\$62.50), Chai (\$18.00), Chang (\$19.00), Chartreuse verte (\$18.00), Chef Anton's Cajun Seasoning (\$22.00), Chef Anton's Gumbo Mix (\$21.35), Chocolate (\$12.75), Côte de Blaye (\$263.50), and Escargots de Bourgogne (\$13.25). The "Average Price" row is highlighted. At the bottom, it says "Record: 13 of 78".

ProductName	UnitPrice
Alice Mutton	\$39.00
Aniseed Syrup	\$10.00
Average Price	\$28.87
Boston Crab Meat	\$18.40
Camembert Pierrot	\$34.00
Carnarvon Tigers	\$62.50
Chai	\$18.00
Chang	\$19.00
Chartreuse verte	\$18.00
Chef Anton's Cajun Seasoning	\$22.00
Chef Anton's Gumbo Mix	\$21.35
Chocolate	\$12.75
Côte de Blaye	\$263.50
Escargots de Bourgogne	\$13.25

Figure B: The summary row usually gets lost in the data.

A screenshot of a Microsoft Access query window titled "qryUNIONCount : Union Query". It displays a table with two columns: "EmployeeID" and "Country". The data rows are: 1 (USA), 2 (USA), 3 (USA), 4 (USA), 5 (UK), 6 (UK), 7 (UK), 8 (USA), 9 (UK), and Country Count (5). The "Country Count" row is highlighted at the bottom of the list.

EmployeeID	Country
1	USA
2	USA
3	USA
4	USA
5	UK
6	UK
7	UK
8	USA
9	UK
Country Count	5

Figure C: Text labels will sort to the bottom of numeric data.

How do I... Display summary values in Microsoft Access using UNION?

Sorting the summary row to the bottom of the detail records can be as easy as adding the ALL keyword. Doing so forces the query to inhibit any sorting. The downside of this method is that it includes all the records from all the data sources, and the query doesn't sort the data. If you can live with that, try the following:

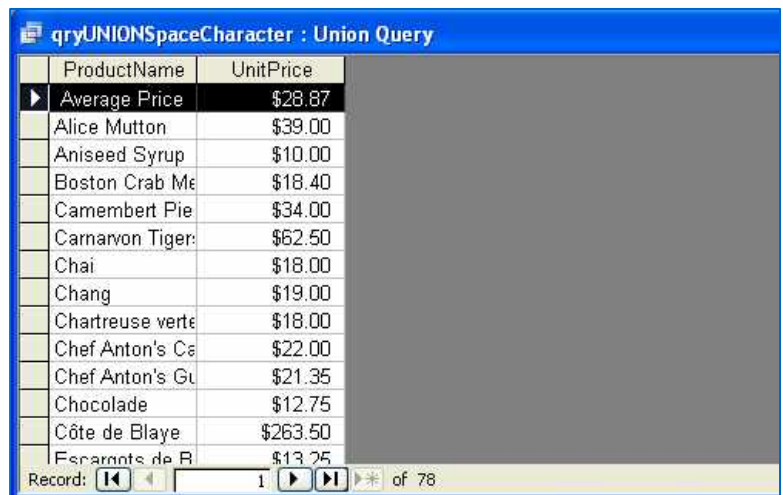
```
SELECT ProductName, UnitPrice
FROM Products
UNION
ALL
SELECT 'Average Price', Avg(UnitPrice)
FROM Products
```

Most likely, you *won't* want to inhibit the sort, as shown in **Figure E**.

The next method works if you don't really care what the summary label is, but you want the data sorted and you want the summary row at the bottom of the sorted data. Simply begin the label with something you know will sort to the bottom, such as the letter Z, as follows:

