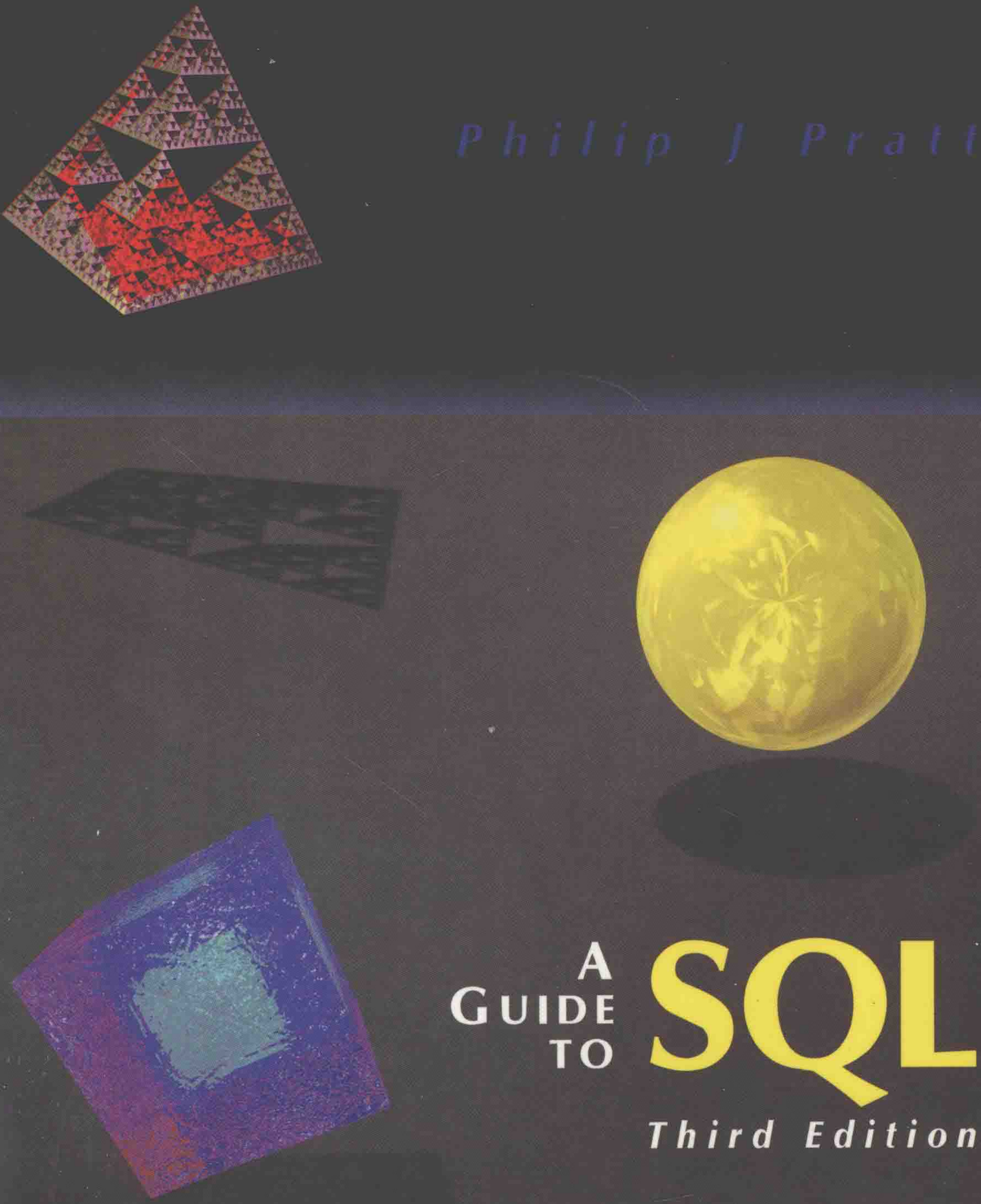

Philip J Pratt

A Guide to SQL

Third Edition

A Guide to SQL

Third Edition

PHILIP J. PRATT

Grand Valley State University

COURSE TECHNOLOGY

ONE MAIN STREET, CAMBRIDGE, MA 02142

an International Thomson Publishing company I(T)P®

Cambridge • Albany • Bonn • Boston • Cincinnati • London • Madrid • Melbourne • Mexico City
New York • Paris • San Francisco • Singapore • Tokyo • Toronto • Washington

