

《数据库原理及应用 A》 实验指导书

西南科技大学 计算机科学与技术学院

《数据库原理及应用》 课程组

目 录

第 1 章 概述.....	3
1.1 课程简介.....	3
1.2 实验环境选择.....	3
1.2.1 Oracle 简介.....	3
1.2.2 Oracle 的体系结构.....	4
1.4 图书管理系统概述.....	10
第 2 章 实验要求.....	13
2.1 实验过程要求.....	13
2.2 考核及评分标准.....	13
第 3 章 实验内容.....	14
3.1 实验 1 数据定义与数据操纵（建立数据库）.....	14
3.1.1 实验类型.....	14
3.1.2 实验目的.....	14
3.1.2 基础知识.....	14
3.1.3 实验内容.....	14
3.1.4 实验步骤.....	15
3.2 实验 2 数据查询.....	20
3.2.1 实验类型.....	20
3.2.2 实验目的.....	20
3.2.3 基础知识.....	20
3.2.4 实验内容.....	20
3.2.5 实验步骤.....	20
3.2.6 实验扩展.....	21
3.3 实验 3 存储过程与触发器.....	23
3.3.1 实验类型.....	23
3.3.2 实验目的.....	23
3.3.3 基础知识.....	23
3.3.4 实验内容.....	25
3.3.5 实验步骤.....	25

3.4 实验 4 数据库恢复与安全性	27
3.4.1 实验类型	27
3.4.2 实验目的	27
3.4.3 基础知识	27
3.4.4 实验内容	31
3.4.5 实验步骤	31
附录 A 实验报告格式	33

第 1 章 概述

1.1 课程简介

随着计算机技术与网络通信技术的发展,数据库技术已成为信息社会中对大量数据进行组织与管理的重要技术手段及软件技术,是网络信息化管理系统的基础。它不仅有完整的理论基础,而且随着硬件技术与软件技术的飞快发展,它的应用也越来越广泛,为此各高校纷纷开设了《数据库原理及应用 A》这门专业课,以适应社会信息化的需求。通过有针对性的实验,能使学生更深层次地理解数据库系统的体系结构,掌握数据库系统的基本理论、技术和方法,培养学生运用所学理论知识,独立解决现实问题的能力。

实验体系见表 1-1 《数据库原理及应用 A》实验体系。

表 1-1 《数据库原理及应用 A》实验体系

序号	学时	实验名称	实验类型	要求完成情况
1	4	实验一 数据定义与数据操纵	设计型	代码
2	4	实验二 数据查询	设计型	代码
3	4	实验三 存储过程与触发器	设计型	代码
4	4	实验四 数据库恢复与安全性	设计型	代码

1.2 实验环境选择

实验环境: Oracle 企业版

1.2.1 Oracle 简介

Oracle 是一种关系型数据库系统。Oracle 是“神谕”的意思,取自 Oracle Bone Inscriptions 的第一个单词。Oracle 公司由 Larry Ellison 于 1977 年创办,目前已发展为全球最大的企业软件公司。数据库是 Oracle 的固有强项,至今在世界保持第一的市场占有率。

Oracle 特点

- 高性能: 伸缩性
- 安全性: 角色、权限、用户
- 并发控制: 表锁定、行锁定
- 海量数据: 表空间、空间数据构件、TB 级
- 数据完整性: 日志、回滚段
- 分布式数据库: 数据库连接

1.2.2 Oracle 的体系结构

为了能使 Oracle 数据库管理系统满足商业用户的要求，Oracle 引入了如图 1-1 所示的复杂的体系结构。

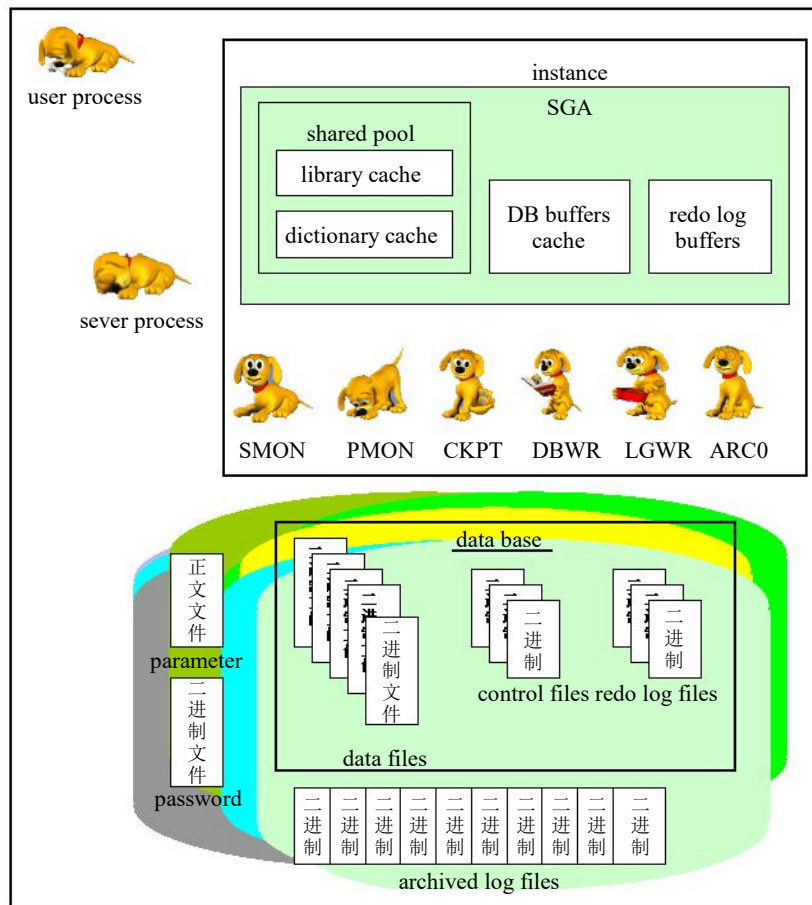


图 1-1 Oracle 体系结构示意图

体系结构主要包括 Oracle 服务器（server）（而服务器又是由 Oracle 实例（instance）和 Oracle 数据库（database）组成），还包括一些其他的关键文件、用户进程和服务器进程等。

Oracle 服务器（server）由 Oracle 实例（instance）和 Oracle 数据库（database）两大部分组成。它是一个数据库管理系统，提供了一致、开放和多样的信息管理的方法和途径。服务器中的一些结构并不在处理 SQL 语句时使用，它们是为了改进数据库系统的效率或数据的恢复等而设计的。

一、Oracle 服务器（server）

Oracle 服务器（server）可以有以下 3 种安装方式。

1. 基于主机方式：在此种配置下，用户直接在安装了数据库的计算机上登录 Oracle 数据库。
2. 客户端-服务器（client-server）（两层模型）方式：数据库和客户终端分别安装在不同的计算机上，用户通过网络从个人计算机（客户端）上访问数据库。
3. 客户端-应用服务器-服务器（client- application server-server）（三层模型）：用户首先从自己的个人计算机登录应用服务器，再通过应用服务器访问真正的数据库。

二、Oracle 实例（instance）

Oracle 实例 (instance) 是一种访问数据库的机制, 它是由内存结构和一些后台进程组成的。它的内存结构也称为系统全局区 (system global area, SGA)。系统全局区是实例的最基本的部件之一。实例的后台进程中有 5 个是必需的, 即这 5 个后台进程中的任何一个没有启动, 实例将自动关闭。这 5 个后台进程分别是 SMON、PMON、DBWR、LGWR 和 CKPT。在 OCP 考题中有时可能会问哪些后台进程是可选的? 除了这 5 个都是可选的。实例一启动就分配系统全局区和启动所需的后台进程。这里应该指出的是, 每个实例只能操作一个数据库, 而且它不可以操作其他的数据库。但是反过来是不成立的, 因为一个数据库可以同时被几个实例操作 (在 Oracle 集群中)。

系统全局区 (SGA) 中包含了以下几个内存结构: 共享池 (shared pool)、数据库高速缓冲区 (database buffer cache)、重做日志缓冲区 (redo log buffer) 和其他的一些结构 (如锁和统计数据) 等。

三、Oracle 数据库

Oracle 数据库是数据的一个集合, Oracle 把这些数据作为一个完整的单位来处理。Oracle 数据库也叫做物理 (外存) 结构, 它为数据库信息提供了真正的物理存储, 它是由以下 3 类操作系统文件组成的。

