1.1 什么是触发器

Oracle数据库中触发器(Trigger)是一个与表相关联的、存储的PL/SQL程序。每当一个特定的数据操作语句(Insert、update、delete)在指定的表上发生时,会自动地执行触发器中定义的代码块。

解释:

- 首先,它也是一段plsql程序。
- 然后,它是来触发与表数据操作相关的(insert, update, delete)。
- 然后,在进行表数据操作的时候,会自动触发执行的一段程序。

触发器就是在执行某个操作(增删改)的时候触发一个动作(一段程序)。

可以把触发器理解为一种特殊的存储过程。而区别就是存储过程是显式调用,可以有参数;触发器是隐式调用(当某种操作被触时调用),不可以有参数。

触发器常用于:

- 1. 数据完整性: 在数据修改前或后验证业务规则(如检查字段合法性)
- 2. 级联操作:自动更新或删除关联表中的数据(如删除主表记录时同步删除子表)。
- 3. 审计跟踪:记录数据变更的日志(如记录谁在何时修改了哪些数据)。
- 4. 数据冗余:维护派生数据(如自动计算并更新某个字段的值)。
- 5. 安全控制: 限制对敏感数据的操作(如禁止在非工作时间删除数据)。

触发器分类:

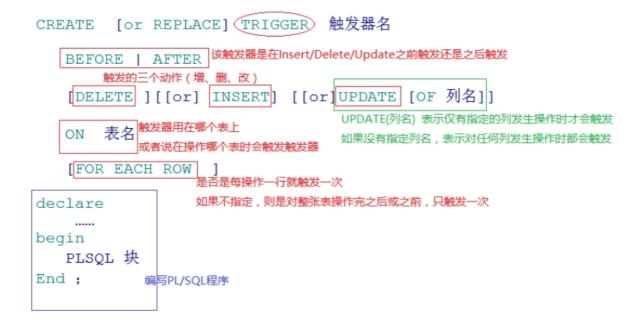
- 前置触发器(BEFORE)
- 后置触发器(AFTER)

1.2 触发器语法

语法:

```
1 CREATE [or REPLACE] TRIGGER 触发器名
2 BEFORE | AFTER
3 [DELETE ][[or] INSERT] [[or]UPDATE [OF 列名]]
4 ON 表名
5 [FOR EACH ROW ]
6 declare
7 .....
8 begin
9 PLSQL 块
10 End;
```

解释说明:



• 行级触发器 (FOR EACH ROW)

触发语句作用的每一条记录都被触发。在行级触发器中使用old和new伪记录变量,识别值的状态

• 语句级触发器 (表级触发器)

在指定的操作语句操作之前或之后执行一次,不管这条语句影响了多少 行

触发器中触发语句与伪记录变量的值:

触发语句	:OLD	:NEW
Insert	所有字段都是空(null)	将要插入的数据
Update	更新以前该行的值	更新后的值
delete	删除以前该行的值	所有字段都是空(null)

1.3. 触发器的使用

1.3.1 前置触发器

需求: 当用户输入本月累计表数后, 自动计算出本月使用数

代码:

```
create or replace
2 trigger tri_update_usenumbefore
  --在表中的数据 update之前执行触发器
  update
      of num1 --当num1字段进行update操作触发器就会执行
  lon
      t_account --触发器建立在t_account表上
  for each row --每update一行,触发器就执行一次(行级触发器)
  declare
11
  begin
12
      :new.usenum := :new.num1 - :new.num0;
13
  end;
14
```

1.3.2 后置触发器

需求: 当用户修改了业主信息表的业主名称时记录修改前与修改后的值

```
1--创建业主名称修改日志表:用于记录业主更改前后的名称2create table t_owners_log(
```

```
updatetime date,
      ownerid number,
      oldname varchar2(30),
      newname varchar2(30)
  );
  1--创建后置触发器,自动记录业主更改前后日志
  create or replace
10
11 trigger tri_owners_logafter
  --后置触发器
12
  update of name --当字段name进行update操作,会执行触发器
13
  on t_owners
                   --触发器建立在t owners表上
14
  for each row --行级触发器
15
16
  declare
17
  begin
   insert into t_owners_logvalues
18
   (sysdate,:old.id,:old.name,:new.name);
19  --注意: 触发器中不需要写commit (隐式提交)
20 end;
```

测试:

```
1 --更新数据
2 update t_owners set name='范冰冰' where id=1;
3 commit;
4 --查询日志表
5 select * from t_owners_log;
```

1.4 触发器案例

案例: 当用户输入本月累计数后,自动计算阶梯水费。

```
1 create or replace trigger tri_account_updatenum1
2 before
3 update of num1
4 on t_account
5 for each row
6 declare
7 v_usenum2 number(10,2);--吨数
```

```
8 begin
9 --使用数赋值
10 :new.usenum:=:new.num1-:new.num0;
11 v_usenum2:= round(:new.usenum/1000,3);--计算吨数
12 :new.money:=fn_calmoney(:new.ownertype,v_usenum2);--对金
额列赋值
13 end;
```

测试:修改记录,观察结果。