SQL FOR dBASE IV
A User's Tutorial

Jonathan Sayles

SQL FOR dBASE IV
A User's Tutorial

Jonathan Sayles
with Edwin F. Kerr

Edited by Meva Eringen

QED Information Sciences, Inc.
Wellesley, Massachusetts

QED Plaza
P.O. Box 181
Wellesley, MA 02181

Library of Congress Catalog Number: 88-35086
International Standard Book Number: 0-89435-289-X

Printed in the United States of America
89 90 91 10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data

Sayles, Jonathan,
SQL for dBASE IV / Jonathan Sayles: with Edwin F. Kerr edited by Meva Eringen
p. cm.
ISBN 0-89435-289-X
1. Data base management. 2. dBASE IV (Computer program) 3. SQL (Computer program language) I. Eringen, Meva. II. Kerr, Edwin F., 1933- . III. Title. IV. Title: SQL for dBASE 4. V. Title: SQL for dBASE four.
QA76.9.D3S26 1989 88-35086
005.75'65— —dc 19 CIP

Preface

Let's see now, you're (select any from the following list where they apply):

( ) too busy
( ) too old
( ) too young
( ) overextended
( ) already way over budget
( ) not convinced of the need to overcome the inertia to learn, or get some training in, SQL and relational database technology.

I sympathize; after all, the "law of diminishing amusement" takes a sizeable amount of octane out of our professional enthusiasm over the years. And it isn't long before the prospect of learning a new computer language rates right up there with doing income taxes.

But you see, learning SQL will be different. It's:

1. *Easy to learn.* Little did you know that you've already begun your first class in SQL! By reading the opening sentence of this preface, you were exposed to four SQL reserved words (SELECT, ANY, FROM, WHERE). What's more, they mean the same thing to SQL as they do to you.

2. *Easy to use.* Whether you're running a query written in SQL or writing a program that contains embedded SQL statements, the language is consistent, logical, and straight-forward.

3. *Necessary.* SQL is IBM's strategic direction in relational database languages and the ANSI standard for relational languages. SQL-based DBMS and 4GL application development tools are proliferating like rabbits on the market, in micro, mini, and mainframe products such as Gupta's *SQLBase®*, *ORACLE*, *INFORMIX*, *INGRES*, *SQL/DS*, dBASE IV, and (lest we forget) *DB2*. In short, SQL is the first language since COBOL that virtually everyone in the industry "agrees on."

This is a book specially tailored for dBASE IV users. In it, we'll cover the heart of SQL instructional material. Along the way, you'll learn what can be done with SQL, how it's used, and how to use it to accomplish SQL query processing. All of the exercises may be typed in and executed on dBASE IV.

In each chapter, you'll get an explanation of a command or function from the book (which is also your personal SQL reference manual). Then, you'll conduct a dry run by writing the statement in the book. Finally, you'll practice by using SQL to access the data tables. An answer key is provided, so that you can check your answers.

The book is divided into twelve chapters.

In Chapter 1, relational terms are defined in simple English, and concepts are introduced to give you a frame of reference regarding relational data objects and the SQL language.

Chapter 2 shows how to switch to SQL from dBASE and how to CREATE, SHOW, START, STOP, and DROP a database.

Chapter 3 introduces the case you will use and shows how to enter the data tables.

Chapter 4 covers the basics of the SELECT statement, and discusses the procedures for using the book.

In Chapter 5 you will learn how to use the SQL WHERE clause to retrieve rows based on search criteria.

Chapter 6 presents the SQL ORDER BY clause, and discusses ways to sort your output.

In Chapter 7, you will learn how to combine two or more search conditions into a single query with AND and OR. Chapter 7 also presents SQL math expressions.

Chapter 8 covers the relational operators BETWEEN, IN, and LIKE.

In Chapter 9, you will learn how to combine tables and merge or combine information from two separate sources in a single query.

Chapter 10, presents an introduction to SQL data definition statements — specifically the CREATE, DROP, and ALTER verbs. In Chapter 10, you will learn how to CREATE tables, delete tables, and modify table structure.

Chapter 11 covers relational data update capabilities: how to maintain the information in your database. Three SQL statements will be presented: INSERT, DELETE, and UPDATE.

Chapter 12 covers some important advanced topics such as views, synonyms, indexing, SQL aggregate (column) functions and GROUP BY/HAVING.

