Preface

Oracle PL/SQL by Example, Fourth Edition, presents the Oracle PL/SQL programming language in a unique and highly effective format. It challenges you to learn Oracle PL/SQL by using it rather than by simply reading about it.

Just as a grammar workbook would teach you about nouns and verbs by first showing you examples and later asking you to write sentences, *Oracle PL/SQL by Example* teaches you about cursors, loops, procedures, triggers, and so on by first showing you examples and later asking you to create these objects yourself.

Who This Book Is For

This book is intended for anyone who needs a quick but detailed introduction to programming with Oracle's PL/SQL language. The ideal readers are those with some relational database experience and with some Oracle experience, specifically with SQL and SQL*Plus, but with little or no experience with PL/SQL or with most other programming languages.

The content of this book is based primarily on the material that was taught in an Introduction to PL/SQL class at Columbia University's Computer Technology and Applications (CTA) program in New York City. The student body was diverse; some students had years of experience with information technology (IT) and programming, but no experience with Oracle PL/SQL, and others with absolutely no experience in IT or programming. The content of this book, like the class, is balanced to meet the needs of both extremes. The exercises in this book can be used as lab and homework assignments to accompany the lectures in a similar PL/SQL course.

How This Book Is Organized

The intent of this workbook is to teach you about Oracle PL/SQL by presenting you with a series of challenges followed by detailed solutions. The basic structure of each chapter is as follows:

> Introduction: New Features of PL/SQL in Oracle 11*g*
>
> Chapter
>
>> Lab
>>
>> Exercises (with answers and detailed discussion)
>>
>> Lab . . .
>>
>> Try It Yourself Questions

The Introduction summarizes new PL/SQL features introduced in Oracle 11*g* along with examples and references to chapters that cover these features.

Each chapter contains interactive labs that introduce topics about Oracle PL/SQL. The topics are discussed and then explored further though exercises, which are the heart of each lab.

Each exercise consists of a series of steps that you follow to perform a specific task, along with the questions and answers section. The questions are designed to help you discover important facets of PL/SQL programming on your own. The answers to the questions consist of more in-depth discussions of the concepts explored.

At the end of each chapter you find a Try It Yourself section, which consists of a series of projects designed to solidify all of the skills you have learned throughout the chapter. If you have successfully completed all of the labs in the chapter, you should be able to tackle these projects with few problems. You find guidance and solutions to these projects in Appendix D and at this book's companion Web site.

About The Companion Web Site

The companion Web site is located at:

http://www.informit.com/title/9780137144228

Two very important objects are found at the companion Web site. First, files you will need before you begin reading the workbook. All of the exercises and questions are based on a sample database called STUDENT. The files required to create and install the STUDENT schema are available from the Web site. Also, answers to the Try It Yourself questions are available to review.

You should visit the companion Web site, download the student schema, and install it in your database.

What You Will Need

Software programs as well as knowledge requirements are necessary to complete the exercise sections of this book. Note that some features covered throughout the book are applicable to Oracle 11g only. However, you will be able to complete a great majority of the exercise sections by using the following products:

Software

- Oracle 10g or higher
- SQL*Plus 10g or higher
- Internet Access
- Windows 2000/ME/XP/Vista

You can use either Oracle Personal Edition or Oracle Enterprise Edition to perform the exercises in this book. If you use Oracle Enterprise Edition, it can be running on a remote server or locally on your own machine. It is recommended that you use Oracle 10g or Oracle 11g in order to perform every exercise in this book. When a feature only works in the latest

version of Oracle database, the book will state it explicitly. Additionally, you should have access to and be familiar with SQL*Plus.

You have a number of options for how to edit and run scripts from SQL*Plus. There are also many third-party programs to edit and debug PL/SQL code. SQL*Plus is used throughout this book, because SQL*Plus comes with all versions of the Oracle database.