1. 控制文件 (control files): 包含了维护和校验数据库一致性所需的信息。
2. 重做日志文件 (redo log files): 包含了当系统崩溃后进行恢复所需记录的变化信息。
3. 数据文件 (data files): 包含了数据库中真正的数据。

四、Oracle 其他的关键文件

除了以上 3 类数据库文件之外, Oracle 服务还需要其他的一些文件, 这些文件不属于数据库。其中包括:

- 初始化参数文件 (parameter files): 定义了实例的特性, 如系统全局区中一些内存结构的大小、DBWR 的个数。
- 密码文件 (password files): 包含了数据库管理员或操作员用户在启动和关闭实例时所需的密码。虽然 Oracle 数据库提供了相当完善的安全管理机制, 但是在 Oracle 数据库没有开启时如何验证要启动数据库的人是真正的数据库管理员或操作员呢? 这就是 Oracle 引入密码文件的原因。
- 归档重做日志文件 (archived redo log files): 是重做日志文件的脱机备份。在系统崩溃后进行恢复时可能需要这些文件。

五、服务器进程

当 Oracle 创建一个服务器进程的同时要为该服务器进程分配一个内存区, 这个内存区称为程序全局区 (program global area, PGA)。与 SGA 不同, PGA 是一个私有的内存区, 是不能共享的, 是只属于一个进程的。它随着进程的创建而被分配, 随着进程的终止而被回收。在专用服务器进程的配置情况下, 程序全局区包括了以下结构:

1. 排序区 (sort area): 用于处理 SQL 语句所需的排序。
2. 游标状态区 (cursor state): 用于指示会话当前所使用的 SQL 语句的处理状态。
3. 会话信息区 (session information): 包括了会话的用户权限和优化统计信息。
4. 堆栈区 (stack space): 包括了其他的会话变量。

如果是共享服务器进程或多线程的配置, 以上这些结构除了堆栈区外大部分将存在 SGA 中。如果有 large pool, 它们就会被存在 large pool, 否则它们就会被存在共享池中。

六、共享池 (shared pool)

SGA 中的共享池 (shared pool) 是由库高速缓存 (library cache) 和数据字典高速缓存 (data dictionary cache) 两部分所组成。服务器进程将 SQL (也可能是 PL/SQL) 语句的正文和编译后的代码 (parsed code) 以及执行计划都放在共享池 (shared pool) 的库高速缓存中。在

进行编译时,服务器进程首先会在共享池中搜索是否有相同的 SQL 或 PL/SQL 语句(正文),如果有就不进行任何后续的编译处理,而是直接使用已存在的编译后的代码和执行计划。

提示:

库高速缓存包含了共享 SQL 区和共享 PL/SQL 区两部分,它们分别存放 SQL 和 PL/SQL 语句以及相关的信息。

要想共享 SQL 或 PL/SQL 语句,第一,库高速缓存(library cache)要足够大,因为只有这样要共享的 SQL 或 PL/SQL 语句才不会很快地淘汰出内存。第二,SQL 或 PL/SQL 语句要是能共享的通用代码(generic code),因为 Oracle 是通过比较 SQL 或 PL/SQL 语句的正文来决定两个语句是否相同的,只有当两个语句的正文完全相同时 Oracle 才重用已存在的编译后的代码和执行计划。

当 Oracle 在执行 SQL 语句时,服务器进程将把数据文件、表、索引、列、用户和其他的数据对象的定义和权限的信息放入数据字典高速缓存。如果在这之后,有进程(用户)需要同样的信息,如表和列的定义,那么所有的这些信息将从数据字典高速缓存中获得。因为以上所说的这些信息都是存在 Oracle 数据库的数据字典中,这也可能就是该部分内存叫做数据字典高速缓存的原因。

表和列的定义等重用的机会要比 SQL 语句大,因此为了能达到共享这些信息的目的,数据字典高速缓存应该尽可能设置得大一些。不过与库高速缓存一样,Oracle 并没有给出直接设置数据字典高速缓存大小的方法,只能通过设置共享池的大小来间接地设置数据字典高速缓存的大小。

七、数据库高速缓冲区(database buffer cache)

如果用户发出了以下的 SQL 语句: `SELECT * FROM emp`, Oracle 又是怎样提取数据库中的数据呢?服务器进程将首先在数据库高速缓冲区(database buffer cache)中搜寻所需的数据,如果找到了就直接使用而不进行磁盘操作,如果没找到就将进行磁盘操作把数据文件中的数据读入到数据库高速缓冲区中。

从以上的讨论可知,为了能够共享数据库中的数据,数据库高速缓冲区要足够的大,因为这样那些可以共享的数据才不会被很快地淘汰出内存。Oracle 也是使用一个叫做 LRU (least recently used)的队列(list)或算法(algorithm)来实现对数据库高速缓冲区的管理。可以使用参数文件中的 `DB_BLOCK_SIZE` 和 `DB_BLOCK_BUFFERS` 两个参数来设置数据库高速缓冲区的大小。其中 `DB_BLOCK_SIZE` 为 Oracle 数据块(内存缓冲区)的大小,在 Oracle 数据库中内存和外存的数据块的大小是相同的。`DB_BLOCK_BUFFERS` 为内存缓冲区的个数。数据库高速缓冲区大小为这两个参数的乘积。但是 `DB_BLOCK_SIZE` 的值是在创建数据库时设定的,如果要改变该参数的值一般需要重建数据库。因此多数情况下只能通过改变 `DB_BLOCK_BUFFERS` 的值来调整数据库高速缓冲区大小。但一定得重新启动 Oracle 数据库。

八、重做日志缓冲区(redo log buffer)

从理论上讲,如果数据库不会崩溃,根本没有必要引入重做日志缓冲区(redo log buffer)。引入重做日志缓冲区的主要目的就是数据的恢复。Oracle 在使用任何 DML 或 DDL 操作改变数据之前都将恢复所需的信息,即在写数据库高速缓冲区之前,先写入重做日志缓冲区。

与执行查询语句有所不同,Oracle 在执行 DML 语句时只有编译(parse)和执行(execute)

两个阶段。以下是 Oracle 执行 UPDATE 语句的步骤:

1. 如果数据和回滚数据不在数据库高速缓冲区中, Oracle 服务器进程将把它们从数据文件中读到数据库高速缓冲区中。
2. Oracle 服务器进程在要修改的数据行上加锁(行一级的锁)。
3. Oracle 服务器进程将数据的变化信息和回滚所需的信息都记录在重做日志缓冲区中。
4. Oracle 服务器进程将回滚所需的原始值和对数据所做的修改都写入数据库高速缓冲区(database buffer cache)。之后在数据库高速缓冲区中所有的这些数据块都将被标为脏缓冲区,因为此时内外存的数据是不同的(不一致的)。

Oracle 处理 INSERT 或 DELETE 语句的步骤与处理 UPDATE 语句的步骤大体相同。

九、大池 (large pool) 和 Java 池 (Java pool)

除了以上所介绍的内存结构之外,SGA 中还有可能包含 large pool 和 Java pool 两个可选的内存结构。

引入 large pool 的主要目的应该是提高效率。large pool 是一个相对简单的内存结构,与 shared pool 不同的是它没有 LRU 队列。在多线程 (MTS) 或共享服务器 (shared server) 连接时,Oracle 服务器进程的 PGA 的大部分区域(也叫 UGA)将放入 large pool (stack space 除外)。另外在大规模 I/O 及备份和恢复操作时可能使用该区。可以通过设置参数 LARGE_POOL_SIZE 的值来配置 large pool 的大小。该参数也是一个动态参数。

引入 Java pool 的目的是能够编译 Java 语言的命令。如果要使用 Java 语言就必须设置 Java pool。Java 语言在 Oracle 数据库中的存储与 PL/SQL 语言几乎完全相同。可以通过设置参数 JAVA_POOL_SIZE 的值来配置 Java pool 的大小。其数字的单位是字节 (bytes)。

十、重做日志写进程 (LGWR) 及快速提交 (fast commit)

重做日志写进程 (LOG writer, LGWR) 负责将重做日志缓冲区的记录顺序地写到重做日志文件中。为了更好地理解 LOG writer 的操作原理,在这里先介绍一下 Oracle 提交 (commit) 语句是如何工作的。

Oracle 服务器是使用了一种称为快速提交 (fast commit) 的技术,该技术既能保证 Oracle 系统的效率又能保证在系统崩溃的情况下所有提交的数据可以得到恢复。为此 Oracle 系统引入了系统变化数 (system change number, SCN)。无论任何时候只要某个事务 (transaction) 被提交,Oracle 服务器都将产生一个 SCN (号码) 并将其赋予该事务的所有数据行。在同一个数据库中 SCN 是单调递增的并且是惟一的。为了避免在进行一致性检验时操作系统时钟可能引发的问题,Oracle 服务器将 SCN 作为 Oracle 的内部时间戳来保证数据文件中的数据同步和数据读的一致性。

