**实 验 报 告**

实验名称：PL/SQL编程

实验课时：2课时

实验地点：

实验时间： 年 月 日 星期 第 周

实验目的及要求：

1. 熟练掌握PL/SQL程序设计的基本知识。
2. 熟练掌握PL/SQL中控制结构的使用。具体包括选择结构语句（IF语句和CASE语句），循环结构（四种循环结构）。
3. 熟练使用PL/SQL中系统函数。
4. 掌握PL/SQL中异常处理语句的使用
5. 掌握PL/SQL中SELECT语句和DML语句的综合运用。

实验内容：

1. 创建表departments和表employees, 并为两张表输入若干数据。

表departments结构：

字段名称 字段类型 约束

Dep\_id number 主键

Dep\_name varchar2(20)

表employees结构：

字段名称 字段类型 约束

emp\_id number 主键

emp\_name varchar2(20)

sex varchar2(20)

title varchar2(20)

wage number

idcard varchar2(12)

dep\_id number 外键

（表的数据需要根据程序要求自行添加）

1. 编写PL/SQL程序，查询5号员工工资，如果工资小于3000，则加200员工资，并提示信息“5号员工工资已更新。”，如果工资大于3000，则提示信息“5号员工工资为XXX，已达到规定标准。”
2. 编写PL/SQL程序，查询1号员工的工资，使用CASE语句输出其工资等级。工资小于等于3000，等级为“低”；工资大于3000，小于5000，等级为”中”；工资大于等于5000，等级为高；
3. 编写PL/SQL程序，查询的departments表中是否有部门号为“6”的记录，如果没有，插入新记录部门号为6，部门名称为“后勤部”。如果有，显示查询结果。

create table departments(

dep\_id number primary key,

dep\_name varchar2(20)

);

create table employees(

emp\_id number primary key,

emp\_name varchar2(20),

sex varchar2(20),

title varchar2(20),

wage number,

idcard varchar2(12),

dep\_id number,

foreign key (dep\_id) references departments(dep\_id)

);

insert into departments values (1,'财务部');

insert into departments values (2,'人力资源部');

insert into departments values (3,'经理办公室');

insert into departments values (4,'研发部');

insert into departments values (5,'市场部');

insert into employees values(1,'王一一','男', '文员', 2500,12345678901,1);

insert into employees values(2,'王二二','男', '文员', 2500,12345678901,2);

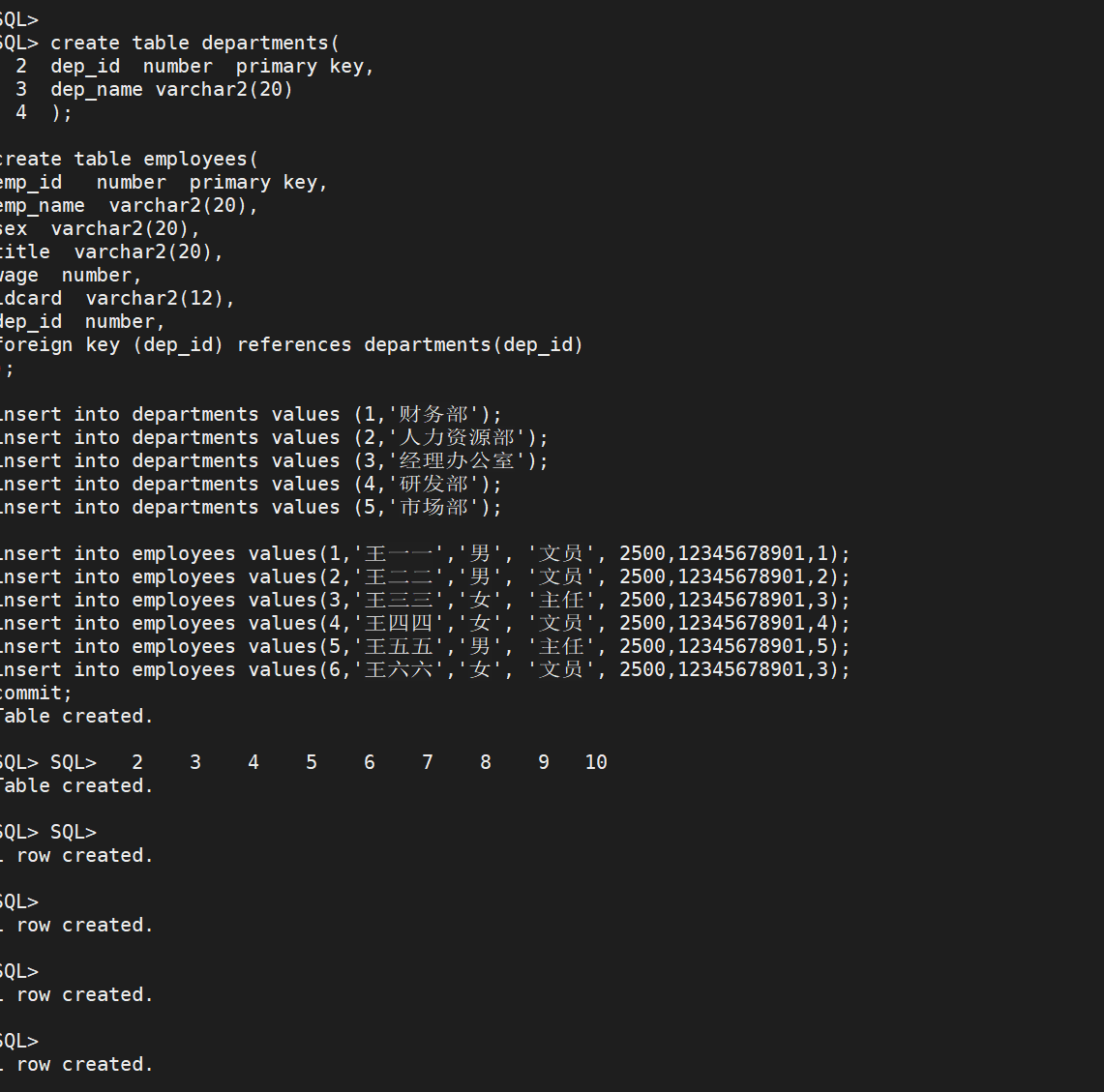
insert into employees values(3,'王三三','女', '主任', 2500,12345678901,3);

