



成都大学
CHENGDU UNIVERSITY

期末项目设计报告

题 目	基于 Oracle 的甜点商城的数据库设计		
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	2017 级
学生姓名	徐元通	学 号	201710414423
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	10	
PL/SQL 设计	存储过程和函数设计正确	25	
备份方案	备份方案设计正确	25	
容灾方案	DataGuard 设计正确	10	
得分合计			

2019 年 11 月 18 日

目录

微信小程序甜点商城.....	3
1. 甜点商城 ER 模型.....	3
1.1 实体模型.....	3
1.2 实体联系模型.....	5
整个甜点微信小程序的 ER 图, 如图 1-10.....	6
2. 甜点商城数据表的设计.....	6
3. 创建数据库.....	8
3.1 配置插接式数据库 PDB.....	8
3.2 通过 Oracle SQL Developer 连接数据库.....	9
4. 用户创建与空间分配.....	9
5. 创建表, 约束和索引.....	11
6. 创建触发器、序列和视图.....	15
7. 创建程序包、函数和过程.....	16
8. 数据库备份.....	21
8.1 开始全备份.....	21
8.2 备份后修改数据.....	28
8.3 删除数据库文件, 模拟数据库文件损坏.....	28
8.4 数据库完全恢复.....	30
8.4.1 重启损坏的数据库到 mount 状态.....	30
8.4.2 开始恢复数据库.....	31
8.5 查询数据是否恢复.....	31
9. 数据库 DataGuard 设计.....	32
9.1 oracle 12C dataguard 搭建环境 (主从库相同, 从库只装软件不建库)	32
9.2 dataguard 主库参数配置.....	33
9.3. 配置两个数据库的网络参数文件 tnsname.ora 和 listener.ora.....	35
9.3.1 配置 tnsnames.ora 参数文件.....	35
9.3.2 配置监听文件.....	36
9.4. 配置数据库参数文件.....	37
9.5.启动 standby database 到 nomount 状态.....	38
9.5.1 检查 tns 连接名是否正常(主备库均需要检查).....	38
9.5.2 启动备库.....	38
参考文献.....	39

微信小程序甜点商城

1. 甜点商城 ER 模型

本系统基于微信小程序的甜点商城系统，主要针对于大学生线上购买的小程序，服务于特定的某一个商家。如果商家在美团或则是饿了么平台上来卖东西的话，则会收取昂贵的平台费，本来大学生的消费就不是很高，再加上平台费，可能在外卖上点甜点就不是很现实。如果有了微信小程序的甜点线上商城，少了平台费，就可以在价格上面带来很大的优势。

1.1 实体模型

根据应用场景分析，共有 5 个原始的实体(Entity)，它们是管理员、用户、会员、订单和甜点。

管理员(admin)包括管理员 id(admin_id)、联系电话(tel)、管理员名(admin_username)和管理员密码(password)，如图 1-1。

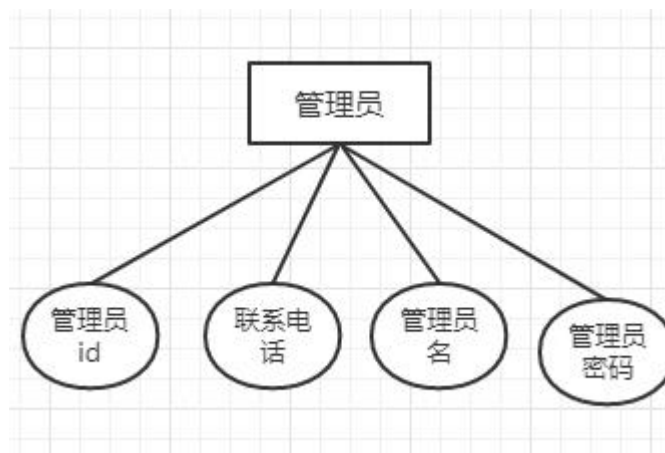


图 1-1 管理员 (admin) 实体

甜点 (dessert) 包括甜点 id (dessert_id)、甜点图片 (image)、甜点名称 (name)、甜点简介 (Introduction)、甜点价格 (price)。管理员可对甜点进行增删查改，主要进行商品的价钱的更新。甜点的实体，如图 1-2。

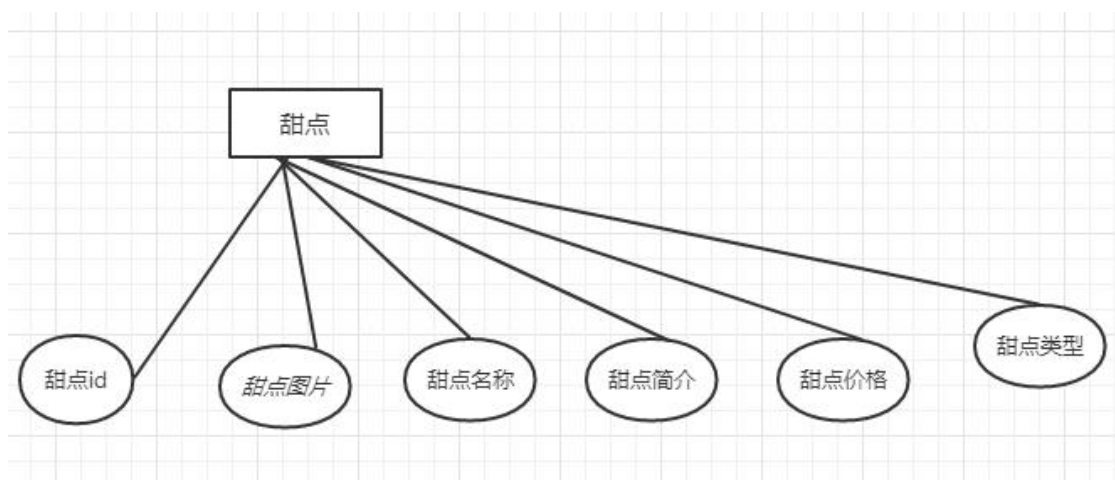


图 1-2 甜点 (dessert) 实体

用户（user）包括用户 id（user_id）、用户名（username）、联系电话（tel）、用户地址（address）。在每次用户下单后，订单上面需要根据用户 id 或是用户名对该订单的详细记录写进数据库。用户实体，如图 1-3。

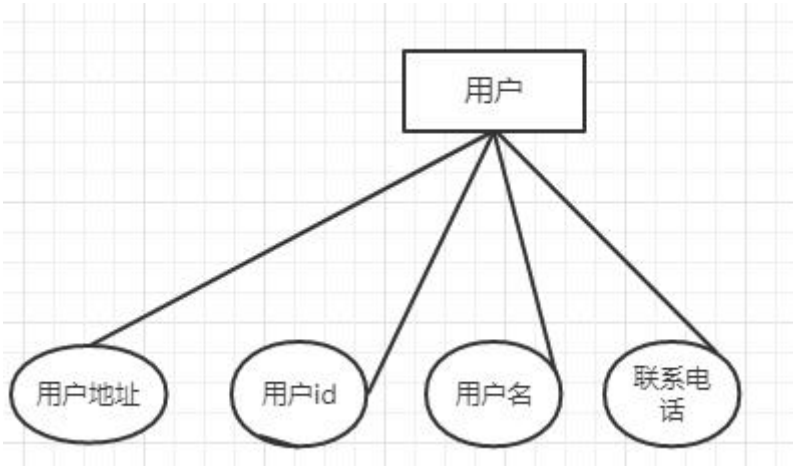


图 1-3 用户（user）实体

会员（member）包括会员 id（member_id）、用户 id（user_id）、会员购买时间（member_start）、会员截止时间（member_end）、购买时长（member_time）。可以根据会员的结束时间来提醒用户是否续费，根据购买时长来计算会员的截止时间。会员实体，如图 1-4。

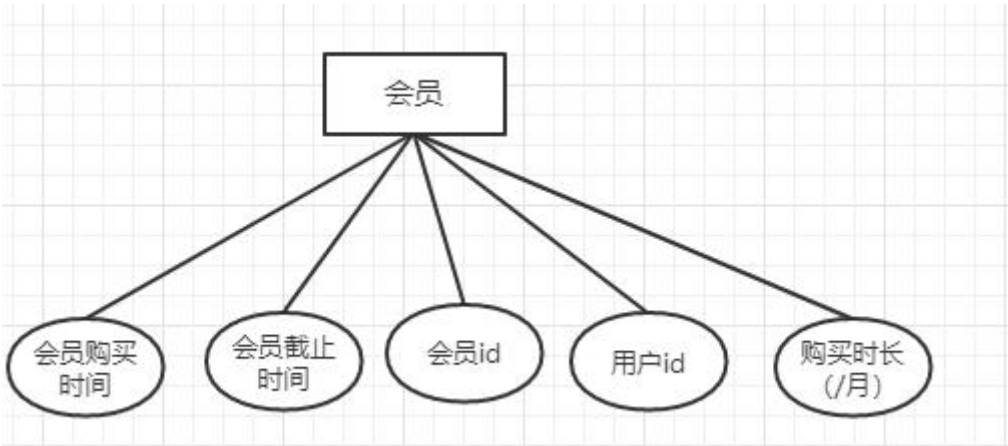


图 1-4 会员（member）实体

订单（order）包括订单编号（order_id）、用户 id（user_id）、用户名（username）、用户地址（address）、下单时间（order_time）、订单总金额（total_price）。订单实体，如图 1-5。

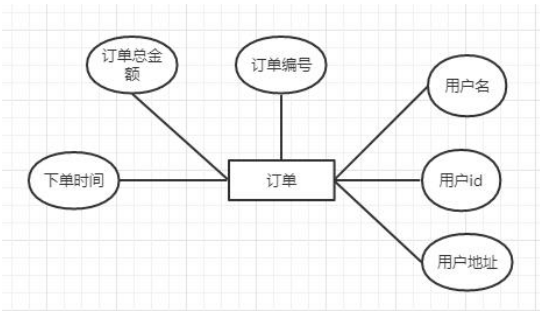


图 1-5 订单（order）实体

1.2 实体联系模型

管理员和甜点的关系是一对多的关系，在这个商城系统，只有一个管理员账号，每个员工可以登录这个管理员账号对商品的甜点信息进行增删查改。

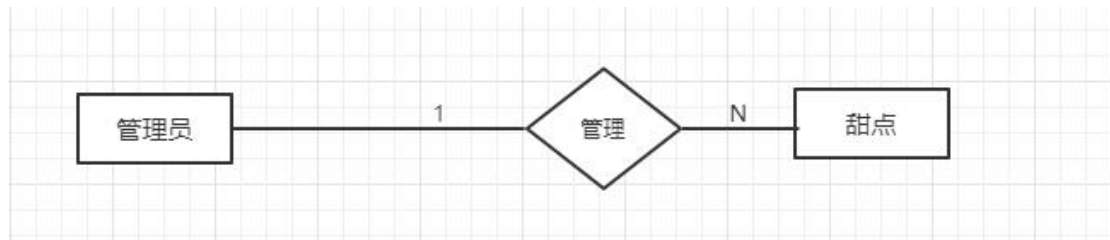


图 1-6 管理员与甜点的关系简图

用户和甜点的关系是多对多，一个用户可以购买多个甜品，同一种甜品可以被多个用户购买。

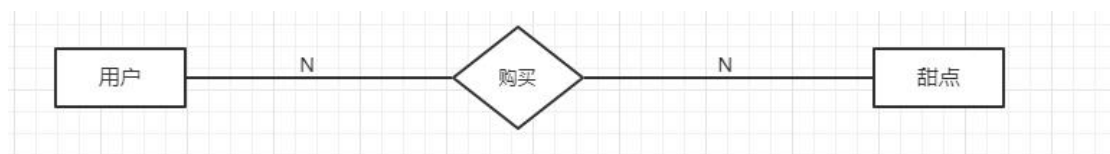


图 1-7 用户和甜点的关系简图

用户和会员的关系是多对一，一个用户只能购买一个会员。

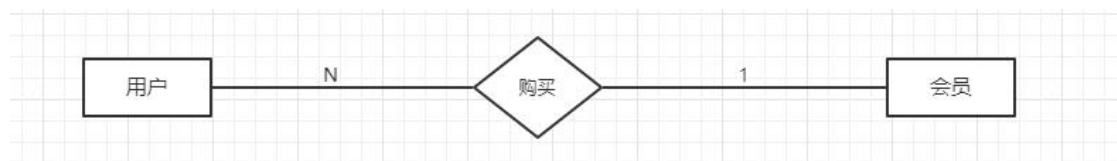


图 1-8 用户和会员的关系简图

用户和订单的关系是多对多，每个人的下单没有限制，一个人可以下多个订单。因为一个用户的下单地址可能不一样，所以在这里每一个订单都必须绑定自己的订单地址。

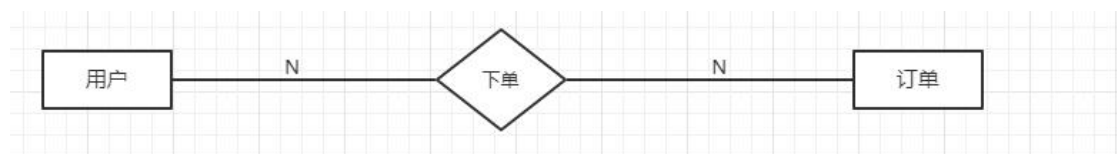


图 1-9 用户和订单的关系简图

整个甜点微信小程序的 ER 图，如图 1-10.