当在 SQL*Plus 中发了 commit 语句之后,Oracle 的内部操作步骤如下:

1. 服务器进程将把提交的记录连同所产生的 SCN (号码) 一起写入重做日志缓冲区中。
2. 重做日志写进程 (LGWR) 将把重做日志缓冲区中一直到所提交的记录(包括该记录)的所有记录连续地写到重做日志文件中。在此之后,Oracle 服务器就可以保证即使在系统崩溃的情况下所有提交的数据也可以得到恢复。
3. Oracle 通知用户(进程)提交已经完成。
4. 服务器进程将修改数据库高速缓冲区中的相关数据的状态并释放资源和打开锁等。

此时可能这些数据并未被写到数据文件中,此时这些数据缓冲区被标为脏缓冲区,因为相同的数据在内外存中为不同的版本。数据库高速缓冲区中的数据是由 DBWR 写到数据文件中的。

曾有不少学生问过这样一个问题:“为什么不同时写两个数据文件呢?” Oracle 的这种解决方案的最大好处是在保证不丢失数据的同时数据库的效率不会受到很大影响。因为重做

日志文件中的记录是以最紧凑的格式存放的，所以它的 I/O 量要比对数据文件的操作少得多。另外 LGWR 是顺序地将重做日志缓冲区中的记录写到重做日志文件中的，这样其 I/O 速度要比将数据块写到数据文件中快得多。

重做日志写进程 (LGWR) 要在下列情况下将重做日志缓冲区的记录 (内存) 顺序地写到重做日志文件 (外存) 中：

- 当某个事务被提交时。
- 当重做日志缓冲区中变化的记录超过一兆字节 (1MB) 时。
- 当重做日志缓冲区中所存的记录已超过缓冲区容量的 1/3。
- 在 DBWR 将数据库高速缓冲区中修改过的数据块写到数据文件之前。
- 每 3 秒钟。

因为在进行数据库恢复时需要重做日志数据，所以重做日志写进程 (LGWR) 只有在重做日志数据写到重做日志文件 (磁盘) 上时才能确定提交已经完成。在 Oracle 8i 之前的版本中，重做日志数据的惟一目的和用处就是数据库恢复。Oracle 在 Oracle 8i 的版本中引入了一个叫做重做日志挖掘器 (logminer) 的工具。该工具可以将重做日志文件或归档重做日志文件中的数据转换成用户能理解的正文信息。在 Oracle 8i 中，该工具只有命令行操作方式。Oracle 9i 加强了此工具的功能并引入了一个称为日志挖掘浏览器 (logminer viewer) 的图形界面。

十一、数据库写进程 (DBWR/DBWn)

数据库的典型操作就是大规模的输入/输出 (I/O)。因此为了提高 Oracle 系统的效率，一要减少 I/O 量，这可能是 Oracle 引入 LGWR 的原因之一；二要减少 I/O 次数，这可能是 Oracle 引入数据库写进程 (DBWR/DBWn) 的主要原因。

提示：

在 Oracle 的英文书中有些将“数据库写进程”用 DBWR 表示，有些将它用 DBWn 表示。这是因为在一个 Oracle 实例中可以启动多个数据库写进程，特别是在要进行大规模输入/输出并且运行在多 CPU 计算机上的 Oracle 数据库系统。Oracle 允许在一个实例上最多启动 10 个数据库写进程，它们分别是 DBW0~DBW9。

数据库写进程负责将数据库高速缓冲区中的脏缓冲区中的数据写到数据文件上。为了提高效率，数据库写进程并不是数据库高速缓冲区中的数据一有变化就写数据文件，而是积累了足够多的数据一次写一大批内存数据块到数据文件上。

数据库写进程将在下列事件之一发生时把数据库高速缓冲区中的数据写到数据文件上：

- 当脏缓冲区的数量超过了所设定的限额。
- 当所设定的时间间隔已到。
- 当有进程需要数据库高速缓冲区而却找不到空闲的缓冲区时。
- 当校验点发生时。
- 当某个表被删除 (drop) 或被截断 (truncate) 时。
- 当某个表空间被设置为只读状态 (read only) 时。
- 当使用类似于 ALTER TABLESPACE users BEGIN BACKUP 的命令对某个表空间进行联机备份时。
- 当某个临时表空间被设置为只脱机状态 (offline) 或正常状态 (normal) 时等。

十二、系统监督进程 (SMON)

从前面的论述中可以知道，由于某种原因 Oracle 系统崩溃了 (如断电)，SGA 中任何没

有来得及写到磁盘中的信息都将丢失,如有些已经提交的数据还没有真正地被写到数据文件中时就会丢失。在这种情况下,当数据库重新开启时,系统监督进程(SMON)将自动地执行 Oracle 实例的恢复工作。其步骤如下:

1. 执行前滚(roll forward),即将已经写到重做日志文件中但还没写到数据文件中的提交数据写到数据文件中(Oracle 是用 SCN 号码来识别提交记录的)。
2. 在前滚完成后立即打开数据库,此时用户就可以登录并使用数据库了。这时在数据文件中可能还有一些没有提交的数据。之所以这样安排主要是为了提高系统的效率。
3. 回滚没有提交的事务(数据)。除了 SMON 进程之外,服务器(server)进程也可能进行回滚没有提交的事务,但该进程只回滚它所用到的加锁的数据行。

除此之外,SMON 进程还要执行如下的磁盘空间的维护工作:

4. 回收或组合数据文件中相连的空闲区。
5. 释放临时段(在执行 SQL 语句时用作排序的磁盘区),将它们还给数据文件以作为空闲区使用。

十三、进程监督进程(PMON)

当某个进程崩溃时(如在正常退出 Oracle 的情况下重新启动了所用的 PC),进程监督进程(PMON)将负责它的清理工作。PMON 进程将负责进行如下的清理工作:

- 回滚用户当前的事务。
- 释放用户所加的所有表一级和行一级的锁。
- 释放用户所有的其他资源等。

十四、校验点(checkpoint)和校验点进程

Oracle 系统为了提高系统的效率和数据库的一致性,引入了一个称为校验点的事件。该事件是在当 DBWR 进程把在 SGA 中所有已经改变了的数据库高速缓冲区中的数据(包括提交的和没提交的数据)写到数据文件上时产生。从理论上讲校验点(checkpoint)和校验点进程可以完全不需要,因为 Oracle 系统利用重做日志数据和 SCN 号是能够保证数据库的完全恢复的。引入校验点可能是为了提高系统的效率。因为所有到校验点为止的变化了的数据都已写到了数据文件中,在实例恢复时校验点之前的重做日志记录已经不再需要,这样实例恢复速度就加快了。

在校验点事件发生时,Oracle 要将校验点号码(Oracle 系统自动产生的)写入所有相关的数据文件的文件头中。还要将校验点号码、重做日志序列号、归档日志名字和 SCN 号都写入控制文件中。

尽管经常产生校验点可以加快实例恢复的速度,但是由于在产生校验点时 Oracle 系统要进行大量的 I/O 操作,所以过于频繁地产生校验点会使数据库正常的联机操作受到冲击。最后数据库管理员要在实例恢复的速度和联机操作之间进行折衷。一般的生产或商业数据库的校验点间隔是在半小时以上。

十五、归档日志(ARCH/ARCn)进程

以上 5 个后台进程都是必需的,即它们中的任何一个停止后实例自动关闭。在可选后台进程中,归档日志(ARCH/ARCn)进程可能是最重要的一个可选后台进程,因为如果 Oracle 数据库的数据文件丢失或损坏,一般数据库要进行完全恢复,Oracle 数据库应运行在归档方式。

在 Oracle 数据库中,重做日志文件被划分为若干个组。当一组重做日志的文件被写满后,Oracle 就开始写下一组重做日志,这被称为日志切换。切换是以循环的方式进行的,即当最后一组写满后,又开始写第一组。因此如果只有重做日志文件,即 Oracle 数据库运行在非归档方式下,当遇到数据文件丢失或损坏时,Oracle 系统很难保证完全恢复数据库中的

