

Video 12-2: 存储过程与函数



存储过程

- ❖ 存储过程：由过程化**SQL**语句书写的一段程序，经编译和优化后，命名并存储在数据库服务器中，使用时只要调用即可。
- ❖ 存储过程的优点
 - (1) 运行效率高
 - (2) 降低了客户机和服务器之间的通信量
 - (3) 方便实施企业规则



存储过程

❖ 过程化SQL块类型

■ 命名块

- 编译后保存在数据库中，可以被反复调用，运行速度较快，**过程和函数是命名块**

■ 匿名块

- 每次执行时都要进行编译，它不能被存储到数据库中，也不能在其他过程化SQL块中调用



存储过程的用户接口

- (1) 创建存储过程
- (2) 执行存储过程
- (3) 修改存储过程
- (4) 删除存储过程



存储过程的用户接口

(1) 创建存储过程

CREATE PROCEDURE 过程名([参数1,参数2,...]) **AS** <过程化SQL块>;

- 过程名：数据库服务器合法的对象标识
- 参数列表：用名字来标识调用时给出的参数值，必须指定值的数据类型。参数也可以定义输入参数、输出参数或输入/输出参数，默认为输入参数
- 过程体：是一个<过程化SQL块>，包括声明部分和可执行语句部分



存储过程的用户接口（续）

- ❖ [例8.8] 利用存储过程来实现下面的应用：从账户1转指定数额的款项到账户2中。

CREATE PROCEDURE

transfer(inAccount INT,outAccount INT,amount FLOAT)

*/*定义存储过程TRANSFER，其参数为转入账户、转出账户、转账额度*/*

AS DECLARE */*定义变量*/*

totalDepositOut Float;

totalDepositIn Float;

inAccountnum INT;



存储过程的用户接口（续）

```
BEGIN                                /*检查转出账户的余额*/  
  
    SELECT Total INTO totalDepositOut FROM Accout  
    WHERE accountnum=outAccount;  
  
    IF totalDepositOut IS NULL THEN  
  
                                /*如果转出账户不存在或账户中没有存款*/  
  
        ROLLBACK;                /*回滚事务*/  
  
        RETURN;  
  
    END IF;
```



存储过程的用户接口（续）

```
IF totalDeposit Out< amount THEN      /*如果账户存款不足*/  
    ROLLBACK;                          /*回滚事务*/  
    RETURN;  
END IF;  
SELECT Accountnum INTO inAccountnum FROM Account  
WHERE accountnum=inAccount;  
IF inAccount IS NULL THEN              /*如果转入账户不存在*/  
    ROLLBACK;                          /*回滚事务*/  
    RETURN;  
ENDIF;
```



存储过程的用户接口（续）

```
UPDATE Account SET total=total-amount  
WHERE accountnum=outAccount;
```

/* 修改转出账户余额，减去转出额 */

```
UPDATE Account SET total=total + amount  
WHERE accountnum=inAccount;
```

/* 修改转入账户余额，增加转入额 */

```
COMMIT;          /* 提交转账事务 */  
END;
```



存储过程的用户接口（续）

（2）执行存储过程

CALL/PERFORM PROCEDURE 过程名([参数1,参数2,...]);

- 使用**CALL**或者**PERFORM**等方式激活存储过程的执行
- 在过程化**SQL**中，数据库服务器支持在过程体中调用其他存储过程



存储过程的用户接口（续）

❖ [例8.9] 从账户01003815868转10000元到01003813828账户中。

CALL PROCEDURE

transfer(01003813828,01003815868,10000);



存储过程的用户接口（续）

（3）修改存储过程

ALTER PROCEDURE 过程名1 RENAME TO 过程名2;



存储过程的用户接口（续）

（4）删除存储过程

DROP PROCEDURE 过程名();



函数

❖ 函数和存储过程的异同

- 同：都是持久性存储模块
- 异：函数必须指定返回的类型



函数（续）

1. 函数的定义语句格式

CREATE FUNCTION 函数名 ([参数1,参数2,...])
RETURNS <类型> **AS** <过程化SQL块>;

2. 函数的执行语句格式

CALL/SELECT 函数名 ([参数1,参数2,...]);

3. 修改函数

■重命名

ALTER FUNCTION 过程名1 **RENAME TO** 过程名2;

■重新编译

ALTER FUNCTION 过程名 **COMPILE**;

