

**信息工程学院**

**《oracle数据库应用》课程设计**

**题 目** 宠物商店管理系统

**学 院**

**专 业**

**班 别**

**指导教师**

**学 号**

**姓 名**

**成 绩**

2019 年 6月 27日

**一、背景与前期设计**

背景：在越来越多人养宠物的情况下，也越来越多了注重起宠物的生活质量与食物安全，所以宠物商店也逐渐兴盛起来。在这种情况下，宠物商店管理系统的建立可以帮助管理人员、商店工作人员减少大量工作，对数据的处理也更加方便。

根据其职能划分，商店管理主要分为：

* 客户订单管理
* 售后管理
* 仓库管理
* 员工工资管理
* 员工职位管理
* 客户反馈管理
* …

该作业针对商店管理中的客户管理模块。客户订单管理模块有如下的表：

表1 商品种类（varity）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| vno | char(4) | varity | 商品种类编号 | FALSE |
| vnamve | char(8) | varity | 货物种类 | FALSE |

表2 客户（client）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| cno | char(8) | client | 客户编号 | FALSE |
| cname | char(20) | client | 客户名 | FALSE |
| tel | char(11) | client | 电话 | FALSE |
| gender | char(2) | client | 性别 | FALSE |

表3 职位（post）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| postno | char(8) | post | 职位编号 | FALSE |
| postname | char(20) | post | 职位名 | FALSE |

表4 薪酬表（wage）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| wno | char(8) | wage | 薪酬编号 | FALSE |
| basepay | char(10) | wage | 基本薪酬 | FALSE |
| insurance | char(10) | wage | 保险 | FALSE |
| allowance | char(10) | wage | 补贴 | FALSE |
| salary | char(10) | wage | 第一工资 | FALSE |
| postname | char(20) | wage | 职位名 | FALSE |

表5 员工（clerk）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| clerkno | char(8) | clerk | 员工编号 | FALSE |
| clerkname | char(20) | clerk | 姓名 | FALSE |
| tel | char(11) | clerk | 手机 | FALSE |
| postno | char(8) | clerk | 职位号 | TRUE |
| wno | char(8) | clerk | 薪酬号 | TRUE |
| bonus | char(10) | clerk | 奖金 | FALSE |
| wages | char(10) | clerk | 最后工资 | FALSE |

表6 订单表（orderlist）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| odno | char(8) | orderlist | 订单编号 | FALSE |
| cno | char(8) | orderlist | 顾客号 | TRUE |
| totalprice | char(10) | orderlist | 总价 | FALSE |
| clerkno | char(8) | orderlist | 负责人 | TRUE |

表7 仓库表（repertory）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| rno | char(8) | repertory | 商品号 | FALSE |
| rname | char(30) | repertory | 商品名 | FALSE |
| capital | char(10) | repertory | 进货价 | FALSE |
| saleprice | char(10) | repertory | 售出价 | FALSE |
| numin | char(10) | repertory | 进货数量 | FALSE |
| numout | char(10) | repertory | 售出数量 | FALSE |
| numnow | char(10) | repertory | 库存 | FALSE |
| clerkno | char(8) | repertory | 负责人 | TRUE |
| vno | char(4) | repertory | 种类号 | TRUE |

表8宠物商品表（goods）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| gno | char(8) | goods | 宠物商品编号 | FALSE |
| odno | char(8) | goods | 商品号 | TRUE |
| rno | char(8) | goods | 商品订单号 | TRUE |
| rname | char(30) | goods | 商品名 | FALSE |
| saleprice | char(10) | goods | 售价 | FALSE |
| num | char(10) | goods | 数量 | FALSE |
| clerkno | char(8) | goods | 商品种类编号 | TRUE |

表 9 宠物表（pets）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| pno | char(8) | pets | 宠物编号 | FALSE |
| odno | char(8) | pets | 商品号 | TRUE |
| rno | char(8) | pets | 订单号 | TRUE |
| rname | char(20) | pets | 商品名 | FALSE |
| num | char(10) | pets | 数量 | FALSE |
| gender | char(10) | pets | 性别 | FALSE |
| age | char(10) | pets | 年龄 | FALSE |
| saleprice | char(10) | pets | 售价 | FALSE |
| clerkno | char(8) | pets | 负责人 | TRUE |

表 10 反馈信息信息表（feedback）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名稱 | 資料類型 | 表格 | 注釋 | 是否主鍵 |
| cno | char(8) | feedback | 顾客号 | TRUE |
| feedback | char(50) | feedback | 评价内容 | FALSE |
| clerkno | char(8) | feedback | 负责人 | TRUE |
| mark | char(10) | feedback | 评分 | FALSE |
| states | char(10) | feedback | 是否解决 | FALSE |

根据以上表得出宠物商店管理系统的E\_R图。



**二、创建表空间**

在现有数据库ORCL或其它数据库中创建表空间，表空间信息和表空间对应数据文件见表10和表11。

表10 表空间分配

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表名 | 所属管理 | 对应表空间 | 初始大小 | 增长方式 |
| 商品种类 | chiarman | Pet\_shop | 200M | 自动 |
| 职位表 | chiarman | Pet\_shop | 200M | 自动 |
| 薪酬表 | chiarman | Pet\_shop | 200M | 自动 |
| 员工表 | chiarman | Pet\_shop | 200M | 自动 |
| 客户表 | Saleuser | Pet\_shop | 200M | 自动 |
| 订单表 | Pet\_shop | 200M | 自动 |
| 宠物商品表 | Pet\_shop | 200M | 自动 |
| 宠物表 | Pet\_shop | 200M | 自动 |
| 仓库表 | Stockuser | tbs\_bio\_foo | 200M | 自动 |
| tbs\_infor\_mati | 200M | 自动 |
| tbs\_art\_fash\_busi | 200M | 自动 |

表11 表空间与数据文件的对应关系

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表空间 | 类型 | 对应文件 | 文件大小 | 增长幅度 | 增长方式 |
| Pet\_shop | 永久/联机 | Pet\_shop1.dbf | 200MB | 10% | 自动 |
| Pet\_shop2.dbf | 200MB | 10% | 自动 |
| normal\_clerck | 永久/联机 | normal \_clerck1.dbf | 200MB | 10% | 自动 |
| normal \_clerck 2.dbf | 200MB | 10% | 自动 |
| pet\_stock | 永久/联机 | pet\_stock 1.dbf | 200MB | 10% | 自动 |
| pet\_stock 2.dbf | 200MB | 10% | 自动 |
| good\_stock | 永久/联机 | good\_stock 1.dbf | 200MB | 10% | 自动 |
| good\_stock 2.dbf | 200MB | 10% | 自动 |
| normal\_stock | 永久/联机 | normal\_stock 1.dbf | 200MB | 10% | 自动 |
| normal\_stock 2.dbf | 200MB | 10% | 自动 |

1. 创建pet\_shop表空间

（1）创建pet\_shop表空间，对应文件为pet\_shop1.dbf

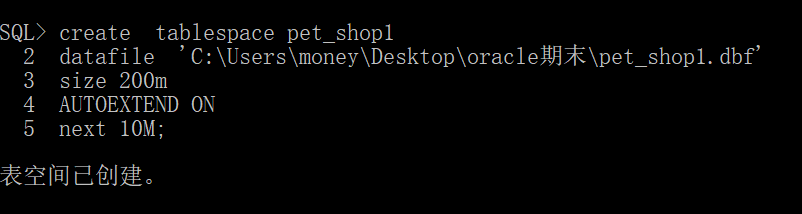
create tablespace pet\_shop1

datafile 'C:\Users\money\Desktop\oracle期末\pet\_shop1.dbf'

size 200m

AUTOEXTEND ON

next 10M;



（2）为pet\_shop表空间添加对应数据文件pet\_shop2.dbf

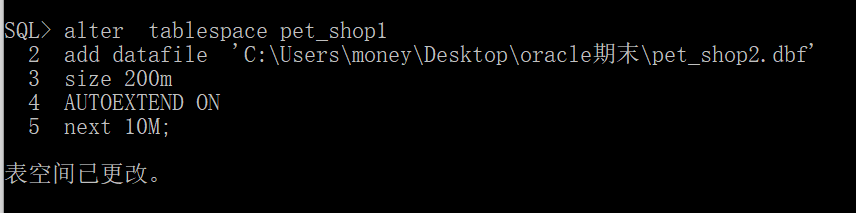
alter tablespace pet\_shop1

add datafile 'C:\Users\money\Desktop\oracle期末\pet\_shop2.dbf'

size 200m

AUTOEXTEND ON

next 10M;



1. 创建pet\_stock表空间

（1）创建pet\_stock表空间，对应数据文件为pet\_stock1.dbf

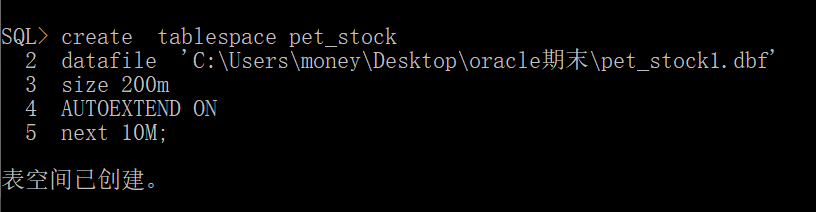
create tablespace pet\_stock

datafile 'C:\Users\money\Desktop\oracle期末\pet\_stock1.dbf'

size 200m

AUTOEXTEND ON

next 10M;