At the time we wrote this book, Ashton-Tate acknowledged that the SQL portion of dBASE IV can produce inconsistencies and inaccurate results. Sometimes the same query can produce two different results. We did not experience any difficulty with the exercises in this book. Therefore, if your answer appears wrong, first check your SQL statement carefully. The major problem we experienced is that SQL for dBASE IV does not support NULLs. A NULL is a blank field (no data) which indicates there is no value for the field. All implementations of SQL that we have worked with except SQL for dBASE IV handle NULLs differently than zero fields. For example, a NULL field is not processed when you use the AVG function. We have designed our exercises so this will not be a problem for you. If Ashton-Tate fixes this problem, then you can enter the tables with NULL (blank) fields where we have placed 0.00.

We cover a great deal of material in this book. We suggest that once you become proficient with the material, you will be able to use the more advanced features provided by dBASE IV SQL.

It is a great pleasure to acknowledge the help received in writing this book. I am grateful first of all to several friends and colleagues, especially Meva Eringen, my editor, for her technical excellence and disciplined approach to the production of this work, and Ed Kerr of QED Information Sciences, Inc., for his relentless support and help. Ed spent hours rewriting my original manuscript, *SQL as a Second Language*,* to tailor it for dBASE IV. Also, my thanks to Beth Roberts of QED for her efforts in coordinating the effort and making it all happen.

*Published by QED Information Sciences, Inc.

I would like to thank the management of The Systems Group, specifically John Coatti and Carl Foster for providing a stimulating and supportive atmosphere for the writing of this book.

Personal thanks are extended to Richard LoBrutto for his encouragement and to Dr. Melvin Goldstein, without whose help this book would not have been possible.

Last, but most importantly, I would like to dedicate this book to my wife, Maryellen. Without her humor, selflessness, and patience, the book could never have been written.

Jonathan Sayles
February, 1989

Table of Contents

SQL COMMAND INDEX

Chapter 1
RELATIONAL TERMS AND CONCEPTS

Chapter 1 presents SQL, its capabilities, relational terms, and concepts. Specifically, relational data objects—what they are, what they do, and how they interact with the system—are discussed from a practical, as well as a theoretical point of view.

1.1 WHAT IS SQL?

The name SQL stands for Structured Query Language.

Quite simply, SQL (pronounced "sequel") is a relational database language. Like any language, it is used to communicate. In this case it can data stored as logical sets called tables that in turn are stored in databases. In a relational database, all data is defined in tables which provide a logical, easy-to-picture way to update and make inquiries. dBASE IV understands SQL, and as you learn to use SQL, you will find yourself doing some fairly difficult data manipulation.

SQL is easy to learn. Despite the fact that SQL is a computer programming language, it is much simpler than traditional programming languages like COBOL, BASIC, FORTRAN or APL. You will probably find learning SQL to be a straight-forward process. Why?

First, SQL uses a free-form, English-like structure for its commands (a nonmathematical syntax). You SELECT your data, FROM some table, WHERE certain conditions are met. Or you INSERT your VALUES, INTO some table, or DELETE data FROM some table, WHERE your conditions are met. It is very logical and easy to follow.

Second, SQL decides how to get your data to and from the database. All you have to specify is what you want, and SQL does the rest. This is called nonnavigational. Sometimes entire programs in traditional data processing systems can be replaced with a single SQL query.

Third, dBASE IV supports both interactive and embedded SQL processing. That is, SQL statements can be executed in an interactive fashion — where you talk directly to dBASE IV — SQL statements can be embedded in dBASE IV program files. By itself, SQL has no commands for screen dialog, or for more than crude report formatting, so this dual-mode feature is very important. dBASE IV commands can be used to design forms, set up the printer and print reports.

Finally, there are less than 30 commands to learn. You do not need all of these if you are not going to be doing complex processing. You will have all the power you need to select, change, delete, and insert data into your tables.

