



ORACLE®

Oracle 技术系列 丛书

Oracle8i

PL/SQL

高级程序设计

Oracle8i Advanced
PL/SQL Programming

Oracle Press™ 授权出版

(美) Scott Urman 著

刘 谦 苏建平 等译



机械工业出版社
China Machine Press

OSBORNE

Oracle 技术系列丛书

Oracle8i PL/SQL 高级程序设计

(美) Scott Urman 著

刘 谦 苏建平 等译



机械工业出版社
China Machine Press

本书原版是机械工业出版社已翻译出版的《Oracle8 PL/SQL程序设计》的修订版。本书充分利用Oracle公司认可的内部技术信息,为读者开发健壮的基于Web的PL/SQL应用提供了功能强大的工具。书中集中介绍了包括动态SQL、外部例程、对象、集合在内的所有Oracle8i数据库的最新功能。除此之外,还为读者提供了开发PL/SQL应用所需的程序跟踪和调试技术。随书提供的光盘还向读者免费提供了4种PL/SQL开发环境工具包的试用版本。本书适合有PL/SQL使用经验,并想深入了解Oracle8i提供的PL/SQL最新功能的程序设计人员阅读。

Scott Urman : Oracle8i Advanced PL/SQL Programming (ISBN 0-07-212146-7).

Original edition copyright©2000 by McGraw-Hill Companies. All rights reserved.

Chinese edition copyright©2001 by China Machine Press. All rights reserved.

本书中文简体字版由美国麦格劳-希尔公司授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

本书版权登记号:图字:01-2000-1319

图书在版编目(CIP)数据

Oracle8i PL/SQL 高级程序设计 / (美)尤曼(Urman, S.)著;刘谦等译. - 北京:机械工业出版社, 2001.2

(Oracle 技术系列丛书)

书名原文: Oracle8i Advanced PL/SQL Programming

ISBN 7-111-08599-X

I. O… II. ①尤… ②刘… III. 关系数据库-数据库管理系统, Oracle8i-程序设计
IV. TP311.138

中国版本图书馆CIP数据核字(2000)第80002号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:张玉强 刘立卿

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2001年2月第1版第1次印刷

787mm×1092mm 1/16·44.25印张

印数:0 001-5 000册

定价:98.00元(附光盘)

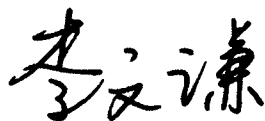
凡购本书,如有倒页、脱页、缺页,由本社发行部调换

贺 辞

祝贺中文版 ORACLE 8 系列丛书的出版。希望她能为中国的广大 ORACLE 用户和对数据库技术感兴趣的读者提供最先进的 ORACLE 技术知识。

ORACLE 软件系统有限公司希望通过中文版 ORACLE 8 系列丛书的出版，更好地建立起与广大用户和数据库工作者之间技术交流的桥梁。

Oracle 中国有限公司总经理

A handwritten signature in black ink, appearing to read '李文谦' (Li Wenqian), written in a cursive style.

1998.6

序

数据库技术的发展，使它已经成为现代信息技术的重要组成部分，成为现代计算机信息系统和计算机应用系统的基础和核心。可以说，如果没有数据库技术的发展，没有优秀的数据库产品的推出和应用，社会信息化的进程将是难以实现的。因此，在衡量一个国家信息化的程度时，其数据库的建设规模、数据库信息量的大小和使用程度也就成为重要的标志之一。

我国引进数据库技术始于70年代末，从微型计算机上运行的数据库到当前的大型数据库系统的引入和应用，已经有20多年的历史。20多年来，虽然在微型计算机数据库知识的普及和应用上取得了很大的进展和成绩，在大型数据库系统的开发和应用上也取得了进步，甚至还有了国产化的数据库软件，但如果从对数据库系统的应用效果和对数据库技术的掌握上来比较，则与发达国家之间仍然存在较大的差距，特别是在大型数据库系统的开发、建设和应用水平方面差距更大。这种差距主要表现在两个方面：第一是数据库的数量及所收集信息的数量和质量与发达国家相比存在很大的差距；第二是对数据库技术和产品的掌握和应用上更有待于提高和加强，应用人才急需培养，经验有待积累和总结。前者的改善应依靠于对数据库应用基础工作的加强，如重视基础数据的收集和整理，即重视数据工程的建设，并制定相应的数据政策；而后者的改善则会更多地依赖于人们对数据库技术和对数据库产品的掌握。为此，我们必须首先占有充足的资料并加以消化。如果数据库厂家和出版机构能提供较完整的、质量较高的技术资料 and 书籍，并为较多的数据库技术人员和应用人员所掌握，无疑将会促进国内数据库技术人才的成长并推动数据库应用水平的提高。

