

Re: BETWEEN in dynamic SQL

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.programming/2004-05/0392.html>

From: Vlad (vovan.c_at_verizon.net)

Date: 05/03/04

Date: Mon, 3 May 2004 09:58:03 -0400

Thank you Tibor.

Yes, I found the way to print instead of exec and now I have the text which doesn't contain single quotes around my parameters:

```
SELECT dbo.[Partial].PartialNumber, dbo.CommissionPayments.CheckDate,
dbo.CommissionPayments.CheckNumber,
        dbo.CommissionPayments.CommissionAmountPaid,
dbo.Salesman.Salesman, dbo.CommissionPayments.CheckAmount
FROM dbo.[Partial] RIGHT OUTER JOIN
        dbo.Salesman RIGHT OUTER JOIN
        dbo.CommissionPayments ON dbo.Salesman.SalesmanID =
dbo.CommissionPayments.SalesmanID ON
        dbo.[Partial].PartialID =
dbo.CommissionPayments.PartialID
WHERE (dbo.CommissionPayments.RowDeleted <> 1) AND
dbo.CommissionPayments.CheckDate BETWEEN CONVERT(DATETIME, 01/01/1999,112)
AND
        CONVERT(DATETIME,05/03/2004,112) ORDER BY
dbo.[Partial].PartialNumber
```

When I try to add them around parameters by adding '+' between CONVERT(DATETIME, and @DateMin as well as after @DateMin I have a mess in printed SQL. It does contain the @DateMin itself instead of its value.

Vlad

"Tibor Karaszi" <tibor_please.no.email_karaszi@hotmail.nomail.com> wrote in message news:OzpkSTRMEHA.2500@TK2MSFTNGP12.phx.gbl...

> *Inside the stored procedure, do:*

>

> *PRINT @sql*

>

> *Instead of EXEC sp_executesql @sql*

>

> *Also, how did you call the procedure. And did you consider making the parameters datetime instead of strings?*

