

第一题：

```
CREATE OR REPLACE TRIGGER TRIG_DEL_DEPARTMENT
BEFORE DELETE ON DEPT
FOR EACH ROW
BEGIN
    DELETE FROM EMP WHERE DEPTNO=:OLD.DEPTNO;
END;
```

第二题：

create or replace package pkg_test is

```
-- Author   : ZhangXingwen
-- Created  : 2015-4-19 18:38:33
-- Purpose  :

    PROCEDURE add_emp(
        EMP_NO NUMBER,
        EMP_NAME VARCHAR2,
        EMP_JOB VARCHAR2 :=NULL,
        EMP_MGR NUMBER :=NULL,
        EMP_HIREDATE DATE :=NULL,
        EMP_SAL NUMBER :=1000,
        EMP_COMM NUMBER :=NULL,
        EMP_DEPTNO NUMBER :=NULL
    );

    PROCEDURE delete_emp(EMP_EMPNO NUMBER);

    FUNCTION get_totalIncome(EMP_EMPNO NUMBER)
        RETURN NUMBER;

    FUNCTION show_deptname(EMP_EMPNO NUMBER)
        RETURN NUMBER;

end pkg_test;
```

create or replace package body pkg_test is

```
PROCEDURE add_emp(  
    EMP_NO NUMBER,  
    EMP_NAME VARCHAR2,  
    EMP_JOB VARCHAR2 :=NULL,  
    EMP_MGR NUMBER :=NULL,  
    EMP_HIREDATE DATE :=NULL,  
    EMP_SAL NUMBER :=1000,  
    EMP_COMM NUMBER :=NULL,  
    EMP_DEPTNO NUMBER :=NULL  
)
```

AS

CNT_PRE NUMBER;

CNT_AFTER NUMBER;

insertException EXCEPTION;

BEGIN

SELECT COUNT(*) INTO CNT_PRE FROM EMP;

INSERT INTO EMP

VALUES(EMP_NO,EMP_NAME,EMP_JOB,EMP_MGR,EMP_HIREDATE,EMP_SAL,EMP_COMM,EMP
_DEPTNO);

COMMIT;

SELECT COUNT(*) INTO CNT_AFTER FROM EMP;

IF NOT (CNT_AFTER-CNT_PRE)=1 THEN

RAISE insertException;

END IF;

EXCEPTION

WHEN insertException THEN

DBMS_OUTPUT.PUT_LINE('员工添加失败');

ROLLBACK;

WHEN OTHERS THEN

DBMS_OUTPUT.PUT_LINE('错误编号：'||SQLCODE||' 错误信息：

'||SQLERRM);

END;

PROCEDURE delete_emp(EMP_EMPNO NUMBER)

AS

del_too_many Exception;

del_no_emp Exception;

```

CNT_PRE NUMBER;
CNT_AFTER NUMBER;
BEGIN
    SELECT COUNT(*) INTO CNT_PRE FROM EMP;
    DELETE FROM EMP WHERE EMPNO = EMP_EMPNO;
    COMMIT;
    SELECT COUNT(*) INTO CNT_AFTER FROM EMP;
    IF (CNT_AFTER-CNT_PRE)>1 THEN
        RAISE del_too_many;
    END IF;

    IF (CNT_AFTER-CNT_PRE)<1 THEN
        RAISE del_no_emp;
    END IF;

    EXCEPTION
        WHEN del_too_many THEN
            DBMS_OUTPUT.PUT_LINE('删除了不止一条记录');
        WHEN del_no_emp THEN
            DBMS_OUTPUT.PUT_LINE('员工删除失败');
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('错误编号：'||SQLCODE||' 错误信息：
'||SQLERRM);
    END;

```

```

FUNCTION get_totalIncome(EMP_EMPNO NUMBER)
    RETURN NUMBER
AS
    CNT NUMBER;
    V_TOTAL_INCOME NUMBER;
    NO_EMP EXCEPTION;
    MORE_EMP EXCEPTION;
BEGIN
    V_TOTAL_INCOME:=0;
    SELECT COUNT(*) INTO CNT FROM EMP WHERE EMP.EMPNO = EMP_EMPNO;
    IF CNT=0 THEN
        RAISE NO_EMP;
    ELSIF CNT>1 THEN
        RAISE MORE_EMP;
    ELSE
        SELECT SAL+NVL(COMM,0)
        INTO V_TOTAL_INCOME
        FROM EMP