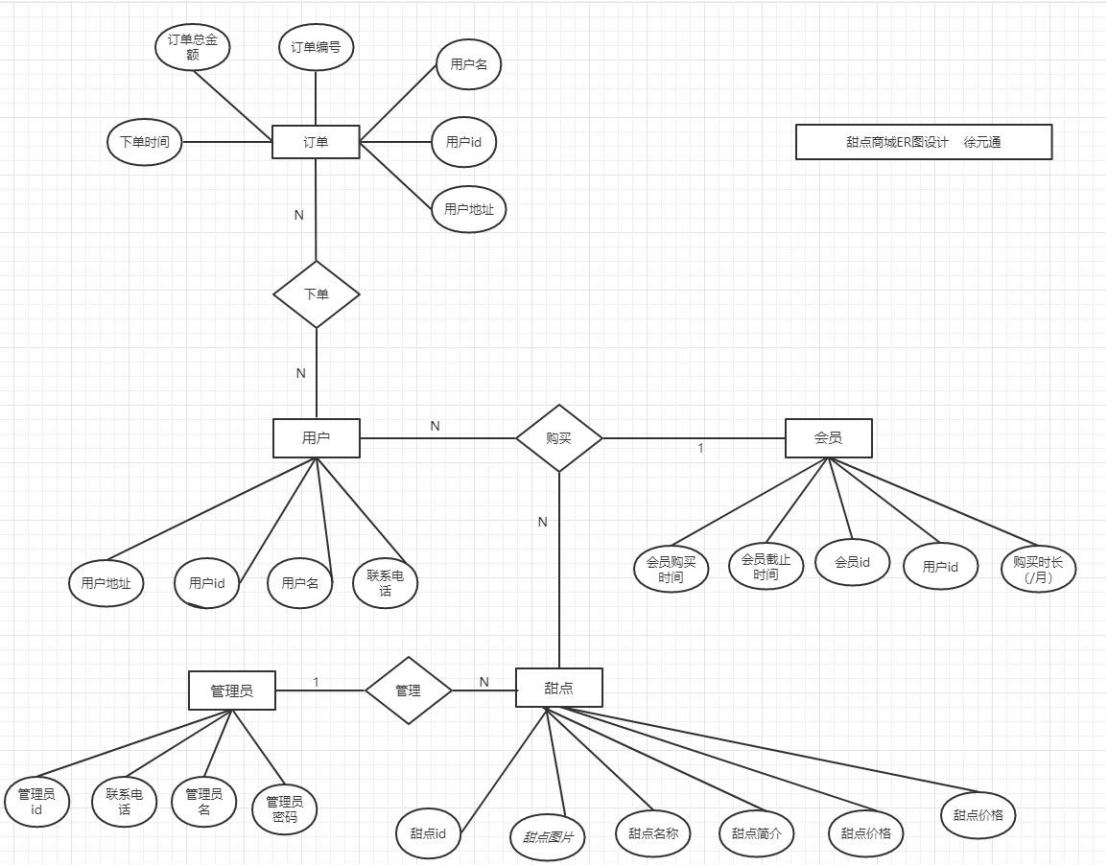


图 1-10 甜点商城 ER 图

2. 甜点商城数据表的设计

ER 模型建立好以后，我们就可以设计 Oracle 的关系表了。在独立实体中找出主要属性设置为主键，比如在订单表中，订单编号（order_id）是主键。由关系派生出的实体要加入外键关系，如订单中要加入外键用户 id（user_id）属性。

管理员(admin)表包括管理员 id(admin_id)、联系电话(tel)、管理员名(admin_name)、管理员密码(password)，见表 2-1。

表 2-1 甜点商城管理员 ADMIN

字段名	数据类型	可以为空	注释	备注
admin_id	varchar	no	管理员 ID	主键
tel	varchar	no	管理员联系方式	
admin_name	varchar	no	管理员名	
password	varchar	no	管理员密码	

用户表 USERS 包括用户 ID (user_id)、用户的联系方式 (tel)、用户名 (username)、用户的联系地址 (address)，其中用户的联系地址可以有多个地址，用逗号 (，) 隔开，见表 2-2。

表 2-2 甜点商城用户 USERNAME

字段名	数据类型	可以为空	注释	备注
user_id	varchar	no	用户 ID	主键
tel	varchar	no	用户联系方式	
username	varchar	no	用户名	
address	varchar	yes	用户地址	可以有多个地址

甜点表 DESERT 包括甜点 ID (desert_id)、甜点图片 (desert_image)、甜点名称 (desert_name)、甜点简介 (introduce)、甜点价格 (price)，见表 2-3。

表 2-3 甜点商城甜点 DESERT

字段名	数据类型	可以为空	注释	备注
desert_id	varchar	no	甜点 ID	主键
desert_image	varchar	no	甜点图片	
desert_name	varchar	no	甜点名称	
introduce	varchar	yes	甜点简介	可以为空
price	float	no	甜点价格	
type	varchar	no	甜点的类型	种类

会员表 MEMBER 包括会员 ID (member_id)、会员购买时间 (member_start)、会员截止时间 (member_end)、购买时长 (member_time)、用户 ID (user_id)，见表 1-4。

表 2-4 甜点商城会员 MEMBER

字段名	数据类型	可以为空	注释	备注
member_id	varchar	no	会员 ID	主键
member_start	date	no	会员开始时间	
member_end	date	no	会员结束时间	
member_time	varchar	no	会员总时间	
user_id	varchar	no	用户 id	外键

订单 (order) 包括订单编号 (order_id)、用户 id (user_id)、用户名 (username)、用户地址 (address)、下单时间 (order_time)、订单总金额 (total_price)。订单实体，如表 2-5。

表 2-5 甜点商城订单 ORDER

字段名	数据类型	可以为空	注释	备注
order_id	varchar	no	订单 ID	主键
user_id	varchar	no	用户 ID	外键
username	varchar	no	用户名	
address	varchar	no	订单地址	
order_time	date	no	下单时间	
total_price	float	no	订单总金额	

3. 创建数据库

3.1 配置插接式数据库 PDB

配置插接式数据库PDB可以通过SQL语句创建插接式数据库，在创建时可以选择创建一个新的空的PDB，不包含任何数据；也可以选择一个现有的PDB克隆成另一个PDB，所谓克隆，是指将原有的PDB的用户信息和数据信息一起复制到新的PDB中，这也是Oracle 12C的重要特性。

在这儿，我们通过dbca 工具创建插接式数据库，首先以Oraclev 用户登录，运行 dbca 命令，然后根据提示操作，要注意输入正确的数据库目录、PDB数据库名称以管理员信息。

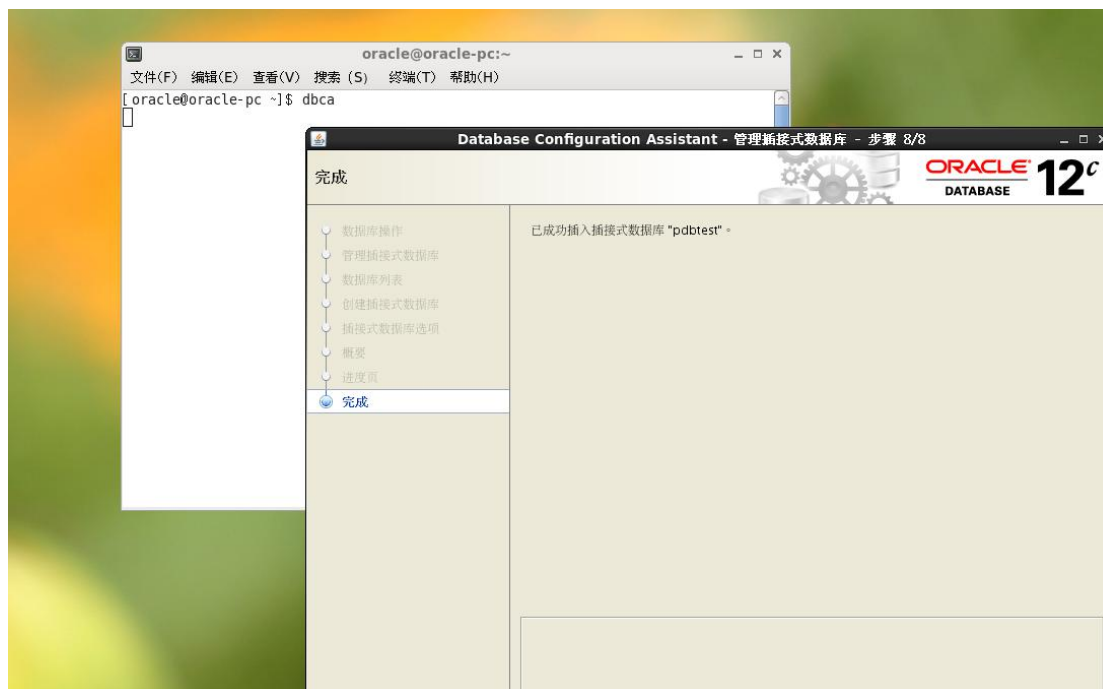


图 3-1 甜点商城通过 dbca 创建插接式数据库

3.2 通过 Oracle SQL Developer 连接数据库

Oracle SQL Developer 是一个免费的 GUI 图形界面的管理和开发工具，可以提高工作效率并简化数据库开发任务。SQL Developer 可以在没有安装数据库的客户端上运行，支持 Windows、Linux、Mac 系统。可以在 Oracle 官网上下载，直接使用，无需安装。Oracle SQL Developer 是基于 Java 的应用程序，如果客户端没有安装 Java，就需要下载自带 Java 的 SQL Developer，如果客户端已经安装了 Java，就可以下载不带 Java 版本。

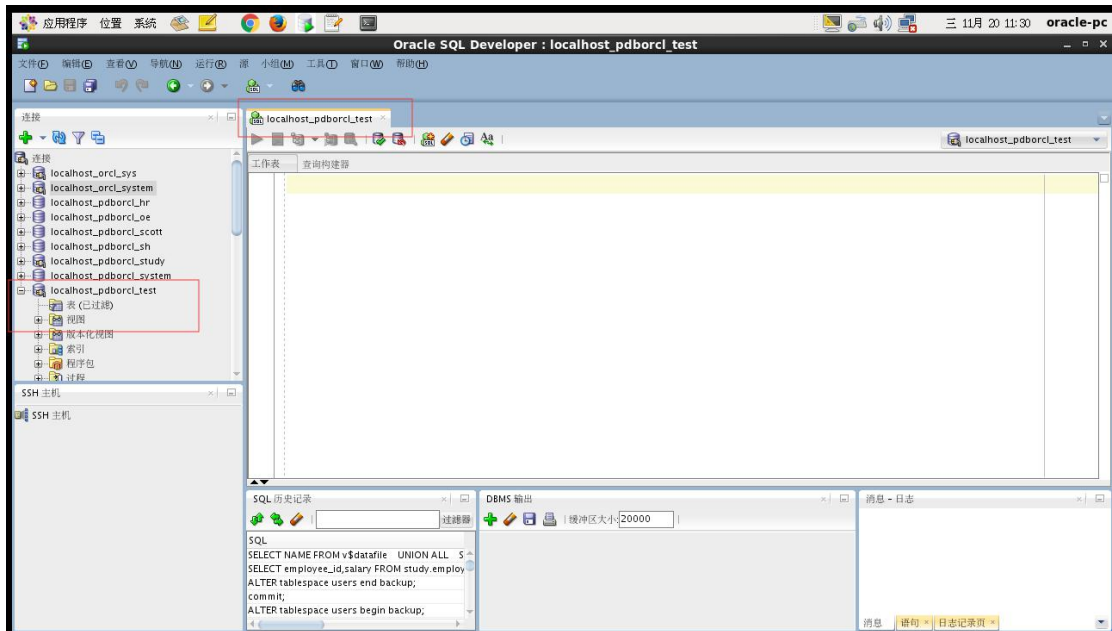


图 3-2 甜点商城通过 SQL Developer 连接 Oracle

4. 用户创建与空间分配

表的结构设计之后，还需要考虑用户和空间的分配问题。我们需要为系统新建一个用户（NICETONG）。另外还需要为甜点商城系统创建一个新表空间NICETONG用户存储订单记录。

