Atitit T-SQL的十一种设计模式

目录

[1.1. ITERATOR(迭代) 游标。 1](#_Toc10343)

[1.2. 二、INTERSECTOR(交集) join 1](#_Toc10473)

[1.3. 三、QUALIFIER(限定) where 2](#_Toc3708)

[1.4. 四、EXECTOR(运行) 动态sql 2](#_Toc4200)

[1.5. 五、Conveyor(传送) 2](#_Toc31464)

[1.6. 六、Restorer(恢复) 2](#_Toc4622)

[1.7. 七、PROTOTYPE(原型) 复制表结构等 3](#_Toc4696)

[1.8. 八、Singleton(单例） 4](#_Toc9014)

[1.9. 九、FACADE（外观） view 4](#_Toc21146)

[1.10. 十、Chain Of Responsibility(职责链) triger链 5](#_Toc6378)

[1.11. 十一、COMMAND(命令) 5](#_Toc6453)

带你深入了解"T-SQL"的十一种设计模式

## ITERATOR(迭代) 游标。

1. 这种模式提供一种在相似对象列表中遍历对象的标准化方法。在SQL Server数据库中的同义词是游标。

## 二、INTERSECTOR(交集) join

1. 这种模式是表示集合交集的一种模板。  
     
   1、推荐方法：  
     
   SELECT c.companyname,o.orderid  
     
   FROM customer c INNER JOIN orders o ON c.customerid = o.customerid

## 三、QUALIFIER(限定) where

1. 限定数据等价于筛选查询所返回的行数。  
     
   1、常用法：WHERE子句限定  
     
   SELECT city,count(\*) AS NumberCity

## 四、EXECTOR(运行) 动态sql

1. 提供创建并执行动态T-SQL字符串的模板

## 五、Conveyor(传送)

1. 提供一种通过存储过程链传送信息的机制。与GoF的责任链模式(Chain of Responsibility)相类似。  
     
   1、传送返回码  
     
   注：上述代码使用了存储过程的结果代码从过程向过程传递原始返回码的方法，即A调用B，B又调用C，C运行时如出现了错误，则将错误代码-1传送给A。  
     
   2、通过输出参数传送消息  
     
   注：可以使用任何数据类型(包括游标)来返回任何想要的信息

## 六、Restorer(恢复)

1. 此模式提供一种在出错时清理资源的机制。为避免孤立一个事务，当事务活动时，适当地处理出错条件极其重要。  
     
   1、出错时回滚事务  
     
   注：此模式的关键部分是将错误码@@error缓存至变量@err中，如果不缓存@@error，下一执行成功的语句将重置@@error，缓存它后，如出现错误，将检查@errr的值并回滚该活动事务。  
     
   2、出错时清除临时表  
     
   CREATE PROC procR  
     
   3、主动执行恢复模式  
     
   注：通过@@TRANCOUNT<>0可知有活动事务，执行ROLLBACK回滚当前活动连接的所有事务。当SQL Server使用连接池时（对WEB服务器而言相当常见），  
     
   在实际应用中编写此种逻辑就非常重要。由于一个虚连接可以留下一个打开的事务，该事务会影响使用同一物理连接的后续用户，因此，通过主动地  
     
   执行Restorer模式，让代码知道如何保护自己免受“无赖”事务和其他意外残余的影响。

## 七、PROTOTYPE(原型) 复制表结构等

1. 此模式的目标：使用一种原型实例指定要创建对象的类型，并且通过复制原型创建新的对象。  
     
   1、最常见的实现方式是SELECT...INTO结构  
     
   2、复制表结构(T-SQL惯例中也曾提过)  
     
   SELECT \*  
     
   INTO newCustomers  
     
   FROM Customers  
     
   WHERE　1 = 2  
     
   或  
     
   SELECT　TOP 0 \*  
     
   INTO newCustomers  
     
   FROM Customers  
     
   注：通过错误的WHERE条件或不存在的行实现了复制表结构的功能  
     
   3、复制表时指定新数据  
     
   SELECT IDENTITY(int,1,1) AS CustNo,\*  
     
   INTO newCustomers  
     
   FROM Customers  
     
   注：还可指定新列、通过联接选取来自其他表或视图的列、约束或函数等许多的可能性。

## 八、Singleton(单例）

1. 此模式目标：确保在任何给定时间只存在一个类实例并且提供访问该实例的路径。  
     
   严格说，在关系数据库中，对于面向对象类的等价物是表。类的一个实例就是表中的一行。因此，Singleton模式的最显而意见的实现就是确保表中只包含一行。  
     
   注：由于触发器的原因，在任何时刻只允许在表中插入一行，如果表中已包含至少一行，在试图插入新的一行时将导致错误并回滚事务。  
     
   实际应用：禁止一个应用的多个实例连接至服务器  
     
   方法1：应用程序锁  
     
   --锁定应用程序资源

## 九、FACADE（外观） view

1. 此模式目标：它给位于子系统的一个接口集合提供统一的接口。  
     
   在T-SQL中与此模式类似的是包含INSTEAD OF触发器的视图(INSTEAD OF触发器接受对视图的更新，并将它们分配给适当的底层表)。

## 十、Chain Of Responsibility(职责链) triger链

1. 此模式目标：为避免通过多个对象提供机会处理请求，合并请求的发送者与接收者。为实现该模式，必须串联接收对象并沿此链传送请求，直到某个对象处理它。  
     
   前面的Conveyor模式已接到责任链模式，这里再作深入讨论。在T－SQL中最接近此模式所描述行为的是嵌套触发器(触发器的执行导致其他触发器激活并实现串联行为的操作)。  
     
   注：SP\_CONFIGURE 'NESTED TRIGGER',0 可以禁止触发器嵌套,同时触发器最多嵌套32次。从功能角度考虑，在触发器未设定激活顺序情况下，插入请求从一个触发器传送给另一个。在任何情况下，如果其中的任一触发器拒绝插入并回滚事务，则整个操作都将被取消。

## 十一、COMMAND(命令)

1. 这种模式目标：将请求一个对象来封装，允许你参数化包含不同请求、队列或日志请求的客户端，并支持可撤消操作。在T－SQL中与此模式对应的是事务。