

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的二手书交易的数据库设计 | | |
| 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 | 刘家银 | 学号 | 201810414313 |
| 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

基于Oracle的二手书交易系统数据库设计

# 1背景

为了满足现如今二手书交易市场的需求，各种书籍交易系统层出不穷。但又因为数据库系统的高效性而诟病，故此，我们选择基于Oracle数据库来设计这样一个系统来满足我们行业现状。

# 2设计过程

## 2.1.创建表空间space\_qhl001、space\_qhl002

--创建表空间space\_qhl001

Create Tablespace space\_qhl001

datafile

'/u01/app/oracle/oradata/orcl/pdbtest\_qhl001\_1.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,

'/u01/app/oracle/oradata/orcl/pdbtest\_qhl001\_2.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED

EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;

--创建表空间space\_qhl002

Create Tablespace space\_qhl002

datafile

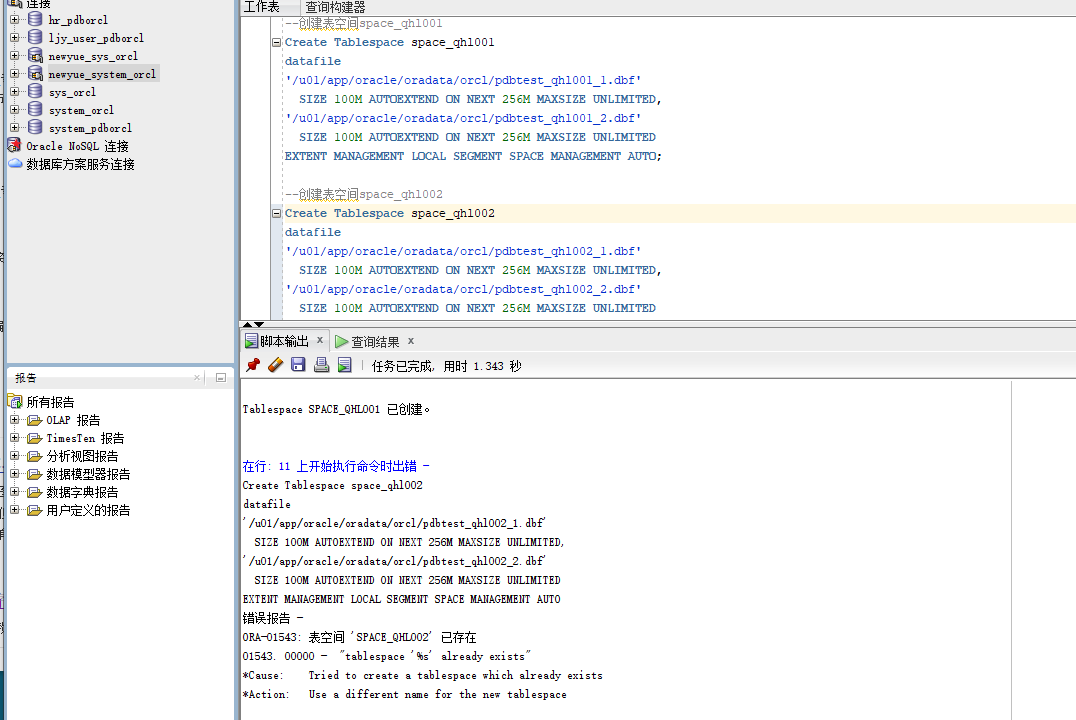
'/u01/app/oracle/oradata/orcl/pdbtest\_qhl002\_1.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,

'/u01/app/oracle/oradata/orcl/pdbtest\_qhl002\_2.dbf'

SIZE 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED

EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;



## 2.2创建用户以及角色

1. 创建角色qhl1将connect,resource,create view授权给qhl1
2. 创建用户qhl\_1
3. 分配50M空间给qhl\_1并将角色qhl1授权给用户qhl\_1

代码：

CREATE ROLE qhl1;