喜闻机械工业出版社华章公司与ORACLE出版公司合作，为配合ORACLE 8在中国的发行，由机械工业出版社买断了ORACLE出版公司出版的ORACLE 8系列丛书的中文简体字版的出版权，并组织国内从事ORACLE应用开发的科技人员和教学人员进行翻译出版，还邀请国内数据库专家对译稿进行了审定，以保证丛书在技术

上的权威性。无疑，这对大型数据库系统特别是ORACLE数据库系统的开发和应用将起到很好的推动作用。

众所周知，ORACLE公司推出的ORACLE 8是一种面向网络计算的数据库(the database for Network Computing)并支持对象关系模型的数据库产品。该系列丛书全面地介绍了ORACLE 8的功能和技术，具体书目见封底。

该丛书内容丰富，涵盖了大型数据库应用开发中的全部技术内容，有的资料，如《Oracle 8数据仓库分析、构建实用指南》、《新版Oracle 8故障解决手册》等都是在国内首次面世，很值得数据库技术工作者参考和阅读。

无疑，这套丛书应该有广泛的读者，它可供大型数据库系统，特别是ORACLE系统应用的系统管理员、应用程序员、系统分析员和设计人员以及广大用户学习和参考，也可做为ORACLE系统培训的教材和高等学校本科和研究生的学习参考书，同时也是掌握大型数据库系统理论和实践的好材料。

应感谢机械工业出版社的领导和组织翻译丛书的先生们和女士们，大家的辛勤的劳动将为我国信息化事业的发展增加动力。ORACLE中国有限公司对丛书的出版给予极大的关注与支持，李文谦总经理并为丛书的出版题词。

预祝丛书的出版和发行获得成功，并得到读者的欢迎。

罗晓沛

中国科学院研究生院 教授
ORACLE大学(中国) 校长

译者序

本书是深入介绍PL/SQL语言的专著。可供数据库系统管理员和应用系统设计师参考。众所周知，PL/SQL是实现从各种平台访问Oracle数据库的高级程序设计语言。本书原版自出版以来，深受专业人员欢迎，仅在美国市场的销售量就多达46 000多本。它不仅为开发健壮的、基于Web的应用提供了深入的信息，而且随书附带的CD-ROM中还为读者提供了样本代码，以及第三方提供的Oracle开发工具。

书中全面介绍了PL/SQL语言的包、触发器、动态SQL、外部例程、对象、集合等基本概念，而且还集中讨论了Oracle8i提供的最新功能。这些新的特性包括如何使用内置包，优化策略，以及PL/SQL程序的调试技术等。本书从以下几个方面深入介绍了Oracle8i程序设计的最新特点：

- 1) Oracle8i和PL/SQL提供的高效开发和运行环境。
- 2) 使用包DBMS_DEBUG和DBMS_TRACE跟踪调试PL/SQL应用。
- 3) 使用包DBMS_JOB调度PL/SQL任务在指定时间自动运行。
- 4) 使用包UTE_FILE直接读写操作系统文件。
- 5) 使用Pro*C和嵌入SQL语言编制外部例程。
- 6) 使用动态SQL语言在运行时创建并执行SQL和PL/SQL语句。
- 7) 允许数据库管道与警告之间进行交互会话，等等。

本书由刘谦、苏建平、王军平、刘丽云、李新、李江月、张君哲、钱云共同翻译完成，由刘谦审校。由于本书内容较新，篇幅较长，再加上译者的时间和水平有限，在翻译过程中难免有疏漏和错误，敬请读者给予批评指正。