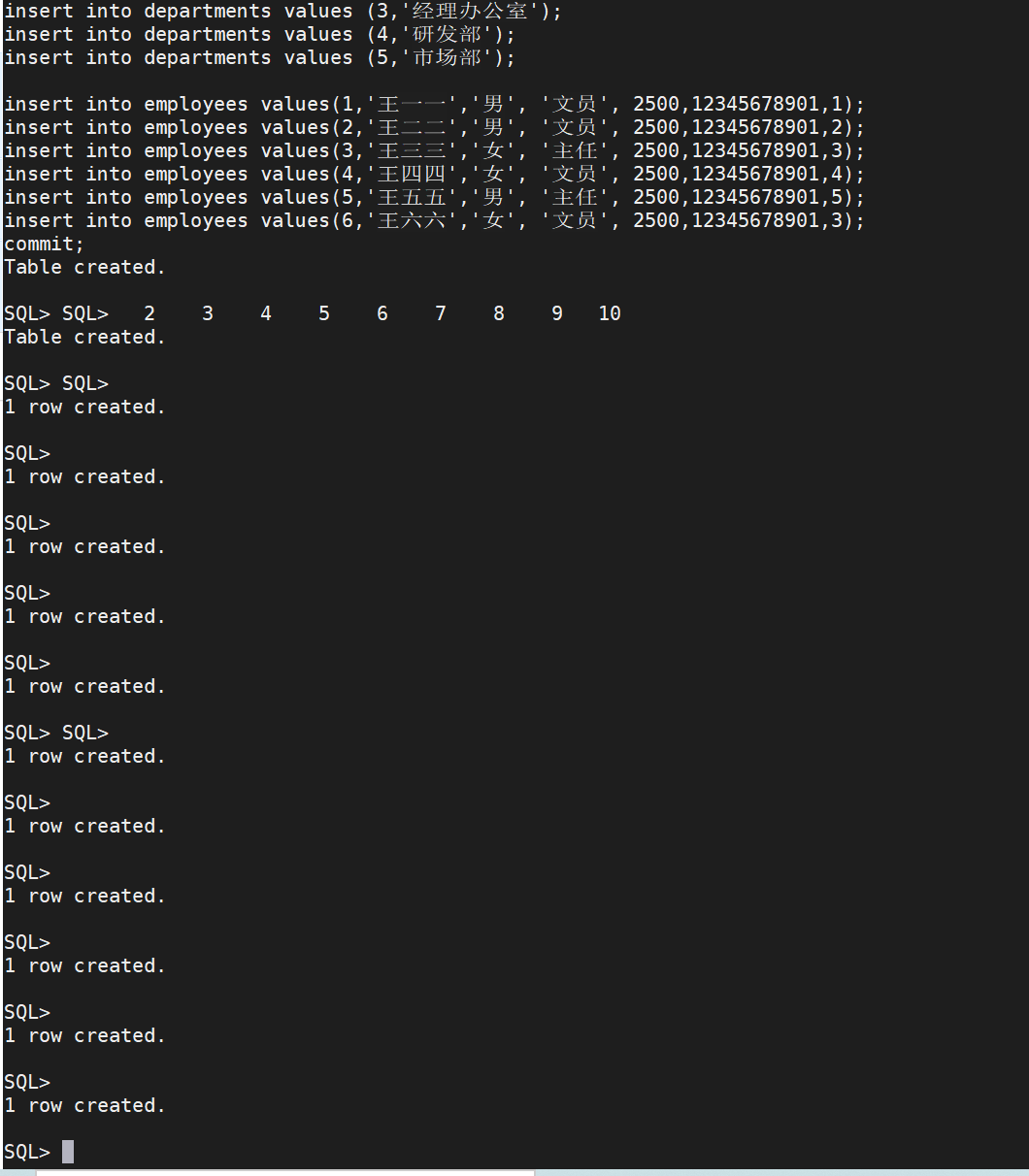
insert into employees values(4,'王四四','女', '文员', 2500,12345678901,4);

