

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的书籍交易系统的数据库设计 | | |
| 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 | 李仟禧 | 学号 | 201810414415 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

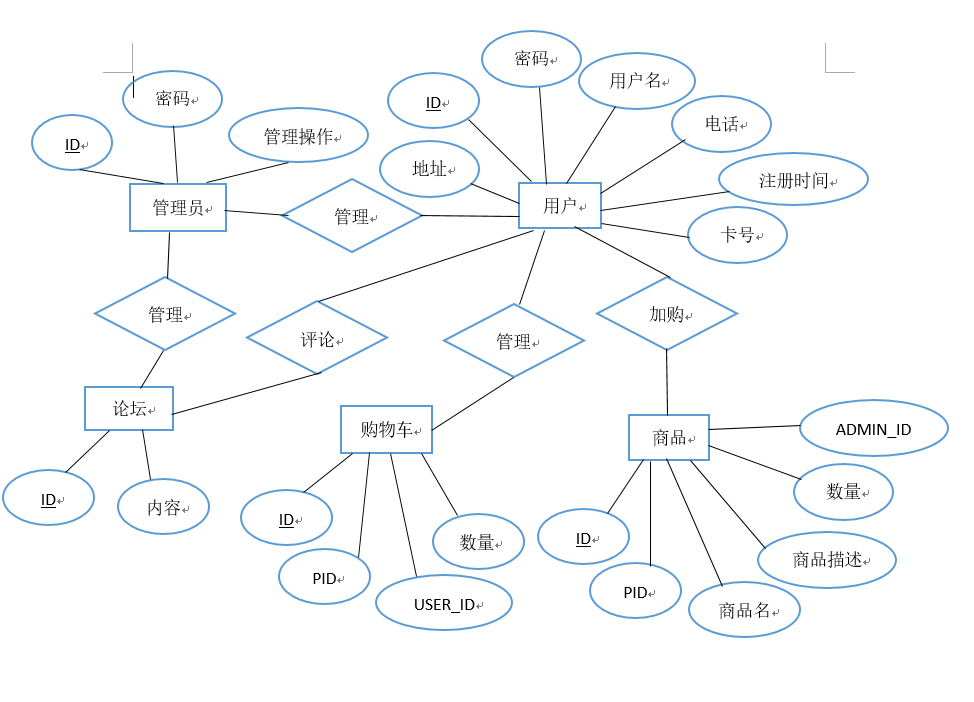
|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 10 |  |
| 表设计 | 表，表空间设计合理，数据合理 | 20 |  |
| 用户管理 | 权限及用户分配方案设计正确 | 20 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 30 |  |
| 备份方案 | 备份方案设计正确 | 20 |  |
| **得分合计** | | |  |

1. 年 6 月 12 日

### 一、书籍交易系统概述

随着Internet的发展和电子商务的日益普及，网络购物可以使顾客足不出户便可以方便快捷地选购自己喜欢的商品，比如图书等。因此，需要网上销售系统为网上交易活动提供一个高效、低成本的平台。该书籍交易系统能够满足用户进行网上交易活动的各项要求，能够使用户足不出户便可以方便快捷地选购自己喜欢的商品。

### 二、概念模型设计（E-R图）



### 三、数据库表

#### 3.1 管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为主键 | 约束 | 备注 |
| ID | int |  | 是 | NOT NULL | 自增 |
| password | varchar | 20 | 否 | NOT NULL | 管理员登录密码 |
| admin | varchar | 20 | 否 | NOT NULL |  |

#### 3.2用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为主键 | 约束 | 备注 |
| ID | int |  | 是 | NOT NULL | 自增 |
| password | varchar | 20 | 否 | NOT NULL | 用户登录密码 |
| username | varchar | 50 | 否 | NOT NULL | 用户名 |
| phone | varchar | 20 | 否 | NOT NULL | 用户手机号码 |
| address | varchar | 30 | 否 | NOT NULL | 用户收货地址 |
| RegistrationDate | Date |  | 否 | NOT NULL |  |
| Cart\_ID | varchar |  | 否 | NOT NULL | 自增 |

#### 3.3商品表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为主键 | 约束 | 备注 |
| ID | int |  | 是 | NOT NULL | 自增 |
| PID | int |  | 否 | NOT NULL | 自增，商品唯一标识 |
| USER\_ID | varchar | 20 | 否 | NOT NULL | 用户id |
| price | int |  | 否 | NOT NULL | 商品价格 |
| Describe | varchar | 50 | 否 | NOT NULL | 商品描述 |
| Num | Date |  | 否 | NOT NULL |  |
| Admin\_ID | int |  | 否 | NOT NULL |  |

#### 3.4购物车表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为主键 | 约束 | 备注 |
| ID | int |  | 是 | NOT NULL | 自增 |
| Amount | int |  | 否 | NOT NULL | 数量 |
| PID | int |  | 否 | NOT NULL |  |
| User\_ID | int |  | 否 | NOT NULL |  |