（2）为pet\_stock表空间添加对应数据文件pet\_stock2.dbf

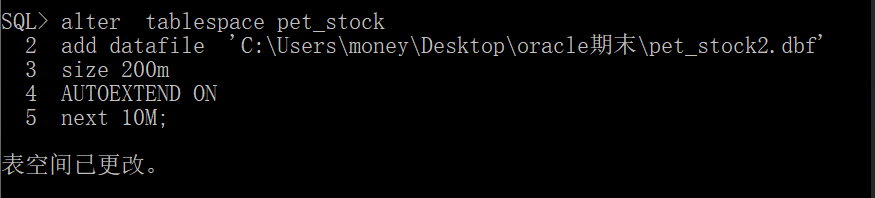
alter tablespace pet\_stock

add datafile 'C:\Users\money\Desktop\oracle期末\pet\_stock2.dbf'

size 200m

AUTOEXTEND ON

next 10M;



1. 创建good\_stock表空间

（1）创建good\_stock表空间，对应数据文件为good\_stock1.dbf

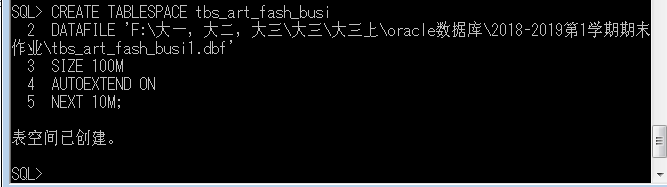
create tablespace good\_stock

datafile 'C:\Users\money\Desktop\oracle期末\good\_stock1.dbf'

size 200m

AUTOEXTEND ON

next 10M;



（2）为good\_stock表空间添加对应数据文件good\_stock2.dbf

alter tablespace good\_stock

add datafile 'C:\Users\money\Desktop\oracle期末\good\_stock2.dbf'

size 200m

AUTOEXTEND ON

next 10M;

**4、** 创建normal\_stock表空间

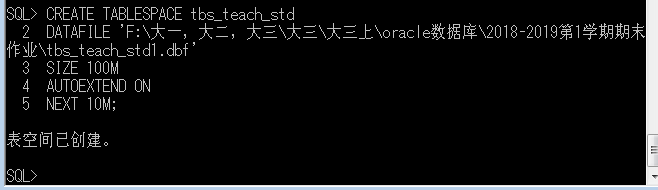
（1）创建normal\_stock表空间，对应数据文件为normal\_stock1.dbf

create tablespace normal\_stock

datafile 'C:\Users\money\Desktop\oracle期末\normal\_stock1.dbf'

size 200m

AUTOEXTEND ON

next 10M;

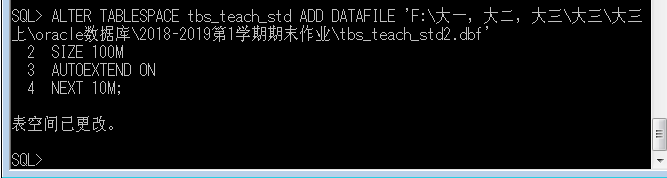
（2）为normal\_stock表空间添加对应数据文件normal\_stock2.dbf

alter tablespace normal\_stock

add datafile 'C:\Users\money\Desktop\oracle期末\normal\_stock2.dbf'

size 200m

AUTOEXTEND ON

next 10M;

**5、** 创建normal\_clerk表空间

（1）创建normal\_clerk表空间，对应数据文件为normal\_clerk1.dbf

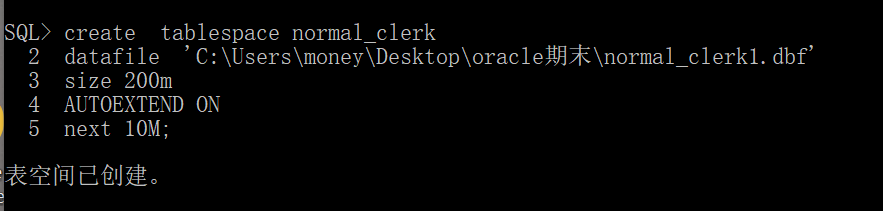
create tablespace normal\_clerk

datafile 'C:\Users\money\Desktop\oracle期末\normal\_clerk1.dbf'

size 200m

AUTOEXTEND ON

next 10M;



（2）为normal\_stock表空间添加对应数据文件normal\_stock2.dbf

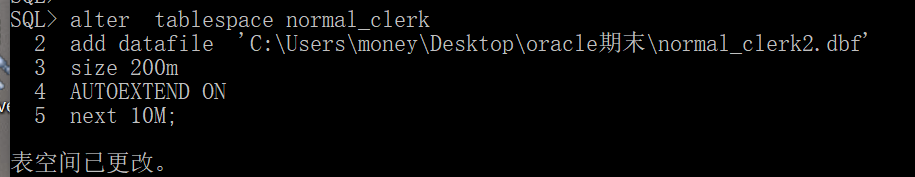
alter tablespace normal\_clerk

add datafile 'C:\Users\money\Desktop\oracle期末\normal\_clerk2.dbf'

size 200m

AUTOEXTEND ON

next 10M;



**三、创建数据库用户。**

学生成绩管理系统中的用户有4个：

（1）经理级用户chairman：拥有所有表及其全部权限。可查看库存、订单、利润及承担发放工资的任务；可更新商店的库存等部分信息，根据店员的售货情况给予奖金，也可辞退员工或辞退某员工。打个比方，某店员在开店以来从未售出一件商品，chairman可把该店员从店铺中辞退（delete），然后更新该店员的信息，找人替代他的工作。

（2）负责销售店员用户saleuser：可对顾客表、订单表进行查找、修改、添加等，当某顾客购买某样商品时，该员工就可在订单表、顾客表添加或修改信息。

（3）负责管理仓库的用户：可对仓库进行查找、修改、添加等，当仓库某样物品缺货是，仓库管理员可修改库存量，也可添加新进货的商品。

（4）商店售后用户feedbackuser：对售后表具有增删改查的权限，当顾客商品有问题时进行记录并解决。

以上全部用户均有执行存储过程、函数、包等数据库对象的权限。用户表空间和表之间所属关系如图所示。

图1 用户、表空间和表之间的关系

表空间、用户和表的对应关系如表12所示。

表12 表空间、用户及表的对应关系

|  |  |  |  |
| --- | --- | --- | --- |
| 数据库用户 | 拥有的表 | 表的默认表空间 | 程序对象 |
| Chairman | varity  post  wage  clerk | Pet\_shop1 | 触发器、存储过程、函数 |
| feedbackuser | feedback | Normal\_clerk | 部分应用程序对象 |
| saleuser | Pets  Goods  Orderlist  client | Normal\_clerk |
| stockuser | repertory(分区表) | Pet\_stock |
| Goods\_stock |
| Normal\_stock |

用户与表的权限关系矩阵如表13所示。

表13 用户与表的权限关系矩阵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据库用户表数据 | Chairman | feedbackuser | saleuser | stockuser |
| varity | DML |  |  | S |
| client | DML |  | SUI |  |
| post | DML | S | S | S |
| wage | DML | S | S | S |
| clerk | DML | S | S | S |
| Pets | S | S | DML |  |
| Goods | S | S | DML |  |
| Orderlist | S | S | DML |  |
| Feedback | S | DML | S |  |
| repertory | SU | S | S | DML |

表13中的权限缩写如表14所示。

表14 表级权限列表

|  |  |  |
| --- | --- | --- |
| 表级权限 | 含义 | 缩写 |
| ALTER | 修改表结构 | A |
| DELETE | 删除记录 | D |
| INDEX | 索引 | X |
| INSERT | 插入数据 | I |
| REFERENCES | 关联 | R |
| SELECT | 查询 | S |
| UPDATE | 更新 | U |
| DML | 增删改查 | DML |

请按照以上信息，给出创建5个用户的语句及其授权语句，授权语句请说明是哪个用户给哪个用户授权。

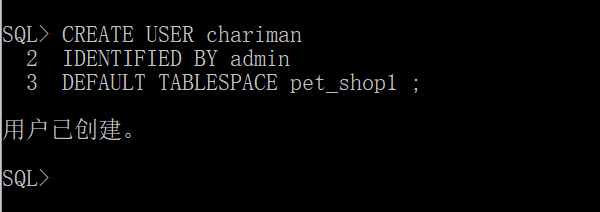
（1）chairman用户

① 创建chairman用户

CREATE USER chairman

IDENTIFIED BY admin

DEFAULT TABLESPACE pet\_shop1 ;

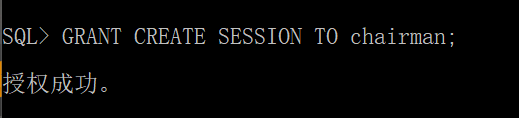


② 为用户chairman授权

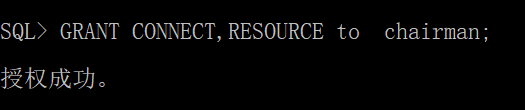
使用system用户授权

为用户chiarman授予连接数据库的CREATE SESSION 权限

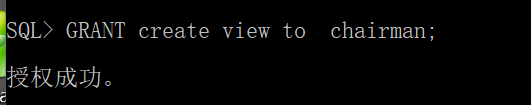
GRANT CREATE SESSION TO chairman;



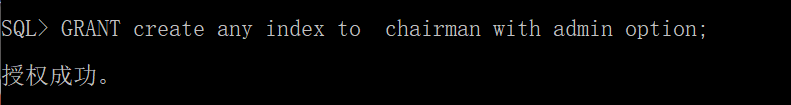
GRANT CONNECT,RESOURCE to chairman;