译者
2000年7月

前 言

Oracle是一种功能异常强大和灵活的关系型数据库系统，然而，随着Oracle数据库功能的不断发展，使得数据库的使用变得愈来愈复杂。为了设计基于Oracle数据库系统的应用，我们有必要了解Oracle数据库系统对其存储在系统中的数据的处理方法。Oracle提供的专门用于处理数据的PL/SQL是一种重要的工具，它不仅可以处理Oracle内部的数据，而且还可以用于用户的应用程序。PL/SQL可以用在不同的环境并具有不同的特点。

我写的第一本有关PL/SQL的专著是1996年出版的“Oracle PL/SQL程序设计”一书,该书的内容是针对Oracle7.3版本的PL/SQL 2.3编制的。1997年，我又对该书的内容进行了扩充，增加了有关Oracle8.0的最新信息，出版了该书的第2版，即《Oracle8 PL/SQL程序设计》（该书已由机械工业出版社翻译出版。——编者注）。

本书是“Oracle PL/SQL程序设计”一书的第3版。书中涉及了Oracle数据库最新版本Oracle8i（与Oracle8.1版相对应）的有关内容。由于篇幅的限制，本书不能覆盖前两版的全部内容，而是集中介绍了PL/SQL的最新功能。如果读者是第一次接触PL/SQL，最好先从本书的前两版开始学习，我认为这是了解PL/SQL的最好的途径。本书的部分内容来自于《Oracle8 PL/SQL程序设计》，但主要内容是介绍Oracle8i的最新技术。如果读者想深入了解Oracle8i的话，本书可为您提供最详尽的信息。

本书的特点

本书的最大特点是集中介绍Oracle8i的最新技术进展。书中自始至终都贯穿了对Oracle8i提供的最新数据库功能的讨论，例如，第8章就集中介绍了动态SQL技术，包括DBMS_SQL包和Oracle8i提供的最新的本地动态SQL的内容。除此之外，本书还深入讨论了有关PL/SQL开发及调试环境的各种问题。

本书附带的CD-ROM

本书的CD-ROM包括下列三类信息：

- 带有本书演示的所有案例的在线代码。在有些情况下，书中使用的程序案例只是CD-ROM中完整代码的某一部分。如果想要了解这些程序的完整代码，读者可以参考CD-ROM中提供的代码。
- 带有来自四个开发商提供的五种不同开发工具的试用版。书中的开发工具分别来自于Quest Software、Compuware Corporation、Embarcadero Technologies以及Sylvain Faust International公司。本书的第2章和第3章将专门讲解这些开发工具的使用细节。
- 包括了第15章和第16章以及附录A～C的电子版文档。除此之外，本书第12章中的某些图形也存储在CD-ROM中供读者参考。

当读者在微软的视窗操作系统下放入本书的CD-ROM时，CD-ROM携带的启动程序将自动提示读者如何安装CD-ROM中的各种开发工具。对于在使用这些开发工具时遇到的任何问题，读者可以直接与开发商联系解决，请在联系时告之开发商所使用的是源自本书的试用版本。有关进一步的信息，请浏览CD-ROM根目录下的超本文档[readme.html](#)。

本书适用的读者

本书既可以作为Oracle8i PL/SQL的用户手册，也可以作为PL/SQL语言参考资料。如果读者已经有使用PL/SQL的经验并有兴趣深入学习Oracle8i提供的有关PL/SQL的新的高级功能，本书是非常适用的教材。如果读者没有接触过PL/SQL语言，我建议读者在阅读了第2版《Oracle8 PL/SQL程序设计》后再来看本书。尽管这两本书的内容有些相互重叠，但我认为这是全面了解PL/SQL的最好方式。

如何使用本书

本书共分为16章和4个附录。

第一部分：PL/SQL介绍及开发环境

本书的第一部分首先介绍了PL/SQL语言及其运行环境。重点介绍CD-ROM提供的PL/SQL开发环境。

第1章：PL/SQL介绍

本章介绍PL/SQL的基本功能，并重点描述该语言的某些主要特点。除此之外，本章还就PL/SQL的版本与数据库版本的对应关系进行了说明，并对本书中作为数据库应用案例使用的数据库模式进行说明。

第2章：PL/SQL开发和运行环境

PL/SQL可以以客户或服务器方式在多种不同的环境下运行。我们将在本章讨论PL/SQL引擎的不同配置，并讲解这些不同引擎间的通信机制。除此之外，我们还要分析CD-ROM中的有关开发工具和屏幕快照。