下面是SYSTEM用户创建的一个表空间nicetong的命令，注意给 nicetong表空间分配了两个数据文件：pdbtest_nicetong1.dbf和pdbtest_nicetong2.dbf，这两个数据文件初始大小都是100M，所以表空间的初始大小是200M。

```
create tablespace nicetong
datafile
'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong1.dbf'
size 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED,
'/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong2.dbf'
size 100M AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

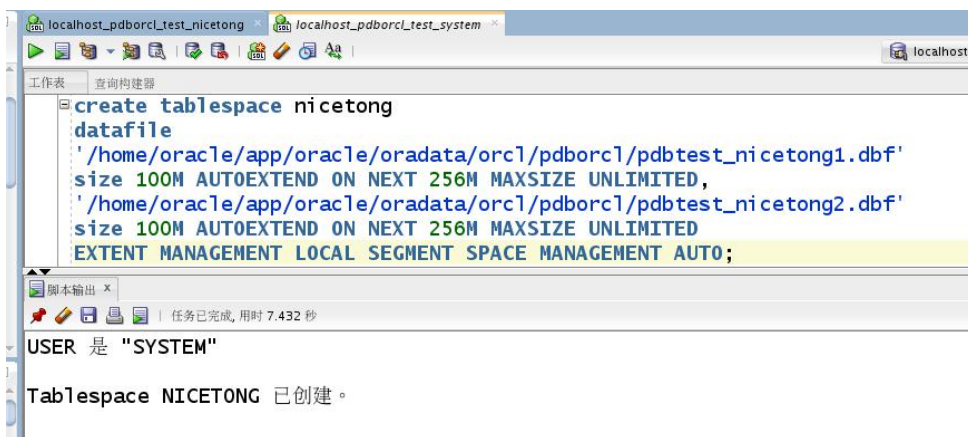


图 4-1 甜点商城创建表空间 nicetong

表空间nicetong创建完成之后，就可以创建用户了。我们可以为本系统设计的用户名称是nicetong。用户创建之后，给用户nicetong分配表空间nicetong的使用配额，再分配角色CONNECT 和RESOURCE，以便该用户可以连接数据库，可以创建资源（表，过程，序列等资源对象），最后再分配一个系统权限：“CREATE VIEW”，以便该用户可以创建视图。

```

//给用户 nicetong
alter user nicetong DEFAULT TABLESPACE "USERS" TEMPORARY TABLESPACE "TEMP";
-- QUOTAS
ALTER USER nicetong QUOTA UNLIMITED ON USERS;
ALTER USER nicetong QUOTA UNLIMITED ON nicetong;
-- ROLES
GRANT "CONNECT" TO nicetong WITH ADMIN OPTION;
GRANT "RESOURCE" TO nicetong WITH ADMIN OPTION;
ALTER USER nicetong DEFAULT ROLE "CONNECT","RESOURCE";
-- SYSTEM PRIVILEGES
GRANT CREATE VIEW TO nicetong WITH ADMIN OPTION;

```

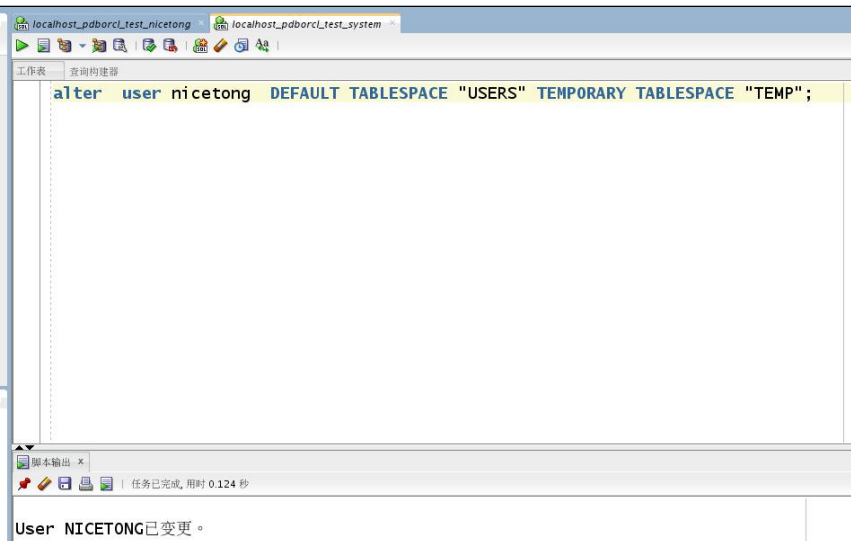


图 4-2 甜点商城创建用户以及分配表空间

到现在，我们有了两个表空间 USERS（原有的）和NICETONG（新建的）。这里我们假定店主的销售订单非常多，订单表的记录非常大，可能上千万条记录，所以我们将ORDERS表设计为基于 REDER_DATA的分区表，以加快局部时间范围的查询速度。存储空间的分如表 4-1。

表 4-1 甜点商城存储空间分配

表	表空间USERS	表空间NICETONG
ORDERS	存储全部数据	
USERS	存储全部数据	
ADMIN	存储全部数据	
MEMBER	存储2019年前（不含）的数据	存储 2019年前（不含）的数据
DESERT	存储全部数据	

5. 创建表，约束和索引

用户和空间分配完成后，可以创建表，约束和索引了。创建表的命令是 CREATE TABLE，由于命令参数和选项很复杂，命令行数非常多，所以这里就不将创建表的 DDL语句全部写出来。这里仅对 ORDERS 和 DESERT两个表的创建作一些说明，ORDERS 表按分区存储，分区类型选择“RANG”范围分区，创建 ORDERS 表的部分语句是：

```
create table ORDERS
(
order_id VARCHAR2(10) PRIMARY KEY NOT NULL,
user_id VARCHAR2(10) NOT NULL,
username VARCHAR2(9) NOT NULL,
address VARCHAR2(10) NOT NULL,
order_time DATE NOT NULL,
total_price FLOAT NOT NULL
);
CREATE TABLE DESERT(
desert_id VARCHAR2(10) PRIMARY KEY NOT NULL,
desert_image VARCHAR2(14) NOT NULL,
desert_name VARCHAR2(13) NOT NULL,
introduce VARCHAR2(14) NOT NULL,
price FLOAT NOT NULL,
type VARCHAR2(14) NOT NULL
);
CREATE TABLE MEMBER
(
member_id VARCHAR2(10) PRIMARY KEY NOT NULL,
member_start DATE NOT NULL,
member_end DATE NOT NULL ,
member_time VARCHAR2(10) NOT NULL,
user_id VARCHAR2(10) NOT NULL
```

```
);
CREATE TABLE USERS
(
user_id VARCHAR2(10) PRIMARY KEY NOT NULL,
tel VARCHAR2(10) NOT NULL,
username VARCHAR2(10) NOT NULL,
address VARCHAR2(10)
);
CREATE TABLE ADMIN
(
admin_id VARCHAR2(10) PRIMARY KEY NOT NULL,
tel VARCHAR2(10) NOT NULL,
admin_name VARCHAR2(10) NOT NULL,
password VARCHAR2(10) NOT NULL
);
```

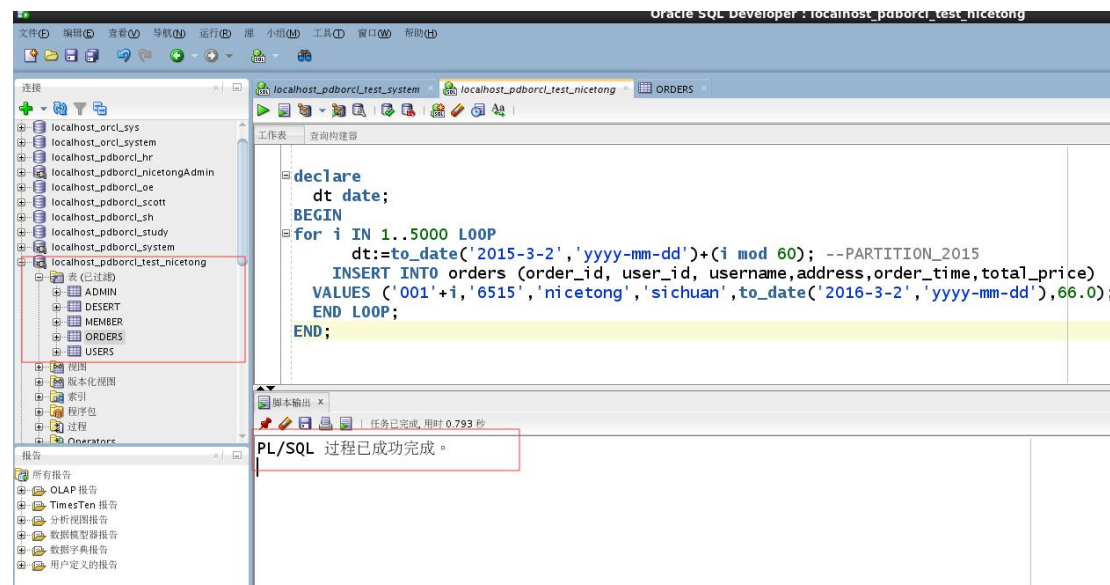


图 5-1 甜点商城创建表及其约束

在这个命令中，核心的语句是“PARTITION BY RANGE (order_time)”，表示按订单时间 ORDER_TIME 的范围进行分区。分区 PARTITION_BEFORE_2019 存储订单时间小于 2019 年的订单记录，这个分区存储在表空间 USERS 中，而分区 PARTITION_BEFORE_2020 存储的是订单时间小于 2020 年的订单记录，由于有分区 PARTITION_BEFORE_2019 的存在，PARTITION_BEFORE_2020 分区实际只存储 2019 年一年的记录。创建 ORDERS 表的部分语句如下：

```
create table ORDERS
(
order_id VARCHAR2(10) PRIMARY KEY NOT NULL,
user_id VARCHAR2(10) NOT NULL,
username VARCHAR2(9) NOT NULL,
address VARCHAR2(10) NOT NULL,
```

```

order_time DATE NOT NULL,
total_price FLOAT NOT NULL
)
TABLESPACE USERS
PCTFREE 10 INITRANS 1
STORAGE ( BUFFER_POOL DEFAULT )
NOCOMPRESS NOPARALLEL
PARTITION BY RANGE (order_time)
(
    PARTITION PARTITION_BEFORE_2016 VALUES LESS THAN (
        TO_DATE(' 2019-01-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS',
            'NLS_CALENDAR=GREGORIAN'))
    NOLOGGING
    TABLESPACE USERS
    PCTFREE 10
    INITRANS 1
    STORAGE
    (
        INITIAL 8388608
        NEXT 1048576
        MINEXTENTS 1
        MAXEXTENTS UNLIMITED
        BUFFER_POOL DEFAULT
    )
    NOCOMPRESS NO INMEMORY
    , PARTITION PARTITION_BEFORE_2017 VALUES LESS THAN (
        TO_DATE(' 2020-01-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS',
            'NLS_CALENDAR=GREGORIAN'))
    NOLOGGING
);

```

向订单表中加入 50000 条数据

```

declare
    dt date;
BEGIN
for i IN 1..50000 LOOP
    dt:=to_date('2015-3-2','yyyy-mm-dd')+(i mod 60); --PARTITION_2015
    INSERT INTO orders (order_id, user_id,
username, address, order_time, total_price)
    VALUES
('001'+i,'6515','nicetong','sichuan',to_date('2016-3-2','yyyy-mm-dd'),66.0);
    END LOOP;
END;

```

查询插入的数据

```
select * from orders;  
select count(order_id) from orders;
```

The image shows two screenshots of a SQL query interface. The top screenshot displays the results of a query to retrieve all data from the 'orders' table. It shows 16 rows with columns: ORDER_ID, USER_ID, USERNAME, ADDRESS, ORDER_TIME, and TOTAL_PRICE. The bottom screenshot shows the results of a query to count the number of rows in the 'orders' table, returning a single row with the count 50000.

	ORDER_ID	USER_ID	USERNAME	ADDRESS	ORDER_TIME	TOTAL_PRICE
1	48717	6515	nicetong	si chuan	02-3月 -16	66
2	48718	6515	nicetong	si chuan	02-3月 -16	66
3	48719	6515	nicetong	si chuan	02-3月 -16	66
4	48720	6515	nicetong	si chuan	02-3月 -16	66
5	48721	6515	nicetong	si chuan	02-3月 -16	66
6	48722	6515	nicetong	si chuan	02-3月 -16	66
7	48723	6515	nicetong	si chuan	02-3月 -16	66
8	48724	6515	nicetong	si chuan	02-3月 -16	66
9	48725	6515	nicetong	si chuan	02-3月 -16	66
10	48726	6515	nicetong	si chuan	02-3月 -16	66
11	48727	6515	nicetong	si chuan	02-3月 -16	66
12	48728	6515	nicetong	si chuan	02-3月 -16	66
13	48729	6515	nicetong	si chuan	02-3月 -16	66
14	48730	6515	nicetong	si chuan	02-3月 -16	66
15	48731	6515	nicetong	si chuan	02-3月 -16	66
16	48732	6515	nicetong	si chuan	02-3月 -16	66

	COUNT(ORDER_ID)
1	50000

图 5-2 向 order 表中插入 50000 条数据

因为订单表的数据很多，我们在这里可以为订单表建立索引。

为 user_id 建立索引

