-- 消费统计（仅考虑全部版本）

-- 创建临时表来保存统计期内的消费记录信息

-- 1.判断临时表是否存在

If Object\_Id('tempdb.dbo.#TempConsume') Is Not NULL--#Test 为临时表名

Begin

truncate table #TempConsume; -- 删除临时表数据

End

Else

Begin

-- 创建临时表

create table #TempConsume

(

OpenID varchar(50), -- 用户OpenId的唯一标示

ConsumeOrder varchar(20) default(''), -- 订单号

ConsumeNewCoin int default(0), -- 消费New币

AppId int default(0), -- 应用id

DeviceFlag varchar(50) default(''), -- 子机型

SubmitTime datetime default(0), -- 提交时间

ConfirmTime datetime default(0), -- 确认时间

Status int default(0), -- 状态，:预留，默认:未处理,2:消费成功,3:消费失败,4:退款

Version varchar(50) default('') -- 版本号

);

End

-- 3.查询数据，并添加到临时表中

declare @sStart varchar(30)

set @sStart = CONVERT(varchar(30), @dStartDate, 120);

set @sql = 'SELECT OpenId,ConsumeOrder,ConsumeNewCoin,AppId,DeviceFlag,SubmitTime,ConfirmTime,Status,Version FROM tradingcenter.AccConsumeRecord where ClientId=12 and ChannelNo=70 and Status=2';

SET @sql = 'SELECT \* FROM OPENQUERY([MYSQL\_DB], ''' + @sql + ''')'

-- 将存储过程的数据赋值给临时的表

insert into #TempConsume(OpenID,ConsumeOrder,ConsumeNewCoin,AppId,DeviceFlag,SubmitTime,ConfirmTime,Status,Version)

exec(@sql);

**创建临时表**  
       方法一：  
     create table #临时表名(字段1 约束条件,  
                      字段2 约束条件,  
                  .....)  
        create table ##临时表名(字段1 约束条件,  
                          字段2 约束条件,  
                      .....)  
        方法二：  
     select \* into #临时表名 from 你的表;  
       select \* into ##临时表名 from 你的表;  
注：以上的#代表局部临时表，##代表全局临时表  
  
**查询临时表**  
     select \* from #临时表名;  
       select \* from ##临时表名;  
  
**删除临时表**  
     drop table #临时表名;  
       drop table ##临时表名;

SQL SERVER临时表的使用

|  |
| --- |
| drop table #Tmp   --删除临时表#Tmp create table #Tmp --创建临时表#Tmp (     ID   int IDENTITY (1,1)     not null, --创建列ID,并且每次新增一条记录就会加1     WokNo                varchar(50),        primary key (ID)      --定义ID为临时表#Tmp的主键       ); Select \* from #Tmp    --查询临时表的数据 truncate table #Tmp --清空临时表的所有数据和约束  相关例子：  Declare @Wokno Varchar(500) --用来记录职工号 Declare @Str NVarchar(4000) --用来存放查询语句 Declare @Count int --求出总记录数       Declare @i int Set @i = 0  Select @Count = Count(Distinct(Wokno)) from #Tmp While @i < @Count      Begin        Set @Str = 'Select top 1 @Wokno = WokNo from #Tmp Where id not in (Select top ' + Str(@i) + 'id from #Tmp)'        Exec Sp\_ExecuteSql @Str,N'@WokNo Varchar(500) OutPut',@WokNo Output        Select @WokNo,@i --一行一行把职工号显示出来        Set @i = @i + 1     End  临时表 可以创建本地和全局临时表。本地临时表仅在当前会话中可见；全局临时表在所有会话中都可见。 本地临时表的名称前面有一个编号符 (#table\_name)，而全局临时表的名称前面有两个编号符 (##table\_name)。  SQL 语句使用 CREATE TABLE 语句中为 table\_name 指定的名称引用临时表：  CREATE TABLE #MyTempTable (cola INT PRIMARY KEY) INSERT INTO #MyTempTable VALUES (1)  如果本地临时表由存储过程创建或由多个用户同时执行的应用程序创建，则 SQL Server 必须能够区分由不同用户创建的表。为此，SQL Server 在内部为每个本地临时表的表名追加一个数字后缀。存储在 tempdb 数据库的 sysobjects 表中的临时表，其全名由 CREATE TABLE 语句中指定的表名和系统生成的数字后缀组成。为了允许追加后缀，为本地临时表指定的表名 table\_name 不能超过 116 个字符。  除非使用 DROP TABLE 语句显式除去临时表，否则临时表将在退出其作用域时由系统自动除去：  当存储过程完成时，将自动除去在存储过程中创建的本地临时表。由创建表的存储过程执行的所有嵌套存储过程都可以引用此表。但调用创建此表的存储过程的进程无法引用此表。  所有其它本地临时表在当前会话结束时自动除去。  全局临时表在创建此表的会话结束且其它任务停止对其引用时自动除去。任务与表之间的关联只在单个 Transact-SQL 语句的生存周期内保持。换言之，当创建全局临时表的会话结束时，最后一条引用此表的 Transact-SQL 语句完成后，将自动除去此表。  在 存储过程或触发器中创建的本地临时表与在调用存储过程或触发器之前创建的同名临时表不同。如果查询引用临时表，而同时有两个同名的临时表，则不定义针对哪 个表解析该查询。嵌套存储过程同样可以创建与调用它的存储过程所创建的临时表同名的临时表。嵌套存储过程中对表名的所有引用都被解释为是针对该嵌套过程所 创建的表，例如：  CREATE PROCEDURE Test2 AS CREATE TABLE #t(x INT PRIMARY KEY) INSERT INTO #t VALUES (2) SELECT Test2Col = x FROM #t GO CREATE PROCEDURE Test1 AS CREATE TABLE #t(x INT PRIMARY KEY) INSERT INTO #t VALUES (1) SELECT Test1Col = x FROM #t EXEC Test2 GO CREATE TABLE #t(x INT PRIMARY KEY) INSERT INTO #t VALUES (99) GO EXEC Test1 GO  下面是结果集：  (1 row(s) affected)  Test1Col     -----------  1  (1 row(s) affected)  Test2Col     -----------  2  当创建本地或全局临时表时，CREATE TABLE 语法支持除 FOREIGN KEY 约束以外的其它所有约束定义。如果在临时表中指定 FOREIGN KEY 约束，该语句将返回警告信息，指出此约束已被忽略，表仍会创建，但不具有 FOREIGN KEY 约束。在 FOREIGN KEY 约束中不能引用临时表。  考虑使用表变量而不使用临时表。当需要在临时表上显式地创建索引时，或多个存储过程或函数需要使用表值时，临时表很有用。通常，表变量提供更有效的查询处理。 |