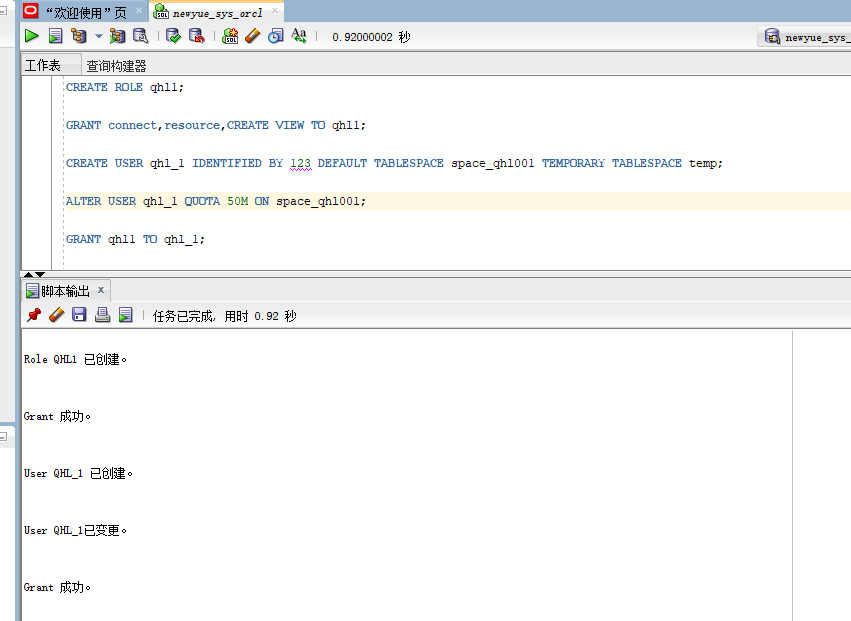
GRANT connect,resource,CREATE VIEW TO qhl1;

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

ALTER USER qhl\_1 QUOTA 50M ON space\_qhl001;

GRANT qhl1 TO qhl\_1;

截图



1. 创建角色qhl2将connect,resource授权给qhl2
2. 创建用户qhl\_2
3. 分配50M空间给qhl\_2并将角色qhl2授权给用户qhl\_2

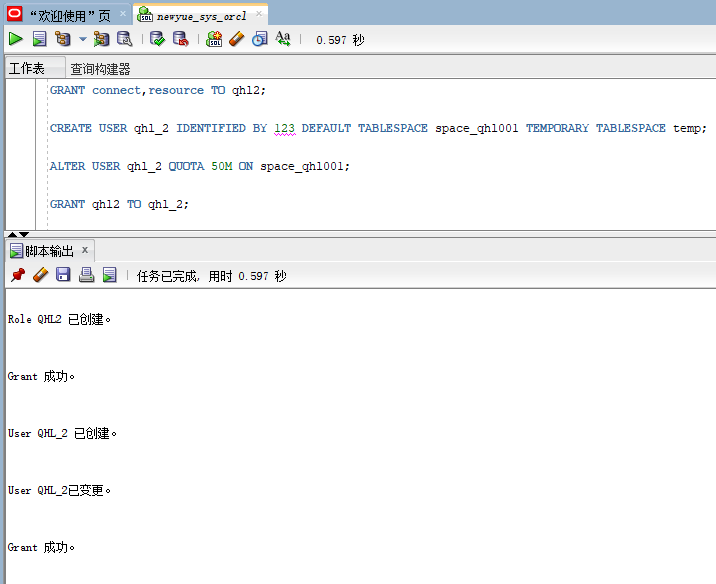
代码：

GRANT connect,resource TO qhl2;

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

ALTER USER qhl\_2 QUOTA 50M ON space\_qhl001;

GRANT qhl2 TO qhl\_2;