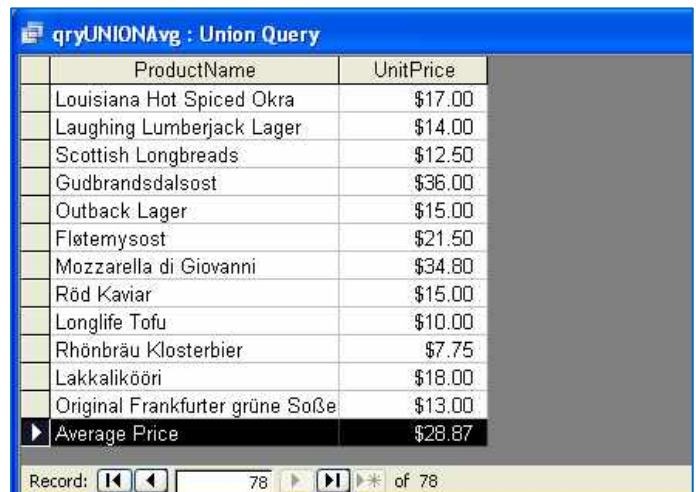
```
SELECT ProductName, UnitPrice
FROM Products
UNION
SELECT 'ZAverage Price', Avg(UnitPrice)
FROM Products
```

Figure F shows sorted records and the summary row neatly tucked in at the bottom of the data.



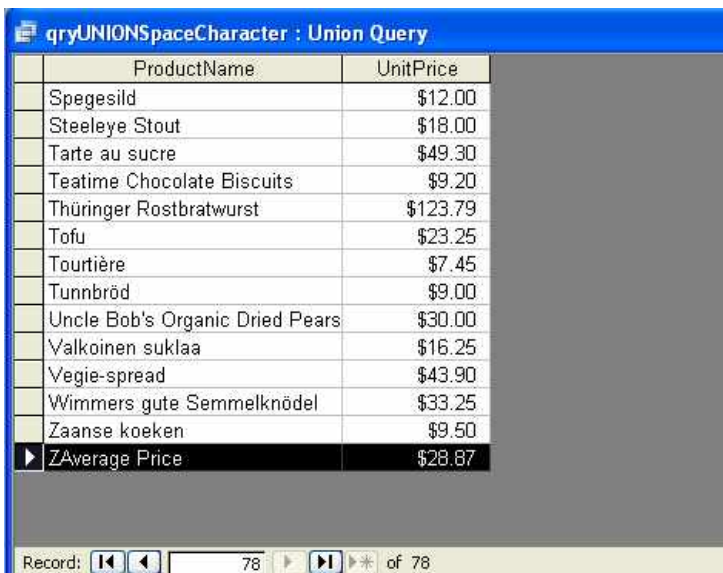
ProductName	UnitPrice
Average Price	\$28.87
Alice Mutton	\$39.00
Aniseed Syrup	\$10.00
Boston Crab Me	\$18.40
Camembert Pie	\$34.00
Camarvon Tiger	\$62.50
Chai	\$18.00
Chang	\$19.00
Chartreuse verte	\$18.00
Chef Anton's Ca	\$22.00
Chef Anton's Gu	\$21.35
Chocolade	\$12.75
Côte de Blaye	\$263.50
Ferarants de B	\$13.25

Figure D: Use a space character to sort the summary row to the top of the data.



ProductName	UnitPrice
Louisiana Hot Spiced Okra	\$17.00
Laughing Lumberjack Lager	\$14.00
Scottish Longbreads	\$12.50
Gudbrandsdalsost	\$36.00
Outback Lager	\$15.00
Fløtemysost	\$21.50
Mozzarella di Giovanni	\$34.80
Röd Kaviar	\$15.00
Longlife Tofu	\$10.00
Rhönbräu Klosterbier	\$7.75
Lakkalikööri	\$18.00
Original Frankfurter grüne Soße	\$13.00
Average Price	\$28.87

Figure E: Use ALL to display the summary row at the bottom of unsorted data.



ProductName	UnitPrice
Spegesild	\$12.00
Steeleye Stout	\$18.00
Tarte au sucre	\$49.30
Teatime Chocolate Biscuits	\$9.20
Thüringer Rostbratwurst	\$123.79
Tofu	\$23.25
Tourtière	\$7.45
Tunnbröd	\$9.00
Uncle Bob's Organic Dried Pears	\$30.00
Valkoinen suklaa	\$16.25
Veggie-spread	\$43.90
Wimmers gute Semmelknödel	\$33.25
Zaanse koeken	\$9.50
ZAverage Price	\$28.87

Figure F: If you don't care about the summary label, use the letter Z to force the summary row to the bottom of the data.

