

Tech Support Tid Bytes: Speaking SQL

By Jennifer Mills

Anyone who has tried to understand the nuances of the English language realizes that the meaning of a single word can vary according to inflection, context, and geographic location. This linguistic marvel persists despite absolute standards set by committees, governments, and industries.

Understanding the meaning of today's ever-increasing technical terminology adds to the existing complications of language. When you are striving to resolve a problem or illustrate a new approach to an application, it is easy to assume that the other person understands. Without knowing it, the two of you may be discussing entirely different concepts. As more and more BASIS customers begin to use SQL and the BASIS ODBC Driver™, it becomes increasingly important for everyone to use the same definitions. This article attempts to clarify a few frequently-used SQL/ODBC terms that you will encounter.

Datasource vs. Database

Datasource and database are two terms that mean the same thing: a location that contains your data.

Database is defined by Merriam Webster as "a usually large collection of data organized especially for rapid search and retrieval." If only Mr. Webster had explained how to organize the data.

Rows, Columns and Tables

Information systems professionals have always referred to database components as files, records and fields, while accountants have always referred to corresponding spreadsheet components as tables, rows, and columns.

One easy way to remember the database-spreadsheet data relationship is to use the following association:

```
Tables ---> Files  
Rows -----> Records  
Columns --> Fields
```

Dependent vs Independent Indices

The terms independent and dependent refer to indices defined in your data dictionary. You can set either index type in ascending or descending order, but only PRO/5® uses independent indices.

A dependent index relies on a single column or multiple concatenated columns. An independent index may span multiple columns, or may be made up of parts of multiple columns.

Inner Join vs Outer Join

A join occurs when you select data from two tables that have one or more columns in common. For example, each table in the following illustration contains a CUSTID column:

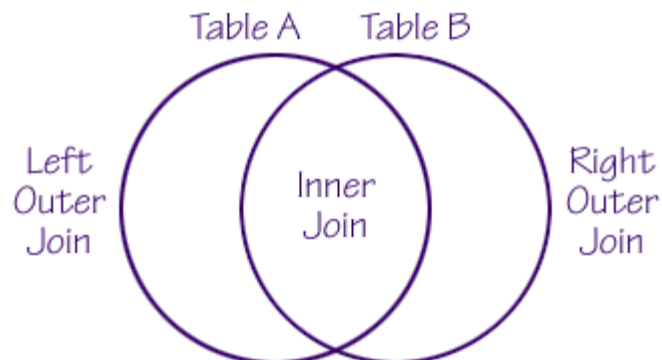
CUST_HEADER	CUST_ORDER
CUSTID	CUSTID
COMPANY	ORDER_NUMBER
CONTACT	PO_NUMBER
TELEPHONE	ITEM_NUMBER
DATE_ESTABLISHED	TOTAL_PRICE
And other information	And other information

If customer XYZ places an order that exceeds its credit limit, you can create a SQL query to retrieve the company's order history and other information to help you decide whether or not to raise the credit limit:

```
SELECT CUST_HEADER.CUSTID, COMPANY, ORDER_NUMBER,  
DATE_ESTABLISHED, PO_NUMBER, TOTAL_PRICE FROM CUST_HEADER,  
CUST_ORDER WHERE CUST_HEADER.CUSTID='12345' AND  
CUST_ORDER.CUSTID=CUST_HEADER.CUSTID
```

From the CUST_HEADER table, this query retrieves the row in which the contents of the CUSTID column equals 12345. From the CUST_ORDER table, the query retrieves all rows in which the contents of the CUSTID column equals 12345, and includes the contents of the ORDER_NUMBER, PO_NUMBER, ITEM_NUMBER, and TOTAL_PRICE columns.

The join occurs on the CUSTID column because both tables contain this column. As displayed in the graphic below, the CUSTID column falls into the **inner join** category. The other areas of the circles fall into the **outer join** category and can be further defined with Right Outer Join and Left Outer Join terminology, although this syntax is currently unsupported by the BASIS ODBC Driver.



Communication, the Most Important Term

Since SQL grammar and syntax play such an important part in the computer industry, it's extremely important to foster clear, concise communication. Terms should be easy to understand and remember. Hopefully, this brief discussion has helped clarify terms you will encounter during your endeavors with SQL and ODBC.