

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的书籍交易的数据库设计 | | |
| 课程 | Oracle数据库应用 | | |
| 学 院 | 信息科学与工程学院 | | |
| 专 业 | 软件工程 | 年级 | 2018级 |
| 学生姓名 | 高洋 | 学号 | 201810414109 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 10 |  |
| 表设计 | 表，表空间设计合理，数据合理 | 20 |  |
| 用户管理 | 权限及用户分配方案设计正确 | 20 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 30 |  |
| 备份方案 | 备份方案设计正确 | 20 |  |
| **得分合计** | | |  |

2021 年 6 月 1 日

目录

[第一章 结构设计 1](#_Toc9168)

[1.1 编写目的 1](#_Toc22412)

[1.2 背景 1](#_Toc11613)

[1.3 参考资料 1](#_Toc12683)

[第二章 结构设计 1](#_Toc9168)

[第三章 运用设计 1](#_Toc24962)

[3.1创建表空间 1](#_Toc30744)

[3.2 创建角色及用户并分配权限 2](#_Toc24004)

[3.3 在用户qhl\_1下建表、视图 4](#_Toc10021)

[3.4 插入用户、商品、购物车、管理员、论坛数据 13](#_Toc14273)

[3.5 创建程序包、存储过程、函数执行分析计划 14](#_Toc27622)

[3.6 备份方案 16](#_Toc22781)

第一章

引言：基于oracle的书籍交易系统的数据库设计

**1.1** 编写目的

本数据库设计的是书籍交易系统。数据库可直接用于书籍交易系统的数据存储。在本数据库中做了表空间限制、用户权限限制，采用了分区表，使用了程序包、存储过程、自定义函数等方案。此外，本数据库还使用了备份、容灾方案。本数据库努力提高数据的安全性，防止数据丢失等情况发生。

**1.2 背景**

数据存储一直是一个很关键的技术，现在的数据存储技术不仅要保证无误的存储数据还要保证数据的安全性。在学习完oracle后，我设计了本数据库方案模拟公司数据存储，在设计中使用了很多oracle的特性用于提高数据查询效率，使用便利，保证数据安全。

**1.3 参考资料**

《Oracle 12C 数据库基础教程》，赵卫东，刘永红，于曦，李力，鄢涛科学出版社

**第二章 结构设计**

**2.1 逻辑结构设计**

本数据库在表空间space\_qhl001下，属于用户qhl\_1。

一共五张表，一个视图。

表BOOKUSER用于存储用户信息，本表根据注册时间采用范围分区存储。

表ADMINISTRATOR用于管理员信息存储。

表COMMODITY 用于存储商品信息。

表CART 用于存储用户购物车信息，本表根据用户id采用引用分区。

表FORUM 用于存储论坛信息。

视图VIEW\_SINGLEPRICESUM用于计算每个用户购物车中单个商品的总价。

1. **运用设计**

**3.1创建表空间**

创建两个表空间：分别为**space\_qhl001**、**space\_qhl002**本数据库用space\_qhl001

Create Tablespace space\_qhl001

datafile

'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest\_qhl001\_1.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,

'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest\_qhl001\_2.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED

EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;

Create Tablespace space\_qhl002

datafile

'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest\_qhl002\_1.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,

'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest\_qhl002\_2.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED

EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;



**3.2 创建角色及用户并分配权限**

用户默认使用表空间space\_qhl001

**3.2.1创建第一个角色和用户**

创建角色qhl1将connect,resource,create view授权给qhl1

创建用户qhl\_1

分配60M空间给qhl\_1并将角色qhl1授权给用户qhl\_1

CREATE ROLE qhl1;

GRANT connect,resource,CREATE VIEW TO qhl1;

CREATE USER qhl\_1 IDENTIFIED BY 123 DEFAULT TABLESPACE space\_qhl001 TEMPORARY TABLESPACE temp;

ALTER USER qhl\_1 QUOTA 60M ON space\_qhl001;

GRANT qhl1 TO qhl\_1;