#### 3.5论坛表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为主键 | 约束 | 备注 |
| ID | int |  | 是 | NOT NULL | 自增 |
| Content | varchar | 50 | 否 | NOT NULL | 论坛发布内容 |

### 四、创建表空间

#### 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;

#### space\_qhl002

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;



### 五、 创建角色及用户

用户默认使用表空间 space\_qhl001

#### 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

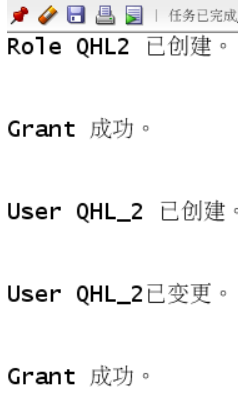
CREATE ROLE qhl1;  
GRANT connect,resource,CREATE VIEW TO qhl1;  
CREATE USER qhl\_1 IDENTIFIED BY 123 DEFAULT TABLESPACE space\_qhl001 TEMPORARY TABLESPACE temp;  
ALTER USER qhl\_1 QUOTA 60M ON space\_qhl001;  
GRANT qhl1 TO qhl\_1;



#### 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;



### 六、在用户 qhl\_1 下创建表

#### 创建管理员表

* id 为主键

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;

#### 创建用户表

* id 为主键
* 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  
 )  
);

#### 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;

#### 创建购物车表

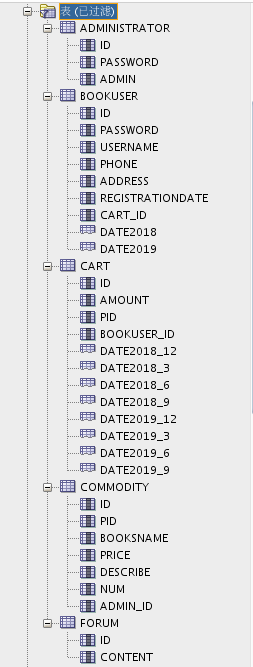
* 用户表字段 BOOKUSER\_ID 为购物车表的外键
* 购物车采用引用分区

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  
);

#### 创建论坛表

CREATE TABLE TABLE1  
(  
 ID INT NOT NULL  
, CONTENT NVARCHAR2(50) NOT NULL  
, CONSTRAINT TABLE1\_PK PRIMARY KEY  
 (  
 ID  
 )  
 ENABLE  
);

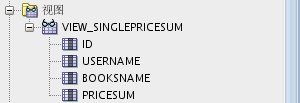
#### 完整表结构如下图所示



#### 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;



#### 用户 qhl\_1 空间不足，修改 qhl\_1 空间大小

ALTER USER qhl\_1 QUOTA 90M ON space\_qhl001;

### 七、插入用户、商品、购物车数据

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..50000  
 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;

### 八、创建程序包、存储过程、函数执行分析计划

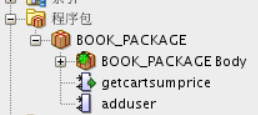
#### 创建程序包

* 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;

#### 创建函数、存储过程

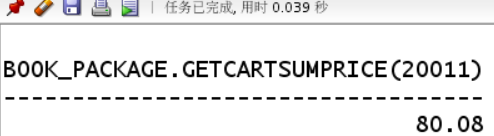
create or replace PACKAGE body book\_package Is  
  
 function getcartsumprice(user\_id number) return number as  
 begin  
 declare cart\_sum number;  
 query\_sql varchar2(200);  
 begin  
 query\_sql:='select sum(pricesum) from view\_SinglePriceSum where ID=' || user\_id;  
 execute immediate query\_sql 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;