Acquisitions Editor:	Anne E. Hamilton
Production Editor:	Jean Bermingham
Composition:	Ruttle, Shaw & Wetherill, Inc.
Interior Design:	Becky Herrington
Cover Design:	Richard Pepper/Flying Pepper Design
Manufacturing Coordinator:	Tracy Megison

© 1995 by Course Technology.
A Division of International Thomson Publishing – ITP®

For more information contact:

Course Technology
One Main Street
Cambridge, MA 02142

International Thomson Publishing Europe
Berkshire House 168-173
High Holborn
London WCIV 7AA
England

Thomas Nelson Australia
102 Dodds Street
South Melbourne, 3205
Victoria, Australia

Nelson Canada
1120 Birchmount Road
Scarborough, Ontario
Canada M1K 5G4

International Thomson Editores
Campos Eliseos 385, Piso 7
Col. Polanco
11560 Mexico D.F. Mexico

International Thomson Publishing GmbH
Königswinterer Strasse 418
53227 Bonn
Germany

International Thomson Publishing Asia
211 Henderson Road
#05-10 Henderson Building
Singapore 0315

International Thomson Publishing Japan
Hirakawacho Kyowa Building, 3F
2-2-1 Hirakawacho
Chiyoda-ku, Tokyo 102
Japan

All rights reserved. This publication is protected by federal copyright law. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, or be used to make a derivative work (such as translation or adaptation), without prior permission in writing from Course Technology.

Trademarks

Course Technology and the open book logo are registered trademarks and CourseKits is a trademark of Course Technology. Custom Editions is a registered trademark of International Thomson Publishing, Inc.

ITP® The ITP logo is a registered trademark of International Thomson Publishing, Inc.

Some of the product names and company names used in this book have been used for identification purposes only and may be trademarks or registered trademarks of their respective manufacturers and sellers.

Disclaimer

Course Technology reserves the right to revise this publication and make changes from time to time in its content without notice.

ISBN 0-87709-520-5

Printed in the United States of America

10 9 8 7 6 5

Structured Query Language (SQL) is becoming increasingly popular. SQL can be used by such diverse groups as home computer owners, owners of small businesses, end-users in large organizations, and programmers.

A Guide to SQL, Third Edition is intended for anyone interested in gaining familiarity with the SQL language. It is appropriate for students in introductory classes in computer science or information systems programs. It is also appropriate for anyone wishing to use SQL to access data in databases on anything from the largest mainframe to the smallest microcomputer.

This text can be used as a textbook for a stand-alone course on SQL or as a companion to a database textbook to furnish additional material on SQL. In either case, the textbook is designed to be covered from start to finish. Students should work the exercises at the ends of the chapters as they go along. The only exception to this would be in a course strictly for end-users. In such a course, instructors may not want to include the last two chapters because the material in them is rather specialized.

SPECIAL FEATURES

Use of Examples

The SQL language is presented through approximately 100 examples. Learning through examples is, for most people, the most effective way to master material.

Case Studies

A case study, the Chazy Associates database, is used throughout the examples. The same case is used in one set of exercises at the end of each chapter. Using a realistic case study makes the examples more interesting, and more real, to students. Using the same case study in a set of exercises gives a high level of continuity.

A second case study, the Movies database, is used in a second set of exercises at the end of each chapter. The second case study gives students a chance to venture out ''on their own'' without the direct guidance of examples from the text, thus enhancing the learning process.

Embedded Exercises

A special type of exercise, called a Q&A, is used throughout the text. These exercises are designed to be answered by students before they proceed with their reading. The intent of many of the Q&As is to make sure students understand key points in the text before they proceed. In other cases, the intent is to stimulate students to consider some special issues on their own before moving on to the presentation of these issues in the text. The answer to the Q&A is given immediately after the question. Students are encouraged to formulate their own answer before reading the one in the text. Students can thus make sure they have sufficient understanding of the material before they proceed.

Exercises

At the end of each chapter there are exercises in which students get a chance to use the features of SQL presented in the chapter to solve realistic problems. Each chapter contains two sets of exercises, one using the Chazy Associates database and one using the Movies database.