第3章：跟踪和调试

本章讨论七种包括CD-ROM提供的图形调试器在内的调试应用程序的不同技术。本章将使用所讨论的每一种调试技术来解决PL/SQL程序中的常见问题。另外，本章还将介绍包括Oracle8i提供的最新工具Profiler在内的跟踪应用程序的几种不同的方法。

第二部分：非对象功能

这一部分主要介绍PL/SQL的关系特性，其中包括过程、触发器、数据库任务、文件输入输出、动态SQL、会话间通信和外部例程等。在这一部分的最后一章还将总结Oracle8i为PL/SQL提供的最新功能。

第4章：创建子程序和包

子程序（包括过程和函数）提供了将PL/SQL代码组织成为可调用的命名块的强大功能，这

些可调用命名块可以存储在数据库中供程序调用。包是一组由相关子程序和声明语句组成的程序存储单位。本章的主要内容是详细讨论创建子程序和包的语法结构。

第5章：使用子程序和包

第5章的内容是第4章的延续，本章主要讨论子程序和包的其他特性，包括从SQL语句调用子程序和包的方法以及子程序和包之间的依赖性。除此之外，我们还将讨论包与共享缓冲区之间的交互方式和Oracle 8i提供的如调用权过程等新增的功能。

第6章：数据库触发器

触发器是一种特殊类型的PL/SQL块，这种PL/SQL块可以在触发事件发生时自动运行。触发事件可以是一个如插入语句类的DML操作，也可以是一个DDL语句，或者是一个系统事件。触发器也可以独立于给定的DML语句单独运行。本章将讨论所有类型的触发器。

第7章：数据库作业和文件输入输出

包DBMS_JOB允许程序员以存储过程的形式调度PL/SQL作业在指定的时间自动运行。包UTL_FILE允许PL/SQL读写操作系统文件。本章将结合程序案例详细讨论这两种包。

第8章：动态SQL

动态SQL是一种可以用来编制功能非常灵活的应用的强有力的程序设计技术。我们在本章将讨论有关包DBMS_SQL和动态SQL语言，这两种工具都可以在运行时直接创建，并发布SQL语句和PL/SQL块。采用这种方法设计的程序可以克服以往在PL/SQL块中只能使用DML语句的限制。

第9章：会话间通信

Oracle8i中提供了两个内置的用于数据库会话间直接进行通信的程序包，它们是DBMS_PIPE和DBMS_ALERT。我们将在本章看到一些使用这两个包的程序实例，并对这两个包提供的功能进行比较。

第10章：外部例程

使用外部例程可以让程序员从PL/SQL中直接调用使用C语言语句（要使用Oracle8以上的版本）或Java语言语句（要求使用Oracle8i以上的版本）编制的函数。调用时，C子程序运行在一个独立的进程中，而Java子程序则被装入数据库中并由Oracle8i内置的Java虚拟机JServer控制在一个PL/SQL进程中运行。本章的主要任务就是详细讨论这两种外部例程。

第11章：Oracle8i的其他功能

本章将全面总结Oracle8i PL/SQL提供的所有最新功能。我们将仔细地分析那些尚未介绍的全新的功能特点以及由此而使系统性能即时得以提升的效果。

第三部分：对象功能和LOB

该部分集中介绍Oracle8和Oracle8i的对象功能，同时还将讨论大型对象（LOB），这些大型对象使用户能在单一数据库列中存储多达4GB的数据。

第12章：引入对象

本章将讨论在Oracle8i中声明对象类型以及如何初始化对象实例和调用方法的语法及语义。除此之外，我们还要介绍一下Oracle8i提供的静态方法。

第13章：数据库中的对象

本章也是介绍数据库对象，其主要内容涉及如何在数据库中存储对象以及如何使用SQL语言和PL/SQL语言来操作对象。除此之外，本章还将讨论对象引用及其使用方法。

第14章：集合

Oracle8i中的集合就是指一组相关的PL/SQL对象，包括索引表、嵌套表和变长数组。这些数据类型类似于其他程序设计语言中的数组结构。本章主要讨论在数据库中存储嵌套表、变长数组以及集合的方法。