This trick is great for you, but you won't want to share it with a user or client. When that's the case, there's one last solution: Force the sort by adding a position column as follows:

```
SELECT ProductName, UnitPrice, 1 As
SortPosition

FROM Products

UNION

SELECT 'Average Price', Avg(UnitPrice), 2 As
SortPosition

FROM Products

ORDER BY SortPosition, ProductName
```

Adding a sort position to the field list allows you to control the sort. **Figure G** shows the result of the previous statement, which sorts the summary row to the bottom of the data. To sort the summary row to the top, reverse the values in the SortPosition columns.

ProductName	UnitPrice	SortPosition
Sirap d'érable	\$28.50	1
Spegesild	\$12.00	1
Steeleye Stout	\$18.00	1
Tarte au sucre	\$49.30	1
Teatime Chocolate Biscuits	\$9.20	1
Thüringer Rostbratwurst	\$123.79	1
Tofu	\$23.25	1
Tourtière	\$7.45	1
Tunnbröd	\$9.00	1
Uncle Bob's Organic Dried Pears	\$30.00	1
Valkoinen suklaa	\$16.25	1
Veggie-spread	\$43.90	1
Wimmers gute Semmelknödel	\$33.25	1
Zaanse koeken	\$9.50	1
Average Price	\$28.87	2

Figure G: Control the sort by adding a sorting position value to the field list clause on both sides of UNION.

Including nonsummarized data

You can display columns that contain no summary value or label, but you must account for them in the summary row's SELECT field list clause. Remember, the number of columns from both sides of the UNION must match. To accommodate this requirement, add a placeholder to represent the additional column(s) to the summary row's SELECT statement as follows:

```
SELECT ProductName, UnitPrice,
UnitsInStock, 1 AS SortPosition

FROM Products

UNION

SELECT 'Average Price',
Avg(UnitPrice), "", 2 AS SortPosition

FROM Products

ORDER BY SortPosition, ProductName
```

The zero-length character ("") acts as a placeholder for the UnitsInStock column on the summary side, as shown in **Figure H**.

ProductName	UnitPrice	UnitsInStock	SortPosition
Sirap d'érable	\$28.50	113	1
Spegesild	\$12.00	95	1
Steeleye Stout	\$18.00	20	1
Tarte au sucre	\$49.30	17	1
Teatime Chocolate Biscuits	\$9.20	25	1
Thüringer Rostbratwurst	\$123.79	0	1
Tofu	\$23.25	35	1
Tourtière	\$7.45	21	1
Tunnbröd	\$9.00	61	1
Uncle Bob's Organic Dried Pears	\$30.00	15	1
Valkoinen suklaa	\$16.25	65	1
Veggie-spread	\$43.90	24	1
Wimmers gute Semmelknödel	\$33.25	22	1
Zaanse koeken	\$9.50	36	1
Average Price	\$28.87		2

Figure H: Control the sort order by using a sort value.

Efficient UNION

In this article, you've learned how to use UNION to display data and simple summary values for that data. Unless you need special formatting and grouping, skip the traditional report and take the easy, quick route—the UNION operator.

Susan Sales Harkins is an independent consultant and the author of hundreds of articles and several books on database technologies. Her most recent book is Mastering SQL Server Express, with Mike Gunderloy, published by Sybex. Other recent books are Automating Microsoft Access 2003 with VBA, and Upgrader's Guide to Microsoft office System 2003. Currently, Susan volunteers as the Publications Director for Database Advisors at <http://www.databaseadvisors.com>. You can reach her at harkins1@setel.com.

Additional resources

- TechRepublic's [Downloads RSS Feed](#) **XML**
- Sign up for TechRepublic's [Downloads Weekly Update](#) newsletter
- Sign up for our [Microsoft Office Suite newsletter](#)
- Check out all of TechRepublic's [free newsletters](#)
- "[Access reference: Working with reports](#)" (TechRepublic download)
- "[Task sheet: Create calculated fields for Access queries](#)" (TechRepublic download)
- "[Hands-On Microsoft Access: A Practical Guide to Improving Your Access Skills](#)" (TechRepublic download)

Version history

Version: 1.0

Published: August 17, 2006

Tell us what you think

TechRepublic downloads are designed to help you get your job done as painlessly and effectively as possible. Because we're continually looking for ways to improve the usefulness of these tools, we need your feedback. Please take a minute to [drop us a line](#) and tell us how well this download worked for you and offer your suggestions for improvement.

Thanks!

—The TechRepublic Downloads Team