```

```

        WHERE EMPNO=EMP_EMPNO;
    END IF;
    EXCEPTION
        WHEN NO_EMP THEN
            DBMS_OUTPUT.PUT_LINE('未找到编号为'||EMP_EMPNO||'的员工资料');
        WHEN MORE_EMP THEN
            DBMS_OUTPUT.PUT_LINE('多于一个员工的编号为'||EMP_EMPNO);
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE(' 错误 编 号 :   '||SQLCODE||' 错误 信息 :
'||SQLERRM);
        RETURN V_TOTAL_INCOME;
    END;

FUNCTION show_deptname(EMP_EMPNO NUMBER)
    RETURN NUMBER
AS
    CNT NUMBER;
    EMP_DNAME VARCHAR2(20);
    NO_DEPT EXCEPTION;

BEGIN
    SELECT COUNT(*),DNAME INTO CNT,EMP_DNAME
    FROM EMP E JOIN DEPT D ON E.DEPTNO=D.DEPTNO
    WHERE E.EMPNO = EMP_EMPNO;
    IF CNT=0 THEN
        RAISE NO_DEPT;
    END IF;
    DBMS_OUTPUT.PUT_LINE('员工'||EMP_EMPNO||'隶属于'||EMP_DNAME||'部门
');
    EXCEPTION
        WHEN NO_DEPT THEN
            DBMS_OUTPUT.PUT_LINE('未找到员工'||EMP_EMPNO||'的所属部门');
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE(' 错误 编 号 :   '||SQLCODE||' 错误 信息 :
'||SQLERRM);
            CNT := -1;
        RETURN CNT;

    END;
end pkg_test;

```

第三题:

```
CREATE OR REPLACE
FUNCTION NORMALIZESALARY RETURN BOOLEAN AS
CURSOR CUREMP IS
    SELECT EMPNO,JOB,SAL FROM EMP;
TYPE RECEMP_TYPE IS RECORD
    (
        EMPNO VARCHAR2(20),
        JOB VARCHAR2(20),
        SALARY NUMBER(20)
    );
RECEMP RECEMP_TYPE;
TYPE RECSALARY_TYPE IS RECORD
    (
        MAX NUMBER,
        MIN NUMBER
    );
TYPE TABLESALARY IS TABLE OF RECSALARY_TYPE INDEX BY VARCHAR2(20);
SALARY_MIN_MAX TABLESALARY;
FLAG BOOLEAN;
BEGIN
    FLAG:=TRUE;
    SALARY_MIN_MAX('CLERK').MIN:=1500;
    SALARY_MIN_MAX('CLERK').MAX:=2500;
    SALARY_MIN_MAX('SALESMAN').MIN:=2501;
    SALARY_MIN_MAX('SALESMAN').MAX:=3500;
    SALARY_MIN_MAX('ANALYST').MIN:=3501;
    SALARY_MIN_MAX('ANALYST').MAX:=4500;
    SALARY_MIN_MAX('OTHERS').MIN:=4501;
    SALARY_MIN_MAX('OTHERS').MAX:=100000;
    OPEN CUREMP;
    FETCH CUREMP INTO RECEMP;
    WHILE CUREMP%FOUND LOOP
        IF RECEMP.JOB='CLERK' THEN
            IF RECEMP.SALARY<SALARY_MIN_MAX('CLERK').MIN THEN
                UPDATE EMP SET SAL=SALARY_MIN_MAX('CLERK').MIN WHERE
EMPNO=RECEMP.EMPNO;
            ELSIF RECEMP.SALARY>SALARY_MIN_MAX('CLERK').MAX THEN
                UPDATE EMP SET SAL=SALARY_MIN_MAX('CLERK').MAX WHERE
EMPNO=RECEMP.EMPNO;
            ELSE
                DBMS_OUTPUT.PUT_LINE('SALARY IS OK');
```

```

        END IF;
    END IF;
    IF RECEMP.JOB='SALESMAN' THEN
        IF RECEMP.SALARY<SALARY_MIN_MAX('SALESMAN').MIN THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('SALESMAN').MIN    WHERE
EMPNO=RECEMP.EMPNO;
        ELSIF RECEMP.SALARY>SALARY_MIN_MAX('SALESMAN').MAX THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('SALESMAN').MAX    WHERE
EMPNO=RECEMP.EMPNO;
        ELSE
            DBMS_OUTPUT.PUT_LINE('SALARY IS OK');
        END IF;
    END IF;
    IF RECEMP.JOB='ANALYST' THEN
        IF RECEMP.SALARY<SALARY_MIN_MAX('ANALYST').MIN THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('ANALYST').MIN    WHERE
EMPNO=RECEMP.EMPNO;
        ELSIF RECEMP.SALARY>SALARY_MIN_MAX('ANALYST').MAX THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('ANALYST').MAX    WHERE
EMPNO=RECEMP.EMPNO;
        ELSE
            DBMS_OUTPUT.PUT_LINE('SALARY IS OK');
        END IF;
    END IF;
    IF RECEMP.JOB='OTHERS' THEN
        IF RECEMP.SALARY<SALARY_MIN_MAX('OTHERS').MIN THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('OTHERS').MIN    WHERE
EMPNO=RECEMP.EMPNO;
        ELSIF RECEMP.SALARY>SALARY_MIN_MAX('OTHERS').MAX THEN
            UPDATE    EMP    SET    SAL=SALARY_MIN_MAX('OTHERS').MAX    WHERE
EMPNO=RECEMP.EMPNO;
        ELSE
            DBMS_OUTPUT.PUT_LINE('SALARY IS OK');
        END IF;
    END IF;
    FETCH CUREMP INTO RECEMP;
    END LOOP;
EXCEPTION
    WHEN OTHERS THEN
        FLAG:=FALSE;
        DBMS_OUTPUT.PUT_LINE('错误编号: '||SQLCODE||' 错误信息: '||SQLERRM);
    RETURN FLAG;
END NORMALIZESALARY;

```