第四部分：大型对象和包DBMS_LOB

第15章：大型对象

Oracle支持的大型对象LOB可以用来存储代表不同字符集的字符和二进制数据。本章主要讨论不同类型的LOB以及访问LOB的接口，特别是将详细分析SQL用于LOB的接口。

第16章：高级LOB内容和DBMS_LOB

本章将继续讨论大型对象LOB。主要将详细讨论包DBMS_LOB，该包可提供从PL/SQL块中对LOB实施操作的主要方法。由于该包是一个用PL/SQL实现的包，所以它可以在任何可以发布PL/SQL块的环境下使用。

第五部分：附录

附录中提供了使用PL/SQL有关参考信息。

附录A：Oracle内置程序包指南

附录A将讨论所有Oracle内置的程序包。通过讨论，我们可以看到这些内置的程序包极大地扩展了PL/SQL语言的能力。除此之外，本附录还将对本书中没有介绍的其他程序包进行总结。

附录B：PL/SQL保留字

本附录列出了PL/SQL使用的所有保留字。用户程序中使用的变量名和PL/SQL中使用的对象都要避免使用这些保留字。

附录C：数据字典

本附录对多个数据字典视图进行了总结，并将详细讨论与PL/SQL有关的视图，对于其他的视图将只进行简单介绍。

附录D：本书CD-ROM介绍

附录D介绍了本书CD-ROM中的内容。

本书英文原书书号：ISBN 0-07-212146-7。

目 录

贺 辞
序
译者序
前言

第一部分 PL/SQL介绍及开发环境

第1章 PL/SQL介绍	1
1.1 为什么要引入PL/SQL语言	1
1.1.1 PL/SQL与网络传输	3
1.1.2 PL/SQL标准	3
1.2 PL/SQL的特点	4
1.2.1 PL/SQL的基本特点	4
1.2.2 PL/SQL的高级功能	7
1.2.3 PL/SQL内置包	11
1.3 本书的约定	11
1.3.1 PL/SQL和Oracle 数据库版本说明	11
1.3.2 Oracle数据库文档	13
1.3.3 本书提供的CD-ROM内容简介	13
1.4 本书案例使用的通用数据库表	14
1.5 小结	21
第2章 PL/SQL开发和运行环境	22
2.1 应用模式和PL/SQL	22
2.1.1 两层模式	22
2.1.2 三层模式	26
2.2 PL/SQL开发工具介绍	26
2.2.1 SQL*Plus	27
2.2.2 Rapid SQL	33
2.2.3 XPEDITER/SQL	35
2.2.4 SQL Navigator	38
2.2.5 TOAD	40
2.2.6 SQL-Programmer	42
2.2.7 PL/SQL开发工具小结	44

2.3 小结	45
第3章 跟踪和调试	46
3.1 问题分析	46
3.1.1 调试指导原则	46
3.1.2 调试程序包	47
3.2 非图形调试技术	47
3.2.1 在程序中插入调试用表	47
3.2.2 将结果打印到屏幕	56
3.3 PL/SQL调试器	65
3.3.1 PL/SQL调试器功能概述	65
3.3.2 问题3	66
3.3.3 问题4	71
3.3.4 问题5	77
3.3.5 问题6	81
3.3.6 问题7	85
3.4 跟踪和配置	89
3.4.1 基于事件的跟踪	92
3.4.2 基于PL/SQL的跟踪	105
3.4.3 基于PL/SQL的配置	110
3.5 小结	113

第二部分 非对象功能

第4章 创建子程序和包	115
4.1 过程和函数	115
4.1.1 创建子程序	116
4.1.2 过程和函数的撤消	120
4.1.3 子程序参数	120
4.1.4 过程与函数的比较	136
4.2 包	137
4.2.1 包的说明	137
4.2.2 包体	138
4.2.3 包和作用域	140