GRANT create view to chairman;



GRANT create any index to chairman with admin option;



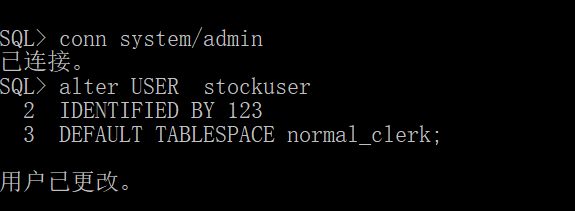
（2）stockuser用户

① 创建stockuser用户

CREATE USER stockuser

IDENTIFIED BY 123

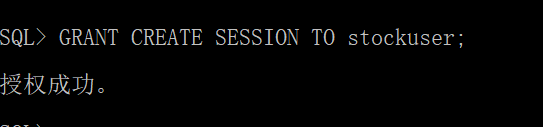
DEFAULT TABLESPACE normal\_clerk;



② 为用户stockuser授权

使用system用户授权

GRANT CREATE SESSION TO stockuser;



使用chiarman用户授权

登陆chiarman账号：

GRANT SELECT, UPDATE,INSERT,DELETE ON

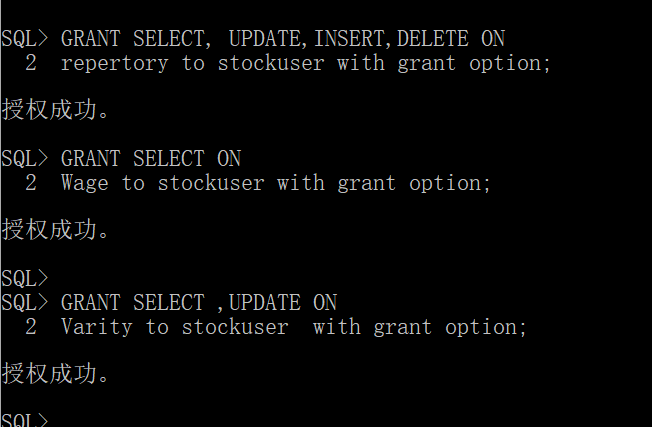
repertory to stockuser with grant option;

GRANT SELECT ON

Wage to stockuser with grant option;

GRANT SELECT ,UPDATE ON

Varity to stockuser with grant option;

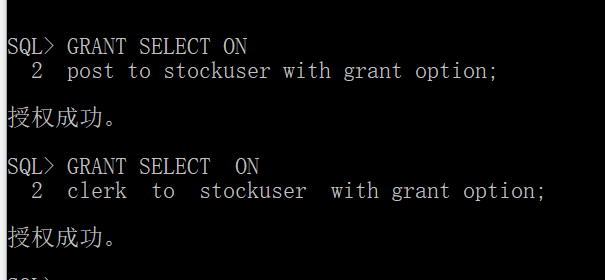


GRANT SELECT ON

post to stockuser with grant option;

GRANT SELECT ON

clerk to stockuser with grant option;



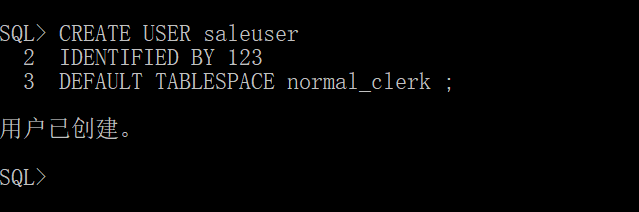
（3）saleuser用户

① 创建saleuser用户

CREATE USER saleuser

IDENTIFIED BY 123

DEFAULT TABLESPACE normal\_clerk ;



② 为用户saleuser授权

使用chiarman用户授权

登陆chiarman账号：

GRANT SELECT ,UPDATE ,INSERT ON

client to saleuser with grant option;

GRANT SELECT ,UPDATE ,INSERT ON

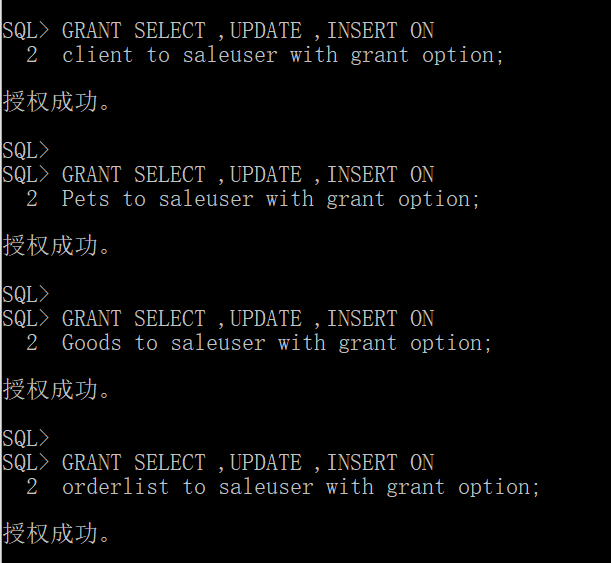
Pets to saleuser with grant option;

GRANT SELECT ,UPDATE ,INSERT ON

Goods to saleuser with grant option;

GRANT SELECT ,UPDATE ,INSERT ON

orderlist to saleuser with grant option;



GRANT SELECT ON

Post to saleuser;

GRANT SELECT ON

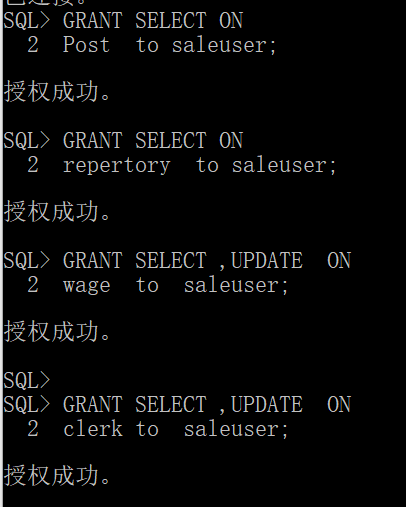
repertory to saleuser;

GRANT SELECT ,UPDATE ON

wage to saleuser;

GRANT SELECT ,UPDATE ON

clerk to saleuser;



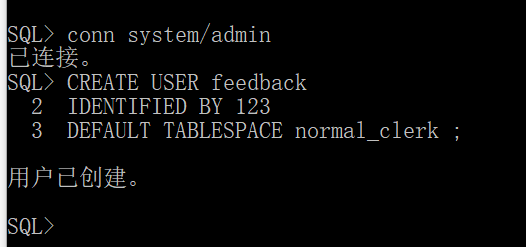
（4）feedback用户

① 创建feedback用户

CREATE USER feedback

IDENTIFIED BY 123

DEFAULT TABLESPACE normal\_clerk ;



② 为用户feedback授权

使用chairman用户授权

GRANT SELECT ,UPDATE ,INSERT ON

Feedback to feedback with grant option;

GRANT SELECT ON

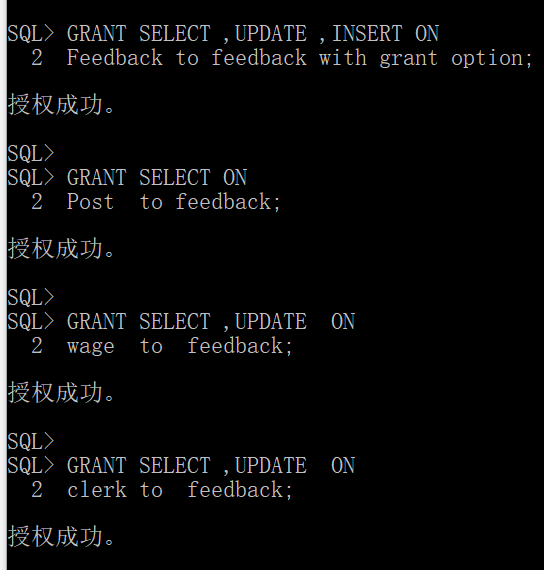
Post to feedback;

GRANT SELECT ,UPDATE ON

wage to feedback;

GRANT SELECT ,UPDATE ON

clerk to feedback;



**四、创建表**

（1）以chairman登录，创建表10个表：feedback(反馈),pets（宠物）, goods（一般商品） ,orderlist（总订单）， repertory（库存） ,client （客户）,varity（种类） , clerk （员工）,wage （工资）,post(职位)并向表插入数据。

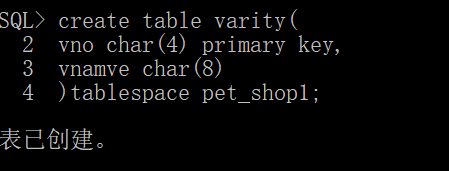
1、建varity（种类）表

create table varity(

vno char(4) primary key,

vnamve char(8)

)tablespace pet\_shop1;



插入数据：

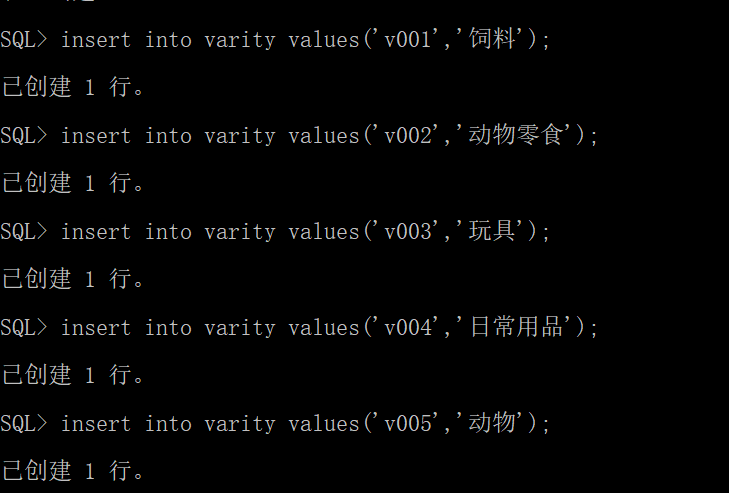
insert into varity values('v001','饲料');

insert into varity values('v002','动物零食');

insert into varity values('v003','玩具');

