

授权与触发器

主要考点：

- 1. 授权 (GRANT)
- 2. 收回授权 (REVOKE)
- 3. 触发器概述
- 4. 创建触发器
- 5. 更改与删除触发器

1、授权

授权 (GRANT)

语句格式：
`GRANT 权限列表 [ON <对象类型><对象名>]
TO <用户>[, <用户>]... /PUBLIC
[WITH GRANT OPTION];`
PUBLIC：表示将权限授予所有人
WITH GRANT OPTION：表示获得了这个权限的用户还可以将权限赋给其他用户。

对象	对象类型	操作权限
属性列	TABLE	SELECT, INSERT, UPDATE, DELETE, ALL PRIVILEGES (4种权限的总和)
视图	TABLE	SELECT, INSERT, UPDATE, DELETE, ALL PRIVILEGES (4种权限的总和)
基本表	TABLE	SELECT, INSERT, UPDATE, DELETE, ALTER, INDEX, ALL PRIVILEGES (6种权限的总和)
数据库	DATABASE	CREATETAB建立表的权限，可由DBA授予普通用户

`GRANT SELECT,INSERT
ON TABLE SC
TO USER_1 WITH GRANT OPTION;`

\\授权 查询，插入

\\的权限到 表 SC 给用户USER_1 ON的使用是给指定用户授权

\\ 此命令允许被赋予权限的用户赋予其他用户自身相同权限

=====

=====

PUBLIC

\\此命令赋予所有用户 GRANT 赋予的权限

2、回收权限

收回权限(REVOKE)

语句格式: **REVOKE** 权限列表 [**ON** <对象类型><对象名>]

FROM <用户>[, <用户>]... /PUBLIC

[**RESTRICT** | **CASCADE**];

RESTRICT: 表示只收回语句中指定的用户的权限

CASCADE: 表示除了收回指定用户的权限外, 还收回该用户赋予的其他用户的权限。

例: 将用户User1及User2对供应商S、零件P、项目J的所有操作权限收回:

```
REVOKE ALL PRIVILEGES ON TABLE S, P, J FROM User1, User2;
```

将所有用户对供应商S的所有查询权限收回:

```
REVOKE SELECT ON TABLE S FROM PUBLIC;
```

将User1用户对供应商S的供应商编号Sno的修改权限收回。

```
REVOKE UPDATE (Sno) ON TABLE S FROM User1;
```

例: 收回用户li对表employee的查询权限, 同时级联收回li授予其他用户的该权限, SQL语句为: (1) select ON TABLE employee FROM li (2);

(1) A. GRANT

B. GIVE

C. CALLBACK

D. REVOKE

(2) A. RESTRICT

B. CASCADE

C. WITH GRANT OPTION

D. WITH CHECK OPTION

```
REVOKE ALL PRIVILEGES
```

```
ON TABLE S, P, J
```

```
FROM User1, User2;
```

3、触发器

触发器概述

• 触发器主要有以下三方面的特点:

(1) 当数据库程序员声明的事件发生时, 触发器被激活。声明的事件可以是对某个特定关系的插入、删除或更新。

(2) 当触发器被事件激活时, 不是立即执行, 而是首先由触发器测试触发条件, 如果事件不成立, 响应该事件的触发器什么都不做。

(3) 如果触发器声明的条件满足, 则与该触发器相连的动作由DBMS执行。动作可以阻止事件发生, 可以撤销事件。

• 创建触发器时需指定:

(1) 触发器名称

(2) 在其上定义触发器的表

(3) 触发事件: 触发器将何时激发

(3) 触发条件: 满足什么条件时执行触发动作

(4) 触发动作: 指明触发器执行时应做的动作

• 触发器可以引用当前数据库以外的对象, 但只能在当前数据库中创建触发器。

• 不能在临时表或系统表上创建触发器, 但触发器可以引用临时表。

创建触发器

```
CREATE TRIGGER <触发器名称> [BEFORE | AFTER]
[DELETE | INSERT | UPDATE OF 列名清单]           //触发事件
ON 表名
[REFERENCING <临时视图名>]
[FOR EACH ROW | FOR EACH STATEMENT]
[WHEN <触发条件>]           //WHEN后面跟触发条件，指明当什么条件满足时，执行下面的触发动作
BEGIN
    <触发动作>           //BEGIN...END 中定义触发动作，即当触发条件满足时，需要数据库做什么
END [触发器名称]
```

BEFORE/AFTER: 指明是在执行触发语句之前激发触发器还是执行触发语句之后激发触发器。

DELETE: 当一个DELETE语句从表中删除行时激发触发器。

INSERT: 当一个INSERT语句向表中插入行时激发触发器。

UPDATE/UPDATE OF(列名清单): 当UPDATE修改表中的值时，激发触发器，也可加 (OF 列名清单) 指定是某些列的值被修改时激发触发器。

REFERENCING: 触发器运行过程中，系统会生成两个临时视图，分别存放更新前和更新后的值，对于行级触发器，为OLD ROW和NEW ROW，对于语句级触发器，为OLD TABLE和NEW TABLE。

例: REFERENCING new row AS nrow / REFERENCING old row AS orow。

FOR EACH ROW: 表示为行级触发器，对每一个被影响的元组（即每一行）执行一次触发过程。

FOR EACH STATEMENT: 表示为语句级触发器，对整个事件只执行一次触发过程，为默认方式。

更改和删除触发器

1、更改触发器

语句格式:

```
ALTER TRIGGER <触发器名称> [BEFORE|AFTER]
DELETE|INSERT|UPDATE OF [列名清单]
ON 表名|视图名
AS
BEGIN
    要执行的SQL语句
END
```

2、删除触发器

语句格式: `DROP TRIGGER <触发器名>`