电子信息学院

实验报告书

课程名： 大型数据库

题 目： 实验十 替代触发器(3)

实验类别 【实践】

班 级： 软件1712

姓 名： 张飒

评语：

实验态度：认真（ ） 一般（ √ ） 较差（ ）

实验结果：正确（ ） 部分正确（√ ）错（ ）

实验理论： 掌握（ √ ） 熟悉（ ） 了解（ ） 生疏（ ）

操作技能：较强（ ） 一般（ √ ） 较差（ ）

实验报告： 较好（ ） 一般（ √ ） 较差（ ）

成绩： 指导教师： 林志杰

批阅时间：2019 年 11月11日

评语：

实验态度：认真（ ） 一般（ ） 较差（ ）

实验结果：正确（ ） 部分正确（ ）错（ ）

实验理论：掌握（ ） 熟悉（ ） 了解（ ） 生疏（ ）

操作技能：较强（ ） 一般（ ） 较差（ ）

实验报告：较好（ ） 一般（ ） 较差（ ）

成绩： 指导教师：

批阅时间： 年 月 日

评语：

实验态度：认真（ ） 一般（ ） 较差（ ）

实验结果：正确（ ） 部分正确（ ）错（ ）

实验理论：掌握（ ） 熟悉（ ） 了解（ ） 生疏（ ）

操作技能：较强（ ） 一般（ ） 较差（ ）

实验报告：较好（ ） 一般（ ） 较差（ ）

成绩： 指导教师：

批阅时间： 年 月 日

1．实验目的

（1）掌握oracle 11g的SQL语言中的触发器的相关操作。

（2）熟练oracle创建数据库的过程，能够对触发器进行深入理解和掌握。

（3）掌握oracle的中触发器的存在的意义和作用。

（4）掌握SQL语言中操作触发器的基本技能。

2．实验目标：

在本次实验中，学生应该能够利用所学触发器的基本理论知识，进行触发器的创建、更新、删除和修改触发器的基本操作，能够在实际应用中熟练运用触发器。

3. 触发器简介

触发器的定义就是说某个条件成立的时候，触发器里面所定义的语句就会被自动的执行。因此触发器不需要人为的去调用，也不能调用。

然后，触发器的触发条件其实在你定义的时候就已经设定好了。

这里面需要说明一下，触发器可以分为语句级触发器和行级触发器。

简单的说就是语句级的触发器可以在某些语句执行前或执行后被触发。而行级触发器则是在定义的了触发的表中的行数据改变时就会被触发一次。

具体说明：

1、 在一个表中定义的语句级的触发器，当这个表被删除时，程序就会自动执行触发器里面定义的操作过程。这个就是删除表的操作就是触发器执行的条件了。

2、 在一个表中定义了行级的触发器，那当这个表中一行数据发生变化的时候，比如删除了一行记录，那触发器也会被自动执行了。

4、触发器语法

触发器的语法：

**create [or replace] tigger 触发器名 触发时间 触发事件**

**on 表名**

**[for each row]**

**begin**

**pl/sql语句**

**end**

其中：

**触发器名**：触发器对象的名称。由于触发器是数据库自动执行的，因此该名称只是一个名称，没有实质的用途。

**触发时间**：指明触发器何时执行，该值可取：

before：表示在数据库动作之前触发器执行;

after：表示在数据库动作之后触发器执行。

**触发事件**：指明哪些数据库动作会触发此触发器：

insert：数据库插入会触发此触发器;

update：数据库修改会触发此触发器;

delete：数据库删除会触发此触发器。

**表 名**：数据库触发器所在的表。

**for each row**：对表的每一行触发器执行一次。

如果没有这一选项，则只对整个表执行一次。

触发器能实现如下功能：

功能：

1、 允许/限制对表的修改

2、 自动生成派生列，比如自增字段

3、 强制数据一致性

4、 提供审计和日志记录

5、 防止无效的事务处理

6、 启用复杂的业务逻辑

视图： 基本表的行列子集视图一般是可更新的。若视图的属性来自集函数、表达式，则该视图肯定是不可以更新的。

（1）若视图的字段是来自字段表达式或常数，则不允许对此视图执行INSERT、UPDATE操作，允许执行DELETE操作；

（2）若视图的字段是来自库函数，则此视图不允许更新；

（3）若视图的定义中有GROUP BY子句或聚集函数时，则此视图不允许更新；

（4）若视图的定义中有DISTINCT任选项，则此视图不允许更新；

（5）若视图的定义中有嵌套查询，并且嵌套查询的FROM子句中涉及的表也是导出该视图的基表，则此视图不允许更新；

（6）若视图是由两个以上的基表导出的，此视图不允许更新；

（7）一个不允许更新的视图上定义的视图也不允许更新；

（8）由一个基表定义的视图，只含有基表的主键或候补键，并且视图中没有用表达式或函数定义的属性，才允许更新。

替代触发器的语法：

**create [or replace] tigger 触发器名 instead of 触发事件**

**on 表名**

**[for each row]**

**begin**

**pl/sql语句**

**end**

5．实验内容与步骤

INSTEAD OF用于对视图的DML触发，由于视图有可能是由多个表联结(JOIN)而成，因而并非所有的视图都是可更新的，但可以按照所需的方式执行更新。

创建INSTEAD OF触发器需要注意以下几点：

只能被创建在视图上，并且该视图没有指定WITH CHECK OPTION选项。

不能指定BEFORE或AFTER选项。

FOR EACH ROW子句是可选的。

1)、替换触发器：

--创建表

CREATE TABLE employee(

empno VARCHAR2(4) NOT NULL,

ename VARCHAR2(15) NOT NULL,

deptno VARCHAR2(4) NOT NULL,

job VARCHAR2(4) NOT NULL,

deptno VARCHAR2(4) NOT NULL,

hiredate DATE,

address VARCHAR2(15)