4.2.4 重载打包子程序	143	7.1 数据库作业	229
4.2.5 包的初始化	145	7.1.1 后台进程	229
4.3 小结	147	7.1.2 运行作业	230
第5章 使用子程序和包	148	7.1.3 其他的DBMS_JOB子程序	235
5.1 子程序位置	148	7.1.4 在数据库视图中观察作业	238
5.1.1 存储子程序和数据字典	148	7.1.5 作业运行环境	238
5.1.2 本地子程序	150	7.2 文件输入输出	239
5.1.3 存储子程序和本地子程序的比较	155	7.2.1 安全	239
5.2 存储子程序和包的几个问题	155	7.2.2 UTL_FILE引发的异常	240
5.2.1 子程序的相关性	155	7.2.3 打开和关闭文件	240
5.2.2 包运行时状态	165	7.2.4 文件输出	242
5.2.3 特权和存储子程序	171	7.2.5 文件输入	245
5.3 在SQL语句中使用存储函数	180	7.2.6 文件操作案例	245
5.3.1 纯层	180	7.3 小结	255
5.3.2 默认参数	186	第8章 动态SQL	256
5.3.3 从Oracle8i的SQL语句中调用函数	186	8.1 在PL/SQL中使用SQL语句	256
5.4 包的辅助功能	189	8.1.1 静态SQL与动态SQL的比较	257
5.4.1 共享缓冲区锁定	189	8.1.2 DBMS_SQL包概述	258
5.4.2 包体长度的限制	190	8.1.3 本地动态SQL概述	261
5.4.3 优化参数	191	8.2 使用DBMS_SQL	263
5.5 小结	194	8.2.1 执行DML、DDL以及ALTER SESSION语句	263
第6章 数据库触发器	195	8.2.2 执行查询语句	269
6.1 触发器的类型	195	8.2.3 执行PL/SQL语句	277
6.1.1 DML触发器	195	8.2.4 Oracle8和Oracle8i数据库中 DBMS_SQL功能的扩充	281
6.1.2 替代触发器	196	8.2.5 其他过程	296
6.1.3 系统触发器	197	8.3 使用本地动态SQL	299
6.2 创建触发器	198	8.3.1 执行非查询语句和PL/SQL块	299
6.2.1 创建DML触发器	198	8.3.2 执行查询语句	306
6.2.2 创建替代触发器	207	8.4 常见问题	313
6.2.3 创建系统触发器	212	8.4.1 动态SQL的特权和角色	313
6.2.4 其他触发器问题	217	8.4.2 调用者的权限	315
6.2.5 触发器与数据字典	222	8.4.3 DDL和动态SQL	317
6.3 变异表	223	8.4.4 ALTER SESSION和动态SQL	318
6.3.1 变异表案例介绍	225	8.4.5 DBMS_SQL和本地动态SQL的比较	319
6.3.2 变异表错误的处理	226	8.5 小结	324
6.4 小结	228		
第7章 数据库作业和文件输入输出	229		

第9章 会话间通信	325
9.1 DBMS_PIPE	325
9.1.1 发送消息	330
9.1.2 接收消息	332
9.1.3 创建和管理管道	333
9.1.4 权限和安全	335
9.1.5 建立通信协议	336
9.1.6 案例介绍	337
9.2 DBMS_ALERT	344
9.2.1 发送警报	345
9.2.2 接收警报	345
9.2.3 其他过程	346
9.2.4 警报和数据字典	347
9.3 DBMS_PIPE和DBMS_ALERT的比较	348
9.4 小结	349
第10章 外部例程	350
10.1 引入外部例程的必要性	350
10.1.1 外部例程举例	350
10.1.2 外部例程的结构	355
10.2 C外部例程	358
10.2.1 调用步骤	358
10.2.2 参数映射	364
10.2.3 C外部函数	379
10.3 Java外部例程	381
10.3.1 创建步骤	381
10.3.2 参数映射	390
10.3.3 Java外部函数	396
10.3.4 Java存储过程和Oracle JServer	397
10.4 数据库回调	403
10.4.1 C语言服务程序	403
10.4.2 在外部例程中执行SQL语句	408
10.4.3 数据库回调的限制	412
10.5 常见问题	413
10.5.1 外部例程的其他位置	414
10.5.2 调用者和定义者的权限	416
10.5.3 外部例程的相关性	423
10.5.4 调试外部例程	426