数据。因为此时所需的重做记录可能因重做日志循环使用而被覆盖掉了。

在归档方式下，ARCn 进程将把切换后的重做日志文件复制到归档日志文件。可以把归档日志文件看成是重做日志文件的备份，但归档日志文件是脱机的，即除了在进行（复制）时，Oracle 数据库在正常运行时是不会关注归档日志文件的。Oracle 系统确保在一组重做日志的归档操作完成之前不会重新使用该组重做日志。在 Oracle 数据库中归档操作一般是自动执行的。利用这些归档日志文件，Oracle 系统就能确保在遇到数据文件丢失或损坏后可以完全恢复数据库中的数据。

1.4 图书管理系统概述

某高校欲开发一个图书管理系统

藏书数量：纸本图书 300 万册

学生人数：2 万

功能：

1. 借书

- (1) 读者自行取书到服务台；
- (2) 工作人员拿到书籍以后办理借阅手续；
- (3) 修改书籍状态；
- (4) 重复 (2) ~ (3) 处理所有图书借阅。

2. 还书

- (1) 读者将书籍递交给服务台还书处工作人员；
- (2) 服务人员扫描书籍信息，办理还书手续；
- (3) 修改书籍状态；
- (4) 重复 (3) ~ (4) 处理所有图书归还。

3. 预约

- (1) 在搜索栏输入需要借阅的书籍；
- (2) 获取该书籍借阅情况，对于读者需预借的每一本书籍；
- (3) 点击预约按钮；
- (4) 填写个人信息以及预借时间完成预借手续；
- (5) 重复 2~4，直到所有的预借书籍处理完毕。

4. 交费

- (1) 读者将借阅卡给服务台交费处工作人员；
- (2) 工作人员扫描借阅卡；
- (3) 读者将钱给工作人员；
- (4) 工作人员输入金额点击提交确认完成交费任务。

5. 查询图书信息

- (1) 进入系统首页；
- (2) 在搜索栏输入需要借阅的书籍信息；
- (3) 点击搜索；
- (4) 系统返回满足搜索条件的书籍列表。

6. 查询借阅情况

- (1) 读者登录系统；
- (2) 点击借阅情况查询；

(3) 系统返回读者借阅情况信息并显示。

7. 个人信息管理

- (1) 读者登录系统；
- (2) 进入个人信息管理窗口；
- (3) 对个人信息进行管理（修改、删除等）；
- (4) 如果个人信息有变动点击保存；
- (5) 系统修改该读者个人信息并返回修改成功。

8. 用户注册/登录

注册

- (1) 读者进入注册页面；
- (2) 输入注册信息；
- (3) 点击提交；
- (4) 系统验证注册信息；
- (5) 如果无错误，提示注册成功；
- (6) 如果有错误，提示注册失败并指明哪些信息填写错误。

登录

- (1) 读者进入登录页面；
- (2) 填写账号和密码；
- (3) 点击登录按钮；
- (4) 系统对账号和密码审核；
- (5) 如果无误，跳转到系统主页；
- (6) 如果有误，提示用户名或密码错误。

9. 用户管理

- (1) 管理员登录系统；
- (2) 进入用户管理窗口；
- (3) 对用户信息进行管理（添加、删除、修改等）。

10. 密码找回

- (1) 读者进入密码找回窗口；
- (2) 输入用户名和注册邮箱；
- (3) 点击找回按钮；
- (4) 系统审核用户名和注册邮箱是否匹配；
- (5) 匹配则将该读者密码发送到该邮箱；
- (6) 不匹配则提示用户名或邮箱错误。

11. 新增图书

- (1) 进入新增图书查看；
- (2) 录入新增图书信息；
- (3) 点击确认将图书信息录入系统并通知相应的图书室领取书籍；
- (4) 重复（2）~（3）完成所有新书的入管工作。

12. 续借

- (1) 进入续借窗口；
- (2) 系统返回已借书籍的信息列表；
- (3) 选取要续借的书籍，点击续借按钮；
- (4) 系统修改该书的到期日期并返回；
- (5) 重复（2）~（4）完成所有想要续借图书的续借工作。

13. 书籍到期提醒

- (1) 系统查询所有借出书籍的到期日期；
- (2) 如果当前日期离书籍的到期日期相差小于 2 天，则搜索该书借阅者的邮箱信息；
- (3) 系统发送到期提醒信息到该邮箱；
- (4) 重复 (2) ~ (3) 完成所有将到期书籍提醒工作。

14. 借阅卡挂失/补办

挂失

- (1) 读者持有效证件到服务台申请挂失；
- (2) 服务台工作人员录入读者借阅卡卡号和有效证件号；
- (3) 点击挂失按钮；
- (4) 系统将借阅卡状态修改为挂失。

[挂失期间读者不能办理借书业务]

解除挂失

- (1) 读者申请解除挂失；
- (2) 工作人员录入卡号；
- (3) 解除挂失。

补办

- (1) 读者申请补办借阅卡；
- (2) 服务台工作人员录入读者借阅卡号；
- (3) 系统返回借阅卡状态；
- (4) 如果状态为挂失状态，补办借阅卡，并将借阅状态恢复到可用状态；
- (5) 如果状态不是挂失状态，读者先办理挂失。

[补办后原来的借阅卡无效]

15. 书籍注销

- (1) 图书馆管理员列出下架注销列表申请书籍注销；
- (2) 馆长审核列表给出审核后的注销列表；
- (3) 对通知各图书室管理员对注销列表中的书籍进行下架处理。

第 2 章 实验要求

2.1 实验过程要求

本课程中，实验者必须服从指导教师和实验室工作人员的安排，遵守纪律与实验制度，爱护设备及卫生。在指定的实验时间内，必须到机房内实验，其余时间可自行设计和分析。

实验前，预习实验，了解实验背景。按照实验指导书内容进行实验，实验时注意观察实验结果，最后分析实验结果，得出结论，按格式写出实验报告（详细格式参照附录 A）。

2.2 考核及评分标准

若实验者在指定上机时间三次不到课，取消考核资格。请假，必须提前出具正式假条，不接受事后假条。如果发现抄袭、篡改、伪造实验数据，或实验报告雷同，涉及的所有学生的该课程成绩计为 0 分。实验主要考核指标如下：

- 实验者是否真实、认真的完成了本次实验；
- 实验代码是否调试通过、运行结果是否正确，是否具备良好可读性；
- 实验报告格式是否规范，是否有抄袭行为等。
- 实验答辩分为自述和教师提问两部分，自述时间不得超过 5 分钟，内容包括：演示、描述本次实验主要步骤、关键代码分析等。考察学生是否思路清晰，逻辑严谨，表达流畅，答辩结束后，答辩老师根据答辩情况并仔细审查实验报告和相关文档之后，参照实验题目的难度以及完成实验的时间，给出本次实验的最终成绩。

第 3 章 实验内容

3.1 实验 1 数据定义与数据操纵（建立数据库）

3.1.1 实验类型

设计型实验

3.1.2 实验目的

- 熟悉 oracle 环境；
- 熟练掌握和使用 PL/SQL 建立数据库基本表；
- 使用 PL/SQL developer 操作数据库；
- 熟练掌握 SQL 建立关系，及增删改数据。

3.1.2 基础知识

数据库的设计：从狭义上讲，是指设计数据库的各级模式并建立数据库，是数据库应用系统涉及的一部分。**数据库设计的基本步骤：**按照规范设计的方法，考虑数据库及其应用系统开发全过程，数据库的设计可以分为 6 个阶段：需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施、数据库运行和维护。

该实验要求使用用例图对系统需求进行用例分析，使用 E-R 图建立数据库概念模型，用 SQL 完成数据库逻辑结构的建立。

创建基本表（CREATE TABLE）语句的语法格式：

```
CREATE TABLE < 表名 > (< 列名 > < 数据类型 > [列级完整性约束条件]
                        [, < 列名 > < 数据类型 > [列级完整性约束条件]]
                        ...
                        [, < 表级完整性约束条件 >])
```

3.1.3 实验内容

- 了解 SQL PLUS 的使用
- 使用 PL/SQL developer 的图形界面，建立图书管理数据库 orcl 中的各个关系
- 在建立的关系中输入有效数据
- 删除以上各关系
- 在 PL/SQL developer 用 SQL 代码建立 orcl 数据库中各关系
- 用 SQL 代码完成数据增、删、改

3.1.4 实验步骤

1. 根据系统需求（1.4 图书管理系统概述），使用在线绘图工具 process on 画系统用例图；
2. 根据需求，使用 Powerdesigner/process on 建立数据库的概念模型（E-R 图）；
3. 以 SYSTEM 登录数据库