Appendix

There is a single appendix giving concise descriptions of the examples from the text. The statement of each example is included together with the SQL solution. This gives students a handy reference to the ideas covered in the text. If they want to use SQL to solve a particular problem, they can rapidly scan this index to find a similar example.

Instructor's Manual/ Transparencies

The accompanying instructor's manual contains detailed teaching tips, answers to exercises in the text, and test questions (and answers). Transparency masters are included for most of the figures in the text.

SQL2

This text includes coverage of the key features of the latest version of SQL, SQL2 (also called SQL-92).

ORGANIZATION OF THE TEXTBOOK

The textbook consists of eight chapters and an appendix.

1. Introduction

Chapter 1 introduces the concept of databases and database management systems. It also introduces the relational model as well as the two cases that are used throughout the text.

2. Data Definition

Chapter 2 covers the process of defining a database using SQL. Included in this chapter is a discussion of the role and use of nulls.

3. Single-Table Queries

Chapter 3 begins the presentation of using SQL to query a database. The queries in Chapter 3 all involve single tables. Included in this chapter are discussions of simple and compound conditions, the SQL word BETWEEN, computed columns, the SQL word LIKE, the SQL word IN, sorting, the SQL built-in functions, nesting queries, and grouping.

4. Multiple-Table Queries

Chapter 4 completes the discussion of querying a database by considering queries involving more than one table. Included in this chapter are discussions of the SQL words IN and EXISTS, subqueries within subqueries, the use of aliases, joining a table to itself, the SQL set operations, and the use of ALL and ANY.

此为试读，需要完整PDF请访问：www.ertongbook.com

5. Updates

Chapter 5 covers how to use SQL to update the data in a database. Included are discussions of how to change current data, add new rows, and create a new table from an existing one. In addition, the manner in which the structure of a database may be changed is examined in detail.

6. Database Administration

Chapter 6 covers the database administration features of SQL. These include views, the GRANT mechanism, indexes, and the use of the system catalog.

7. Embedded SQL

Chapter 7 covers the manner in which SQL can be embedded in a procedural language such as COBOL. Although COBOL is used as a vehicle to illustrate the concepts in this chapter, the material would apply equally well to any language that supports such an embedding. Included in this chapter are discussions of the use of embedded SQL to insert new rows, and change and delete existing rows. Also included is a discussion of how to retrieve single rows using embedded SQL and how to use cursors to retrieve multiple rows.

8. SQL2

Chapter 8 covers the important features of the latest version of SQL, SQL2.

Appendix: List of Examples

The appendix contains a concise list of all the examples presented in the text together with the associated SQL commands.

GENERAL NOTES TO THE STUDENT

Embedded Questions

At a number of places in the text, special questions have been inserted. Sometimes the purpose of these questions is to ensure that you understand some crucial material before you proceed. In other cases, the questions are designed to give you the chance to consider some special concept in advance of its actual presentation. In all cases, the answer to each question is given immediately after the question. You could simply read the question and its answer, but you will receive maximum benefit from the text if you take the time to work out the answer to the question and then check your answer against the one given in the text before you proceed with your reading.

End-of-Chapter Material

The end-of-chapter material consists of a summary and exercises. The summary briefly describes the material covered in the chapter. Scan the summary and make sure all the concepts are familiar to you. Following the summary are two sets of exercises. The first set uses the same Chazy Associates database that was used in the examples in the chapter. First work the exercises in this set to make sure you understand how the commands presented in the chapter are to be used. Then move on to the second set of exercises. The second set uses the Movies database and gives you a chance to apply what you have learned to a database that is not quite so familiar to you. (The answers to the odd-numbered exercises in both sets are given in the text.)

ACKNOWLEDGMENTS

I would like to acknowledge several individuals for their contributions in the preparation of this book.

I appreciate the efforts of the following individuals who reviewed the text and made many helpful suggestions: Ranjan B. Kini, University of North Carolina at Greensboro; Robert B. Norton, San Diego Mesa College; David Paradice, Texas A&M University; and Darleen V. Pigford, Western Kentucky University.