**3.2.2创建第二个角色和用户**

创建角色qhl2，将connect,resource权限给qhl2

创建用户qhl\_2

分配60M空间给qhl\_2并将角色qhl2授权给用户qhl\_2

CREATE ROLE qhl2;

GRANT connect,resource TO qhl2;

CREATE USER qhl\_2 IDENTIFIED BY 123 DEFAULT TABLESPACE space\_qhl001 TEMPORARY TABLESPACE temp;

ALTER USER qhl\_2 QUOTA 60M ON space\_qhl001;

GRANT qhl2 TO qhl\_2;



**3.3 在用户qhl\_1下建表、视图**

**3.3.1创建管理员表**

CREATE TABLE ADMINISTRATOR

(

ID NUMBER(\*, 0) NOT NULL

, PASSWORD VARCHAR2(20 BYTE) NOT NULL

, ADMIN VARCHAR2(20 BYTE) NOT NULL

, CONSTRAINT ADMINISTRATOR\_PK PRIMARY KEY

(

ID

)

USING INDEX

(

CREATE UNIQUE INDEX ADMINISTRATOR\_PK ON ADMINISTRATOR (ID ASC)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 2

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOPARALLEL

)

ENABLE

)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NO INMEMORY

NOPARALLEL;

**3.3.2创建用户表**

根据注册日期按范围分区

分为2018和2019年两个分区，每年按季度划4个子分区

CREATE TABLE BOOKUSER

(

ID NUMBER(\*, 0) NOT NULL

, PASSWORD VARCHAR2(20 BYTE) NOT NULL

, USERNAME VARCHAR2(50 BYTE) NOT NULL

, PHONE VARCHAR2(20 BYTE) NOT NULL

, ADDRESS VARCHAR2(30 BYTE) NOT NULL

, REGISTRATIONDATE DATE NOT NULL

, CART\_ID NUMBER(\*, 0) NOT NULL

, CONSTRAINT U\_PK PRIMARY KEY

(

ID

)

USING INDEX

(

CREATE UNIQUE INDEX U\_PK ON BOOKUSER (ID ASC)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 2

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOPARALLEL

)

ENABLE

)

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NOPARALLEL

PARTITION BY RANGE (REGISTRATIONDATE)

SUBPARTITION BY RANGE (REGISTRATIONDATE)

(

PARTITION DATE2018 VALUES LESS THAN (TO\_DATE(' 2018-12-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

(

SUBPARTITION DATE2018\_3 VALUES LESS THAN (TO\_DATE(' 2018-03-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2018\_6 VALUES LESS THAN (TO\_DATE(' 2018-06-30 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2018\_9 VALUES LESS THAN (TO\_DATE(' 2018-09-30 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2018\_12 VALUES LESS THAN (TO\_DATE(' 2018-12-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

)

, PARTITION DATE2019 VALUES LESS THAN (TO\_DATE(' 2019-12-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

(

SUBPARTITION DATE2019\_3 VALUES LESS THAN (TO\_DATE(' 2019-03-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2019\_6 VALUES LESS THAN (TO\_DATE(' 2019-06-30 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2019\_9 VALUES LESS THAN (TO\_DATE(' 2019-09-30 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

, SUBPARTITION DATE2019\_12 VALUES LESS THAN (TO\_DATE(' 2019-12-31 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS\_CALENDAR=GREGORIAN'))

NOCOMPRESS NO INMEMORY

)

);

**3.3.3创建商品表**

CREATE TABLE COMMODITY

(

ID NUMBER(\*, 0) NOT NULL

, PID NUMBER(\*, 0) NOT NULL

, BOOKSNAME VARCHAR2(20 BYTE) NOT NULL

, PRICE NUMBER NOT NULL

, DESCRIBE VARCHAR2(50 BYTE) NOT NULL

, NUM NUMBER(\*, 0) NOT NULL

, ADMIN\_ID NUMBER(\*, 0) NOT NULL

, CONSTRAINT COMMODITY\_PK PRIMARY KEY

(

ID

)