打开 PL/SQL developer，以用户：SYSTEM，密码：orcl，登录 orcl 数据库，如图 3-1 所示：



图 3-1 登录数据库

4. 注册用户

打开 Users 文件夹，新建用户。用户名：Sxxxxxxx（其中：xxxxxxx 代表你的 8 位学号），密码自行设定，其他选项请参看图 3-2：

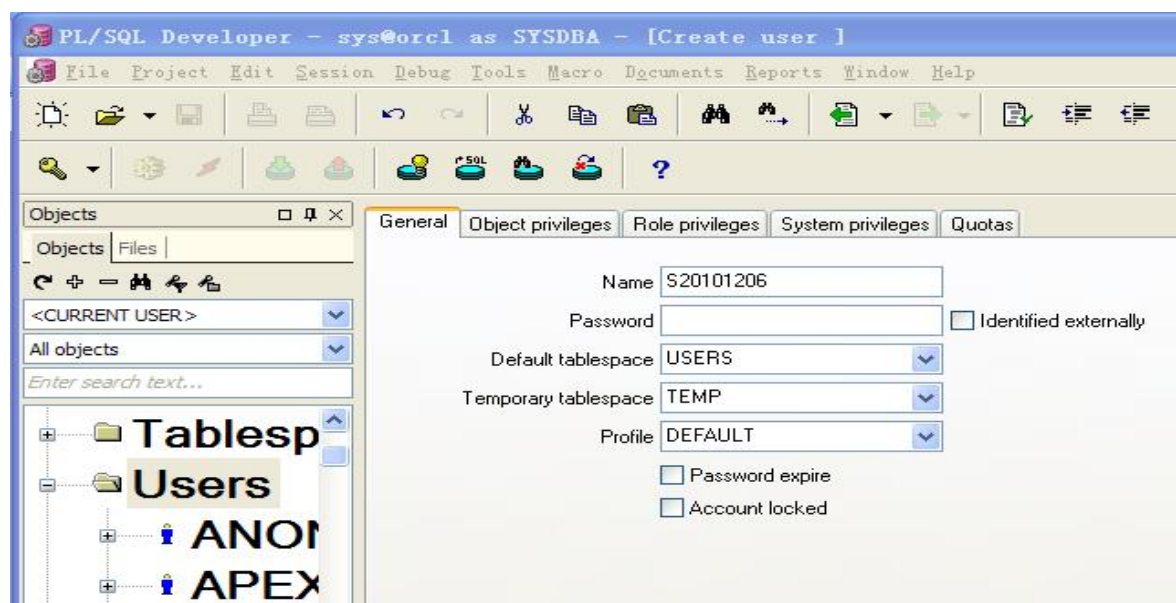


图 3-2 注册新用户

在角色权限（Role privileges）中选择：CONNECT 和 DBA

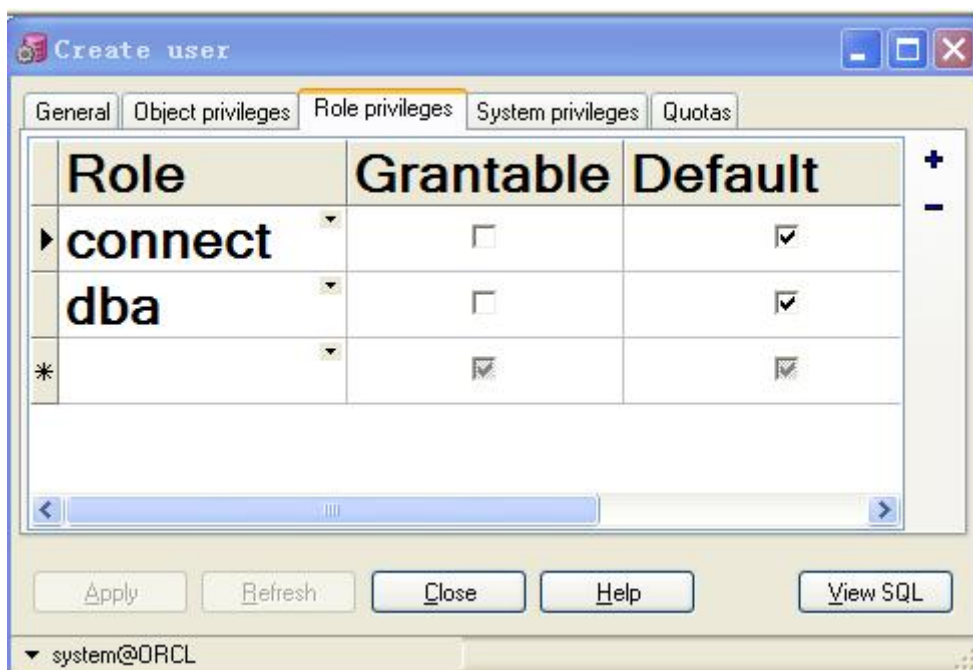


图 3-3 角色权限

5. 重新以新用户登录数据库



图 3-4 以新用户登录数据库

6. 建立数据库表

使用 SQL 建立以下各关系：

图书分类（图书分类号，类名）

书目（ISBN，书名，作者，出版单位，单价，图书分类号）

图书（图书编号，ISBN，是否借出，备注）

读者（借书证号，姓名，单位，性别，地址，联系电话，身份证编号）

借阅（借阅流水号，借书证号，图书编号，借书日期，归还日期，罚款分类号，备注）

罚款分类（罚款分类号，罚款名称，罚金）

预约（预约流水号，借书证号，ISBN，预约时间）



注意：要求根据图书馆管理业务规范，建立各关系的实体完整性，参照完整性以及用户自定义完整性。

7. 使用 SQL 为各关系输入数据如下：

图书分类（图书分类号，类名）

图书分类号	类名
100	文学
200	科技
300	哲学

书目（ISBN，书名，作者，出版单位，单价，图书分类号）

ISBN	书名	作者	出版单位	单价	图书分类号
7040195836	数据库系统概论	王珊	高等教育出版社	39.00	200
9787508040110	红楼梦	曹雪芹	人民出版社	20.00	100
9787506336239	红楼梦	曹雪芹	作家出版社	34.30	100
9787010073750	心学之路	张立文	人民出版社	33.80	300

图书（图书编号，ISBN，是否借出，备注）

图书编号	ISBN	是否借出	备注
2001231	7040195836	否	
2001232	7040195836	是	
1005050	9787506336239	否	
1005063	9787508040110	是	
3007071	9787010073750	是	

读者（借书证号，姓名，单位，性别，地址，联系电话，身份证编号）

借书证号	姓名	单位	性别	地址	联系电话	身份证编号
20051001	王菲	四川绵阳西科大计算机学院	女
20062001	张江	四川绵阳中心医院	男
20061234	郭敬明	四川江油 305	男
20071235	李晓明	四川成都工商银行	男
20081237	赵鑫	四川广元广元中学	女

借阅（借阅流水号，借书证号，图书编号，借书日期，归还日期，罚款分类号，备注）

借阅流水号	借书证号	图书编号	借书日期	归还日期	罚款分类号	备注
1	20081237	3007071	2010/09/19	2010/09/20		
2	20071235	1005063	2010/10/20	2011/02/20	1	
3	20071235	2001232	2011/09/01			

4	20061234	1005063	2011/9/20			
5	20051001	3007071	2011/9/10			
6	20071235	1005050	2011/10/20	2012/02/20	1	