#### 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','wwj','123456','hongkong','2021-06-12');  
end;

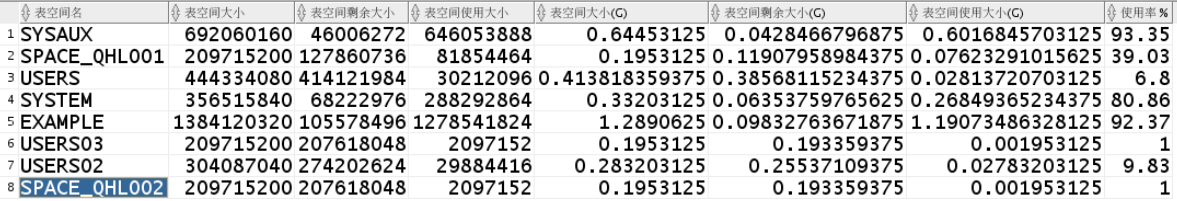
pict8

#### 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');

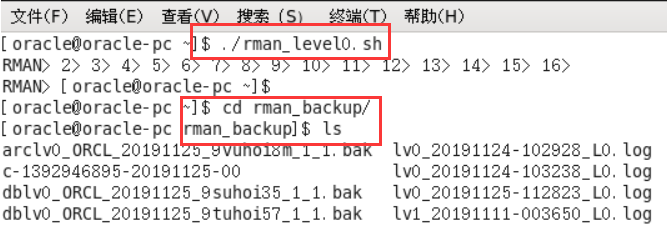
#### 表空间使用状况

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

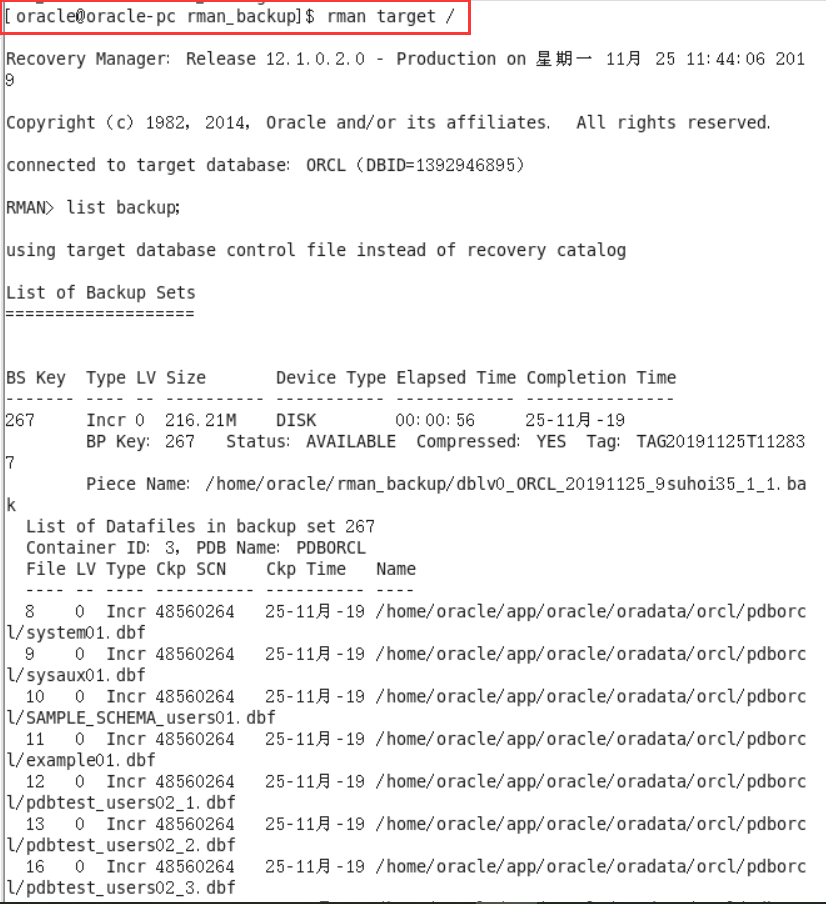
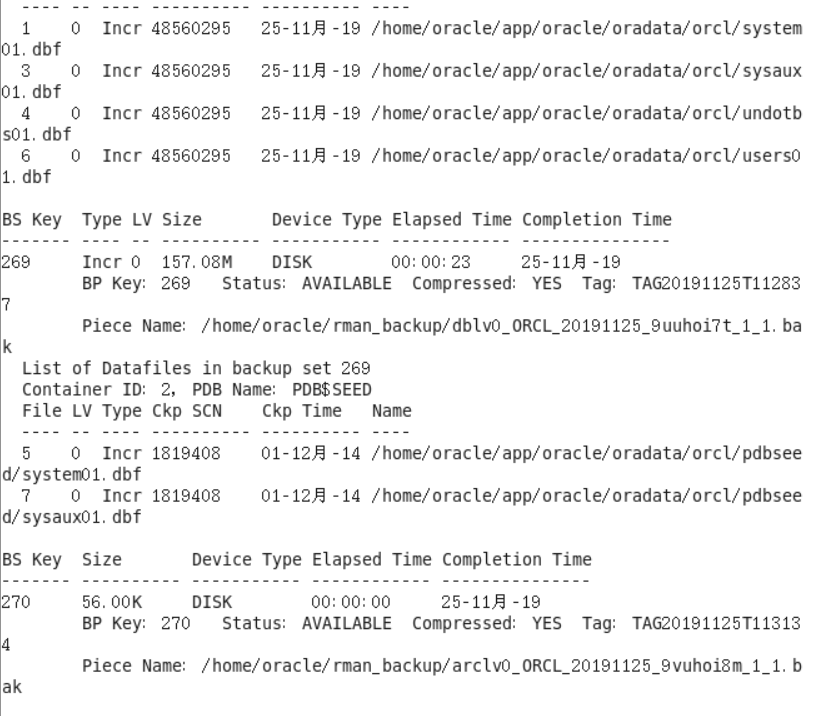