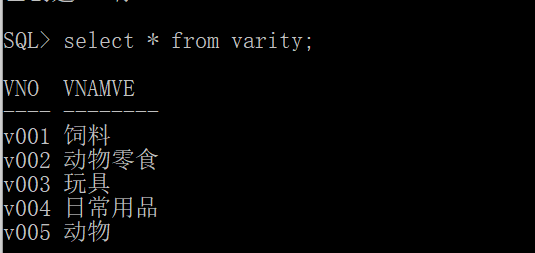
insert into varity values('v004','日常用品');

insert into varity values('v005','动物');



查询数据：

Select \* from varity;



2、建client（客户）表

create table client(

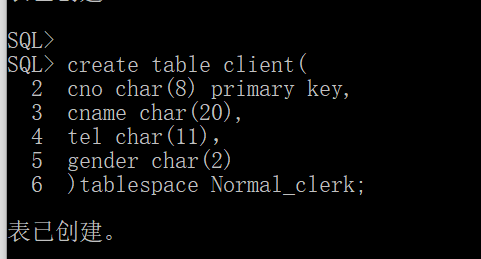
cno char(8) primary key,

cname char(20),

tel char(11)，

gender char(2)

)tablespace Normal\_clerk;



插入数据：

-----------------------------------------

insert into client values('c001','安安','12345678901','0');

insert into client values('c002','包包','12333333333','1');

insert into client values('c003','常常','13443222145','0');

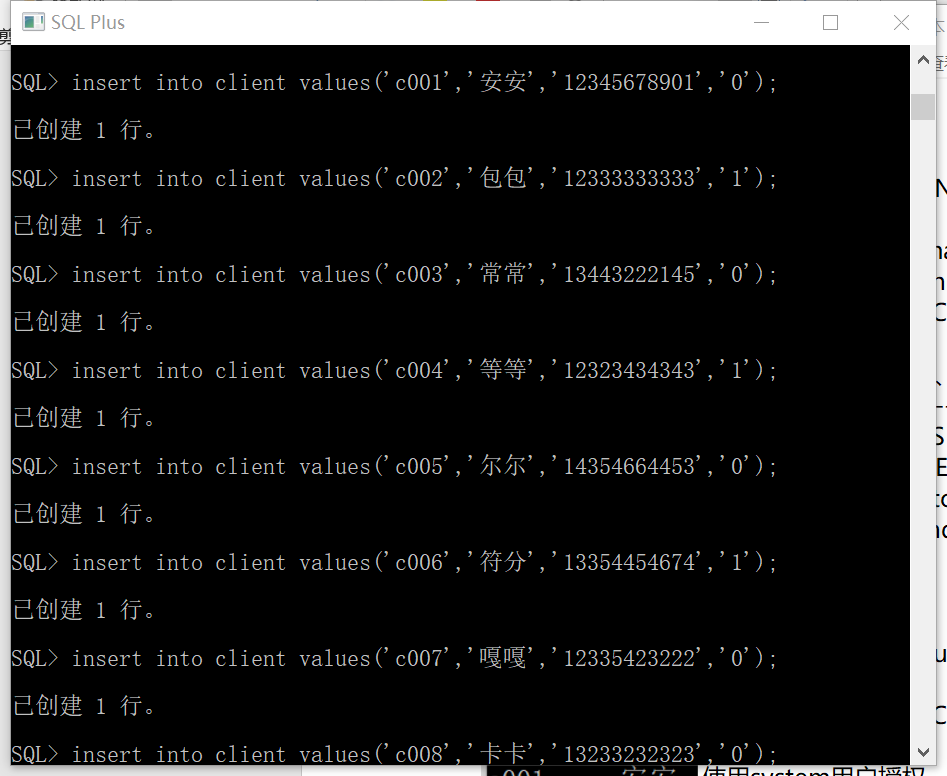
insert into client values('c004','等等','12323434343','1');

insert into client values('c005','尔尔','14354664453','0');

insert into client values('c006','符分','13354454674','1');

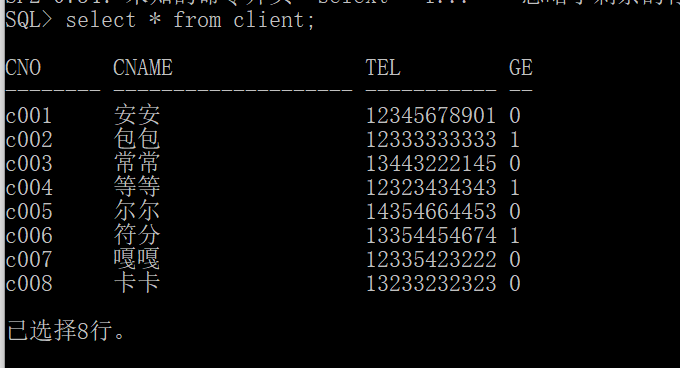
insert into client values('c007','嘎嘎','12335423222','0');

insert into client values('c008','卡卡','13233232323','0');



查询数据：

Select \* from client;



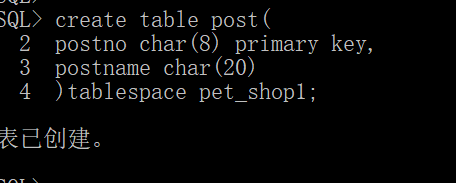
3、建post（职位）表

create table post(

postno char(8) primary key,

posdtname char(20)

)tablespace pet\_shop1;



插入数据：

-----------------------------------------

insert into post values('ps01','经理');

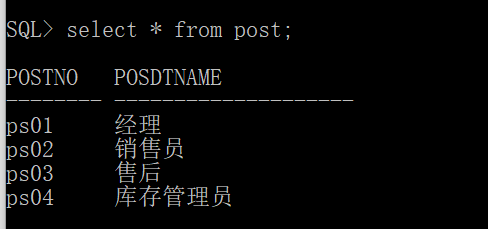
insert into post values('ps02','销售员');

insert into post values('ps03','售后');

insert into post values('ps04','库存管理员');

查询数据：

Select \* from post;



4、建wage(工资)表

create table wage(

wno char(8) primary key,

basepay char(10),

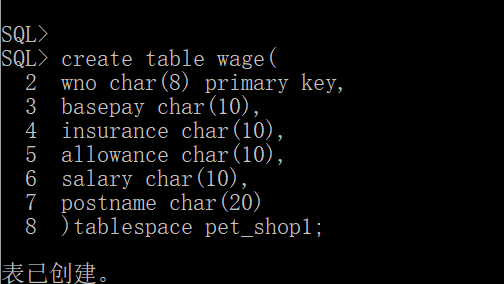
insurance char(10),

allowance char(10),

salary char(10),

postname char(20)

)tablespace pet\_shop1;



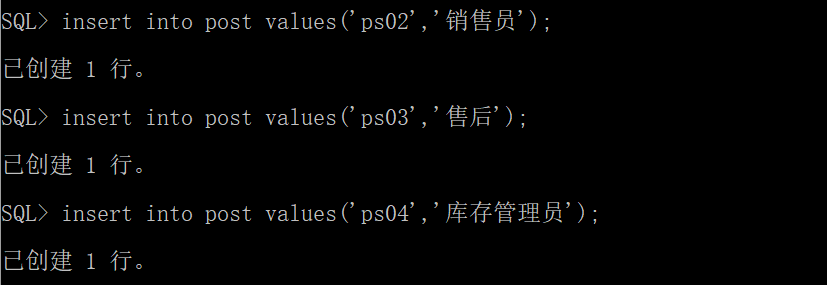
插入数据：

insert into wage values('w001','5200','1000','100','4300','门店经理');

insert into wage values('w002','4200','1000','50','3150','售货员');

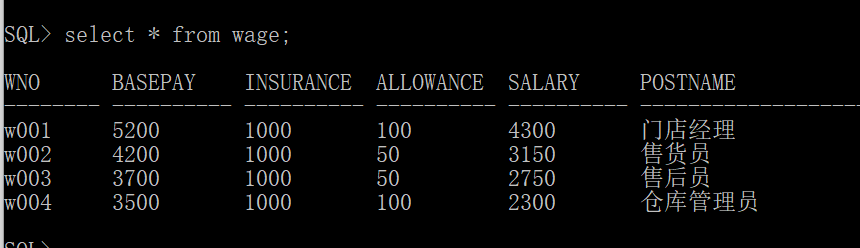
insert into wage values('w003','3700','1000','50','2750','售后员');

insert into wage values('w004','3200','1000','100','2300','仓库管理员');



查询数据：

Select \* from wage;