insert into employees values(5,'王五五','男', '主任', 2500,12345678901,5);

insert into employees values(6,'王六六','女', '文员', 2500,12345678901,3);

commit;





create or replace procedure pr\_find\_emp\_wage

is

ewage number;

begin

select wage into ewage from employees where emp\_id=5;

if ewage <3000 then

update employees set wage=wage+200 where emp\_id=5;

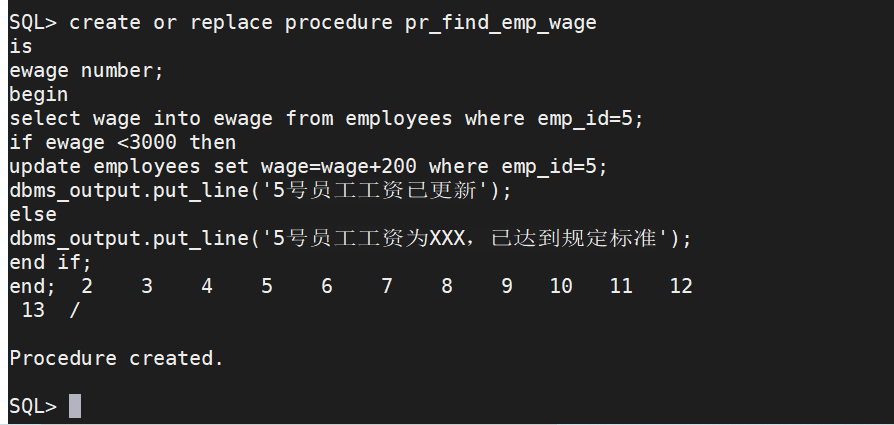
dbms\_output.put\_line('5号员工工资已更新');

else

dbms\_output.put\_line('5号员工工资为XXX，已达到规定标准');

end if;

end;



create or replace function fn\_salary\_level(e\_id number)

return varchar2

is

salary\_level varchar2(5);

begin

select case when wage<=3000 then '低'

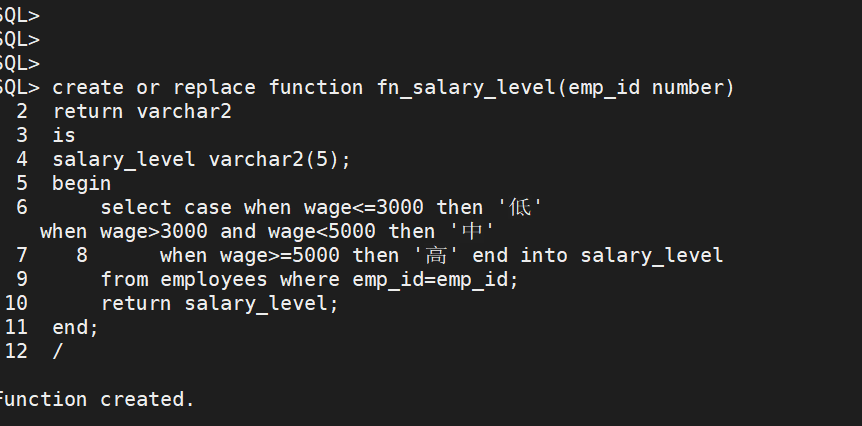
when wage>3000 and wage<5000 then '中'

when wage>=5000 then '高' end into salary\_level

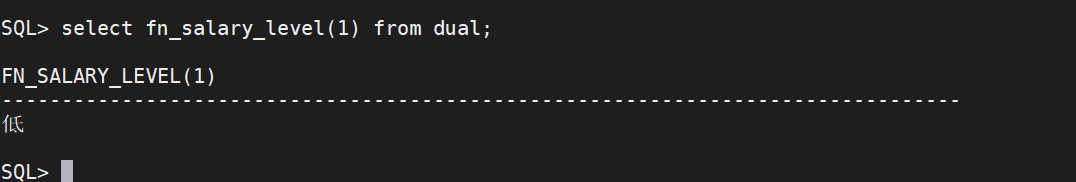
from employees where emp\_id=e\_id;

return salary\_level;

end;



select fn\_salary\_level(1) from dual;



create or replace procedure pr\_dept\_find

is

dept\_num number;

dept\_name varchar2(20);

begin

select count(1) into dept\_num from departments where dep\_id=6;

select dep\_name into dept\_name from departments where dep\_id=6;

if dept\_num =0 then

insert into departments values(6,'后勤部');

else

dbms\_output.put\_line(dept\_name);

end if;

end;