);

CREATE TABLE dept(

deptno VARCHAR2(4) NOT NULL,

dname VARCHAR2(15) NOT NULL

);

例子：首先创建一个视图

create view view\_emp\_dept

as

select empno,ename,dept.deptno,dname,job,hiredate from emp,dept

where emp.deptno=dept.deptno;

注意：在没有创建“替换触发器”时，试图向该视图中插入数据，是会有错误的ORA-01776：无法通过连接视图修改多个基表

创建一个关于view\_emp\_dept视图的替换触发器，在该触发器的主体中实现向emp表和dept表中插入两行相互关联的数据

create or replace trigger tri\_insert\_view

instead of insert

on view\_emp\_dept

for each row

declare

row\_dept dept%rowtype;

begin

select \* into row\_dept from dept where deptno=:new.deptno;

if sql%notfound then

insert into dept(deptno,dname) values(:new.deptno,:new.dname);

end if;

insert into emp(empno,ename,deptno,job,hiredate) values

(:new.empno,:new.ename,:new.deptno,:new.job,:new.hiredate);

end tri\_insert\_view;

/

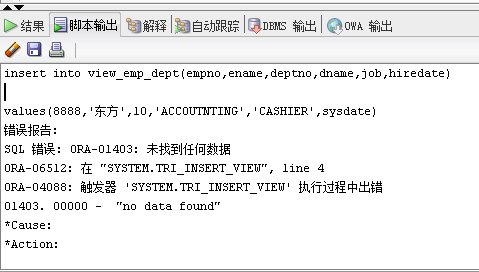
触发该触发器：

insert into view\_emp\_dept(empno,ename,deptno,dname,job,hiredate)

values(8888,'东方',10,'ACCOUTNTING','CASHIER',sysdate);

这样，便可以插入一条记录到视图；

如果这个deptno不在dept表中，那么会在dept表中插入相关记录，但是，测试时，却出现了错误。



2)

首先创建序列

CREATE sequence my\_seq

increment by 1

start with 1

nomaxvalue nocycle cache 20;

创建触发器，在employee表中插入数据时，自动使用序列编号：

create or replace TRIGGER EMP\_INSERT\_ID

BEFORE INSERT ON employee FOR EACH ROW

BEGIN

SELECT my\_seq.NEXTVAL INTO :NEW.empno FROM dual;

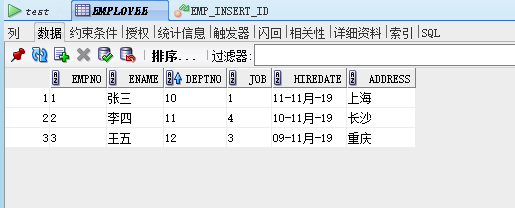
END;

测试语句：

INSERT INTO "SYSTEM"."EMPLOYEE" (ENAME, DEPTNO, JOB, HIREDATE, ADDRESS) VALUES ('张三', '10', '1', TO\_DATE('11-11月-19', 'DD-MON-RR'), '上海')

INSERT INTO "SYSTEM"."EMPLOYEE" (ENAME, DEPTNO, JOB, HIREDATE, ADDRESS) VALUES ('李四', '11', '4', TO\_DATE('10-11月-19', 'DD-MON-RR'), '长沙')

INSERT INTO "SYSTEM"."EMPLOYEE" (ENAME, DEPTNO, JOB, HIREDATE, ADDRESS) VALUES ('王五', '12', '3', TO\_DATE('09-11月-19', 'DD-MON-RR'), '重庆')



3） 创建触发器，在多表联接的视图中插入数据：

-- 创建视图

CREATE OR REPLACE VIEW vw\_emp AS

SELECT e.ename ename, e.address, d.dname dname

FROM employee e, dept d

WHERE e.empno = d.deptno;

-- 创建触发器

CREATE TRIGGER emp\_insert\_trigger

INSTEAD OF INSERT ON vw\_emp

DECLARE

v\_did dept.deptno%TYPE;

BEGIN

SELECT deptno INTO v\_did FROM dept WHERE dname = :NEW.dname;

INSERT INTO emp (ename, address, deptno) VALUES (:NEW.ename, :NEW.address, v\_did);

END emp\_insert\_trigger;

4) 描述替代触发器的在做什么？

CREATE OR REPLACE TRIGGER emp\_delete\_trigger

INSTEAD OF DELETE ON vw\_emp

FOR EACH ROW

DECLARE

v\_did dept.deptno%TYPE;

BEGIN

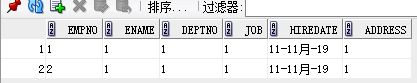
SELEC deptno INTO v\_did FROM dept WHERE dname=:OLD.dname;

DELETE FROM employee WHERE deptno = v\_did;

END emp\_delete\_trigger;

当删除视图vv\_emp中数据时，同时也删除employee表中对应的数据。

DELETE FROM vw\_emp WHERE EMPNO='3'



4.结果分析与实验体会