```
create index info_order_id on orders (user_id);
```

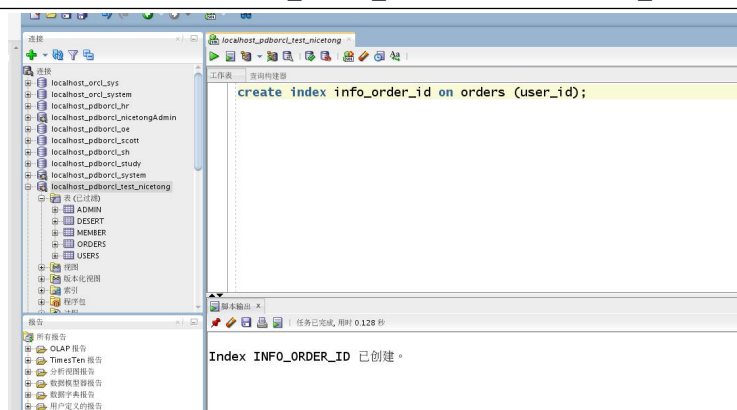


图 5-3 甜点商城创建索引

6. 创建触发器、序列和视图

主要用于级联更新，如更新 users 表中的 users_id 时，member 表的 user_id 也更新。

```
CREATE OR REPLACE TRIGGER cascade_trigger
AFTER UPDATE OF user_id ON users
FOR EACH ROW
BEGIN
    UPDATE member SET user_id=:new.user_id WHERE user_id=:old.user_id;
END;
```

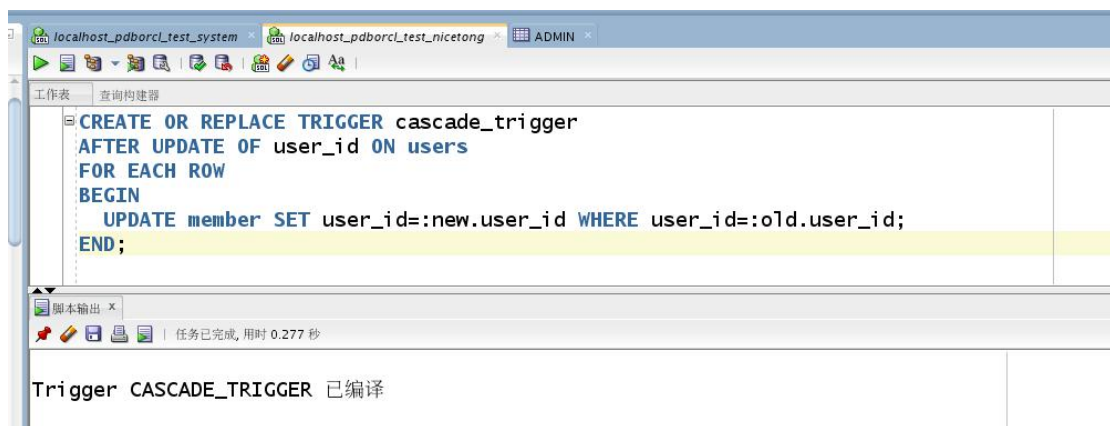


图 6-1 甜点商城创建级联更新用户 id 的触发器

创建序列：为 orders 表创建自增序列。

```
CREATE SEQUENCE orders_sequence
INCREMENT BY 1
START WITH 1
NOMAXVALUE
NOCYCLE
CACHE 10;
```

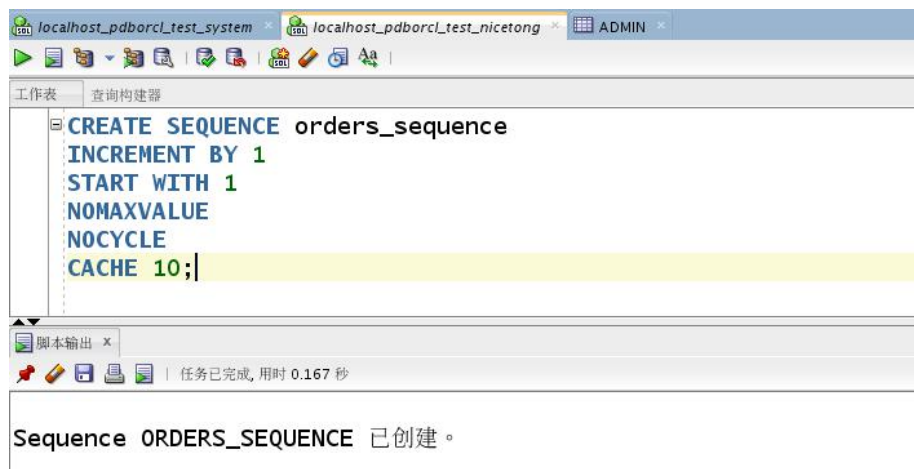


图 6-2 甜点商城创建序列

7. 创建程序包、函数和过程

7.1 创建一个存储过程，修改 admin 表中的 admin_name 字段。

```
create or replace procedure update_admin
(
v_admin_id  varchar2,
v_admin_name varchar2
) is
begin
update admin set admin_name=v_admin_name where admin_id=v_admin_id;
end update_admin;
```

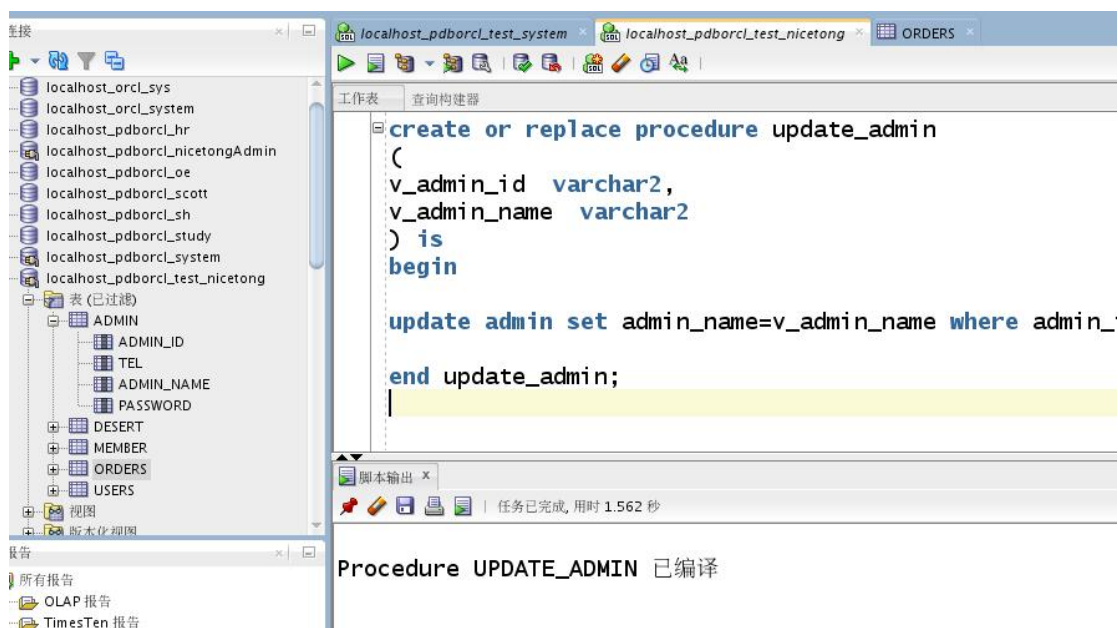


图 7-1 甜点商城创建更新管理员密码的存储过程

测试存储过程：

```
select * from admin;
exec update_admin('001','test');
select * from admin;
```

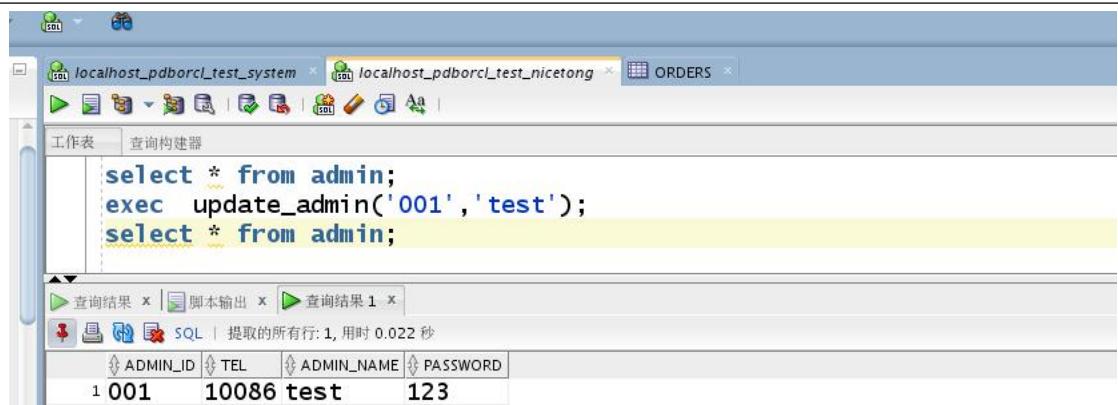


图 7-2 甜点商城测试更新管理员密码的存储过程

7.2 创建一个存储过程，修改 admin 表中的 tel 字段。

```
create or replace procedure update_adminTel
(
v_admin_id varchar2,
v_admin_tel varchar2
) is
begin
update admin set tel=v_admin_tel where admin_id=v_admin_id;
end update_adminTel;
```

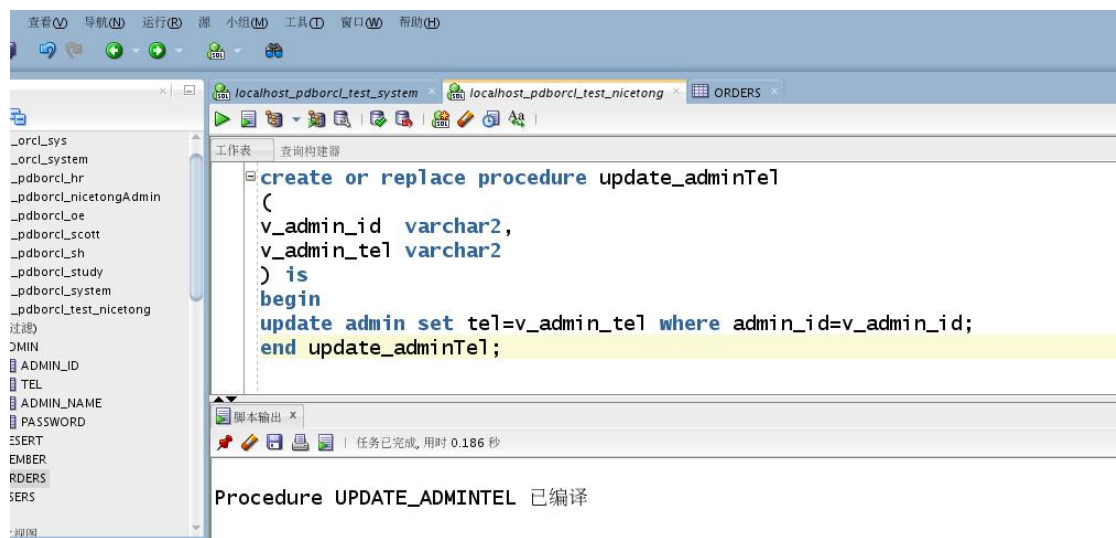


图 7-3 甜点商城创建更新管理员联系电话的存储过程

测试存储过程：

```
select * from admin;
exec update_adminTel('001','123456');
select * from admin;
```

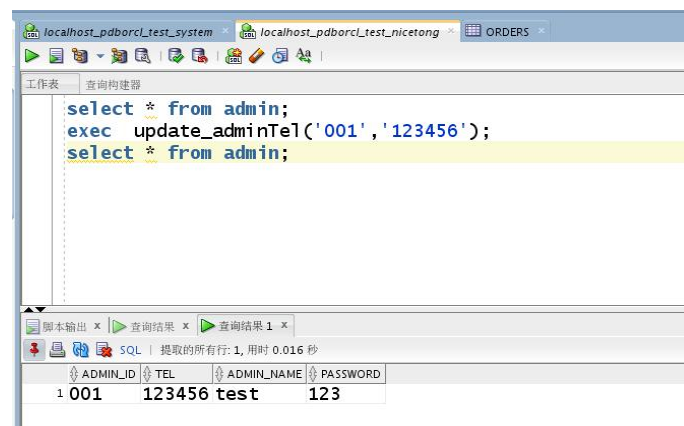


图 7-4 甜点商城测试更新管理员联系电话的存储过程

7.3 创建一个存储过程，查看 admin 表中根据 admin_id 查询出 admin 的 name