USING INDEX

(

CREATE UNIQUE INDEX COMMODITY\_PK ON COMMODITY (ID ASC)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 2

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOPARALLEL

)

ENABLE

)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NO INMEMORY

NOPARALLEL;

**3.3.4创建购物车表**

用户表字段BOOKUSER\_ID为购物车表的外键

购物车采用引用分区

CREATE TABLE CART

(

ID NUMBER(\*, 0) NOT NULL

, AMOUNT NUMBER(\*, 0) NOT NULL

, PID NUMBER(\*, 0) NOT NULL

, BOOKUSER\_ID NUMBER(\*, 0) NOT NULL

, CONSTRAINT CART\_PK PRIMARY KEY

(

ID

)

USING INDEX

(

CREATE UNIQUE INDEX CART\_PK ON CART (ID ASC)

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 2

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOPARALLEL

)

ENABLE

, CONSTRAINT CART\_BOOKUSER FOREIGN KEY

(

BOOKUSER\_ID

)

REFERENCES BOOKUSER

(

CART\_ID

)

ENABLE

)

PCTFREE 10

PCTUSED 40

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NOPARALLEL

PARTITION BY REFERENCE (CART\_BOOKUSER)

(

PARTITION DATE2018\_3

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2018\_6

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2018\_9

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2018\_12

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2019\_3

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2019\_6

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2019\_9

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

, PARTITION DATE2019\_12

LOGGING

TABLESPACE SPACE\_QHL001

PCTFREE 10

INITRANS 1

STORAGE

(

BUFFER\_POOL DEFAULT

)

NOCOMPRESS NO INMEMORY

);

**3.3.5论坛表**

CREATE TABLE TABLE1

(

ID INT NOT NULL

, CONTENT NVARCHAR2(50) NOT NULL

, CONSTRAINT TABLE1\_PK PRIMARY KEY

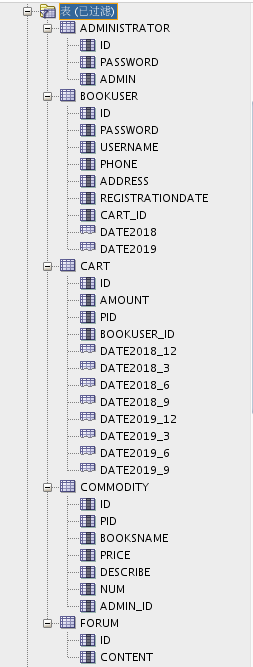
(

ID

)

ENABLE

);



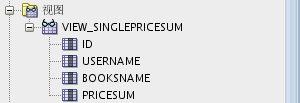
**3.3.6创建视图计算每个用户购车中单个商品的总价**

create or replace view view\_SinglePriceSum

as

select b.id,b.username,co.booksname,(co.price\*ca.amount) pricesum from COMMODITY co,cart ca,BOOKUSER b where co.pid=ca.pid and ca.BOOKUSER\_ID =b.id;

select \* from view\_SinglePriceSum;



**3.4 插入用户、商品、购物车、管理员、论坛数据**

**3.4.1自动插入60000条数据**

declare

id number(38,0);

username varchar2(50);

phone varchar2(20);

address varchar2(30);

REGISTRATIONDATE date;

booksname varchar2(50);

price number(5,2);

num number(38,0);

amount number(38,0);

begin

for i in 1..20000

loop

if i mod 2 =0 then

REGISTRATIONDATE:=to\_date('2018-5-6','yyyy-mm-dd')+(i mod 60);

else

REGISTRATIONDATE:=to\_date('2019-5-6','yyyy-mm-dd')+(i mod 60);

end if;

--插入用户

id:=SEQ\_ORDER\_ID.nextval; --应该将SEQ\_ORDER\_ID.nextval保存到变量中。

username := 'aa'|| 'aa';

username := 'wang' || i;

phone := '131785693' || i;

booksname := '唐诗三百首版本号' || i;

address :='成都'|| '四川';

price :=(dbms\_random.value() \* 100);

num :=(i mod 5);

insert /\*+append\*/ into bookuser (id,password,username,phone,address,REGISTRATIONDATE,cart\_id)

values (id,username,username,phone,address,REGISTRATIONDATE,id);