Using SQL*Plus

You should be familiar with using SQL*Plus to execute SQL statements (if not, then refer to another book in the Prentice Hall Interactive Oracle Series on this topic, Alice Rishchert's *Oracle SQL by Example, Fourth Edition*. There are a few key differences between executing SQL statements in SQL*Plus and executing PL/SQL statements in SQL*Plus. You will be introduced to these differences so that you can work with the exercises in this book.

You can end a SQL Command in SQL*Plus in one of three ways:

- With a semicolon (;)
- With a forward slash (/) on a line by itself
- With a blank line

The semicolon (;) informs SQL*Plus that you want to run the command that you have just entered. You type the semicolon at the end of the SELECT statement and then press return. SQL*Plus will process what is in the SQL Buffer (described next).

For Example

```
SQL> SELECT sysdate
 2 FROM dual
 3 ;

SYSDATE
---------
30-APR-08
```

The SQL Buffer

SQL*Plus will store the SQL command or PL/SQL block that you have most recently entered in an area of memory known as the SQL Buffer. The SQL Buffer remains unchanged until you enter a new command or exit your SQL*Plus session. You can easily edit the contents of the SQL Buffer by typing the EDIT command at the SQL prompt. The default text editor opens with the contents of the SQL Buffer. You can edit and save the file and then exit the editor. This causes the contents of the SQL Buffer to change to your last saved version.

SQL*Plus commands such as SET SERVEROUTPUT ON are not captured into the SQL Buffer, nor does SQL*Plus store the semicolon or the forward slash you type to execute a command in the SQL Buffer.

When you create stored procedures, functions, or packages, you begin with the CREATE command. When you begin a PL/SQL block, you start by entering the word DECLARE or BEGIN. Typing either BEGIN, DECLARE, or CREATE will put the SQL*Plus session into PL/SQL mode.

Running PL/SQL Blocks in SQL*Plus

Once you are in PL/SQL mode, you will not be able to end the block in the same manner that you ended a SQL block. The semicolon (;) can be used multiple times in a single PL/SQL block; thus when you end a line with a semicolon, you will not terminate the block. You can terminate the PL/SQL block in the SQL Buffer by entering a period (.). This will end the block and leave it in the SQL Buffer, but it will not execute it. At this point you have a choice of typing the EDIT command to edit the block or executing it with a forward slash (/) or a SQL*Plus command RUN.

For Example

You can enter and execute a PL/SQL subprogram as follows:

```
SQL> BEGIN
  2 DBMS_OUTPUT.PUT_LINE ('This is a PL/SQL Block');
  3 END;
  4 .
SQL> /
This is a PL/SQL Block

PL/SQL procedure successfully completed.
```

If want to run a script file at a later date, you must remember to terminate it with a period (.) and/or a forward slash (/) before saving it on your computer. If you simply want to put the code into the SQL Buffer and then execute it, you can end the script with a forward slash (/).

You should terminate PL/SQL blocks stored in the script file with the period if you want to put the code in the SQL Buffer. You should end the script with forward slash (/) if you want the PL/SQL code in the file to execute. *Failing to end your PL/SQL block with a period (.) and/or a forward slash (/) will prevent your block from executing.*

About The Sample Schema

The STUDENT schema contains tables and other objects meant to keep information about a registration and enrollment system for a fictitious university. There are ten tables in the system that store data about students, courses, instructors, and so on. In addition to storing contact information (addresses and telephone numbers) for students and instructors and descriptive information about courses (costs and prerequisites), the schema also keeps track of the sections for particular courses, and the sections in which students have enrolled.

The SECTION is one of the most important tables in the schema because it stores data about the individual sections that have been created for each course. Each section record also stores

information about where and when the section meets and which instructor teaches the section. The section table is related to the COURSE and INSTRUCTOR tables.

The ENROLLMENT table is equally important because it keeps track of which students have enrolled in which sections. Each enrollment record also stores information about the student's grade and enrollment date. The enrollment table is related to the STUDENT and SECTION tables.

The schema also has a number of other tables that manage grading for each student in each section.

The detailed structure of the STUDENT schema is described in Appendix B, Student Database Schema.