The efforts of the following members of the staff of boyd & fraser have been invaluable and have made this third edition possible: Tom Walker, President and Publisher; Anne Hamilton, Acquisitions Editor; Jean Bermingham, Production Editor; and Becky Herrington, Production Director.

BRIEF CONTENTS

CHAPTER 1: INTRODUCTION 1

CHAPTER 2: DATA DEFINITION 21

CHAPTER 3: SINGLE-TABLE QUERIES 31

CHAPTER 4: MULTIPLE-TABLE QUERIES 51

CHAPTER 5: UPDATES 71

CHAPTER 6: DATABASE ADMINISTRATION 83

CHAPTER 7: EMBEDDED SQL 109

CHAPTER 8: SQL2 133

CONTENTS

PREFACE xi

CHAPTER 1 Introduction 1

OBJECTIVES 1
INTRODUCTION 1
INTRODUCTION TO CHAZY ASSOCIATES 2
INTRODUCTION TO THE RELATIONAL DATABASES 5

Structure 5
Operations (Introduction to SQL) 8

THE MOVIES DATABASE 9
SUMMARY 16
EXERCISES (CHAZY ASSOCIATES) 17
EXERCISES (MOVIES) 18

CHAPTER 2 Data Definition 21

OBJECTIVES 21
INTRODUCTION 21
DATABASE CREATION 21

Database Description 21
Data Types 23

NULLS 23

What Are Nulls? 23
Implementation of Nulls 24

LOADING A DATABASE 24

The Basic INSERT Command 24
The INSERT Command with Nulls 26

SUMMARY 26
EXERCISES (CHAZY ASSOCIATES) 26
EXERCISES (MOVIES) 28

CHAPTER 3 Single-Table Queries 31

OBJECTIVES 31
INTRODUCTION 31
SIMPLE RETRIEVAL 31

Retrieve Certain Columns and All Rows 32
Retrieve All Columns and All Rows 32
Use of the WHERE Clause—Simple Conditions 33
Compound Conditions 34
Use of BETWEEN 35
Use of Computed Columns 36
Use of LIKE 37
Use of IN 38

SORTING 38

Use of ORDER BY 39
Sorting with Multiple Keys, Descending Order 39

BUILT-IN FUNCTIONS 39

Use of the Built-in Function COUNT 40
Use of SUM 41
Use of DISTINCT 41

NESTING QUERIES 43
GROUPING 44

Using GROUP BY 44
Using HAVING 45

HAVING vs. WHERE 46
NULLS 47
SUMMARY 48
EXERCISES (CHAZY ASSOCIATES) 48
EXERCISES (MOVIES) 49

CHAPTER 4 Multiple-Table Queries 51

OBJECTIVES 51
INTRODUCTION 51
QUERYING MULTIPLE TABLES 51

Joining Two Tables Together 51
Comparison of JOIN, IN, and EXISTS 53
Using IN 54
Using EXISTS 55
Subquery within a Subquery 57
A Comprehensive Example 58

USING AN ALIAS 59
MORE INVOLVED JOINS 60

Joining a Table to Itself 60
An Example Involving Joining All Five Tables 61

SET OPERATIONS 63

Union 64

Intersection 64
Difference 65

ALL AND ANY 65

Using ALL 65
Using ANY 66

SUMMARY 66
EXERCISES (CHAZY ASSOCIATES) 67
EXERCISES (MOVIES) 68

CHAPTER 5 Updates 71

OBJECTIVES 71
INTRODUCTION 71
CHANGING EXISTING DATA 72

Changing Existing Data in the Database 72
Using a Compound Condition 72

ADDING NEW DATA 73
DELETING DATA 74
CREATING A NEW TABLE FROM AN EXISTING TABLE 75
SETTING A COLUMN TO NULL 76
CHANGING THE DATABASE STRUCTURE 77

Alter 77
Difficult Changes 79
Drop 79

SUMMARY 79
EXERCISES (CHAZY ASSOCIATES) 80
EXERCISES (MOVIES) 80

CHAPTER 6 Database Administration 83

OBJECTIVES 83
INTRODUCTION 83
VIEWS 84

Defining and Using Views 84
Advantages of Views 87
Current Systems 91
Dropping a View 92