5、建clerk(员工)表

create table clerk(

clerkno char(8) primary key,

clerkname char(20),

tel char(11),

postno char(8),

wno char(8),

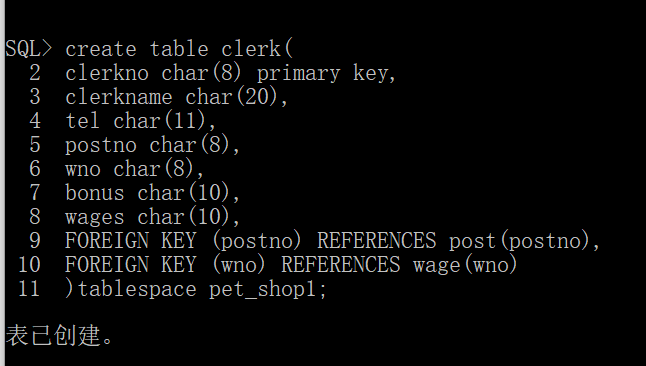
bonus char(10),

wages char(10),

FOREIGN KEY (postno) REFERENCES post(postno),

FOREIGN KEY (wno) REFERENCES wage(wno)

)tablespace pet\_shop1;



插入数据：

-----------------------------------------

insert into clerk values('ck01','拉拉','12637354751','ps01','w001','0','4300');

insert into clerk values('ck02','九九','12637354753','ps02','w002','0','3150');

insert into clerk values('ck03','一一','12637354754','ps02','w002','0','3150');

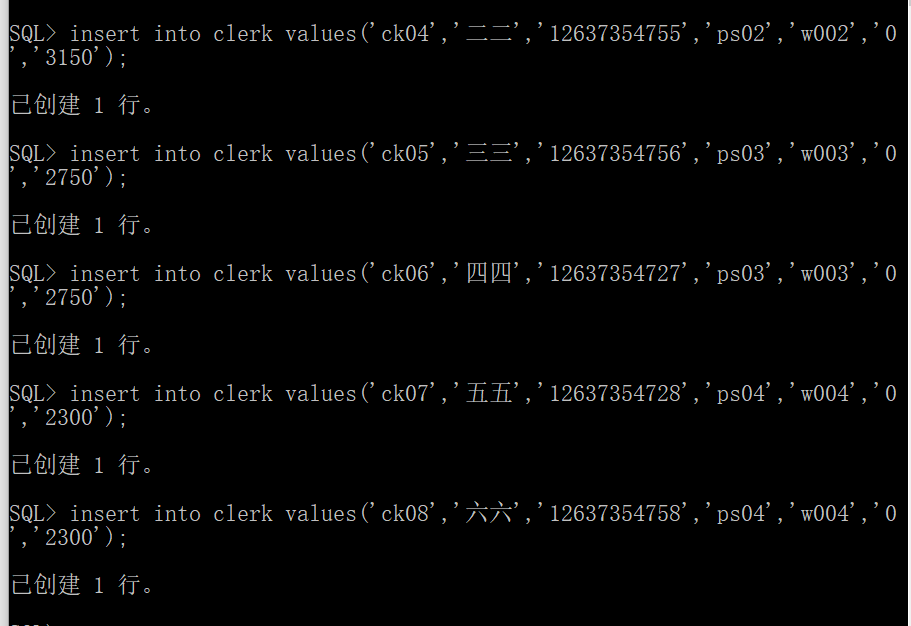
insert into clerk values('ck04','二二','12637354755','ps02','w002','0','3150');

insert into clerk values('ck05','三三','12637354756','ps03','w003','0','2750');

insert into clerk values('ck06','四四','12637354727','ps03','w003','0','2750');

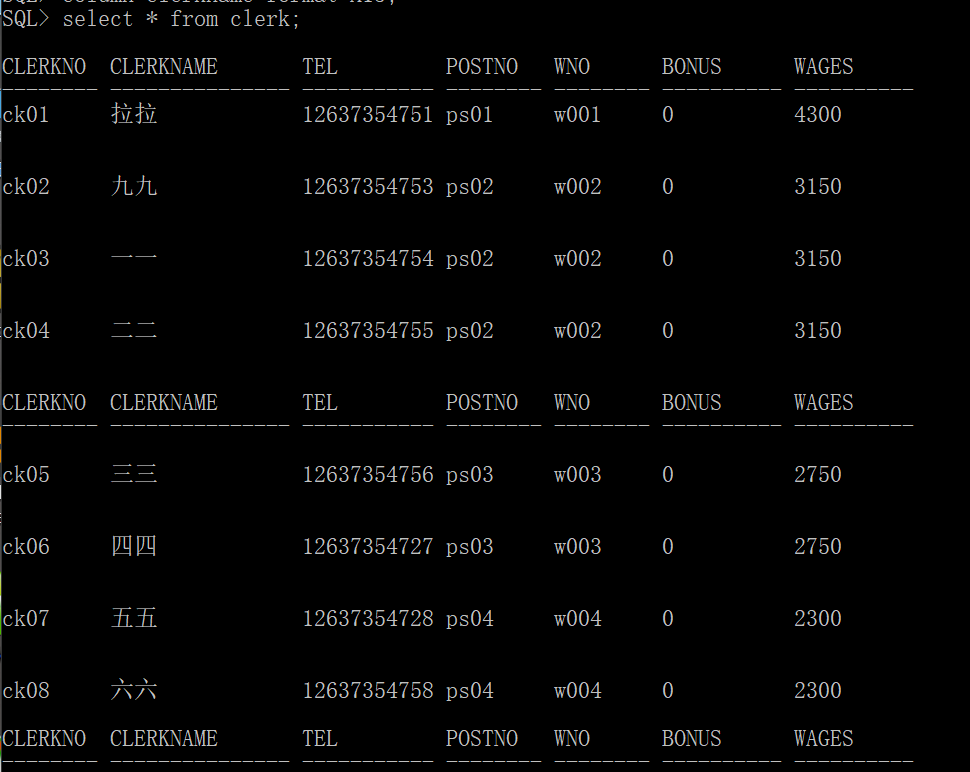
insert into clerk values('ck07','五五','12637354728','ps04','w004','0','2300');

insert into clerk values('ck08','六六','12637354758','ps04','w004','0','2300');



查询数据：

Select \* from clerk;



6、建orderlist (订单)表

create table orderlist（

odno char(8) primary key,

cno char(8),

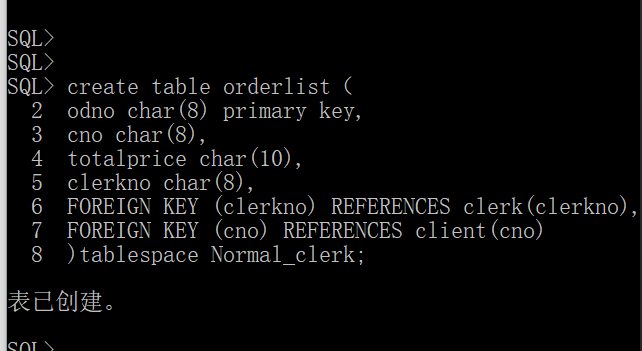
totalprice char(10),

clerkno char(4),

FOREIGN KEY (clerkno) REFERENCES clerk(clerkno),

FOREIGN KEY (cno) REFERENCES client(cno)

)tablespace Normal\_clerk;



插入数据：

insert into orderlist values('o001','c001','0','ck03');

insert into orderlist values('o002','c002','0','ck02');

insert into orderlist values('o003','c003','0','ck03');

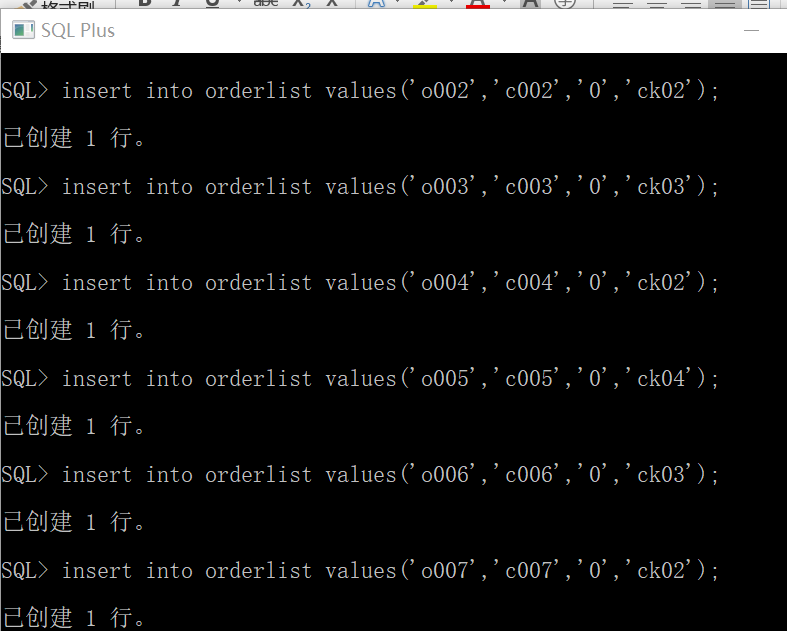
insert into orderlist values('o004','c004','0','ck02');

insert into orderlist values('o005','c005','0','ck04');

insert into orderlist values('o006','c006','0','ck03');

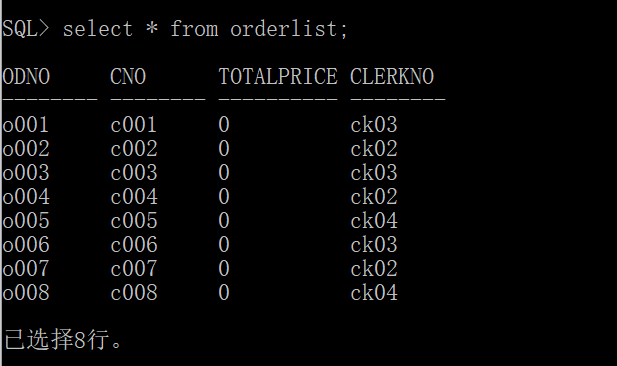
insert into orderlist values('o007','c007','0','ck02');

insert into orderlist values('o008','c008','0','ck04');



