

## 实验七 存储过程和简单系统

### 一、实验目的

- 1、掌握创建存储过程的方法
- 2、掌握修改、删除存储过程的方法
- 3、掌握调用存储过程的方法
- 4、掌握带参存储过程的创建和调用
- 5、了解系统存储过程的用途
- 6、掌握简单系统的实现方法

### 二、预备知识

存储过程是一组为了完成特定功能的SQL语句集，经编译后存在数据库中。用户通过指定存储过程的名字并给出参数如果该存储过程带有参数来执行它。经过了一次创建以后，可以被无数次调用。

在SQL Server的系列版本中，存储过程分为两类：

- 系统存储过程：系统过程主要存储在master 数据库中并以sp\_为前缀，并且系统存储过程主要是从系统表中获取信息从而为系统管理员管理SQL Server 提供支持。
- 用户自定义存储过程：用户自定义存储过程是由用户创建并能完成某一特定功能如查询用户所需数据信息的存储过程。

SQL Server 的存储过程类似于其它编程语言里的过程（procedure），体现在：

- 存储过程能接收参数，并以输出参数的形式返回多个参数给调用存储过程的过程或批处理；
- 存储过程可以容纳对数据进行查询、修改的编程语句，也可以调用其他的存储过程；
- 可以返回执行存储过程的状态值以反映存储过程的执行过程的执行情况。

用系统过程 sp\_helptext+存储过程，可以列出存储过程的源代码、视图、触发器、例程、默认值。

### 三、应用示例

【例 14-1】创建存储过程 auselect，要求先显示表 Student 中计算机系的学生，然后显示表 Student 中英语系的学生。执行后看看其在 SQL server 的哪个地方

```
use Stu_Cou
go
create procedure auselect
as
BEGIN
    select * from Student
    where Sdept='计算机'
    select * from Student
    where Sdept='英语'
END
```

【例 14-2】执行存储过程 auselect

```
use Stu_Cou
```

```
exec aselect      /aselect
```

【例 14-3】查看存储过程 aselect 的源代码

```
use Stu_Cou
```

```
exec sp_helptext 'aselect' /aselect
```

【例 14-4】在实际应用中，常常需要和用户交互，让用户有选择性地进行操作，这就需要创建带参数的存储过程。下面语句创建存储过程 sdept\_pro，它根据用户传递的参数只显示指定系指定性别的学生情况

```
use Stu_Cou
```

--一般情况下，先执行下面的语句：即删除了同名存储过程后再建立新的存储过程

```
if exists (select name from sysobjects
           where name='sdept_pro' and type='p')
    drop procedure sdept_pro
```

```
go
```

如何删除存储过程？

--下面的 dept、sex 为变量名称

```
create procedure sdept_pro (@dept varchar(20) , @sex char(8))
```

```
as
```

```
select * from Student
```

```
where Sdept = @dept and Ssex = @sex
```

@是干什么？

【例 14-5】执行 sdept\_pro 存储过程，列出计算机系男生的情况，然后列出英语系女生的情况

```
use Stu_Cou
```

```
exec sdept_pro '计算机','男'
```

```
go
```

```
exec sdept_pro '英语','女'
```

```
go
```

【例 14-6】如果计算机系学生不止一名，则列出计算机系女生的情况

```
use Stu_Cou
```

```
if exists ( select Sdept, COUNT(Sno) from Student
           group by Sdept
           having Sdept='计算机' and COUNT(Sno)>1 )
```

```
exec sdept_pro '计算机','女'
```

程序扩展了？

【例 14-7】下面语句创建存储过程 name\_pro，它根据用户传递的参数只显示指定姓氏的学生的学号、姓名及选修情况

```
use Stu_Cou
```

```
if exists ( select name from sysobjects
           where name = 'name_pro' and type = 'p')
    drop procedure name_pro
```

```

go
create procedure name_pro ( @name varchar(10))
as
select Student.Sno , Sname , Cno , Grade from
student left outer join sc on student.sno=sc.sno
where Sname like @name

```

--注：这里应用了“外连接”，为什么？

【例 14-8】执行存储过程 name\_pro，列出姓张的学生情况

```

exec name_pro '张%'
go

```

【例 14-9】下面建立存储过程 out\_pro，它返回用户指定学号的学生选修课程门数及平均成绩。其中将应用到“OUTPUT”选项。

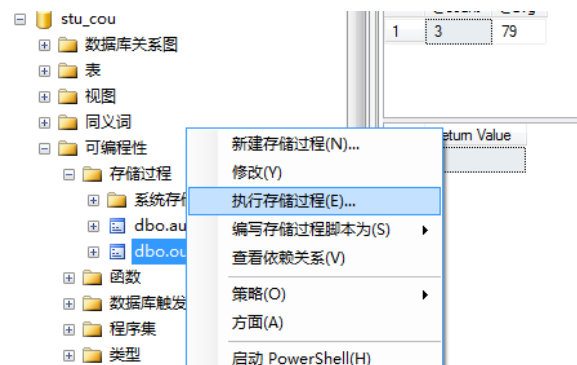
注：OUTPUT 选项表示声明的存储过程参数为“返回参数”，其值可以返回给调用它的 EXEC 语句；但 text 数据类型的参数不能用作存储过程的返回参数。

```

use Stu_Cou
if exists (select name from sysobjects
          where name='out_pro' and type='p')
drop procedure out_pro
go
create procedure out_pro
    (@count int output, @avg real output, @sno char(10))
as
select @count = COUNT(*), @avg = AVG (Grade)
from SC
where Sno = @sno

```

【例 14-10】下面例子说明如何在应用程序中处理存储过程所返回的执行状态信息和参数变量值(也可以通过编译工具反映,例如 C#,C,VB.NET 等,自己查找)，如果测试用自带工具亦可以，如：。



下面是 SQL 代码:

```
use Stu_Cou
go
declare @return_status int
declare @quantity int
declare @avg_grade real
declare @buf char(100)

exec @return_status = out_pro @quantity output, @avg_grade output, '95001'

if @return_status = 0
begin
    select @buf = '95001 号学生共选修' + STR(@quantity, 2) + '门, 平均成绩为: ' +
        STR(@avg_grade, 6, 2)

    print @buf
end
else
begin
    select @buf = '过程执行失败, 其错误代码为' + STR(@return_status, 3)
    print @buf
end
```

注:

字符串函数 STR( float_expression [, length[, decimal ]])
将一个数值型数据表达式 float_expression 转换为字符串, length 为字符串的长度, 其中包括正负号、小数点和数字、空格等, decimal 为小数点后所保留的位数

#### 四、习题

- 1、创建一个存储过程 no\_grade, 显示有所选课程没有成绩的学生的学号、姓名、所选全部课程名称及其成绩。
- 2、如果在 SC 表中存在有学生的课程没有成绩, 则执行存储过程 no\_grade。
- 3、创建一个存储过程 c\_pro, 显示用户指定课程类型且课程名称中包含指定文字的课程的课程号、课程名称、课程类型、学分数。