10.5.5 指导原则	430
10.5.6 限制	430
10.6 小结	431
第11章 Oracle8i 的新增功能	432
11.1 新增功能概述	432
11.1.1 性能	432
11.1.2 应用程序开发	434
11.1.3 Web和外部例程的集成	436
11.2 新的特点	437
11.2.1 批量绑定	437
11.2.2 自动事务	446
11.2.3 错误信息 “Numeric or Value Error” 的含义	454
11.2.4 CALL语句	455
11.3 性能比较	457
11.3.1 批量绑定性能比较	459
11.3.2 NOCOPY限定符	461
11.3.3 本地动态SQL	462
11.4 小结	464

第三部分 对象功能和LOB

第12章 引入对象	465
12.1 引入对象的背景	465
12.1.1 面向对象程序设计的基础	465
12.1.2 关系对象数据库	468
12.2 对象类型	469
12.2.1 定义对象类型	469
12.2.2 声明和初始化对象	472
12.2.3 对象的方法	474
12.2.4 变更及撤消类型	496
12.2.5 对象的相关性	498
12.3 小结	499
第13章 数据库中的对象	500
13.1 Oracle8 数据库中的对象	500
13.1.1 对象定位	500
13.1.2 对象标识和引用	504
13.1.3 对象类型和表之间的相关性	505

13.2 对象和SQL	506	15.1.3 临时LOB	569
13.2.1 访问列对象	506	15.1.4 LOB接口	569
13.2.2 访问行对象	511	15.2 SQL 访问	580
13.2.3 Refs和Values	516	15.2.1 内部LOB和SQL	580
13.2.4 有关对象的其他问题	525	15.2.2 外部LOB和SQL	586
13.3 小结	526	15.3 其他LOB问题	588
第14章 集合	527	15.3.1 LOB和触发器	588
14.1 声明和使用集合类型	527	15.3.2 发布LOB	589
14.1.1 索引表	527	15.3.3 说明LOB存储	589
14.1.2 嵌套表	531	15.4 小结	590
14.1.3 变长数组	534	第16章 高级LOB内容和DBMS_LOB	591
14.1.4 不同集合类型间的比较	537	16.1 包DBMS_LOB	591
14.2 集合方法	538	16.1.1 包LOBUtils	591
14.2.1 方法EXISTS	539	16.1.2 包DBMS_LOB的子程序	593
14.2.2 方法COUNT	540	16.1.3 读写LOB数据	597
14.2.3 方法LIMIT	541	16.1.4 BFILE	614
14.2.4 方法FIRST和LAST	542	16.1.5 打开和关闭LOB	619
14.2.5 方法NEXT和PRIOR	542	16.1.6 临时LOB	621
14.2.6 方法EXTEND	543	16.1.7 DBMS_LOB预定义的异常	624
14.2.7 方法TRIM	545	16.2 其他LOB内容	624
14.2.8 方法DELETE	547	16.2.1 定位指针的读一致性	624
14.3 数据库中的集合	549	16.2.2 LOB操作和事务	626
14.3.1 存储集合的隐含	549	16.3 LONG和LOB类型间的转换	630
14.3.2 操作整个集合	553	16.3.1 DBMS_SQL和DBMS_LOB	630
14.3.3 处理单个集合元素	558	16.3.2 TO_LOB	634
14.4 小结	564	16.4 小结	635

第四部分 大型对象和包DBMS_LOB

第15章 大型对象	565
15.1 什么是LOB	565
15.1.1 LOB的存储	566
15.1.2 LOB的类型	567

第五部分 附录

附录A Oracle内置程序包指南	637
附录B PL/SQL保留字	670
附录C 数据字典	673
附录D 本书CD-ROM介绍	692

第一部分 PL/SQL介绍及开发环境

第1章 PL/SQL介绍

PL/SQL是一种高级数据库程序设计语言，该语言专门用于在各种环境下对Oracle数据库进行访问。由于该语言集成于数据库服务器中，所以PL/SQL代码可以对数据进行快速高效的处理。除此之外，可以在Oracle数据库的某些客户端工具中，使用PL/SQL语言也是该语言的一个特点。本章的主要内容是讨论引入PL/SQL语言的必要性和该语言的主要特点，以及了解PL/SQL语言的重要性的数据库版本问题。还要介绍一些贯穿全书的更详细的高级概念，并在本章的最后就我们在本书案例中使用的数据库表的若干约定做一说明。