查询数据：

Select \* from orderlist;



7、建repertory（仓库）表

create table repertory（

rno char（8) primary key,

rname char(10),

capital char(10),

saleprice char(10),

numin char(10),

numout char(10),

numnow char(10),

clerkno char(8),

vno char(8),

FOREIGN KEY (vno) REFERENCES varity(vno),

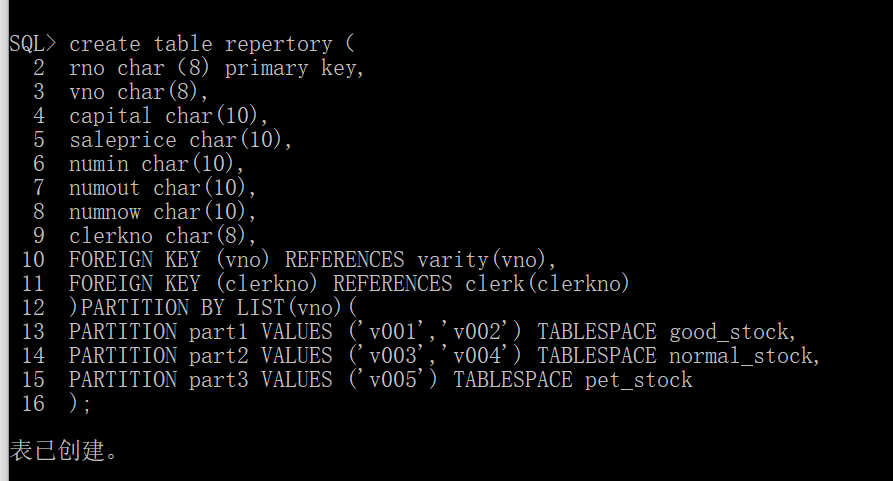
FOREIGN KEY (clerkno) REFERENCES clerk(clerkno)

)PARTITION BY LIST(vno)(

PARTITION part1 VALUES ('v001','v002') TABLESPACE good\_stock,

PARTITION part2 VALUES ('v003','v004') TABLESPACE normal\_stock,

PARTITION part3 VALUES ('v005') TABLESPACE pet\_stock

); 

插入数据：

insert into repertory values('r001','苗苗牌猫粮1','20','85','100','35','65','ck07','v001');

insert into repertory values('r002','苗苗牌猫粮2','15','60','100','70’,'30','ck07','v001');

insert into repertory values('r003','苗苗牌狗粮1','40','100','100','28',72','ck07','v001');

insert into repertory values('r004','苗苗牌狗粮2','45','120','100','65','35','ck07','v001');

insert into repertory values('r005','逗猫棒','15','55','100','40','60','ck07','v006');

……………………………

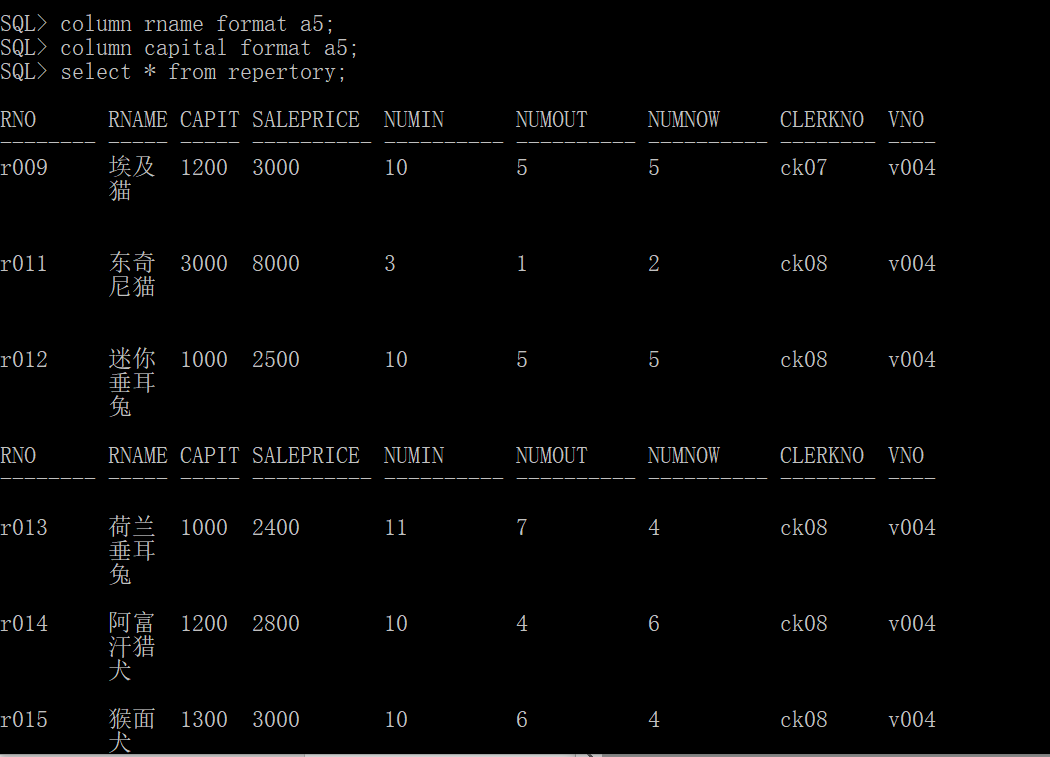
insert into repertory values('r014','阿富汗猎犬','1200','2800','10','4','6','ck08','v004');

insert into repertory values('r015','猴面犬','1300','3000','10','6','4','ck08','v004');(共15行)



查询数据：

Select \* from repertory;



8、建立goods表

create table goods (

gno char(8) primary key,

odno char(8),

rno char(8),

rname char(30),

saleprice char(10),

num char(10),

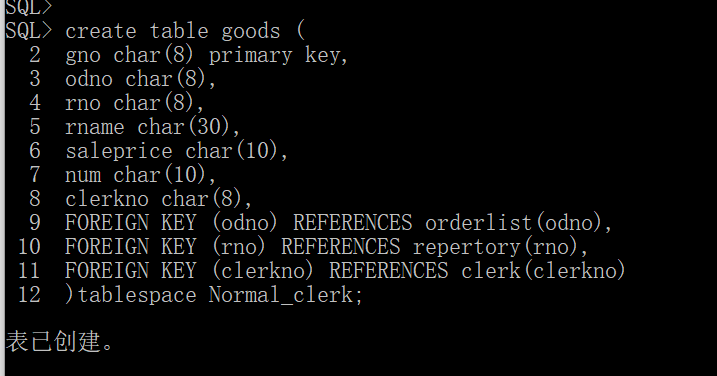
clerkno char(8),

FOREIGN KEY (odno) REFERENCES orderlist(odno),

FOREIGN KEY (rno) REFERENCES repertory(rno),

FOREIGN KEY (clerkno) REFERENCES clerk(clerkno)

)tablespace normal\_clerk;



插入数据：

insert into goods values('gd01','o001','r001', '苗苗牌猫粮1','85','2','ck02');

insert into goods values('gd02','o001','r006','苗苗牌猫粮2','60','1','ck02');

insert into goods values('gd03','o002','r002','苗苗牌猫粮2','60','3','ck04');

insert into goods values('gd04','o003','r005','逗猫棒','55','1','ck04');

insert into goods values('gd05','o003','r007','猫狗牌沐浴露1','155','1','ck04');

insert into goods values('gd06','o004','r008','猫狗牌沐浴露2','205','2','ck02');

insert into goods values('gd07','o005','r002','苗苗牌猫粮2','50','5','ck03');

insert into goods values('gd08','o005','r006','逗狗球','55','5','ck03');

insert into goods values('gd09','o001','r007','猫狗牌沐浴露2','205','2','ck02');

insert into goods values('gd10','o006','r002','苗苗牌猫粮2','50','5','ck03');

insert into goods values('gd11','o007','r006','逗猫棒','55','5','ck03');

insert into goods values('gd12','o008','r007','猫狗牌沐浴露1','80','2','ck02');



Select \* from goods;



9、建立pets表

create table pets(

pno char(8) primary key,

rno char（8)，

odno char(8)，

vname char(20),

num char(10)，

gender char(2),

age char(4),

saleprice char(10),

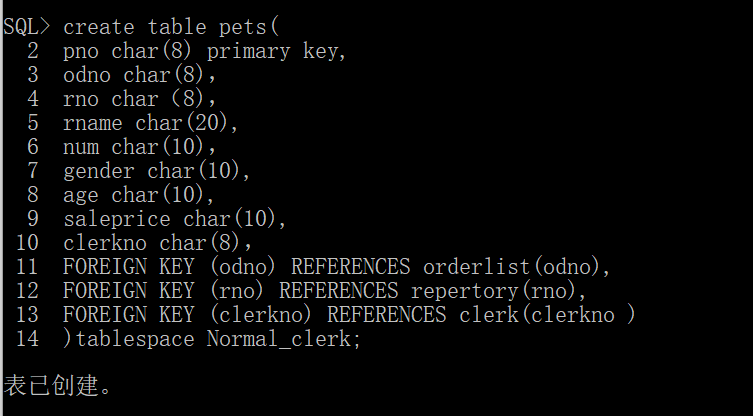
clerkno char(8)，

FOREIGN KEY (odno) REFERENCES orderlist(odno),

FOREIGN KEY (rno ) REFERENCES repertory(rno ),

FOREIGN KEY (clerkno) REFERENCES clerk(clerkno )