--插入货品

insert into commodity(id,pid,booksname,price,describe,num,admin\_id)

values (id,id,booksname,price,'good',num,1);

--插入购物车

amount :=(id mod 3 ) + 1;

insert into cart(id,amount,pid,bookuser\_id)

values (id,amount,id,id);

IF I MOD 1000 =0 THEN

commit; --每次提交会加快插入数据的速度

END IF;

end loop;

end;

**3.5 创建程序包、存储过程、函数执行分析计划**

**3.5.1创建程序包**

函数getcartsumprice计算每个用户的购物车商品总金额

存储过程adduser插入用户信息

create or replace PACKAGE book\_package Is

function getcartsumprice(user\_id number) return number;

procedure adduser(password varchar2,username varchar2,phone varchar2,address varchar2,registerdate VARCHAR2);

end book\_package;

**3.5.2创建函数、存储过程**

create or replace PACKAGE body book\_package Is

function getcartsumprice(user\_id number) return number as

begin

declare cart\_sum number;

query\_sql varchar2(200);

begin

query\_sql:='select sum(pricesum) from view\_SinglePriceSum where ID=' || user\_id;

execute immediate query\_sql into cart\_sum;

return cart\_sum;

end;

end getcartsumprice;

procedure addUser(password varchar2,username varchar2,phone varchar2,address varchar2,registerdate varchar2) as

begin

declare maxId number;

begin

select max(id) into maxId from bookuser;

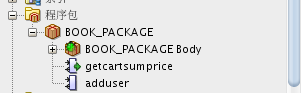
insert into bookuser values(maxId+1,password,username,phone,address,to\_date(registerdate,'yyyy-mm-dd'),maxId+1);

commit;

end;

end adduser;

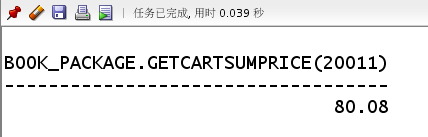
end book\_package;



**3.5.3 存储过程、函数执行分析**

使用自定义函数getcartsumprice（）查询id号为20011的用户购物车商品总价

select BOOK\_PACKAGE.getcartsumprice(20011) from dual;



使用存储过程adduser插入用户数据

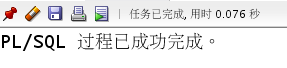
set serveroutput on

declare

begin

BOOK\_PACKAGE.addUser('131','cwd','125626','hongkong','2019-05-02');

end;

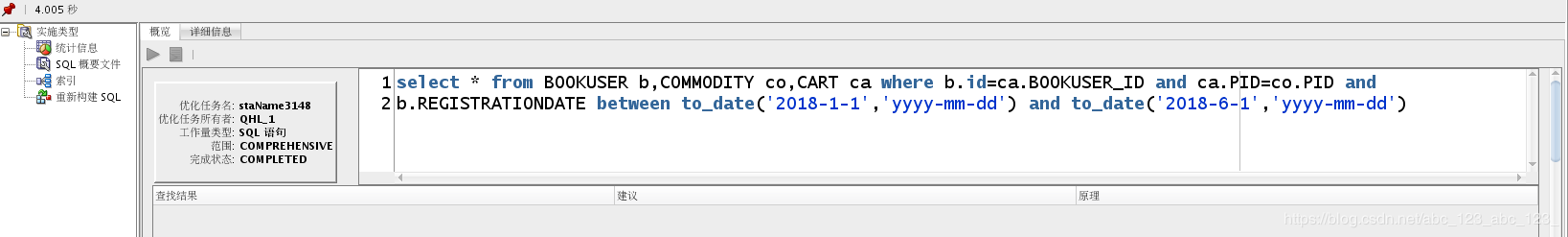




执行计划分析

select \* from BOOKUSER b,COMMODITY co,CART ca where b.id=ca.BOOKUSER\_ID and ca.PID=co.PID and

b.REGISTRATIONDATE between to\_date('2018-1-1','yyyy-mm-dd') and to\_date('2018-6-1','yyyy-mm-dd');