罚款分类（罚款分类号，罚款名称，罚金）

罚款分类号	罚款名称	罚金
1	延期	10
2	损坏	20
3	丢失	50

预约（预约流水号，借书证号，ISBN，预约时间）

预约流水号	借书证号	ISBN	预约时间
1	20081237	9787508040110	2011/09/11

🔊 注意：日期型的格式需要进行设置，在 PL/SQL developer 中设置如下图：

选择 User defined:yyyy/mm/dd

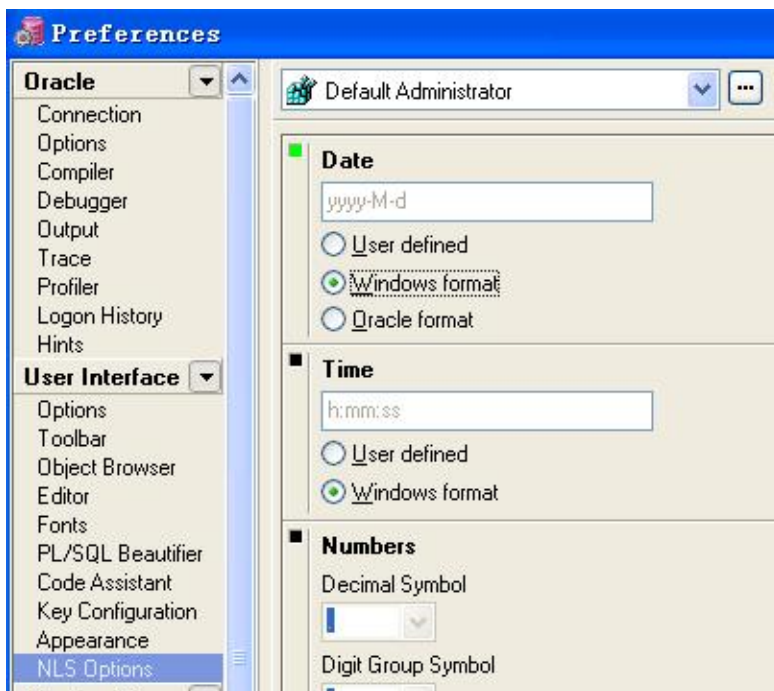


图 3-5 设置日期型格式

🔊 注意：在 Insert 语句中如何实现指定格式日期数据的添加，例如：Insert into 预约 values(1, '20081237', '2001232', to_date('2011-09-11', 'yyyy/mm/dd'))；

3.1.5 实验扩展

试根据下面的完整性约束要求，用 SQL 对上面已经建立好的数据库表进行完整性约束定义。

读者关系中属性 **联系电话** 取值为 11 位数字

身份证编号 取值为 18 位，并且满足身份证编号规则

图书关系中属性 **是否借出** 取值为：‘是’或‘否’

借阅关系中属性 **借书日期** 取值不为空。

3.2 实验 2 数据查询

3.2.1 实验类型

设计型实验

3.2.2 实验目的

- 掌握查询语句的一般格式；
- 熟练掌握单表查询、连接查询、集合查询、统计查询和嵌套查询。

3.2.3 基础知识

数据查询：数据库的查询是数据库的核心操作。PL/SQL 提供了 SELECT 语句进行数据库的查询，该语句具有灵活的使用方式和丰富的功能。它的一般格式如下：

```
SELECT [ ALL | DISTINCT ] < 目标列表表达式 > [, < 目标列表表达式 > ] ...  
FROM < 表名或视图名 > [, < 表名或视图名 > ] ...  
[ WHERE < 条件表达式 > ]  
[ GROUP BY < 列名 1 > [ HAVING < 条件表达式 > ] ]  
[ ORDER BY < 列名 2 > [ ASC | DESC ] ]
```

整个 SELECT 语句的含义是，根据 WHERE 子句的条件表达式，从 FROM 子句指定的基本表或视图中找出满足条件的元组，再按 SELECT 子句中的目标列表表达式，选出元组中属性值形成结果表。

如果有 GROUP BY 子句，则将结果按<列名 1>的值进行分组，该属性值相等的元组为一个组。通常会在每组中作用聚集函数。如果 GROUP BY 子句带 HAVING 短语，则只有满足指定条件的组才予以输出。

如果有 ORDER BY 子句，则结果表还将按照<列名 2>的值升序或降序排序。

3.2.4 实验内容

- 单表查询
- 连接查询
- 嵌套查询
- 集合查询

3.2.5 实验步骤

1、 查询“红楼梦”目前可借的各图书编号，及所属版本信息。（是否借出为‘否’的图

书)

- 2、查找高等教育出版社的所有书目及单价，结果按单价降序排序。
- 3、统计“红楼梦”各版的藏书数量（ISBN 不同则版本不同）。
- 4、查询学号“20061234”号借书证借阅未还的图书的信息。
- 5、查询各个出版社的图书最高单价、平均单价。
- 6、要查询借阅了两本和两本以上图书的读者的个人信息。
- 7、查询“王菲”的单位、所借图书的书名和借阅日期。
- 8、查询每类图书的册数和平均单价。
- 9、统计从未借书的读者人数。
- 10、统计参与借书的人数。
- 11、找出所有借书未还的读者的信息及所借图书编号及名称。
- 12、检索书名是以“Internet”开头的所有图书的书名和作者。
- 13、查询各图书的罚款总数。
- 14、查询借阅及罚款分类信息，如果有罚款则显示借阅信息及罚款名称、罚金，如果没有罚款则罚款名称、罚金显示空（左外连接）
- 15、查询借阅了所有“文学”类书目的读者的姓名、单位。


3.2.6 实验扩展

- 1、在书目关系中新增“出版年份”，并在该属性下添加数据。（使用 SQL 完成）

ISBN	书名	作者	出版单位	出版年份	单价	图书分类号
7040195836	数据库系统概论	王珊	高等教育出版社	2005	39.00	200
9787508040110	红楼梦	曹雪芹	人民出版社	1983	20.00	100
9787506336239	红楼梦	曹雪芹	作家出版社	2008	34.30	100
9787010073750	心学之路	张立文	人民出版社	2009	33.80	300

- 2、求总藏书量、藏书总金额、最高价、最低价。
- 3、列出藏书在 5 本以上的书目（书名、作者、出版社、出版年份）。

- 4、列出年份最久远的书?
- 5、 目前实际已借出多少册书?
- 6、哪一年的图书最多?
- 7、 哪本借书证未归还的图书最多?
- 8、平均每本借书证的借书册数。
- 9、哪个单位的读者平均借书册数最多?
- 10、 最近两年都未被借过的书。

注意：Oracle 系统日期时间 sysdate，更多关于 oracle 日期函数请自行查阅。

- 11、今年未借过书的借书证。

3.3 实验 3 存储过程与触发器

3.3.1 实验类型

设计型实验

3.3.2 实验目的

- 了解存储过程的概念、优点
- 熟练掌握创建存储过程的方法
- 熟练掌握存储过程的调用方法
- 了解触发器的概念、优点
- 掌握触发器的方法和步骤
- 掌握触发器的使用

3.3.3 基础知识

存储过程：存储过程是指用于特定操作的 PL/SQL 块，是由控制流和 SQL 语句书写的过程。存储过程经编译和 SQL 优化后存储在数据库服务器中，使用时只要调用即可。在 oracle 数据库中，若干个有联系的存储过程，可以组合在一起构成包。

存储过程的优点：

1. 存储过程是预编译过的，并且经优化后存储于 SQL 内存中，使用时无需再次编译，提高了工作效率；
2. 存储过程的代码直接存放于数据库中，一般由客户端直接通过存储过程的名字进行调用，减少了网络流量，加快了系统执行速度；
3. 使用存储过程可以减少 SQL 注入式攻击，提高了系统的安全性，执行存储过程的用户要具有一定的权限才能使用存储过程；
4. 在同时进行主、从表及多表间的数据维护及有效性验证时，使用存储过程比较方便，而且可以有效利用 SQL 中的事务处理机制；
5. 使用存储过程，可以实现存储过程设计和编码工作分开进行，只要将存储过程名、参数及返回信息告诉编程人员即可。

创建存储过程：PL/SQL 提供了 CREATE 语句进行存储过程的创建，它的一般格式如下：

```
CREATE 【OR REPLACE】 PROCEDURE procedure_name
[(parameter1 【model】 datatype1,parameter2 【model】 datatype2 ...)]
IS [AS]
BEGIN
    PL/SQL Block;
END [procedure_name];
```

说明：使用 replace 关键字表示如果要创建的存储过程已经存在，则将其替换为当前定义的存储过程。

执行存储过程：当需要执行存储过程时，可使用 CALL 或 EXECUTE 语句，它的一般格式如下：

```
[[ EXEC [ UTE ] ] procedure_name ;
```

```
CALL procedure_name ;
```

触发器：是一种特殊的存储过程，它在插入，删除或修改特定表中的数据时触发执行，它比数据库本身标准的功能有更精细和更复杂的数据控制能力。

触发器的作用：

1、安全性。可以基于数据库的值使用户具有操作数据库的某种权利。

(1) 可以基于时间限制用户的操作，例如不允许下班后和节假日修改数据库数据。

(2) 可以基于数据库中的数据限制用户的操作，例如不允许股票的价格的升幅一次超过 10%。

2、审计。可以跟踪用户对数据库的操作。

(1) 审计用户操作数据库的语句。

(2) 把用户对数据库的更新写入审计表。

3、实现复杂的数据完整性规则。

(1) 实现非标准的数据完整性检查和约束。触发器可产生比规则更为复杂的限制。与规则不同，触发器可以引用列或数据库对象。例如，触发器可回退任何企图吃进超过自己保证金的期货。

