

数据库原理

第4章 SQL --DCL

辽东学院 鲁 琴

本节要点

数据库基础概念

数据库原理

数据库新技术

关系数据库

关系数据模型

关系数据语言

数据库设计

数据库管理

关系代数

SQL

DDL

DML

DCL

INDEX

VIEW

嵌入式SQL

SELECT

INSERT

UPDATE

DELETE

数据控制语言

- ◆ 概述
- ◆ 授权
- ◆ 收回权限
- ◆ 小结

概述

数据控制亦称为数据保护

- ✓ 安全性控制
- ✓ 完整性控制
- ✓ 并发控制
- ✓ 恢复

SQL语言的数据控制功能

SQL语言提供了数据控制功能，能够在一定程度上保证数据库中数据的

- ✓ 完整性
- ✓ 并发控制
- ✓ 恢复
- ✓ 安全性

1. 完整性

- ✓ **数据库的完整性**是指数据库中数据的有效性、正确性与相容性。
- ✓ **SQL语言定义完整性约束条件**

CREATE TABLE语句

ALTER TABLE语句

- 码
- 取值唯一的列
- 参照完整性
- 其他约束条件

2. 并发控制

- ✓ **并发控制**指的是当多个用户并发地对数据库进行操作时，对他们加以控制、协调，以保证并发操作正确执行，并保持数据库的一致性
- ✓ SQL语言提供了并发控制能力

3. 恢复

恢复指的是当发生各种类型的故障，使数据库处于不一致状态时，将数据库恢复到一致状态的功能。

SQL语言提供了恢复的功能，支持事务、提交、回滚等概念。

4. 安全性

- ✓ **数据库的安全性**是指保护数据库，防止不合法的使用所造成的数据泄露和破坏。
- ✓ 数据库系统保证数据安全性的主要措施
 - ✓ 进行**存取控制**，

DBMS实现存取控制的过程

- ✓ 用户或DBA把授权决定告知系统，这是由SQL的 **GRANT**和**REVOKE**语句来完成的
- ✓ DBMS把授权的结果存入数据字典
- ✓ 当用户提出操作请求时，DBMS根据授权情况进行检查，以决定是否执行操作请求

SQL的DCL

不同的用户对不同的数据应具有何种操作权力，是由DBA和表的建立者（即表的属主）根据具体情况决定的，SQL语言则为DBA和表的属主定义与回收这种权力提供了手段

SQL的DCL

- 授权 **GRANT**
- 回收权力 **REVOKE**

一、SQL的授权功能

GRANT语句的一般格式：

GRANT <权限>[, <权限>]...

[ON <对象类型> <对象名>]

TO <用户>[, <用户>]...

[WITH GRANT OPTION];

功能：将对指定操作对象的指定操作权限授予指定的用户。

(1) 操作权限

对象	对象类型	操 作 权 限
属性列	TABLE	SELECT, INSERT, UPDATE DELETE, ALL PRIVIEGES
视图	TABLE	SELECT, INSERT, UPDATE DELETE, ALL PRIVIEGES
基本表	TABLE	SELECT, INSERT, UPDATE DELETE ALTER, INDEX, ALL PRIVIEGES
数据库	DATABASE	CREATETAB

(2) 用户的权限

数据库的建立表（**CREATETAB**）的权限属于**DBA**，可由DBA授予普通用户，普通用户拥有此权限后可以建立基本表。

基本表或视图的**属主**拥有对该表或视图的一切操作权限。

(3) 接受权限的用户

一个或多个具体用户

PUBLIC（全体用户）

(4) WITH GRANT OPTION子句

如果指定了WITH GRANT OPTION子句，则获得某种权限的用户还可以把这种权限再授予别的用户。

如果没有指定WITH GRANT OPTION子句，则获得某种权限的用户只能使用该权限，但不能传播该权限

例题

例1 把查询Student表权限授给用户U1

```
GRANT  SELECT  
ON  TABLE  Student  
TO  U1;
```

例题（续）

例2 把对Student表和Course表的全部权限授予用户U2和U3

```
GRANT ALL PRIVILIGES  
ON TABLE Student, Course  
TO U2, U3;
```

例题（续）

例3 把对表SC的查询权限授予所有用户

```
GRANT SELECT  
ON TABLE SC  
TO PUBLIC;
```

例题（续）

例4 把查询Student表和修改学生姓名的权限授给用户U4

```
GRANT UPDATE(Sname), SELECT  
ON TABLE Student  
TO U4;
```

例题（续）

例5 把对表SC的INSERT权限授予U5用户，并允许他
再将此权限授予其他用户

```
GRANT INSERT  
ON TABLE SC  
TO U5  
WITH GRANT OPTION;
```

传播权限

执行例5后，U5不仅拥有了对表SC的INSERT权限，还可以传播此权限：

```
GRANT INSERT ON TABLE SC TO U6
```

```
WITH GRANT OPTION;
```

同样，U6还可以将此权限授予U7：

```
GRANT INSERT ON TABLE SC TO U7;
```

但U7不能再传播此权限。

例题（续）

例6 DBA把在数据库S_C中建立表的权限授予用户U8

```
GRANT CREATETAB  
ON DATABASE S_C  
TO U8;
```

GRANT语句小结

一次向一个用户授权（例1）这是最简单的一种授权操作

一次向多个用户授权（例2、例3）；

一次传播多个同类对象的权限（例2）；

一次可以完成对基本表、视图和属性列这些不同对象的授权（例4）；

授予关于DATABASE的权限必须与授予关于TABLE的权限分开。

二、SQL收回权限的功能

REVOKE语句的一般格式为：

REVOKE <权限>[, <权限>]...

[ON <对象类型> <对象名>]

FROM <用户>[, <用户>]...;

功能：从指定用户那里收回对指定对象的指定权限

例题

例7 把用户U4修改学生学号的权限收回

```
REVOKE UPDATE (Sno)  
ON TABLE Student  
FROM U4;
```

例题（续）

例8 收回所有用户对表SC的查询权限

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

例题（续）

例9 把用户U5对SC表的INSERT权限收回

REVOKE INSERT

ON TABLE SC

FROM U5;

权限的级联回收

系统将收回直接或间接从U5处获得的对SC表的INSERT权限

小结

SQL提供了非常灵活的授权机制

用户对自己建立的基本表和视图拥有全部的操作权限，并且可以用GRANT语句把其中某些权限授予其他用户。

被授权的用户如果有“继续授权”的许可，还可以把获得的权限再授予其他用户。

小结（续）

DBA拥有对数据库中所有对象的所有权限，并可以根据应用的需要将不同的权限授予不同的用户。

所有授予出去的权力在必要时又都可以用REVOKE语句收回

。