3.5.6表空间使用状况

SELECT a.tablespace\_name "表空间名",

total "表空间大小",

free "表空间剩余大小",

(total - free) "表空间使用大小",

total / (1024 \* 1024 \* 1024) "表空间大小(G)",

free / (1024 \* 1024 \* 1024) "表空间剩余大小(G)",

(total - free) / (1024 \* 1024 \* 1024) "表空间使用大小(G)",

round((total - free) / total, 4) \* 100 "使用率 %"

FROM (SELECT tablespace\_name, SUM(bytes) free

FROM dba\_free\_space

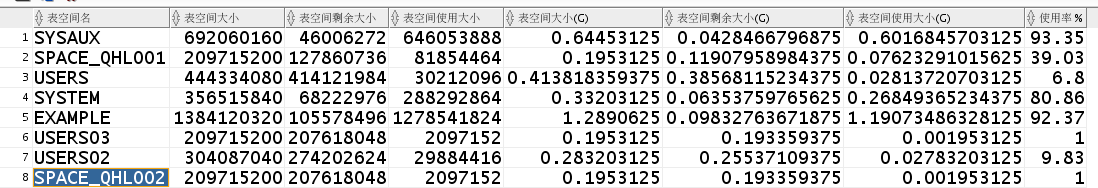
GROUP BY tablespace\_name) a,

(SELECT tablespace\_name, SUM(bytes) total

FROM dba\_data\_files

GROUP BY tablespace\_name) b

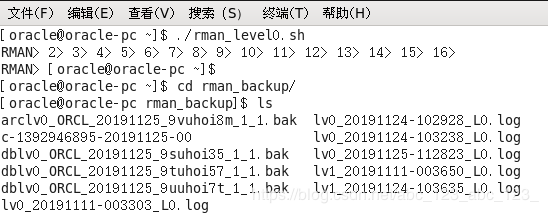
WHERE a.tablespace\_name = b.tablespace\_name