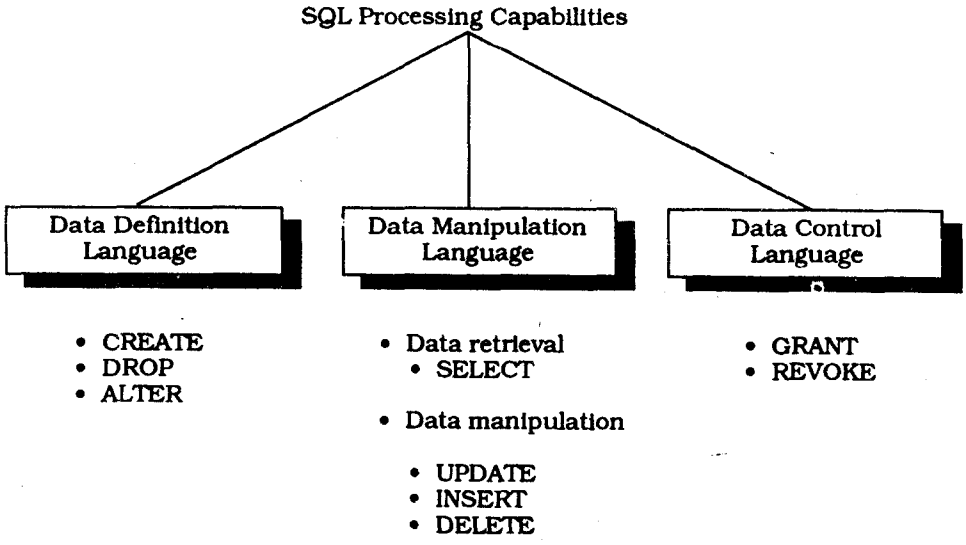

Figure 1.1. SQL processing capabilities.

1.2 SQL PROCESSING CAPABILITIES

SQL is composed of a data definition language, a data manipulation language, and a data control language (see Figure 1.1). These three languages support the complete spectrum of relational data processing activity.

Data definition allows creation, deletion, and modification of the data structures needed for our system. These structures include tables, databases, indexes—the "structural" parts of the database system.

Data manipulation is divided into three categories: retrieving data, manipulating data, and updating data. *Retrieving* data means querying the database. A selection of data stored on tables is presented on the screen. *Manipulating* data refers to the SQL features that allow us to perform statistical functions on data such as averaging and summing columns, and other arithmetic functions like multiplying columns. *Updating* data refers to inserting and deleting rows on tables and changing values in columns; in other words, database maintenance.

Data control allows us definition of a security mechanism or scheme for protecting the data in a system from unauthorized access. It consists of *Granting* and *Revoking* processing privileges on our data objects to other system users. Data access statements allow for as simple or elaborate a set of controls as is needed for the system.

An important feature of SQL statements is that they can be executed dynamically; that is, they may be executed any time the system is operating. To delete an existing database and define a totally new structure does not require the Herculean effort it currently takes using hierarchical or network database systems. Relational databases are simple to define, and once defined, simple to modify.

1.3 DATABASES

The difference between a database, a database management system, and a data access language can be a little confusing at first. Perhaps this description will help:

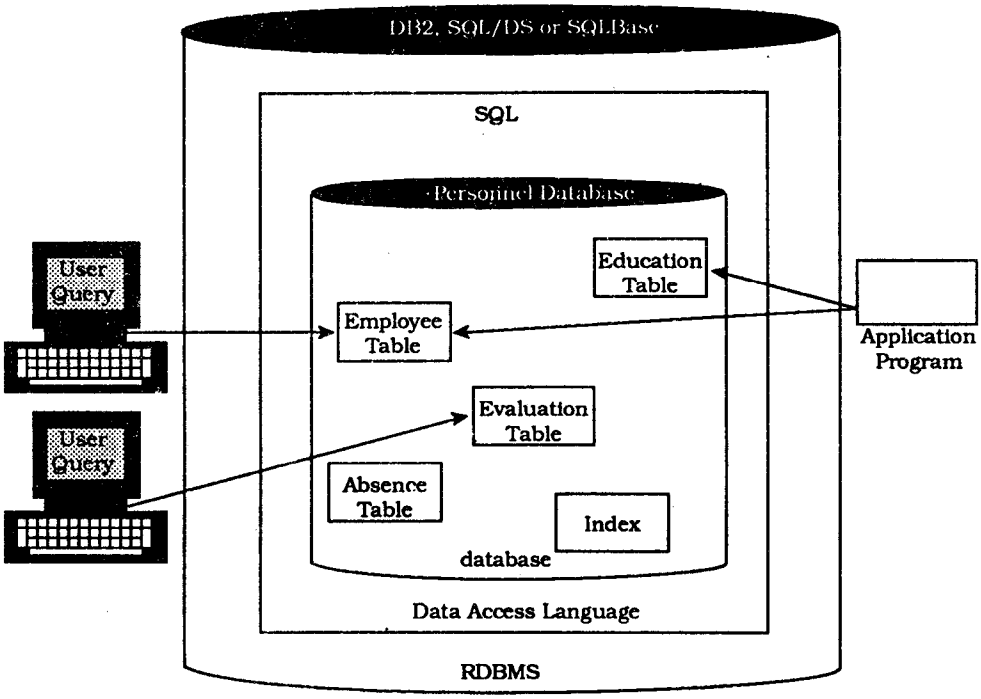

Figure 1.2. Personnel database.

- A *database* is simply an integrated collection of related files or related tables ;
- A *database management system* (DBMS) refers to the entire computerized file-keeping system such as dBASE IV.
- A *data access language* (SQL for instance) is the language system users employ to communicate with the DBMS to accomplish their work.