(2) 提供可变的缺省值。

4、实现复杂的非标准的数据库相关完整性规则。触发器可以对数据库中相关的表进行连环更新。例如，在 auths 表 author_code 列上的删除触发器可导致相应删除在其它表中的与之匹配的行。

(1) 在修改或删除时级联修改或删除其它表中的与之匹配的行。

(2) 在修改或删除时把其它表中的与之匹配的行设成 NULL 值。

(3) 在修改或删除时把其它表中的与之匹配的行级联设成缺省值。

(4) 触发器能够拒绝或回退那些破坏相关完整性的变化，取消试图进行数据更新的事务。当插入一个与其主键不匹配的外部键时，这种触发器会起作用。

创建触发器：PL/SQL 提供了 CREATE 语句进行触发器的创建，它的一般格式如下：

```
create [or replace] trigger 触发器名 触发时间 触发事件  
on 表名  
[for each row]  
pl/sql 语句
```

其中：

触发器名：触发器对象的名称。由于触发器是数据库自动执行的，因此该名称只是一个名称，没有实质的用途。

触发时间：指明触发器何时执行，该值可取：

before---表示在数据库动作之前触发器执行；

after---表示在数据库动作之后触发器执行。

触发事件：指明哪些数据库动作会触发此触发器：

insert：数据库插入会触发此触发器；

update：数据库修改会触发此触发器；

delete：数据库删除会触发此触发器。

表名：数据库触发器所在的表。

for each row：对表的每一行触发器执行一次。如果没有这一选项，则只对整个表执行一次。

3.3.4 实验内容

1. 建立存储过程
2. 调用存储过程
3. 掌握触发器的建立和应用

3.3.5 实验步骤

- 1、建立存储过程完成图书管理系统中的**借书**功能，并调用该存储过程实现**借书**功能。
功能要求：
 - 借书时要求输入**借阅流水号**，**借书证号**，**图书编号**。（即该存储过程有 3 个输入参数）
 - 借书时，借书日期为系统时间。
 - 图书的是否借出改为‘是’
 - 2、建立存储过程完成图书管理系统中的**预约**功能。
 - 预约时要求输入**预约流水号**，**借书证号**，**ISBN**。（即该存储过程有 3 个输入参数）
 - 存储过程先检查输入的 ISBN 版本的图书是否都已借出，如果是则进行预约，否则提示“该书目有可借图书，请查找”。
 - 预约时间为系统时间。
 - 3、建立存储过程完成图书管理系统中的**还书**功能。
 - 还书时要求输入**借书证号**，**图书编号**，**罚款分类号**（即该存储过程有 3 个输入参数）。
 - 还书日期为系统时间。
 - 图书的是否借出改为‘否’。
 - 4、通过序列和触发器实现借阅表中借阅流水号字段的自动递增。
 - 5、修改**借书**功能的存储过程。
该存储过程要求：
 - (1) 借书时输入借书证号，图书编号。（即该函数有 2 个输入参数）
 - (2) 借书时，借书日期为系统时间。

*该存储过程主体部分只有 insert into 语句。
- 实验拓展题：**
- (3) 还书时，如果超期（该图书借阅时间超过两个月），那么借阅表中该记录的罚款

分类号值为 1。

6、建立与借书存储过程相对应的触发器，当借阅表中加入借阅信息时，该触发器触发，自动修改所借图书的是否借出改为‘是’。

实验拓展题：

7、建立丢失存储过程，如果图书丢失，那么借阅表中该记录的罚款分类号的值为 3，图书表中该书的是否借出为“是”（本来为‘是’，不用修改），备注为“丢失”。

3.4 实验 4 数据库恢复与安全性

3.4.1 实验类型

设计型实验

3.4.2 实验目的

- 了解 oracle 的物理备份
- 掌握 oracle 数据库逻辑备份方法
- 掌握 oracle 数据库恢复的方法
- 学会使用 exp 备份数据库、使用 imp 恢复数据库
- 了解 flashback 的使用
- 学会使用 PL/SQL developer 工具完成导入导出
- 掌握 ORACLE 中有关用户创建的方法
- 熟练掌握 PL/SQL 的数据控制语言，能通过自主存取控制进行权限管理
- 熟悉用户资源文件的使用
- 熟悉 ORACLE 中角色管理
- 熟悉视图机制在自主存取控制上的应用

3.4.3 基础知识

数据库的备份与恢复是两个相对应的概念，备份是恢复的基础，恢复是备份的目的。

数据库备份：是指系统管理员定期或不定期地将数据库部分或全部内容复制到磁带或另一个磁盘上保存起来的过程。备份可分为静态备份和动态备份。

数据库恢复：是指在数据库遭到破坏时使数据库从有效的备份中恢复正常。

备份期间不允许对数据库进行任何存取、修改活动的备份方式称为静态备份。备份期间允许对数据库进行存取或修改，即备份和用户事务可以并发执行的备份方式称为动态备份。

Oracle 数据库的备份分为：

1. 物理备份（主）：将数据库各类文件，进行操作系统级别的复制。
 - 冷备份
 - 热备份
2. 逻辑备份（辅）：利用 ORACLE 提供的导出工具，将数据库中的数据、表、模式等以二进制文件的形式存储。
 - 利用 Export 可将数据从数据库中提取出来，利用 Import 则可将提取出来的数据送回到 Oracle 数据库中去。
 - Oracle 支持三种方式类型的输出：
 - (1)表方式(T 方式)，将指定表的数据导出。
 - (2)用户方式(U 方式)，将指定用户的所有对象及数据导出。

(3)全库方式(Full 方式)，将数据库中的所有对象导出。

a) 导出自己的表

```
d:\programme\oracle\bin>exp          userid=user1/m123@orcl          table=(emp)
file=d:\emp.dmp
```

b) 导出其它用户的表

```
exp userid=scott/tiger@myorcl owner=scott file=d:\scott.dmp
```

c) 导出数据库

```
exp userid=system/manager@myorcl full=y inctype=complete
file=d:\all.dmp
```

```
exp userid=system/manager@myorcl full=y inctype=incremental
file=d:\inall.dmp
```

d) 导入自己的表

```
imp userid=scott/tiger@myorcl tables=(emp) file=d:\xx.dmp
```

e) 导入表到其它用户

要求该用户具有 dba 的权限，或是 imp_full_database

```
imp userid=system/tiger@myorcl tables=(emp) file=d:\xx.dmp touser=scott
```

数据库的安全性：是指保护数据库以防止不合法的使用所造成的数据泄露、更改或破坏。

用户标识与鉴别：是系统提供的最外层安全保护措施。其方法是由用户提供一定的方式让用户标识自己的名字或身份。每次用户要求进入系统时，由系统进行核对，通过鉴定后才提供机器使用权。

对于获得上机权的用户若要使用数据库时数据库管理系统还要进行用户标识和鉴定。用户标识和鉴定的方法有很多种，而且在一个系统中往往是多种方法并举，以获得更强的安全性。常用的方法有：

(1) 用户标识

用一个用户名或者用户标识号来标明用户身份。系统内部记录着所有合法用户的标识，系统鉴别该用户是否是合法用户，若是，则可以进入下一步的核实；若不是，则不能使用系统。

(2) 口令

为了进一步核实用户，系统常常要求用户输入口令。为保密起见，用户在终端上输入的口令不显示在屏幕上。系统核对口令以鉴别用户身份。口令虽然简单易行，但容易被人窃取。

(3) 计算过程或者函数

系统提供一个随机数，用户根据自己预先约定的计算过程或者函数进行计算。系统根据用户计算结果是否正确鉴定用户身份。

利用 PL/SQL 创建数据库新用户

创建新的用户，其语法格式为：

```
CREATE USER username
IDENTIFIED BY password
DEFAULT TABLESPACE tablespace
TEMPORARY TABLESPACE tablespace
PROFILE profile
QUOTA integer|UNLIMITED ON tablespace
```

各选项含义如下：

IDENTIFIED BY password: 用户口令；

DEFAULT TABLESPACE tablespace: 默认表空间；

TEMPORARY TABLESPACE tablespace: 临时表空间；

PROFILE profile|DEFAULT: 用户资源文件；

QUOTA integer[K|M]|UNLIMITED ON tablespace: 用户在表空间上的空间使用限额，可以指定多个表空间的限额。

存取控制机制：数据库安全性所关心的主要是 DBMS 的存取控制机制。数据库安全最重要的一点就是确保只授权给有资格的用户访问数据库的权限，同时令所有未被授权的人员无法接近数据，这主要是通过数据库系统的存取控制机制实现。