)tablespace normal\_clerk;



插入数据：

insert into pets values('pt01','o001','r009','迷你垂耳兔','1','雌性','四个月','2200','ck02');

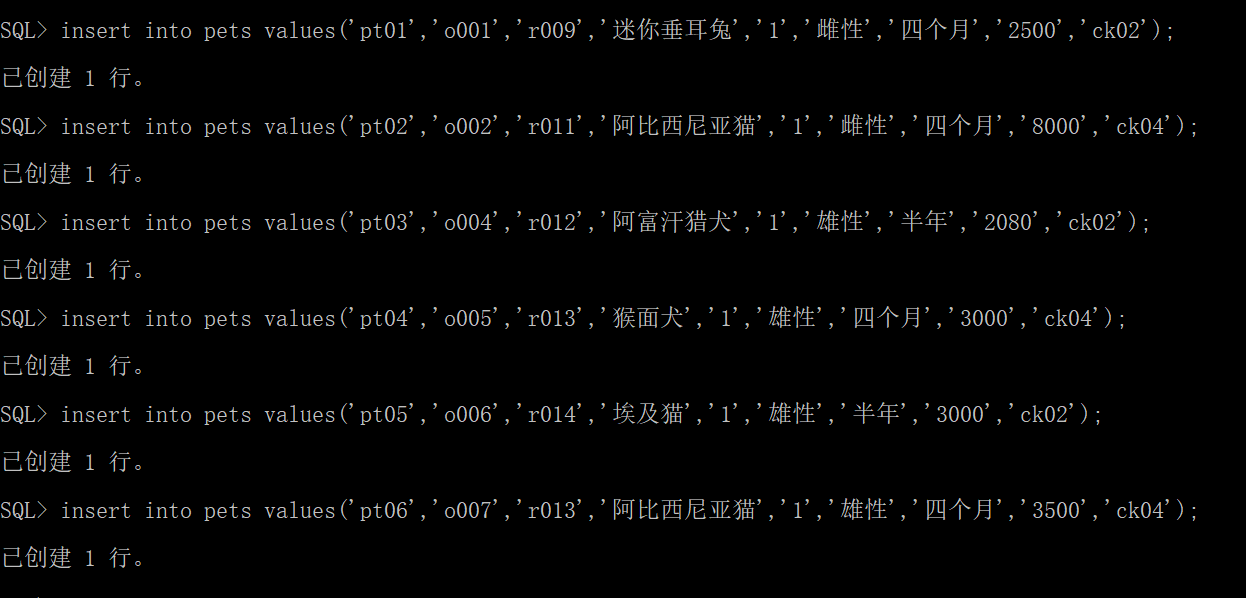
insert into pets values('pt02','o002','r011','阿比西尼亚猫','1','雌性','四个月','8000','ck04');

insert into pets values('pt03','o004','r012','阿富汗猎犬','1','雄性','半年','2080','ck02');

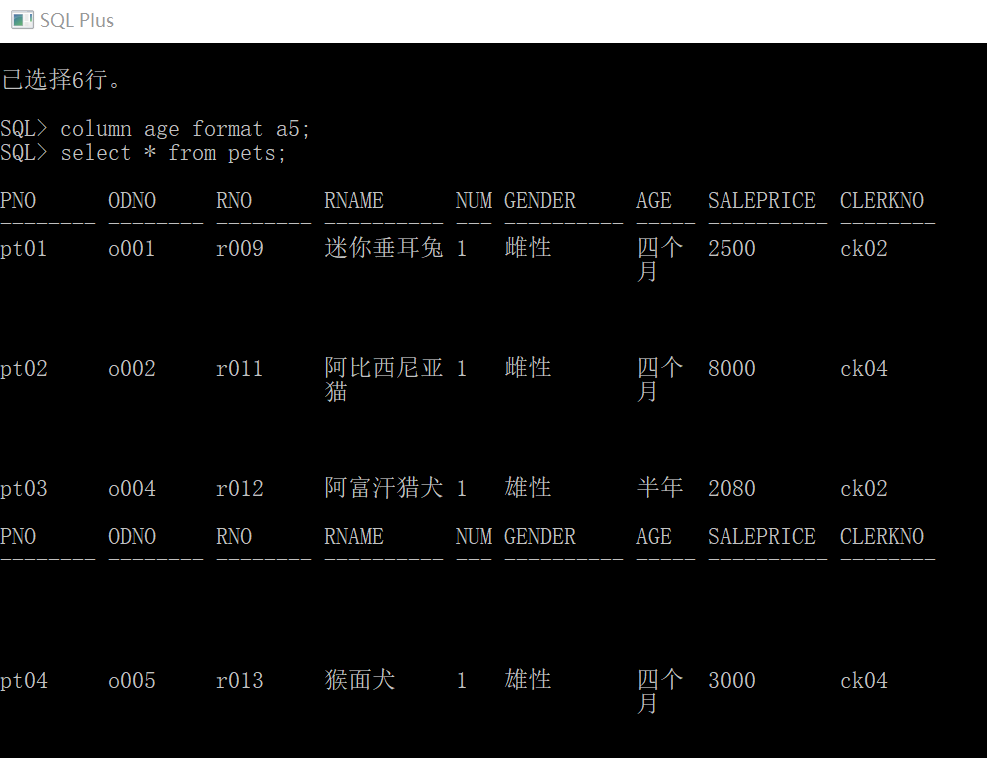
insert into pets values('pt04','o005','r013','猴面犬','1','雄性','四个月','3000','ck04');

insert into pets values('pt05','o006','r014','埃及猫','1','雄性','半年','3000','ck02');

insert into pets values('pt06','o007','r013','阿比西尼亚猫','1','雄性','四个月','3200','ck04');



Select \* from pets;



10、建立feedback(反馈)表

create table feedback（

cno char(8),

feedback char(50) ,

clerkno char(8),

mark char(10),

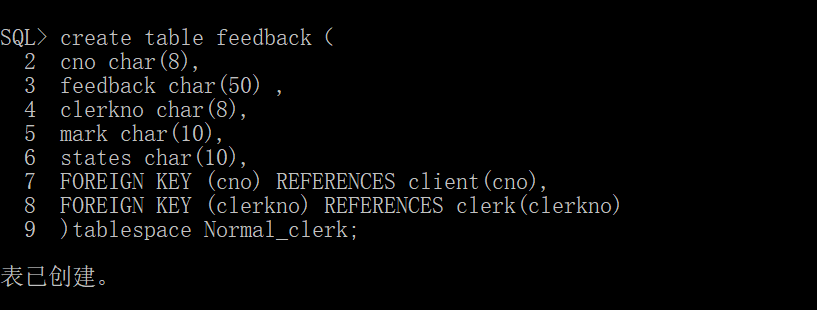
states char(10)，

solve char(20),

FOREIGN KEY (cno) REFERENCES client(cno),

FOREIGN KEY (clerkno) REFERENCES clerk(clerkno)

)tablespace normal\_clerk;



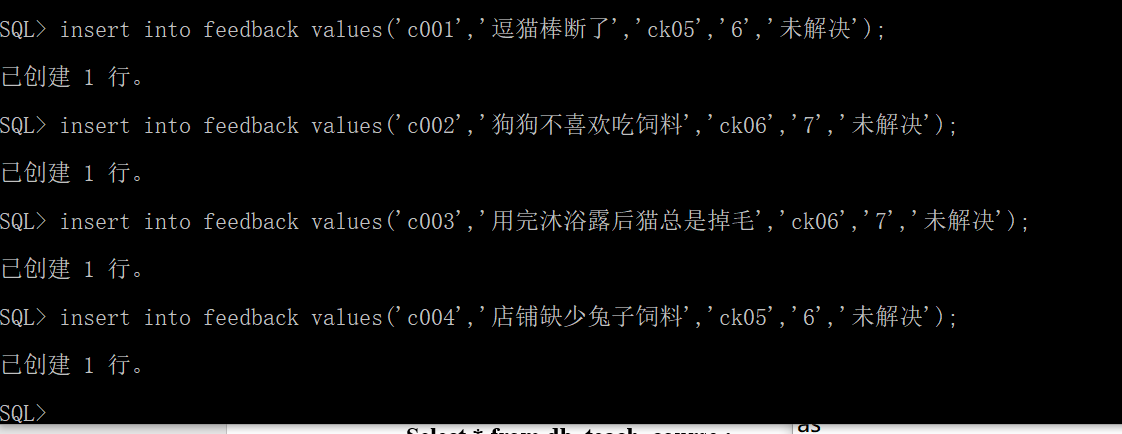
插入数据：

insert into feedback values('c001','逗猫棒断了','ck05','6','未解决');

