**第06章 事务及视图**

**本章内容**

* 事务
* 视图

# 事务

## 什么是事务

事务也称工作单元，是一个或多个SQL语句组成的序列，这些个SQL操作作为一个完整的工作单元要么全部执行，要么全不执行。

例如：

Bank 转账

A*------B 2000*

1） A -2000

*-------------*

2） B +2000

## 事务的特性

1、原子性：不可再分的。

2、一致性：不管是谁操作了数据库，那么在数据库中的数据状态是一致的。大家看到的结果是一致的。

3、隔离性：A操作的时候，不影响B，A在提交（commit）之前，可以查看到修改后的结果，但是B看不到。（commit之后，A的操作才真正执行到数据库，在commit之前可以通过rollback撤消操作）

4、持久性：一旦提交，对数据的修改是永久的。

## 事务常用的关键字

### 提交事务(commit)

作用：对表执行DML语句后，实际上数据并没有在磁盘数据文件发生变化，而是改变内存中的显示，这时的数据，我们称之为赃数据，当我们执行了commit之后，此时数据才真正的写入了数据文件，其它会话才能看到变化的数据。

### 自动提交事务

1. 当执行DDL语句,如create,alter,drop table..
2. 当执行DCL语句, grant , revoke,commit、rollback...
3. 当退出(执行exit) sqlplus

### 回滚事务(rollback)

作用：可以对之前的dml语句误操作进行反悔。

### 只读事务的操作(read only)

set transaction read only;

注:只读事务它只能是在事务操作前的第一句话,它跟其它会话提交事务无关.但是事务的提交则可以同步到其它会话改变的数据,什么时候使用只读事务?一般在某个时间点统计数据的时候进行使用.

实验1：冻结视图

### 保存点的操作(savepoint)

-设置保存点:savepoint 保存点名称

-回退到保存点：rollback to 保存点名称

-回退所有的事务操作:rollback;

通过保存点的操作，可以有效的保证损失降到最低.我们可以通过保存点的设置来进行数据的回退.

实验2：保存点操作示例

**提交后的数据状态**

* 1. 数据的改变已经保存到了数据库中。
  2. 所有的用户可以看到结果。
  3. 所有的保存点被释放。
  4. 锁被释放，其他用户可以操作涉及到的数据。

# 视图

## 概述

视图是一个虚拟表，其内容由查询定义，同真实的表一样，视图包含一系列带有名称的列和行数据。但是，视图并不在数据库中以存储的数据值集形式存在。行和列数据来自由定义视图的查询所引用的表，并且在引用视图时动态生成。（视图不是真实存在磁盘上的）

## 视图与表的区别

* + 表需要占用磁盘空间，视图不需要
  + 视图不能添加索引（所以查询速度略微慢点）
  + 使用视图可以简化复杂查询
  + 视图的使用利于提高安全性
  + 视图能对查询出来的记录进行修改，并将修改后的数据回送到源表中

## 创建视图

--语法结构：

CREATE[OR REPLACE][FORCE][NOFORCE]VIEW view\_name

[(column\_name)[,….n]]

AS Select\_statement

[WITH CHECK OPTION[CONSTRAINT constraint\_name]]

[WITH READ ONLY]

--参数说明

CREATE OR REPALCE：--用于创建和修改视图

WITH CHECK OPTION ：--用于创建限制数据访问的视图

WITH READ ONLY ：--用于创建只读视图

--简化：

create view 视图名 as select 语句;

## 创建或修改视图

create or replace view 视图名 as select 语句 [with read only]

## 删除视图

drop view 视图名

备注：当表查询非常复杂时，请使用视图吧！

--示例：显示与SMITH 同部门的所有员工？映射到该视图（view）

SELECT \* FROM emp WHERE

deptno = (select deptno from emp WHERE ename ='SMITH');

## with\_check\_option演示

## 相关数据字典

--查看当前用户下的视图

SELECT \* FROM USER\_VIEWS;

--查看数据库下所有视图

SELECT \* FROM DBA\_VIEWS;