SECURITY 92
INDEXES 94

Purpose of an Index 94
Creating and Dropping Indexes 98

此为试读，需要完整PDF请访问：www.ertongbook.com

Dropping an Index 99
Index Discussion 99

THE CATALOG 99
INTEGRITY IN SQL 101
SUMMARY 103
EXERCISES (CHAZY ASSOCIATES) 104
EXERCISES (MOVIES) 105

CHAPTER 7 Embedded SQL 109

OBJECTIVES 109
INTRODUCTION 109
DATA DIVISION 110
PROCEDURE DIVISION 111

Retrieve a Single Row and Column 112
Retrieve a Single Row and All Columns 112
Retrieve a Single Row from a Join 116
Insert a Row into a Table 116
Change a Single Row in a Table 118
Change Multiple Rows in a Table 119
Delete a Single Row from a Table 119
Delete Multiple Rows from a Table 119
Multiple-Row Select—Cursors 119
More Complex Cursors 125
Advantages of Cursors 125
Updating Cursors 126

ERROR HANDLING 127

Checking SQLCODE 127
Using WHENEVER 128

SUMMARY 129
EXERCISES (CHAZY ASSOCIATES) 130
EXERCISES (MOVIES) 130

CHAPTER 8 SQL2 133

OBJECTIVES 133
INTRODUCTION 133
TRANSACTIONS 134
CASE 134
DOMAINS 135
JOINS 136
UNION, INTERSECTION, DIFFERENCE 139
INTEGRITY SUPPORT 140

USING SQLSTATE **143**
SUMMARY **145**
EXERCISE (CHAZY ASSOCIATES) **145**
EXERCISE (MOVIES) **146**

APPENDIX List of Examples **147**

ANSWERS TO ODD-NUMBERED EXERCISES **166**

INDEX **188**

CHAPTER

1

Introduction

OBJECTIVES

When you have completed this chapter, you should understand the following:

1. The Chazy Associates database, the database that will be used in examples and exercises throughout the rest of the text.
2. The structure of a relational database.
3. The Movies database, a database that will be used in exercises throughout the text.

INTRODUCTION

A **database** is a structure that can house information about many different types of objects as well as the relationships among those objects. This text uses two databases as examples. In the Chazy Associates database, the objects are personal computers (PCs), employees, software packages, and so on. In the Movies database, which will be used in some of the exercises, the objects are directors, movies, stars, videotapes, and so on.

In addition, both databases contain information about relationships among the objects. The Chazy Associates database, for example, contains information indicating which PCs are assigned to which employees, which packages are installed on which PCs, and so forth. The Movies database has information such as which director directed which movies, which stars appeared in which movies, which movies are on which tapes, and so on.

We need a special tool to manipulate such information easily. This tool is called a **database management system** or **DBMS**. A DBMS must furnish us a method for storing and manipulating information about a variety of objects and the relationships among the objects. A DBMS is often categorized by the general approach that it takes to do this. We are interested in the general category of

DBMS called **relational**. We are also interested in the most popular approach for manipulating data in a relational DBMS, the language called SQL.

We begin this chapter by examining the requirements of Chazy Associates. The database for this organization will be used in the examples throughout the text, so it is important for you to become comfortable with it. Next we investigate the basic concepts of relational databases. We look at how data is structured and how data in a relational database is accessed using SQL. In the remainder of the text, we examine SQL in detail.

INTRODUCTION TO CHAZY ASSOCIATES

A large, multinational corporation, Chazy Associates needs to maintain the following information about the PCs its employees use to do their jobs.

1. For each computer, Chazy Associates needs to store the manufacturer's name and model, the processor type, and the computer ID.
2. For each employee, it needs to store the employee's number, name, and phone number.
3. For each PC, the corporation needs to store the PC's inventory tag number, computer ID, location, and the number of the employee who is the primary user of the PC.
4. For each software package, Chazy Associates needs to store the package's ID, name, version, type, and current cost.
5. For each software package installed on a PC, the corporation needs to store the software package's ID and cost, inventory tag number, and the software installation date.

Figure 1.1 represents a relational model database for Chazy Associates. Figure 1.2 shows the same ,database, but filled in with sample data that we will use throughout the book.