### 九、备份恢复

#### 备份./rman\_level0.sh



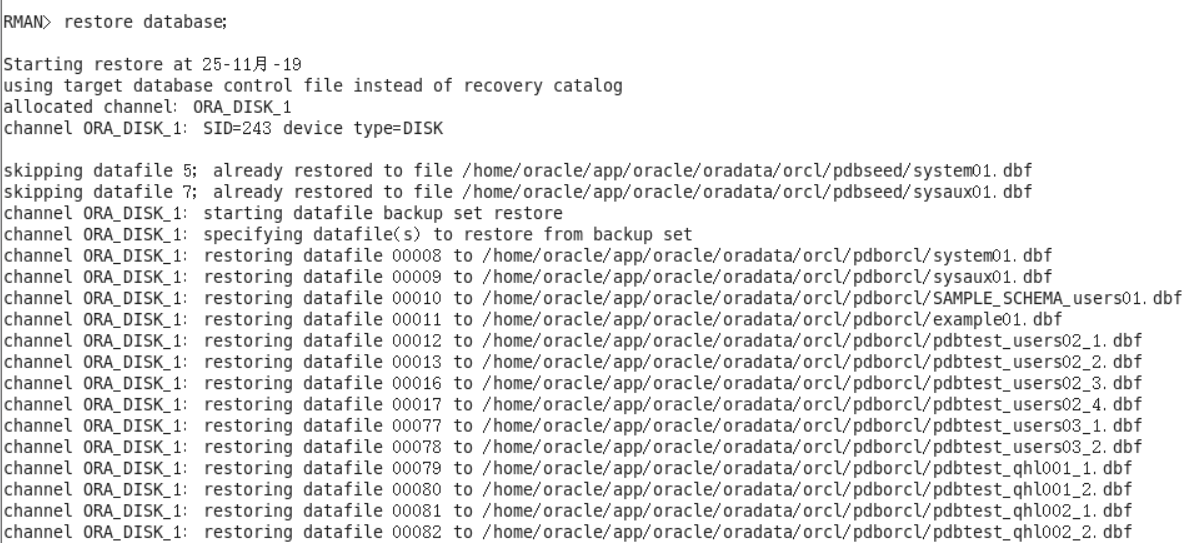
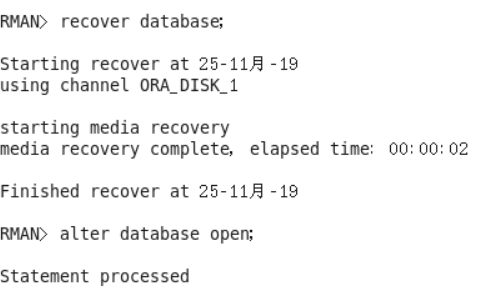
#### 查看备份内容

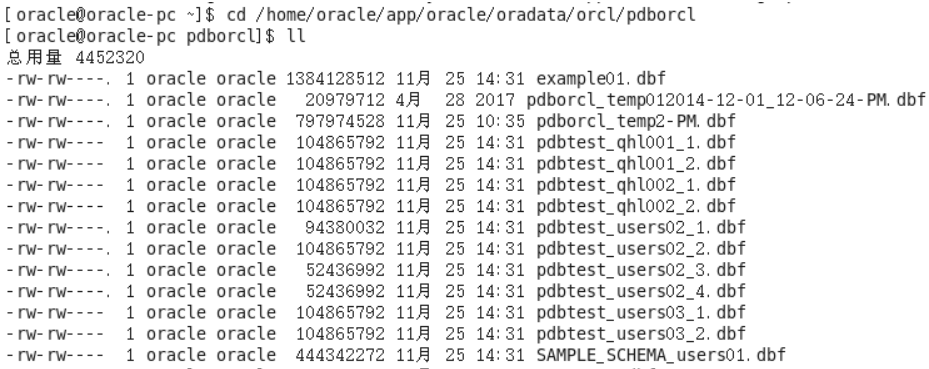
#### 删除数据

pict15

#### 恢复备份

#### 恢复完成



### 十、体会

在完成此书籍交易系统的课程设计中，我花时间在网上查阅了大量资料，一步一步的实现书籍交易系统的数据库设计，我对数据库的各部分的设计有了更全面的认识，对需求分析的重要性和需求分析的任务和过程有了更深的认识，对流程图的描绘应与实际相联系有了进一步的了解，对概念设计的步骤掌握的更加清晰。大体上知道一个系统应该从哪些方面进行分析和设计。进行建表时属性应与实际情况相联系等。页面与数据库后台连接也有了比较清晰的理解。了解到了一个系统大概的步骤。