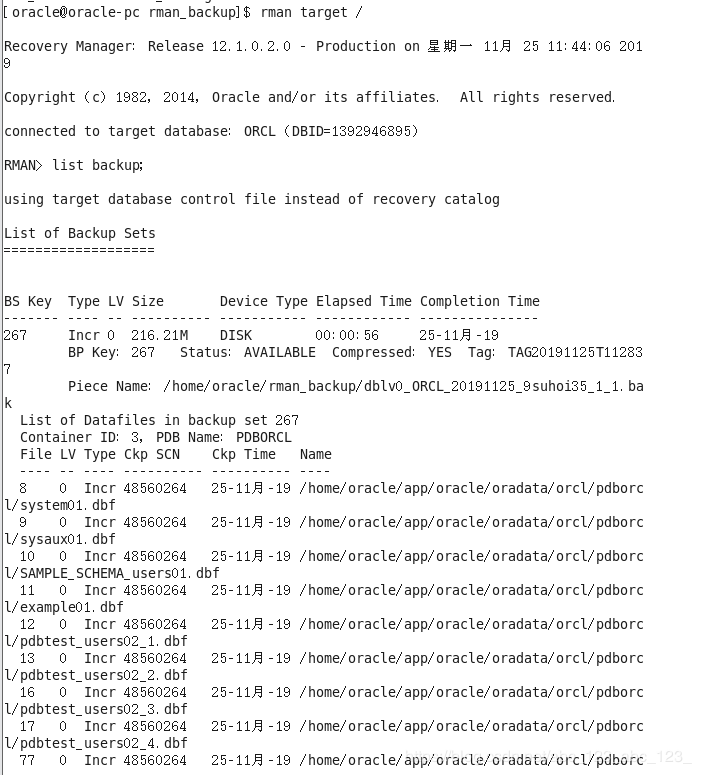
**3.6 备份方案**

**3.6.1备份**

./rman\_level0.sh



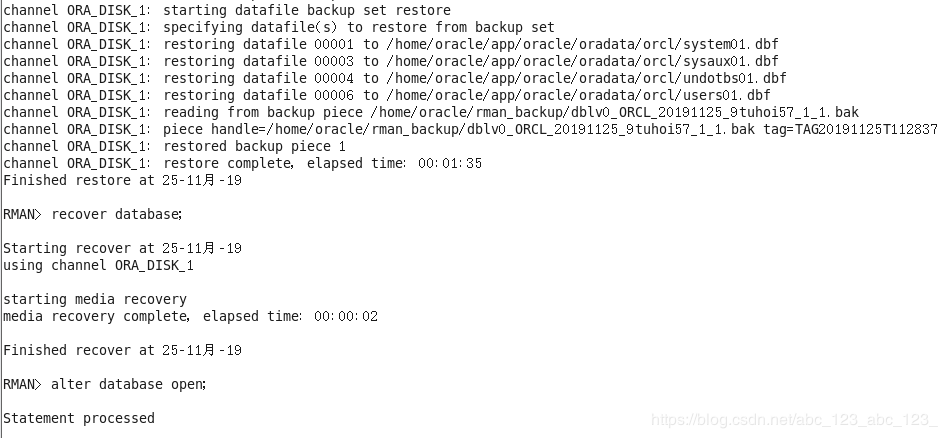
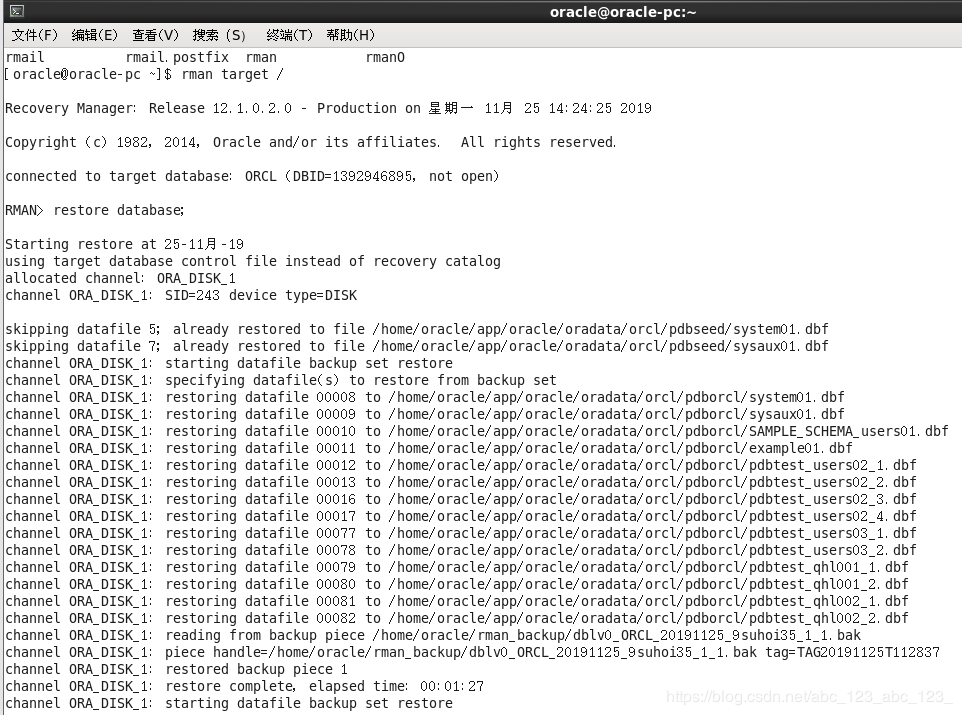
**3.6.2查看备份内容**



**3.6.3删除数据**



**3.6.4恢复备份**



**3.6.5数据已恢复**