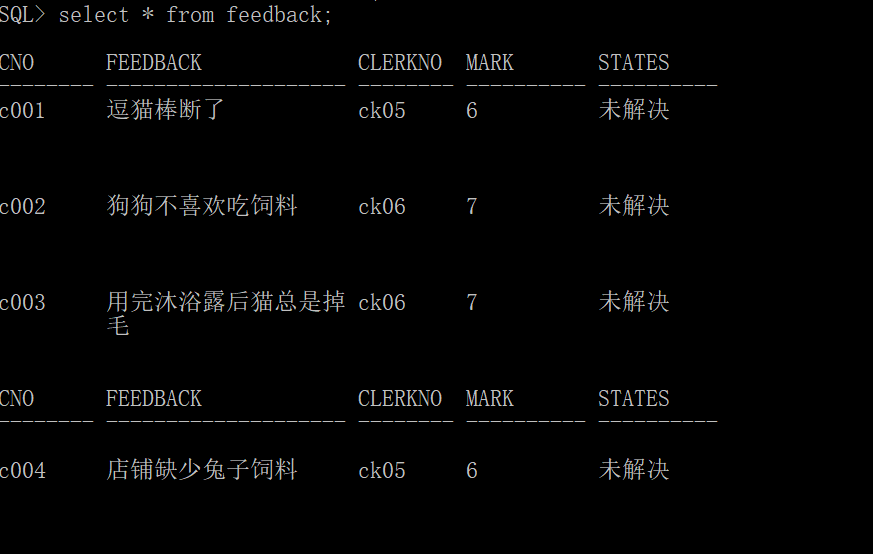
insert into feedback values('c002','狗狗不喜欢吃饲料','ck06','7','未解决');

insert into feedback values('c003','用完沐浴露后猫总是掉毛','ck06','7','未解决');

insert into feedback values('c004','店铺缺少兔子饲料','ck05','6','未解决');



Select \* from feedback;



**五、功能实现**

以chiarman登录，创建一个函数clerk\_ma11，用于返回某个员工的工资。

create function clerk\_ma11

(ck\_no varchar2)

return number as

ck\_wages clerk.wages%TYPE;

begin

select wages into ck\_wages from clerk

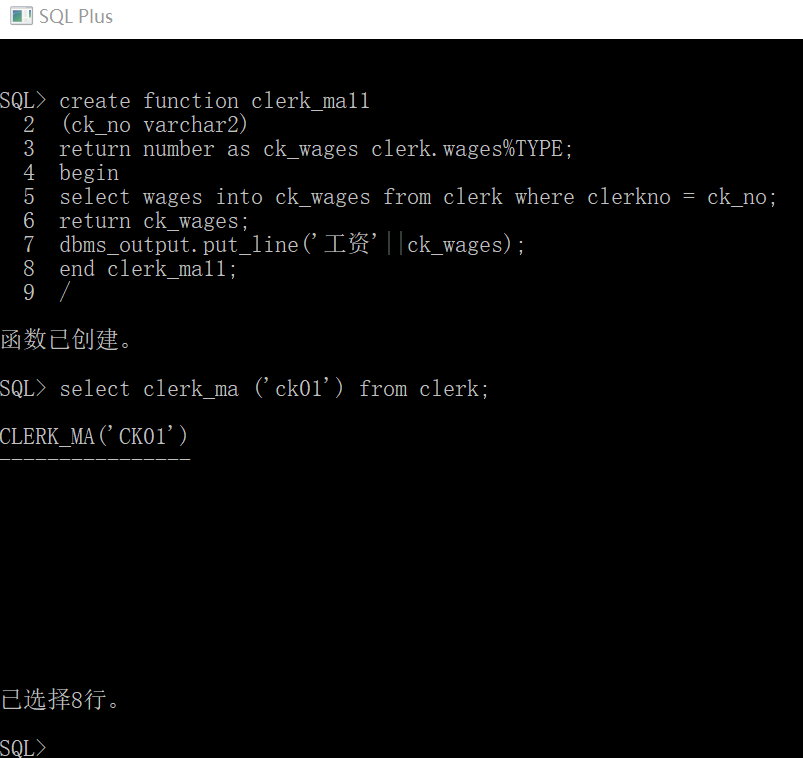
where clerkno = ck\_no;

return ck\_wages;

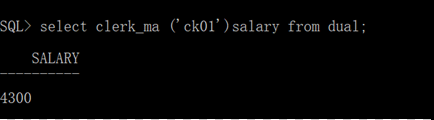
dbms\_output.put\_line('工资'||ck\_wages);

end clerk\_ma11;

/



select fun\_std\_avg\_gra('201231000078')count from dual;



1. 以chiarman登录，创建一个存储过程add\_wage，查询是否有一半程序员的工资在6000元之上,如果不到分别每次给每个店员加薪300元,至之一半店员的工资达到6000以上，给出测试结果截图。

create or replace procedure add\_wage

(base in number)

as

allPeople number;

renshu number;

addsalary number;

i number;

begin

addsalary:=0;

select count(\*) into allPeople from clerk;

select count(\*) into renshu from clerk

where wno in(

select wno

from wage

where wages>=base);

while renshu<(allPeople/2)

loop

select count(\*) into renshu from clerk

where wno in(

select wno

from wage

where wages>=base);

update wage set salary=salary+300,basepay=basepay+300;

update clerk set wages=wages+300;

addsalary:=addsalary+300;

end loop;

dbms\_output.put\_line('共加工资:'|| addsalary);

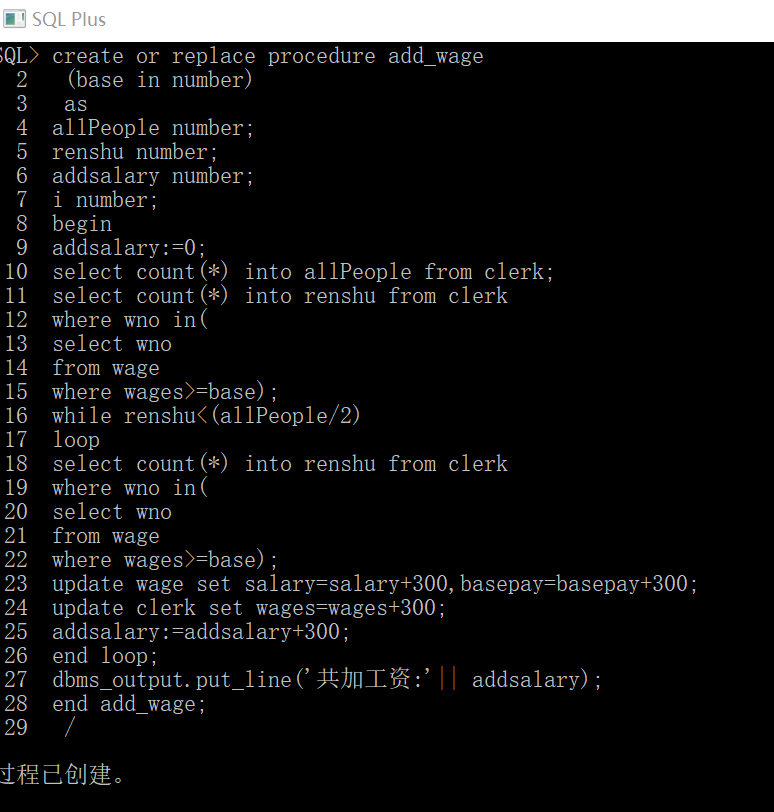
end add\_wage;

/end if;

dbms\_output.put\_line('共加工资:'|| addsalary);

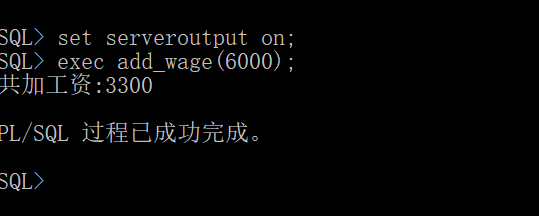
end add\_wage;

/



set serveroutput on;

exec add\_wage(6000);



1. 以chiarman登录，创建一个触发器clerks，当辞退某员工时，把该员工负责的事务找另外员工替代，并把相关信息删除

create or replace trigger clerks

before delete on clerk

for each row

begin

update orderlist

set clerkno='ck02'

where clerkno=:new.clerkno;

update repertory

set clerkno=' ck02'

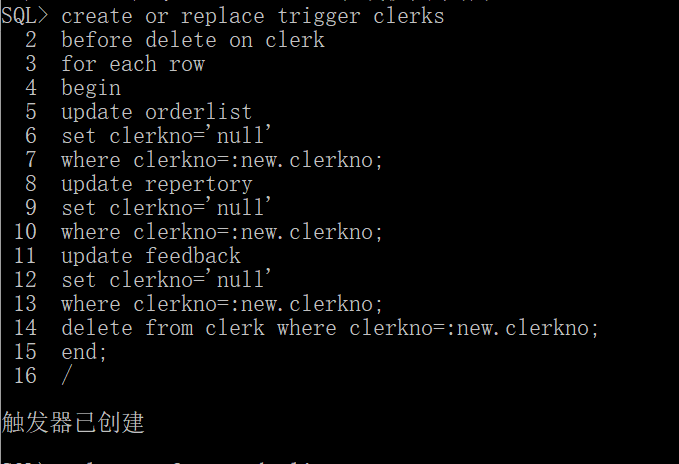
where clerkno=:new.clerkno;

update feedback

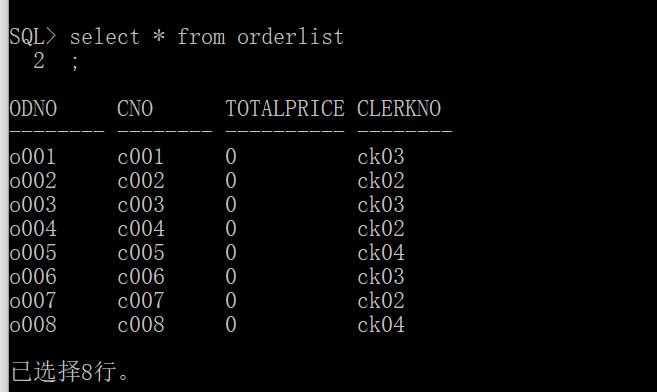
set clerkno='ck02'

where clerkno=:new.clerkno;

delete from clerk where clerkno=:new.clerkno;



Select \* from orderlist;



Delete from clerk where clerkno=’cko3’

d