```
create or replace procedure admin_out_name
(
  v_admin_id in VARCHAR2,
  v_name out VARCHAR2
) is
vname VARCHAR2(10) ;
begin
  select admin_name into vname from admin where admin_id=v_admin_id;
  v_name:=vname;
end;
```

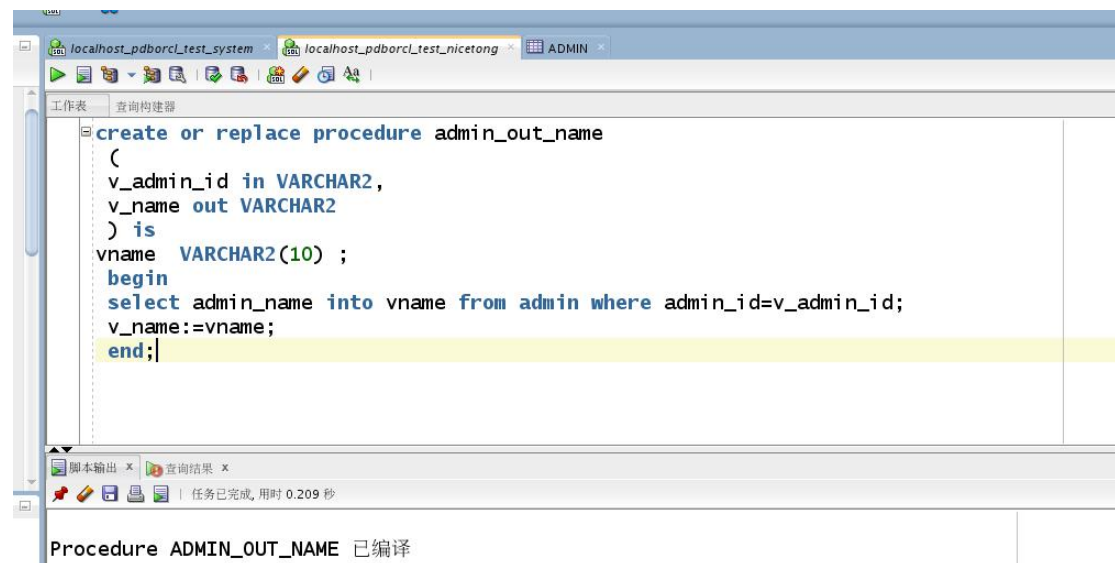


图 7-5 甜点商城创建查看管理员 name 的存储过程

测试存储过程：

```
var vname VARCHAR2
exec admin_out_name('001',:vname);
```

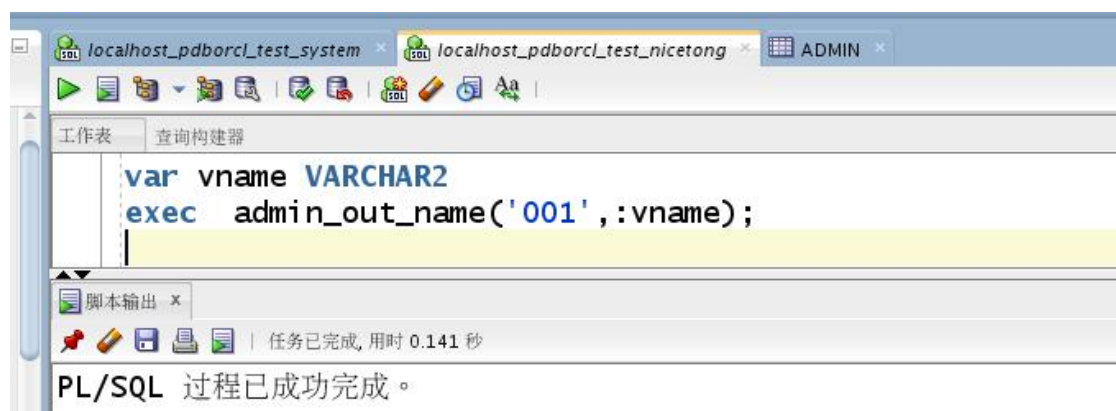


图 7-6 甜点商城测试查看管理员 name 的存储过程

7.4 创建程序包

本系统设计了一些函数及过程，共用的一些程序包放在程序包MyPack中。其他的程序包放在自己的包中。包用于组合逻辑相关的过程和函数，它由包规范和包体两个部分组成。包规范用于定义公用的常量、变量、过程和函数，创建包规范可以使用CREATE PACKAGE命令，创建包体可以使用CREATE PACKAGE BODY。

```
create package admin_pkg is
procedure admin_update_Tel(v_admin_id varchar2,v_admin_tel varchar2);
function admin_get_name(v_admin_id varchar2) return varchar2;
end;
```

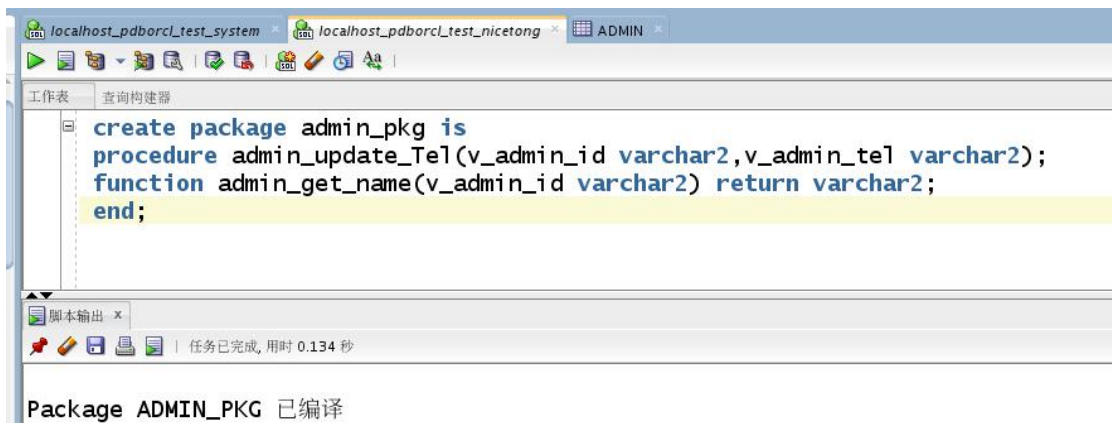


图 7-7 甜点商城创建关于 admin 表的程序包

创建关于 admin 表程序包 admin_pkg 的包体：

```
create or replace package body admin_pkg
is
  procedure admin_update_Tel
  (
    v_admin_id  varchar2,
    v_admin_tel varchar2
  ) is
  begin
    update admin set tel=v_admin_tel where admin_id=v_admin_id;
  end;

  function admin_get_name
  (
    v_admin_id varchar2
  )
  return varchar2 is
    vname varchar2(10);
  begin
    select admin_name into vname from admin where admin_id= v_admin_id;
    return vname;
  end;
```

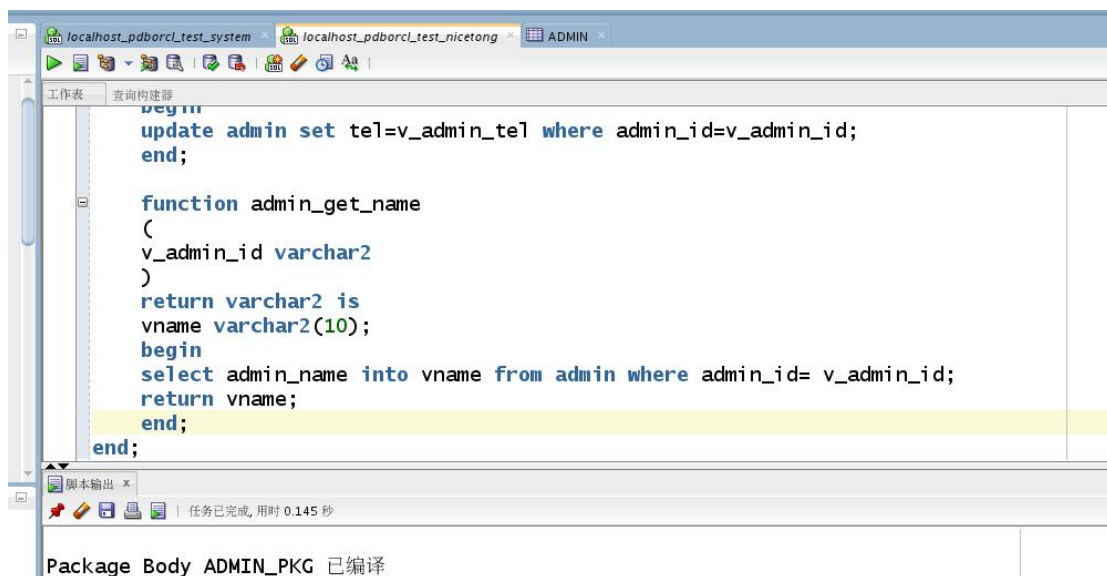


图 7-8 甜点商城创建关于 admin 表的程序包的包体

注意：

在此提示，在没有创建包规范就创建包体，会失败，要使用包，必须先创建包规范，然后在创建包体当要调用包的过程和函数时，在过程和函数的名称前加上包名作为前缀（包名字程序名称），而如果要访问其他方案的包时需要在包的名称前加上方案的名称（方案名称.包名.子程序名称）

测试 admin_pkg 的程序包：

```
select admin_pkg.admin_get_name('001') from dual;
```

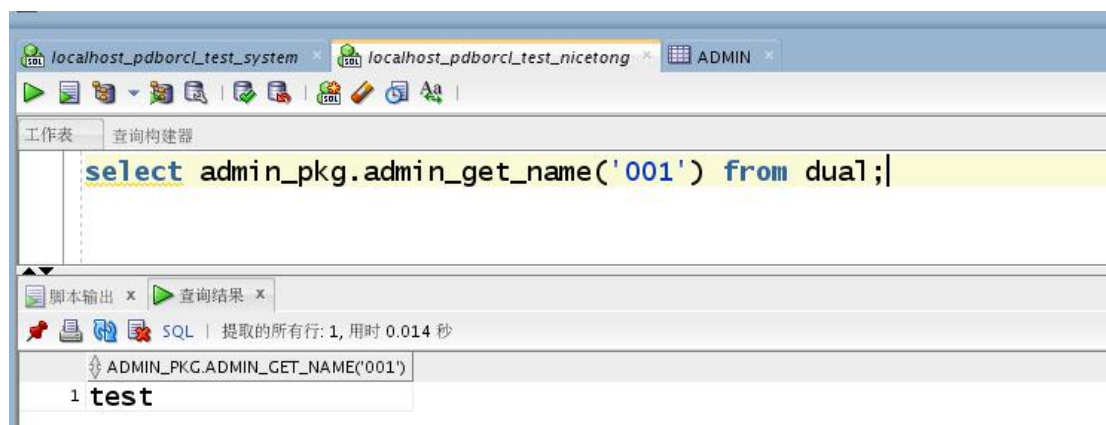


图 7-9 甜点商城测试 admin 程序包的函数

```
exec admin_pkg.admin_update_Tel('001','3333');
select * from admin where admin_id='001';
```



图 7-10 甜点商城测试 admin 程序包的函数

8. 数据库备份

本实验使用老师的虚拟机,通过 rman_level0.sh 和 rman_level1.sh 脚本对数据库进行全备份和全恢复,在数据库出现异常时候,不损失任何数据!

8.1 开始全备份

通过 rman_level0.sh 和 rman_level1.sh 脚本对数据库进行全备份和全恢复。

```
[oracle@oracle-pc ~]$ cat rman_level0.sh
[oracle@oracle-pc ~]$ ./rman_level0.sh
```



```
oracle@oracle-pc:~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
configure controlfile autobackup format for device type disk to '/home/oracle/rman_backup/%F';
configure default device type to disk;
crosscheck backup;
crosscheck archivelog all;
allocate channel c1 device type disk;
backup as compressed backupset incremental level 0 database format '/home/oracle/rman_backup/dblv0_%d_%T_%U.bak'
plus archivelog format '/home/oracle/rman_backup/arclv0_%d_%T_%U.bak';
report obsolete;
delete noprompt obsolete;
delete noprompt expired backup;
delete noprompt expired archivelog all;
release channel c1;
}
EOF
exit
[oracle@oracle-pc ~]$ ./rman_level0.sh
RMAN> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11> 12> 13> 14> 15> 16>
RMAN> [oracle@oracle-pc ~]$
[oracle@oracle-pc ~]$
[oracle@oracle-pc ~]$
```

图 8-1 甜点商城开始全备份

每天定时开始增量备份 (可选)

```
[oracle@oracle-pc ~]$ cat rman_level1.sh
[oracle@oracle-pc ~]$ ./rman_level1.sh
```