## 2.3在用户qhl\_1下创建表

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

(

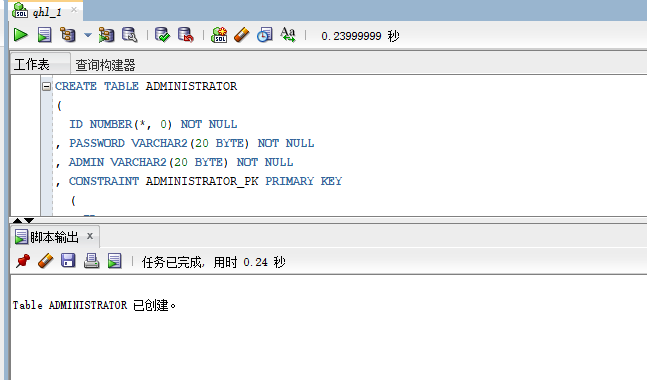
BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NO INMEMORY

NOPARALLEL;



1. 创建用户表

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'))

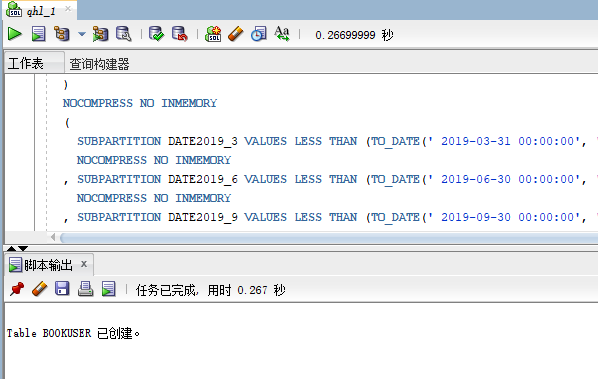
NOCOMPRESS NO INMEMORY

, 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

)

);



1. 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

(

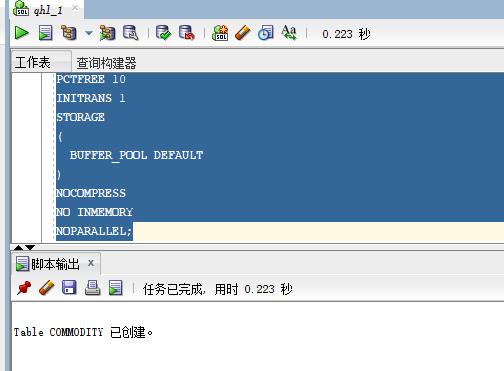
BUFFER\_POOL DEFAULT

)

NOCOMPRESS

NO INMEMORY

NOPARALLEL;



1. 创建购物车表
2. 用户表字段BOOKUSER\_ID为购物车表的外键
3. 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

);

1. 创建论坛表

CREATE TABLE TABLE1

(

ID INT NOT NULL

, CONTENT NVARCHAR2(50) NOT NULL

, CONSTRAINT TABLE1\_PK PRIMARY KEY

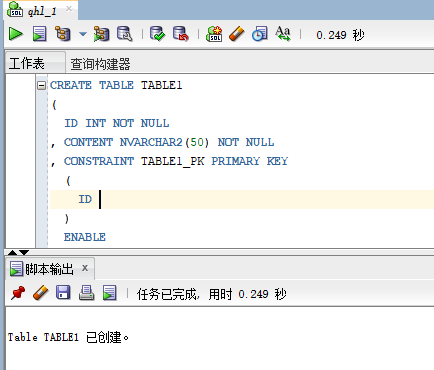
(

ID

)

ENABLE

);



## 2.4插入用户、商品、购物车数据

代码如下：

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;

## 2.5创建程序包、存储过程、函数执行分析计划

### 2.5.1创建程序包

1. 函数getcartsumprice计算每个用户的购物车商品总金额
2. 存储过程adduser插入用户信息

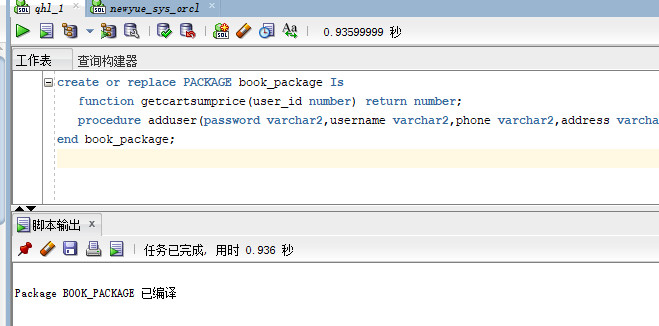
**代码及结果图**

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;