For computers, we have columns for the computer ID, manufacturer's name and model, and processor type. For employees, we have the number, name, and phone number. For PCs, we have the inventory tag number and location. In addition, we have columns in the *PC* table for computer ID and employee number. Using the computer ID in the *PC* table, we can find in the *COMPUTER* table the specific manufacturer name and model and processor type for a particular PC. Thus, we see that PC 37691 (located in Sales) has a computer ID of B121, which is a Bantam 48X 486DX. On the other hand, by looking for all the PCs that have some specific number in the *COMPID* column, we can find all similar PCs. We see then that Chazy Associates has two M759 computers (Lemmin GRL 486SX) with tag numbers 32808 (located in Accounting) and 77740 (located in the home of employee 567).

In a similar way, we can use the *EMPNUM* column in the *PC* table to find

Figure 1.1 Chazy Associates relational database structure

COMPUTER

COMPID	MFGNAME	MFGMODEL	PROCTYPE

EMPLOYEE

EMPNUM	EMPNAME	EMPPHONE

PC

TAGNUM	COMPID	EMPNUM	LOCATION

PACKAGE

PACKID	PACKNAME	PACKVER	PACKTYPE	PACKCOST

SOFTWARE

PACKID	TAGNUM	INSTDATE	SOFTCOST

the primary user of each PC. As examples, the primary user of PC 32808 (located in Accounting) is employee 611 (Melissa Dinh), and employee 124 (Ramon Alvarez) is the primary user of PC 37691 (located in Sales) and 59836 (located in his home).

For software packages, we have columns in the *PACKAGE* table for the ID, name, version, type, and cost. Finally, for software packages installed on Chazy Associates' PCs, we have columns in the *SOFTWARE* table for the package ID that represents the specific software package installed, the tag number that identifies the particular PC, the software package installation date, and the cost of the software package. The first row of the *SOFTWARE* table tells us, for example, that software package AC01 from Boise Accounting was installed on 09/13/95 at a cost of $754.95 on PC 32808 (located in Accounting).

To test your understanding of the relational model database for Chazy Associates, answer the following questions using the data in Figure 1.2.

Q & A

Questions:

1. Give the tag numbers for all PCs that have the software package Words & More installed.
2. Give the location of the PC on which Chazy Associates has installed software

Figure 1.2

Chazy Associates sample data

COMPUTER

COMPID	MFGNAME	MFGMODEL	PROCTYPE
B121	Bantam	48X	486DX
B221	Bantam	48D	486DX2
C007	Cody	D1	486DX
M759	Lemmin	GRL	486SX

EMPLOYEE

EMPNUM	EMPNAME	EMPPHONE
124	Alvarez, Ramon	1212
567	Feinstein, Betty	8716
611	Dinh, Melissa	2963

PC

TAGNUM	COMPID	EMPNUM	LOCATION
32808	M759	611	Accounting
37691	B121	124	Sales
57772	C007	567	Info Systems
59836	B221	124	Home
77740	M759	567	Home

PACKAGE

PACKID	PACKNAME	PACKVER	PACKTYPE	PACKCOST
AC01	Boise Accounting	3.00	Accounting	725.83
DB32	Manta	1.50	Database	380.00
DB33	Manta	2.10	Database	430.18
SS11	Limitless View	5.30	Spreadsheet	217.95
WP08	Words & More	2.00	Word Processing	185.00
WP09	Freeware Processing	4.27	Word Processing	30.00

SOFTWARE

PACKID	TAGNUM	INSTDATE	SOFTCOST
AC01	32808	09/13/95	754.95
DB32	32808	12/03/95	380.00
DB32	37691	06/15/95	380.00
DB33	57772	05/27/95	412.77
WP08	32808	01/12/96	185.00
WP08	37691	06/15/95	227.50
WP08	57772	05/27/95	170.24
WP09	59836	10/30/95	35.00
WP09	77740	05/27/95	35.00

package AC01. Give the name of the employee who is the primary user of this PC.

3. List all the software packages installed on PC tag number 57772. For each package, give the installation date, name, version, and type.
4. Why is the column *SOFTCOST* part of the *SOFTWARE* table? Can't we just take the *PACKID* and look up the cost in the *PACKAGE* table?

此为试读，需要完整PDF请访问：www.ertongbook.com