```
oracle@oracle-pc:~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
run{
configure retention policy to redundancy 1;
configure controlfile autobackup on;
configure controlfile autobackup format for device type disk to '/home/oracle/rman_backup/%F';
configure default device type to disk;
crosscheck backup;
crosscheck archivelog all;
allocate channel c1 device type disk;
backup as compressed backupset incremental level 1 database format '/home/oracle/rman_backup/dblv1_%d_%T_%U.bak'
plus archivelog format '/home/oracle/rman_backup/arclv1_%d_%T_%U.bak';
report obsolete;
delete noprompt obsolete;
delete noprompt expired backup;
delete noprompt expired archivelog all;
release channel c1;
}
EOF
exit
[oracle@oracle-pc ~]$ ./rman_level1.sh
RMAN> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11> 12> 13> 14> 15> 16> RMAN> [oracle@oracle-pc ~]$
[oracle@oracle-pc ~]$
```

图 8-2 甜点商城每天定时开始增量备份

查看备份文件

```
[oracle@oracle-pc ~]$ cd rman_backup
[oracle@oracle-pc rman_backup]$ ls
arclv0_ORCL_20191124_9fuhmdgk_1_1.bak  dblv1_ORCL_20191124_9iuhmdp6_1_1.bak
arclv1_ORCL_20191124_9huhmdp5_1_1.bak  dblv1_ORCL_20191124_9juhmdrr_1_1.bak
arclv1_ORCL_20191124_9muhmdud_1_1.bak  dblv1_ORCL_20191124_9kuhmdtj_1_1.bak
c-1392946895-20191124-01                lv0_20191111-003303_L0.log
dblv0_ORCL_20191124_9buhmd53_1_1.bak    lv0_20191124-155112_L0.log
dblv0_ORCL_20191124_9cuhmd92_1_1.bak    lv1_20191111-003650_L0.log
dblv0_ORCL_20191124_9duhmdce_1_1.bak    lv1_20191124-160237_L0.log
dblv0_ORCL_20191124_9euhmde6_1_1.bak
```



图 8-3 甜点商城查看备份文件

查看备份文件的内容

```
[oracle@oracle-pc ~]$ rman target /

Recovery Manager: Release 12.1.0.2.0 - Production on 星期日 11月 24 16:11:44 2019

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights reserved.

connected to target database: ORCL (DBID=1392946895)

RMAN> list backup;

using target database control file instead of recovery catalog
```

List of Backup Sets

=====

BS Key	Type	LV Size	Device Type	Elapsed Time	Completion Time
251	Incr 0	215.20M	DISK	00:01:59	24-11 月-19
BP Key: 251 Status: AVAILABLE Compressed: YES Tag: TAG20191124T155203					Name:
Piece					

/home/oracle/rman_backup/dblv0_ORCL_20191124_9buhmd53_1_1.bak

List of Datafiles in backup set 251

Container ID: 3, PDB Name: PDBORCL

File	LV Type	Ckp SCN	Ckp Time	Name
------	---------	---------	----------	------

8		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/system01.dbf					
9		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/sysaux01.dbf					
10		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/SAMPLE_SCHEMA_users01.dbf					
11		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/example01.dbf					
12		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_1.dbf					
13		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_2.dbf					
16		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_3.dbf					
17		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_4.dbf					
77		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users03_1.dbf					
78		0	Incr	47254373	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users03_2.dbf					

BS Key	Type	LV Size	Device Type	Elapsed Time	Completion Time
252	Incr 0	369.77M	DISK	00:01:42	24-11 月-19
BP Key: 252 Status: AVAILABLE Compressed: YES Tag: TAG20191124T155203					Name:
Piece					

/home/oracle/rman_backup/dblv0_ORCL_20191124_9cuhmd92_1_1.bak

List of Datafiles in backup set 252

File	LV Type	Ckp SCN	Ckp Time	Name
------	---------	---------	----------	------

1	0	Incr	47254431	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/system01.dbf						
3	0	Incr	47254431	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/sysaux01.dbf						
4	0	Incr	47254431	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/undotbs01.dbf						
6	0	Incr	47254431	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/users01.dbf						

BS Key	Type	LV	Size	Device	Type	Elapsed Time Completion Time

253	Incr	0	160.44M	DISK		00:00:47 24-11月-19
BP Key: 253 Status: AVAILABLE Compressed: YES Tag: TAG20191124T155203						
Piece Name:						
/home/oracle/rman_backup/dblv0_ORCL_20191124_9duhmdce_1_1.bak						
List of Datafiles in backup set 253						
Container ID: 4, PDB Name: LOCALHOST_PDBORCL_TEST						
File	LV	Type	Ckp	SCN	Ckp Time	Name

79	0	Incr	47254493	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/system01.dbf						
80	0	Incr	47254493	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/sysaux01.dbf						
81	0	Incr	47254493	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/localhost_pdborcl_test_users01.dbf						
82	0	Incr	47254493	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong1.dbf						
83	0	Incr	47254493	24-11	月	-19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong2.dbf						

BS Key	Type	LV	Size	Device	Type	Elapsed Time Completion Time

254	Incr	0	157.08M	DISK		00:01:11 24-11月-19
BP Key: 254 Status: AVAILABLE Compressed: YES Tag: TAG20191124T155203						
Piece Name:						
/home/oracle/rman_backup/dblv0_ORCL_20191124_9euhmde6_1_1.bak						
List of Datafiles in backup set 254						
Container ID: 2, PDB Name: PDB\$SEED						
File	LV	Type	Ckp	SCN	Ckp Time	Name

5	0	Incr	1819408	01-12	月	-14
/home/oracle/app/oracle/oradata/orcl/pdbseed/system01.dbf						

7	0	Incr	1819408	01-12	月	-14
/home/oracle/app/oracle/oradata/orcl/pdbseed/sysaux01.dbf						
BS Key	Size	Device	Type	Elapsed Time	Completion Time	

255	67.50K	DISK		00:00:00	24-11 月-19	
BP Key: 255				Status: AVAILABLE	Compressed: YES	Tag: TAG20191124T155811
Piece				Name:		
/home/oracle/rman_backup/arclv0_ORCL_20191124_9fuhmdgk_1_1.bak						
List of Archived Logs in backup set 255						
Thrd	Seq	Low SCN	Low Time	Next SCN	Next Time	

1	1616	47254352	24-11 月-19	47254544	24-11 月-19	
BS Key	Size	Device	Type	Elapsed Time	Completion Time	

257	1.91M	DISK		00:00:00	24-11 月-19	
BP Key: 257				Status: AVAILABLE	Compressed: YES	Tag: TAG20191124T160245
Piece				Name:		
/home/oracle/rman_backup/arclv1_ORCL_20191124_9huhmdp5_1_1.bak						
List of Archived Logs in backup set 257						
Thrd	Seq	Low SCN	Low Time	Next SCN	Next Time	

1	1616	47254352	24-11 月-19	47254544	24-11 月-19	
1	1617	47254544	24-11 月-19	47255360	24-11 月-19	
BS Key	Type	LV	Size	Device	Type	Elapsed Time

258	Incr	1	328.00K	DISK		00:01:24
BP Key: 258				Status: AVAILABLE	Compressed: YES	Tag: TAG20191124T160246
Piece				Name:		
/home/oracle/rman_backup/dblv1_ORCL_20191124_9iuhmdp6_1_1.bak						
List of Datafiles in backup set 258						
Container ID: 3, PDB Name: PDBORCL						
File	LV	Type	Ckp	SCN	Ckp Time	Name

8			1	Incr	47255370	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/system01.dbf						
9			1	Incr	47255370	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/sysaux01.dbf						
10			1	Incr	47255370	24-11 月 -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/SAMPLE_SCHEMA_users01.dbf						

11	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/example01.dbf
12	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_1.dbf
13	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_2.dbf
16	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_3.dbf
17	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users02_4.dbf
77	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users03_1.dbf
78	1	Incr	47255370	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_users03_2.dbf
BS Key Type LV Size Device Type Elapsed Time Completion Time -----							
259	Incr 1	2.34M	DISK	00:00:49	24-11 月-19		
	BP Key: 259 Status: AVAILABLE Compressed: YES Tag: TAG20191124T160246						
	Piece						Name:
	/home/oracle/rman_backup/dblv1_ORCL_20191124_9juhmdrr_1_1.bak						
	List of Datafiles in backup set 259						
	File LV Type Ckp SCN	Ckp Time	Name				
	-----	-----	-----				
1	1	Incr	47255497	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/system01.dbf
3	1	Incr	47255497	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/sysaux01.dbf
4	1	Incr	47255497	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/undotbs01.dbf
6	1	Incr	47255497	24-11	月	-19	/home/oracle/app/oracle/oradata/orcl/users01.dbf
BS Key Type LV Size Device Type Elapsed Time Completion Time -----							
260	Incr 1	208.00K	DISK	00:00:23	24-11 月-19		
	BP Key: 260 Status: AVAILABLE Compressed: YES Tag: TAG20191124T160246						
	Piece						Name:
	/home/oracle/rman_backup/dblv1_ORCL_20191124_9kuhmdtj_1_1.bak						
	List of Datafiles in backup set 260						
	Container ID: 4, PDB Name: LOCALHOST_PDBORCL_TEST						
	File LV Type Ckp SCN	Ckp Time	Name				
	-----	-----	-----				
79	1	Incr	47255538	24-11	月	-19	

```

/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/system01.dbf
80          1          Incr      47255538          24-11    月    -19
/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/sysaux01.dbf
81          1          Incr      47255538          24-11    月    -19
/home/oracle/app/oracle/oradata/orcl/localhost_pdborcl_test/localhost_pdborcl_t
est_users01.dbf
82          1          Incr      47255538          24-11    月    -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong1.dbf
83          1          Incr      47255538          24-11    月    -19
/home/oracle/app/oracle/oradata/orcl/pdborcl/pdbtest_nicetong2.dbf

BS Key   Size      Device Type Elapsed Time Completion Time
-----
261      90.00K    DISK        00:00:00      24-11 月-19
      BP Key: 261   Status: AVAILABLE Compressed: YES Tag: TAG20191124T160533
      Piece
      Name:
/home/oracle/rman_backup/arclv1_ORCL_20191124_9muhmdud_1_1.bak

List of Archived Logs in backup set 261
Thrd Seq      Low SCN      Low Time     Next SCN     Next Time
-----
1      1618        47255360    24-11 月-19 47255551     24-11 月-19

BS Key   Type LV Size      Device Type Elapsed Time Completion Time
-----
262      Full    17.77M    DISK        00:00:00      24-11 月-19
      BP Key: 262   Status: AVAILABLE Compressed: NO Tag: TAG20191124T160534
      Piece Name: /home/oracle/rman_backup/c-1392946895-20191124-01
      SPFILE Included: Modification time: 24-11 月-19
      SPFILE db_unique_name: ORCL
      Control File Included: Ckp SCN: 47255560      Ckp time: 24-11 月-19

RMAN>

```

由上面的“list backup;”输出可以看出，备份集中的文件内容是：

/home/oracle/rman_backup/dblv0_ORCL_20191120_d7uhb2ap_1_1.bak 是插接数据库 PDBORCL 的备份集

/home/oracle/rman_backup/dblv0_ORCL_20191120_d8uhb2c6_1_1.bak 是 ORCL 的备份集

/home/oracle/rman_backup/dblv0_ORCL_20191120_d9uhb2ei_1_1.bak 是 PDB\$SEED 的备份集

/home/oracle/rman_backup/arclv0_ORCL_20191120_dauhb2fm_1_1.bak 是归档文件的备份集

/home/oracle/rman_backup/c-1392946895-20191120-01 是参数文件 (SPFILE) 和控制文件 (Control File) 的备份集

8.2 备份后修改数据

```
[oracle@oracle-pc ~]$ sqlplus study/123@pdborcl
SQL> create table t1 (id number,name varchar2(50));
Table created.
SQL> insert into t1 values(1,'zhang');1 row created.
SQL> commit;Commit complete.
SQL> select * from t1;
      ID NAME-----
      1 zhang
SQL> exit
```



图 8-4 甜点商城备份后修改文件

8.3 删除数据库文件，模拟数据库文件损坏

```
[oracle@oracle-pc~]$rm
/home/oracle/app/oracle/oradata/orcl/pdborcl/SAMPLE_SCHEMA_users01.dbf
```

删除数据库文件后修改数据

删除数据文件后，仍然可以增加一条数据。这是因为增加的数据并没有写入数据文件，而是写到了日志文件中。如果增加的数据较多的时候，就会出问题了。

```
SQL> insert into t1 values(2,'wang');
1 row created.
SQL> commit;
Commit complete.
```

```

SQL> select * from t1;

      ID NAME
-----
      2 wang
      1 zhang
SQL> declare
      2   n number;
      3   begin
      4     for n in 1..10000 loop
      5       insert into t1 values(n,'name' || n);
      6     end loop;
      7   end;
      8   /
declare
*
ERROR at line 1:
ORA-01116: 打开数据库文件 10 时出错 ORA-01110:
数      据      文      件                               10:
' /home/oracle/app/oracle/oradata/orcl/pdborcl/SAMPLE_SCHEMA_users01.dbf'
ORA-27041: 无法打开文件
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
ORA-06512: 在 line 5
SQL> select * from t1;