1.1 为什么要引入PL/SQL语言

Oracle数据库是一种关系型数据库。通常我们把用于访问这种关系型数据库的程序设计语言叫做结构化查询语言，即SQL语言。SQL是一种灵活高效的查询语言，其主要功能是对关系数据库中的数据进行操作和处理。例如，下面的SQL语句可以从数据库中将学习营养专业的全部学生一次删除：

```
DELETE FROM students
WHERE major = 'Nutrition';
```

（本章的最后一节将对本书中使用的包括students表在内的各种数据库表进行说明。）

SQL是先进的第四代程序设计语言，使用这种语言只需对要完成的任务进行描述，而不必指定实现任务的具体方法。以上面例子中的DELETE语句为例，我们并不知道SQL语言是如何找到学习营养专业的学生的。虽然按一般语言的做法推测，数据库服务器要按某种顺序逐个访问数据库表中的所有学生记录以决定删除满足条件的学生记录。但实际上，我们无法知道这些删除操作的细节。

第三代程序设计语言如C语言和COBOL语言等是面向过程的语言。用第三代语言（3GL）编制的程序是一步一步地实现程序功能的。例如，我们可以用下面的程序段来实现上述的删除操作：

```
LOOP over each student record
  IF this record has major = 'Nutrition' THEN
    DELETE this record;
  END IF;
END LOOP;
```

面向对象的程序设计语言如C++或Java也属于第三代程序设计语言。虽然这类语言采用了面向对象的程序结构，但程序中算法的实现还是要用各种语句逐步指定。

各种语言都有其自身的优缺点。相对于第三代程序设计语言来说, SQL一类的第四代程序语言使用起来非常简单, 语言中语句的种类也比较少。但这类语言将用户与实际的数据结构和算法隔离开来, 对数据的具体处理完全由该类语言的运行时系统实现。而在某些情况下, 第三代语言使用的过程结构在表达某些程序过程来说是非常有用的。这也就是引入PL/SQL语言的原因, 即PL/SQL语言将第四代语言的强大功能和灵活性与第三代语言的过程结构的优势融为一体。

PL/SQL代表面向过程化的语言与SQL语言的结合。我们可以从该语言的名称中看出, PL/SQL是在SQL语言中扩充了面向过程语言中使用的程序结构, 如:

- 变量和类型 (即可以预定义也可以由用户定义)
- 控制语句 (如IF-THEN-ELSE) 和循环
- 过程和函数
- 对象类型和方法 (PL/SQL8.0版本以上)

PL/SQL语言实现了将过程结构与Oracle SQL的无缝集成, 从而为用户提供了一种功能强大的结构化程序设计语言。例如, 我们假设要修改一个学生的专业。如果没有该学生的记录的话, 我们就为该学生创建一个新的记录。用PL/SQL编制的程序代码可以实现我们的要求, 如下所示:

节选自在线代码3gl_4gL.sql

```
DECLARE
  /* Declare variables that will be used in SQL statements */
  v_NewMajor VARCHAR2(10) := 'History';
  v_FirstName VARCHAR2(10) := 'Scott';
  v_LastName VARCHAR2(10) := 'Urman';
BEGIN
  /* Update the students table. */
  UPDATE students
    SET major = v_NewMajor
    WHERE first_name = v_FirstName
    AND last_name = v_LastName;
  /* Check to see if the record was found. If not, then we need
    to insert this record. */
  IF SQL%NOTFOUND THEN
    INSERT INTO students (ID, first_name, last_name, major)
      VALUES(student_sequence.NEXTVAL, v_FirstName, v_LastName,
        v_NewMajor);
  END IF;
END;
```

上面的例子中有两个不同的SQL语句 (UPDATE和INSERT), 这两个语句是第四代程序结构, 同时该段程序中还使用了第三代语言的结构 (变量声明和IF条件语句)。

注意 为了运行上面的程序例子, 先要创建程序中引用的数据库对象 (即表students和序列student_sequence)。可以使用本书CD-ROM中提供的脚本文件relTable.sql来实现上述工作。有关创建上述对象的进一步信息, 请参见1.3.3节的内容。

PL/SQL语言在将SQL语言的灵活性及功能与第三代语言的可配置能力相结合方面是独一无