存取控制机制主要包括两方面：定义用户权限，并将用户权限登记到数据字典中；合法权限检查。

存取控制方法分类：自主存取控制法和强制存取控制法

自主存取控制法

相对于强制存取方法，其不同点体现在：

- 同一用户对于不同的数据对象有不同的存取权限。
- 不同的用户对同一对象也有不同的存取权限
- 用户还可将其拥有的存取权限转授给其他用户

大型 DBMS 几乎都支持自主存取控制，目前的 SQL 标准也对自主存取控制提供支持，这主要通过 SQL 的数据控制语句来实现。

GRANT：对指定操作对象的指定操作权限授予指定的用户，其语法格式为：

```
GRANT < 权限 > [, < 权限 > ] ...
ON < 对象类型 > < 对象名 > [, < 对象类型 > < 对象名 > ] ...
TO < 用户 > [, < 用户 > ] ...
[ WITH GRANT OPTION ]
```

REVOKE：取消授予用户的权限，其语法格式为：

```
REVOKE < 权限 > [, < 权限 > ] ...
ON < 对象类型 > < 对象名 > [, < 对象类型 > < 对象名 > ] ...
FROM < 用户 > [, < 用户 > ] ...
```

角色：数据库角色是被命名的一组与数据库操作相关的权限，角色是权限的集合。因此可以为一组具有相同权限的用户创建一个角色，使用角色来管理数据库权限可以简化授权的过程。

```
Create role rolename
```

```
grant 权限 on 对象 to rolename
grant 角色 to rolename
```

视图：视图是虚表，是从一个或几个基本表（或视图）导出的表，在数据库中只存放视图的定义，不会出现数据冗余。当基本表中的数据发生变化，从视图中查询出的数据也随之改变。

建立视图：PL/SQL 提供了 CREATE 语句进行视图的创建，它的一般格式如下：

```
CREATE VIEW < 视图名 > (< 列名> [,< 列名>] ...)  
AS < 子查询 >  
[ WITH CHECK OPTION ]
```

查询视图：PL/SQL 提供了 SELECT 语句进行视图的查询。从用户的角度，对视图进行查询与对基本表的查询是一样的。

更新视图：是指通过视图来插入（INSERT）、删除（DELETE）和修改（UPDATE）数据。由于视图是不实际存储数据的虚表，因此对视图的更新，最终要转换为对基本表的更新操作。

删除视图：PL/SQL 提供了 DROP 语句进行视图的删除，它的一般格式如下：

```
DROP VIEW < 视图名 >
```

用户概要文件：用户资源文件用来对用户的资源存取进行限制，包括：cpu 使用时间限制、内存逻辑读个数限制、每个用户同时可以连接的会话数限制、一个会话的空间和时间限制、一个会话的持续时间限制、每次会话的专用 SGA 空间限制。

创建用户概要文件的语法如下：

```
CREATE PROFILE filename LIMIT  
FAILED_LOGIN_ATTEMPTS integer  
PASSWORD_LOCK_TIME integer  
SESSION_PER_USER integer  
CPU_PER_SESSION integer  
USER_PER_CALL integer  
CONNECT_TIME integer  
.....
```

其中：

FAILED_LOGIN_ATTEMPTS：用户连续数次登录失败后，则锁定该账户；

PASSWORD_LOCK_TIME：锁定账户自动解锁需经过的天数；

SESSION_PER_USER：用户可以同时连接的会话数量限额；

CPU_PER_SESSION：用户在一次数据库会话期间可占用的 CPU 时间总量限额，单位为百分之一秒；

USER_PER_CALL：用户一次 SQL 调用可用的 CPU 时间总量限额，单位为百分之一秒；

CONNECT_TIME：一次连接的时间总量限额，单位为分钟，连接时间超过此值时，连接被断开；

PRIVATE_SGA：用户私有的 SGA 区的大小，单位为数据库块，默认值为 UNLIMITED；

COMPOSITE_LIMIT: 这是一项由上述限制参数构成的组合资源项。

3.4.4 实验内容

1. 使用 exp 导出数据库
2. 使用 imp 导入数据库
3. 使用 flashback 闪回表
4. 使用 PLSQL/developer 工具完成导出
5. 使用 PLSQL/developer 工具完成导入
6. 创建数据库用户
7. 实现自主存取控制
8. 角色管理
9. 视图机制在自主存取控制上的应用
10. 用户概要文件的使用

3.4.5 实验步骤

1、逻辑备份

- (1) 导出自己用户中的“预约”表
在运行中输入: exp 用户名/密码@orcl
按照提示进行导出
 - (2) 删除自己用户中的“预约”表
 - (3) 进行导入数据库操作
在运行中输入: IMP 用户名/密码@orcl
按照提示进行导入
 - (4) 查询导入的“预约”表中的信息。
 - (5) 导出数据库 (以全库方式导出)。
- 必须是 DBA 才能执行完整数据库或表空间导出操作。

2、使用 Flashback

- (1) 设置行可移动
SQL>ALTER TABLE 读者 ENABLE ROW MOVEMENT
- (2) 在读者表中添加多条记录 (或者删除没有借书的读者记录)。
- (3) 闪回到改变前 (TO_TIMESTAMP 函数完成对非时间戳类型数据的转换)
SQL>FLASHBACK TABLE 读者 TO_TIMESTAMP TO_TIMESTAMP(...)

3、使用 PLSQL/developer 来完成 SQL 导出

- (1) 打开 PLSQL/developer, 选择菜单“工具”→“导出表”
- (2) 点击你要导出的表 (如: “预约”表), 然后选择标签 SQL 插入
- (3) 选中复选框创建表, 浏览或者输入输出文件, 然后点击导出
- (4) 在你输入的目录下找到你的导出文件 (SQL 文件)
- (5) 删除自己表空间中的“预约”表
- (6) 通过“工具”→“导入表”, 利用 SQL 插入导入“预约”表。
- (7) 查询导入的“预约”表, 检查导出是否正确。

4、使用 PLSQL/developer 来完成 PLSQL/developer 方式导出

- (1) 打开 PLSQL/developer, 选择菜单“工具”→“导出表”

- (2) 点击你要导出的表（如：“预约”表），然后选择标签 PLSQL/developer
 - (3) 浏览或者输入输出文件，然后点击导出。
 - (4) 在你输入的目录下找到你的导出文件。
 - (5) 删除自己表空间中的“预约”表
 - (6) 通过“工具”→“导入表”，PLSQL/developer 方式导入“预约”表。
 - (7) 查询导入的“预约”表，检查导出是否正确。
- 5、以 SYSTEM 登录数据库，为你的帐号增加系统角色 DBA.
- 6、(1) 创建一个数据库用户，用户名为“自己名字全拼”（如：sashixuan），指定默认表空间，设置密码，该用户拥有 CONNECT, RESOURCE 角色，以及对图书管理数据库中所有表的增删改查权限。
- (2) 查看该用户拥有的角色
 - (3) 查看该用户拥有的系统权限
 - (4) 查看该用户的对象权限
- 7、建立角色：用户_role（用户为“自己名字全拼”），该角色拥有调用存储过程借书、还书、预约的权限，以及 CONNECT 系统角色权限。
- （注：执行存储过程的授权语句 Grant execute on procedure_name to user/role）
- 8、创建一个数据库用户：用户_oper（用户为“自己名字全拼”）
为该用户授权角色：用户_role。以该用户登录，完成借书，还书功能。
- 9、以自己的帐号登录，建立视图 VIEW_READER，该视图包含书目（ISBN，书名，作者，出版单位，图书分类名称）（数据来自书目和图书表）
- 10、创建一个数据库用户：用户_USER1（用户为“自己名字全拼”）
该用户具有对视图 VIEW_READER 查询的权限。
- 11、创建一个概要文件，如果 用户_USER1 连续 3 次登录失败，则锁定该账户，10 天后该账户自动解锁。以该用户登录进行权限测试。
- 12、手工解锁：用户_USER1。
- 13、撤销用户（用户_USER1）权限，删除该用户。

附录 A 实验报告格式

数据库原理及应用 A 实验报告

实验名称: _____

实验类型: _____

指导教师: _____

专业班级: _____

姓 名: _____

学 号: _____

联系电话: _____

成绩: _____

1. 实验目的

2. 实验环境

3. 实验内容

4. 实验代码

5. 实验结果

6. 实验心得