      ID NAME
-----
      2 wang
      1 zhang

```

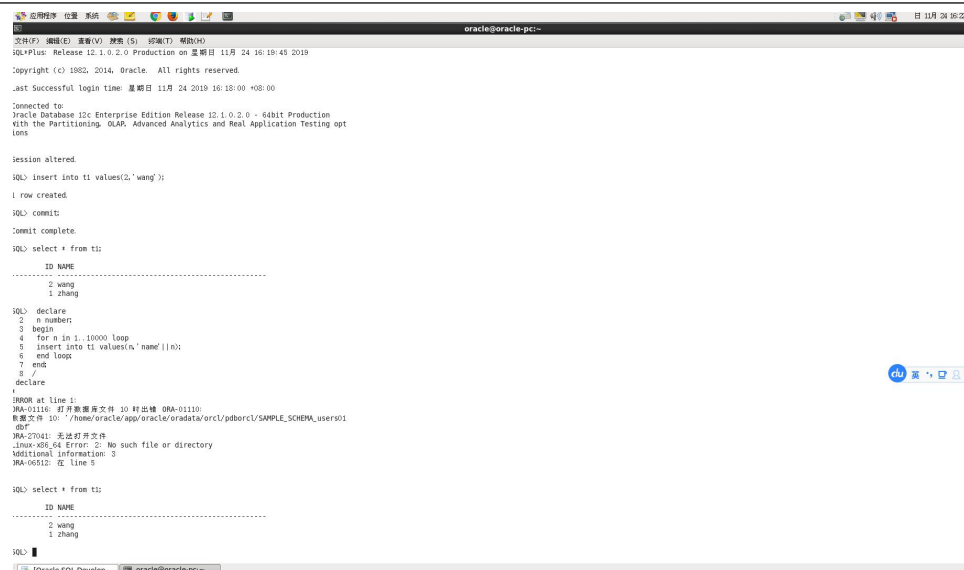


图 8-5 甜点商城删除数据库文件

8.4 数据库完全恢复

8.4.1 重启损坏的数据库到 mount 状态

通过 shutdown immediate 无法正常关闭数据库，只能通过 shutdown abort 强制关闭。然后将数据库启动到 mount 状态。

```
[oracle@oracle-pc ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0 Production on 星期日 11 月 24 16:25:29 2019
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Session altered.
SQL> shutdown immediate
ORA-01116: 打开数据库文件 10 时出错
ORA-01110: 数据文件 10:
'/home/oracle/app/oracle/oradata/orcl/pdborcl/SAMPLE_SCHEMA_users01.dbf'
ORA-27041: 无法打开文件
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
SQL> shutdown abort
ORACLE instance shut down.
SQL> startup mount
ORACLE instance started.

Total System Global Area 1577058304 bytes
Fixed Size 2924832 bytes
Variable Size 738201312 bytes
Database Buffers 654311424 bytes
Redo Buffers 13848576 bytes
In-Memory Area 167772160 bytes
Database mounted.
SQL> exit
```

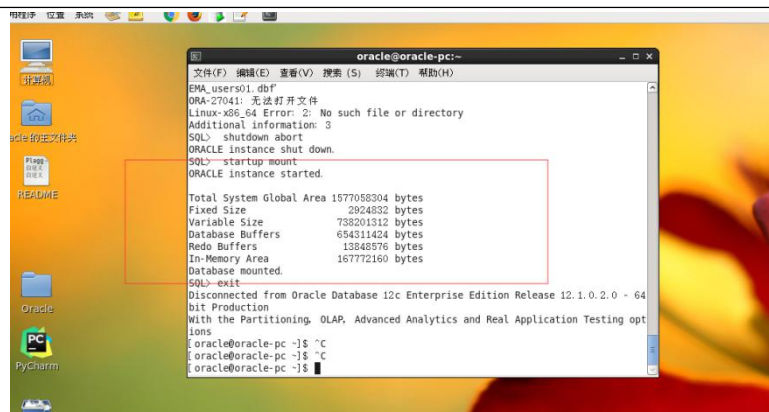


图 8-6 甜点商城重启损坏的数据库到 mount 状态

8.4.2 开始恢复数据库

```
[oracle@oracle-pc ~]$ rman target /
```

第一步:

```
RMAN> restore database ;
```

第二步:

```
RMAN> recover database;
```

第三步:

```
RMAN> alter database open;
```

第四步:

```
RMAN> exit
```

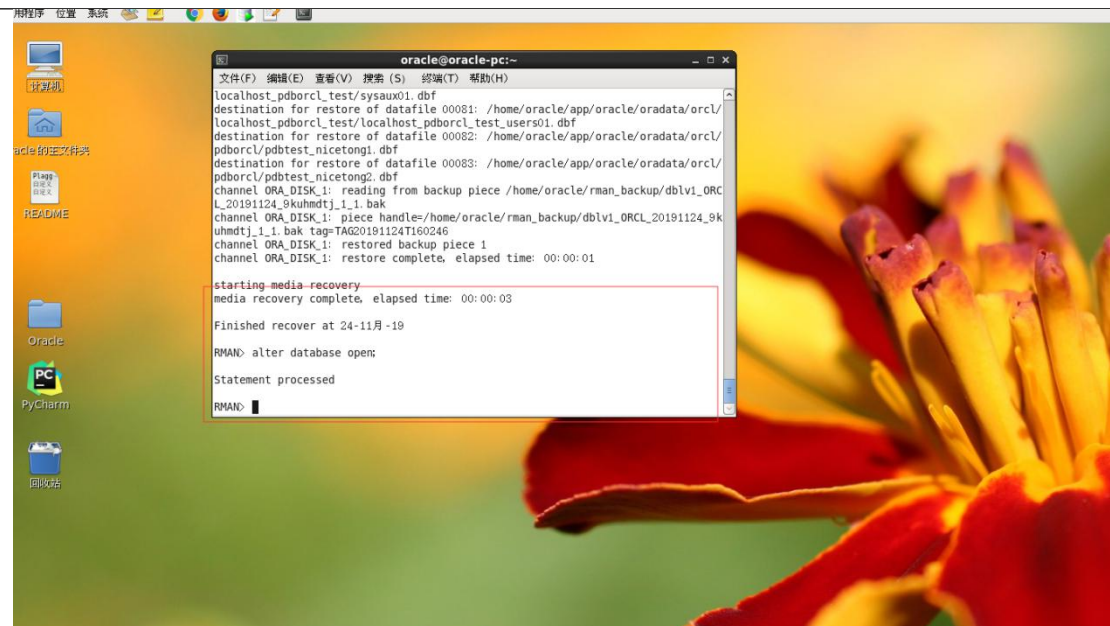


图 8-7 甜点商城开始恢复数据库

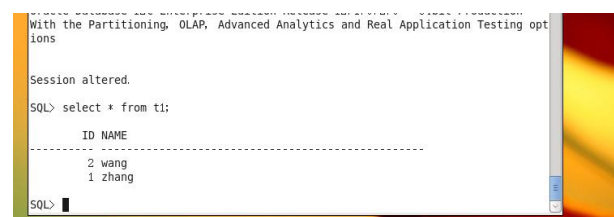
8.5 查询数据是否恢复

```
[oracle@oracle-pc ~]$ sqlplus study/123@pdborcl
```

```
SQL> select * from t1;
```

ID	NAME
1	zhang
2	wang

```
SQL>
```



由以上查询结果可见，数据 100%恢复了!

9. 数据库 DataGuard 设计

9.1 oracle 12C dataguard 搭建环境（主从库相同，从库只装软件不建库）

```
[oracle@oracle-pc ~]$ more /etc/redhat-release
CentOS release 6.9 (Final)
[oracle@oracle-pc ~]$ sqlplus
SQL*Plus: Release 12.1.0.2.0 Production on 星期日 11月 24 16:59:13 2019
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Enter user-name: system
Enter password: ***
Last Successful login time: 星期日 11月 24 2019 12:38:12 +08:00
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Session altered.
SQL> SELECT * FROM V$VERSION;
BANNER
CON_ID
-----
-
-----
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
0
PL/SQL Release 12.1.0.2.0 - Production
0
CORE 12.1.0.2.0 Production
0
TNS for Linux: Version 12.1.0.2.0 - Production
0
NLSRTL Version 12.1.0.2.0 - Production
0
SQL>
```

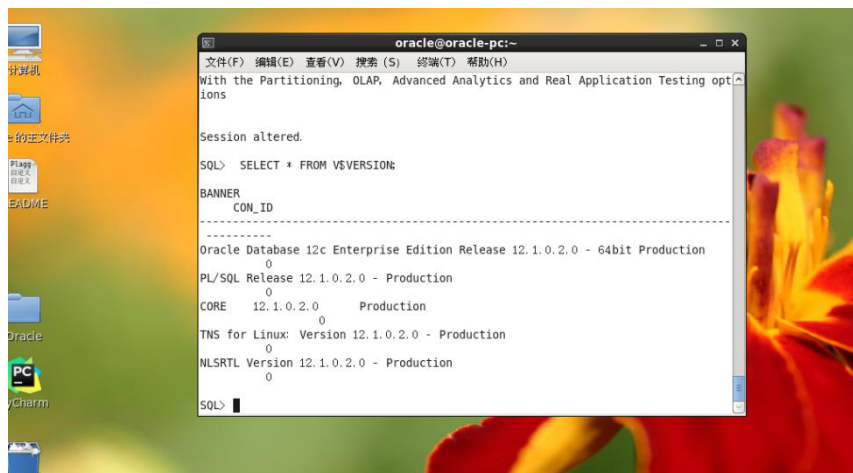


图 9-1 甜点商城查看 OS 和 ORACLE 配置信息

9.2 dataguard 主库参数配置

9.2.1 检查主库状态，pdb 是否为 open

```
[oracle@oracle-pc ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0 Production on 星期日 11月 24 17:08:52 2019
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Session altered.
SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                                READ ONLY  NO
      3 PDBORCL                                  READ WRITE NO
      4 LOCALHOST_PDBORCL_TEST                  READ WRITE NO

SQL>
```

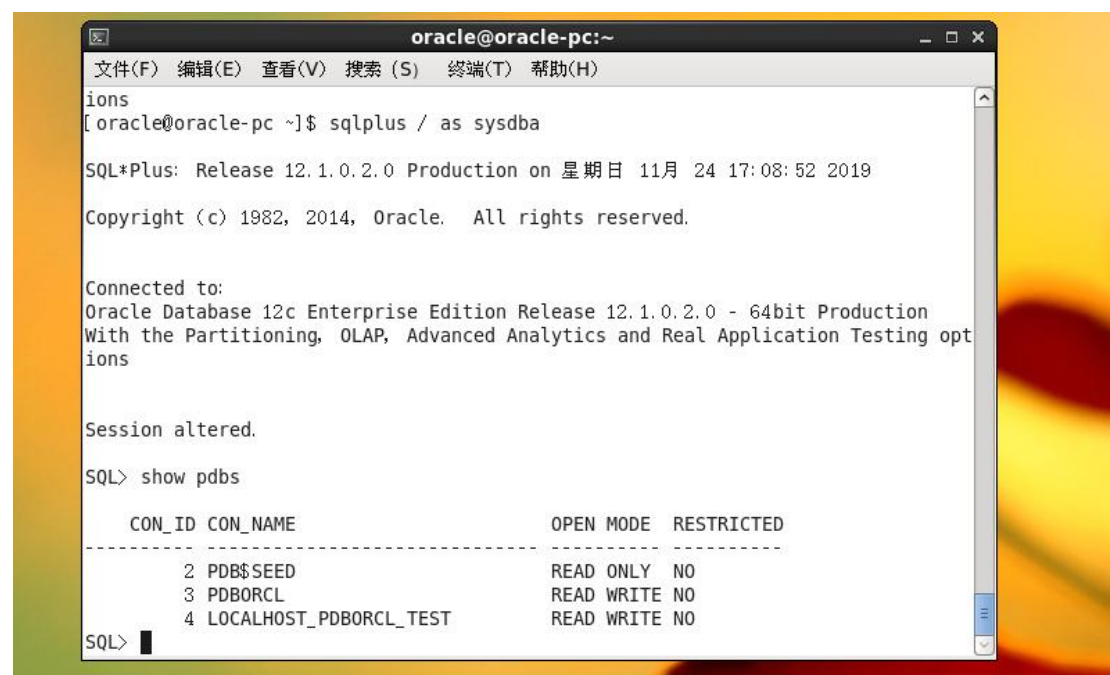


图 9-2 甜点商城查看检查主库状态

9.2.2 检查主库是否开启 force_logging, 否则要执行以下命令