### 2.5.2创建存储过程

**代码及截图**

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;

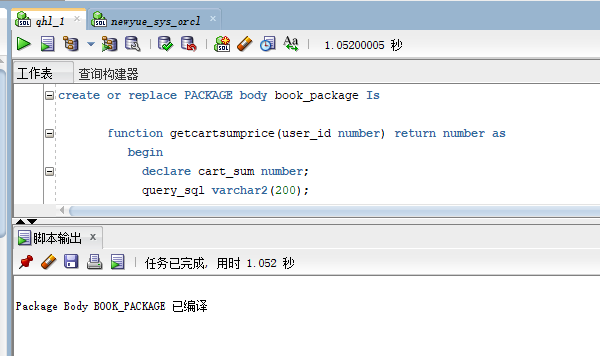
insert into bookuser values(maxId+1,password,username,phone,address,to\_date(registerdate,'yyyy-mm-dd'),maxId+1);

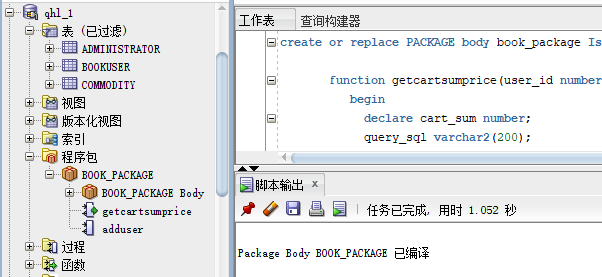
commit;

end;

end adduser;

end book\_package;





### 2.5.3使用存储过程adduser插入用户数据

**代码如下：**

set serveroutput on

declare

begin

BOOK\_PACKAGE.addUser('1223','cdu','125666','hongkong','2019-05-02');

end;

### 2.5.4执行计划分析

**代码如下：**

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');

### 2.5.5表空间使用情况

**代码如下：**

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

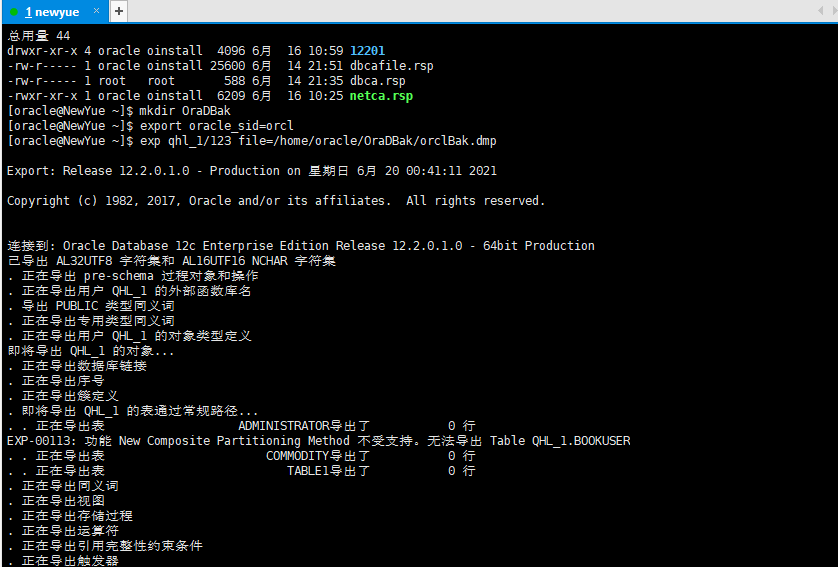
FROM dba\_data\_files

GROUP BY tablespace\_name) b

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

## 2.6手动备份方案

**过程图如下：**



# 3总结

此次项目设计基本涵盖Oracle数据库的常用技术，尽管并没有做到每一步都很完美，尤其是最后的容灾设计方案由于条件和时间等因素影响并没有去实现，但其它的设计过程对自己也是一个挑战。完成此次的课程设计对于个人学习Oracle数据库有着很大的帮助，加深了我对Oracle数据库的理解和认识。