游标

     游标可以对一个select的结果集进行处理，或是不需要全部处理，就会返回一个对记录集进行处理之后的结果。

     1、游标实际上是一种能从多条数据记录的结果集中每次提取一条记录的机制。游标可以完成：

          # 允许定位到结果集中的特定行

          # 从结果集的当前位置检索一行或多行数据

          # 支持对结果集中当前位置的进行修改

     由于游标是将记录集进行一条条的操作，所以这样给服务器增加负担，一般在操作复杂的结果集的情况下，才使用游标。SQL Server 2005有三种游标：T-SQL游标、API游标、客户端游标。

     2、游标的基本操作

          游标的基本操作有定义游标、打开游标、循环读取游标、关闭游标、删除游标。

     A、 定义游标

declare cursor\_name --游标名称  
cursor [local | global] --全局、局部  
[forward only | scroll] --游标滚动方式  
[read\_only | scroll\_locks | optimistic] --读取方式  
for select\_statements --查询语句  
[for update | of column\_name ...] --修改字段

     参数：

     forward only | scroll：前一个参数，游标只能向后移动；后一个参数，游标可以随意移动

     read\_only：只读游标

     scroll\_locks：游标锁定，游标在读取时，数据库会将该记录锁定，以便游标完成对记录的操作

     optimistic：该参数不会锁定游标；此时，如果记录被读入游标后，对游标进行更新或删除不会超过

     B、 打开游标

          open cursor\_name;

          游标打开后，可以使用全局变量@@cursor\_rows显示读取记录条数

     C、 检索游标

          fetch cursor\_name;

          检索方式如下：

             fetch first; 读取第一行

             fetch next; 读取下一行

             fetch prior; 读取上一行

             fetch last; 读取最后一行

             fetch absolute n; 读取某一行

                如果n为正整数，则读取第n条记录

                如果n为负数，则倒数提取第n条记录

                如果n为，则不读取任何记录

             fetch pelative n

                如果n为正整数，则读取上次读取记录之后第n条记录

                如果n为负数，则读取上次读取记录之前第n条记录

                如果n为，则读取上次读取的记录

     D、 关闭游标

          close cursor\_name;

     E、 删除游标

          deallocate cursor\_name;

     3、游标操作示例

--创建一个游标  
declare cursor\_stu cursor scroll for  
 select id, name, age from student;  
--打开游标  
open cursor\_stu;  
--存储读取的值  
declare @id int,  
 @name nvarchar(20),  
 @age varchar(20);  
--读取第一条记录  
fetch first from cursor\_stu into @id, @name, @age;  
--循环读取游标记录  
print '读取的数据如下：';  
--全局变量  
while (@@fetch\_status = 0)  
begin  
 print '编号：' + convert(char(5), @id) + ', 名称：' + @name + ', 类型：' + @age;  
 --继续读取下一条记录  
 fetch next from cursor\_stu into @id, @name, @age;  
end  
--关闭游标  
close area\_cursor;  
  
--删除游标  
--deallocate area\_cursor;