```
SQL> select force_logging from v$database;
SQL> alter database force logging;
Database altered.
SQL> select force_logging from v$database;
FORCE_LOGGING
-----
YES
SQL>
```

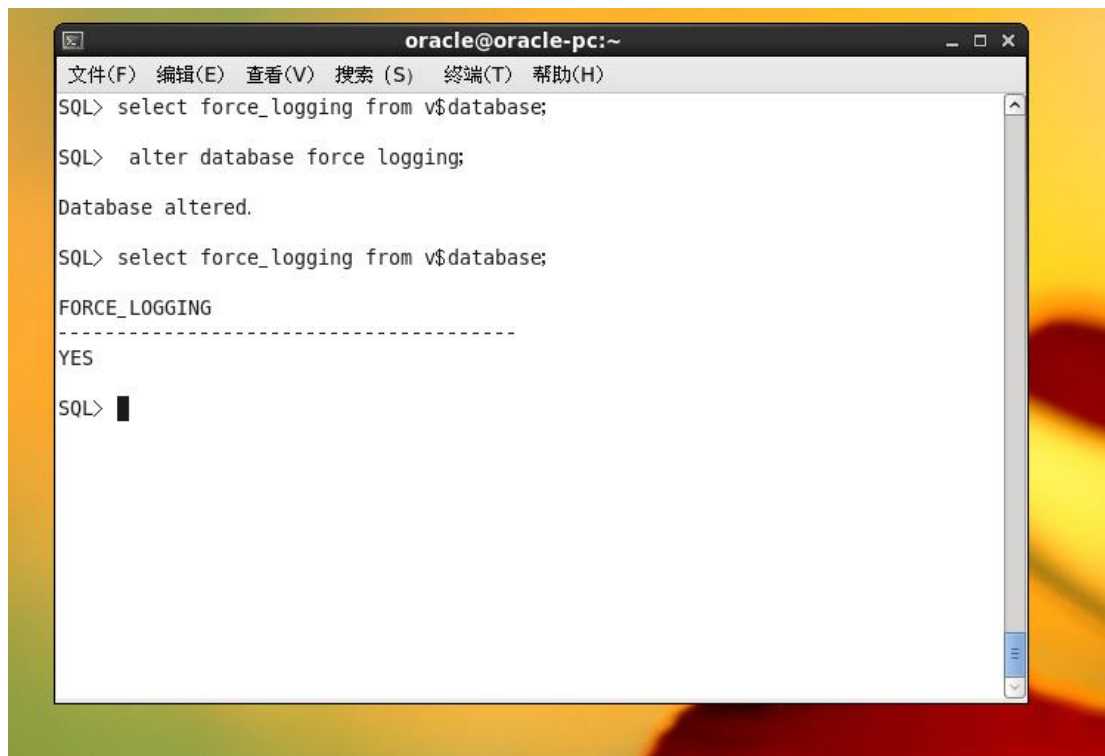


图 9-3 甜点商城查看检查主库是否开启 force_logging

9.2.3 检查主库归档模式开启情况

```

SQL> archive log list;

Database log mode           Archive Mode
Automatic archival         Enabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 1618
Next log sequence to archive 1620
Current log sequence        1620
SQL>

```

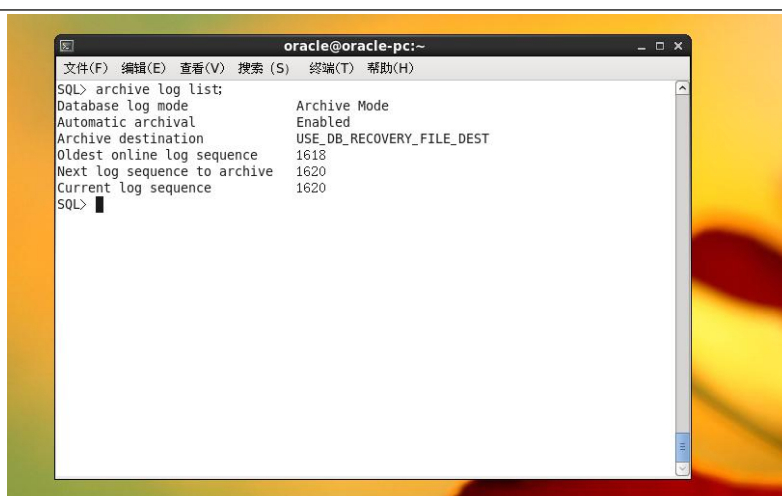


图 9-4 甜点商城查看检查主库检查主库归档模式开启情况

9.2.4 检查在线日志 redo log，确认 member 和 group#，根据在线日志添加 standby logfile
(以下操作均在 CBD 下操作)

```
SQL> select group#, members, bytes from v$log;
```

GROUP#	MEMBERS	BYTES
1	2	52428800
2	1	52428800
3	1	52428800

```
SQL>
```

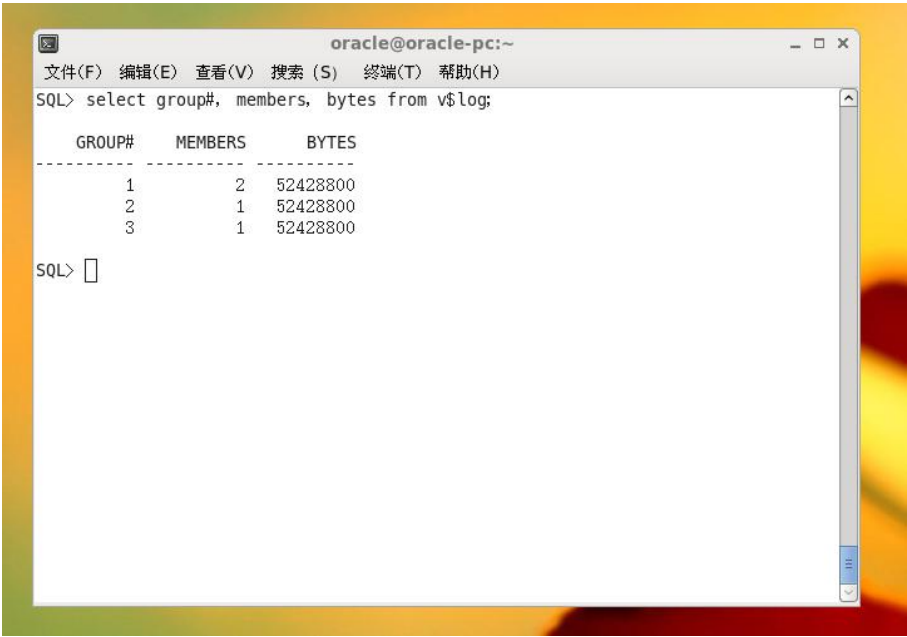


图 9-5 甜点商城检查在线日志 redo log

添加 standby logfile (3+1) :

```
ALTER DATABASE ADD STANDBY LOGFILE
'/home/app/oracle/oradata/orcl/online/redo01.log' size 50M;
ALTER DATABASE ADD STANDBY LOGFILE
'/home/app/oracle/oradata/orcl/online/redo02.log' size 50M;
ALTER DATABASE ADD STANDBY LOGFILE
'/home/app/oracle/oradata/orcl/online/redo03.log' size 50M;
ALTER DATABASE ADD STANDBY LOGFILE
'/home/app/oracle/oradata/orcl/online/redo04.log' size 50M;
```

9.3. 配置两个数据库的网络参数文件 tnsname.ora 和 listener.ora
primary 和 Standby 上的 tnsnames.ora 相同。

9.3.1 配置 tnsnames.ora 参数文件

```
#primary database
porcl =
```

```

(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP) (HOST = oracle-one) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = 10.11.12.1) (PORT = 1521))
  )
  (CONNECT_DATA =
    (SERVICE_NAME = orcl)
  )
)
#standby database
sorcl=
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP) (HOST = 10.11.12.2) (PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = orcl)
  )
  (UR=A)
)
)
#pluggable database
cdb =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP) (HOST = oracle-one) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = 10.11.12.1) (PORT = 1521))
  )
  (CONNECT_DATA =
    (SERVICE_NAME = cdb)
  )
)
)

```

9.3.2 配置监听文件

```

#          listener.ora          Network          Configuration          File:
/home/app/oracle/product/12.2.0/dbhome_1/network/admin/listener.ora
# Generated by Oracle configuration tools.
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = orcl)
      (ORACLE_HOME = /home/app/oracle/product/12.2.0/dbhome_1)
      (SID_NAME = ORCL)
    )
  )
)
LISTENER =
  (DESCRIPTION_LIST =

```

```

(DESCRIPTION =
(AADDRESS = (PROTOCOL = TCP) (HOST = oracle-one) (PORT = 1521))
)
(DESCRIPTION =
(AADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
)
)
ADR_BASE_LISTENER = /u01/app/oracle
LISTENERD =
(DESCRIPTION_LIST =
(DESCRIPTION =
(AADDRESS = (PROTOCOL = TCP) (HOST = oracle-one) (PORT = 1521))
)
)
)

```

9.4. 配置数据库参数文件

注意：数据库参数文件的调整时注意 PDB 和 CDB 的切换运行，否则会报如下错误

9.4.1 主库配置完后参数文件：如下（如果是从 pfile 指定文件启动，启动完成后要创建 spfile 文件）

```

orcl.__data_transfer_cache_size=0
orcl.__db_cache_size=234881024
orcl.__inmemory_ext_roarea=0
orcl.__inmemory_ext_rwarea=0
orcl.__java_pool_size=4194304
orcl.__large_pool_size=8388608
orcl.__oracle_base='/home/app/oracle'#ORACLE_BASE set from environment
orcl.__pga_aggregate_target=272629760
orcl.__sga_target=507510784
orcl.__shared_io_pool_size=20971520
orcl.__shared_pool_size=222298112
orcl.__streams_pool_size=0
*.audit_file_dest='/u01/app/oracle/admin/orcl/adump'
*.audit_trail='db'
*.compatible='12.2.0'
*.control_files='/home/app/oracle/oradata/ORCL/controlfile/o1_mf_dqflr6xp_.ctl'
, '/home/app/oracle/fast_recovery_area/orcl/ORCL/controlfile/o1_mf_dqflr78t_.ctl'
,
*.db_block_size=8192
*.db_create_file_dest='/home/app/oracle/oradata'
*.db_file_name_convert='/home/app/oracle/oradata/ORCL/datafile/', '/u01/app/oracle/oradata/ORCL/datafile/'
*.db_name='orcl'
*.db_recovery_file_dest='/home/app/oracle/fast_recovery_area/orcl'
*.db_recovery_file_dest_size=8016m
*.db_unique_name='porcl'

```

```

*.diagnostic_dest='/home/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=orclXDB)'
*.enable_pluggable_database=true
*.fal_client='PORCL'
*.fal_server='SORCL'
*.log_archive_config='DG_CONFIG=(porcl,sorcl)'
*.log_archive_dest_1='location=/home/app/oracle/oradata/arch
db_unique_name=porcl'
*.log_archive_dest_2='service=sorcl LGWR ASYNC noaffirm reopen=60
valid_for=(online_logfiles,primary_role) db_unique_name=sorcl'
*.log_archive_dest_state_2='enable'
*.log_file_name_convert='/home/app/oracle/oradata/ORCL/onlinelog/', '/home/app/o
racle/oradata/ORCL/onlinelog/', '/home/app/oracle/fast_recovery_area/orcl/ORCL/o
nlinelog/',
/home/app/oracle/fast_recovery_area/orcl/ORCL/onlinelog/'
*.memory_target=744m
*.open_cursors=300
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.service_names='ORCL'
*.standby_file_management='auto'
*.undo_tablespace='UNDOTBS1'

```

9.4.2 将主库修改后的参数文件传送到备库

9.4.3 修改从库数据库参数文件并使用该参数文件从 pfile 启动

```

*.db_unique_name=
*.log_archive_config=
*.log_file_name_convert=
*.db_file_name_convert=
*.log_archive_dest_1=
*.log_archive_dest_2=
*.standby_file_management=
*.fal_server=
*.fal_client=

```

9.4.4 主库密码文件传送到备库

```
[oracle@oracle-pc ~]$ scp orapworcl orcl-one:/dbs/
```

9.5.启动 standby database 到 nomount 状态

9.5.1 检查 tns 连接名是否正常(主备库均需要检查)

```

tnsping porcl
tnsping sorcl
lsnrctl

```

9.5.2 启动备库

```
startup pfile='/home/oracle/standbyfile.ora' nomount
```

启动后创建 spfile

```
create spfile from pfile='/home/oracle/standbyfile.ora'
```

参考文献

- [1]Oracle 12c 数据库基础教程[M]. 赵卫东、刘永红，于曦.北京：科学出版社，2017.
- [2]oracle 12C dataguard 搭建 .
<https://blog.csdn.net/s754520480/article/details/79789771>
- [3]Oracle 数据库应用. <https://github.com/zwdcdu/oracle>