>

> --
> *Tibor Karaszi, SQL Server MVP*
> <http://www.karaszi.com/sqlserver/default.asp>
>
>
> "Vlad" <vovan.c@verizon.net> wrote in message
news:OV8MgQRMEHA.624@TK2MSFTNGP11.phx.gbl...
>> *Thank you, Tibor*
>> *Unfortunately I do not know how to print @sql without executing stored*
>> *procedure.*
>> *I tried to use*
>> *SET @sql = @sql + N' AND dbo.CommissionPayments.CheckDate BETWEEN '+*
>> *CONVERT(DATETIME, @DateMin,112) + ' AND '+*
>> *CONVERT(DATETIME, @DateMax,112)*
>> *Before I changed @DateMin and @DateMax datatype to smalldatetime*
>> *and it gives me an error*
>> *Server: Msg 241, Level 16, State 1, Procedure CRCommissionPayments, Line*
29
>> *Syntax error converting datetime from character string.*
>>
>> *So I don't know what is wrong now*
>>
>> *Vlad*
>>
>> "Tibor Karaszi" <tibor_please.no.email_karaszi@hotmail.nomail.com> wrote
in
>> message news:O3GujCRMEHA.2704@TK2MSFTNGP10.phx.gbl...
>>> *Why do you use varchar as parameters for the datetime values (@DateMin*
and
>> *DatetimeMax)? Why not use datetime?*
>>>
>>> *Also, if you print the @sql variable instead of executing it, it is*
much
>> *easier to provide assistance. My*
>>> *guess is that you convert the datetime variable to datetime, but then*
>> *concatenate them to a string, so they*
>>> *will be converted to string. And that string representation will not*
be
>> *language neutral (see the article*
>>> *about datetime on my web site). I suggest you pass these as datetime,*
and
>> *then use CONVERT so you get a*
>>> *language neutral datetime format (probably code 112).*
>>>
>>> --
>>> *Tibor Karaszi, SQL Server MVP*
>>> <http://www.karaszi.com/sqlserver/default.asp>
>>>
>>>
>>> "Vlad" <vovan.c@verizon.net> wrote in message
>> news:%23FZOsxQMEHA.2556@TK2MSFTNGP10.phx.gbl...

```

>>>> I have this stored procedure which uses dynamic SQL:
>>>> CREATE PROCEDURE CRCommissionPayments
>>>>
>>>> @Salesman int=null,
>>>> @CheckNumber varchar(20)=null,
>>>> @DateMin varchar(20)=null,
>>>> @DateMax varchar(20)=null,
>>>> @SortBy varchar(50)=null,
>>>> @SortOrder varchar(10)=null
>>>> AS
>>>> DECLARE @sql nvarchar(1000)
>>>> DECLARE @localsortby nvarchar(50)
>>>>
>>>> SET NOCOUNT ON
>>>> --the constant part of the SQL Statement
>>>> SET @sql = N'SELECT dbo.[Partial].PartialNumber,
>>>> dbo.CommissionPayments.CheckDate,
>>>> dbo.CommissionPayments.CheckNumber,
>>>> dbo.CommissionPayments.CommissionAmountPaid,
>>>> dbo.Salesman.Salesman, dbo.CommissionPayments.CheckAmount
>>>> FROM dbo.[Partial] RIGHT OUTER JOIN
>>>> dbo.Salesman RIGHT OUTER JOIN
>>>> dbo.CommissionPayments ON
>>>> dbo.Salesman.SalesmanID
>>>> =
>>>> dbo.CommissionPayments.SalesmanID ON
>>>> dbo.[Partial].PartialID =
>>>> dbo.CommissionPayments.PartialID
>>>> WHERE (dbo.CommissionPayments.RowDeleted <> 1)'
>>>>
>>>> --Dynamic Part
>>>> --Selection
>>>>
>>>>
>>>> if (@Salesman is null) AND (@CheckNumber is null)
>>>> SET @sql = @sql + N' AND (dbo.CommissionPayments.CheckDate BETWEEN
>>>> CONVERT(DATETIME, '+@DateMin+N') AND
>>>> CONVERT(DATETIME, '+@DateMax+N'))'
>>>> else if NOT(@Salesman is null) AND (@CheckNumber is null)
>>>> SET @sql = @sql + N' AND dbo.Salesman.SalesmanID = ' +
>>>> cast(@Salesman
>>>> as
>>>> varchar(10)) + N' AND (dbo.CommissionPayments.CheckDate BETWEEN
>>>> CONVERT(DATETIME, '+@DateMin+N') AND
>>>> CONVERT(DATETIME, '+@DateMax+N'))'
>>>> else if (@Salesman is null) AND NOT(@CheckNumber is null)
>>>> SET @sql = @sql + N' AND dbo.CommissionPayments.CheckNumber = ' +
>>>> @CheckNumber + N' AND (dbo.CommissionPayments.CheckDate BETWEEN
>>>> CONVERT(DATETIME, '+@DateMin+N') AND
>>>> CONVERT(DATETIME, '+@DateMax+N'))'
>>>> else if NOT(@Salesman is null) AND NOT(@CheckNumber is null)

```

```

>>>> SET @sql = @sql + N' AND dbo.Salesman.SalesmanID = ' +
cast(@Salesman
>> as
>>>> varchar(10)) + N' AND dbo.CommissionPayments.CheckNumber = ' +
>>>> @CheckNumber + N' AND (dbo.CommissionPayments.CheckDate BETWEEN
>>>> CONVERT(DATETIME, '+@DateMin +N') AND
>>>> CONVERT(DATETIME, '+@DateMax+N'))'
>>>>
>>>> --Sorting
>>>>
>>>> if (@SortBy is null) AND (@SortOrder is null)
>>>> SET @sql = @sql
>>>> else if NOT(@SortBy is null) AND (@SortOrder is null) --default
Order is
>> ASC
>>>> BEGIN
>>>> if @SortBy = 'Partial Number'
>>>> SET @localsortby = N'dbo.[Partial].PartialNumber '
>>>> else if @SortBy = 'Check Date'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckDate '
>>>> else if @SortBy = 'Check Number'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckNumber '
>>>> else if @SortBy = 'Salesman'
>>>> SET @localsortby=N'dbo.Salesman.Salesman '
>>>> else if @SortBy = 'Comm. Amt. Paid'
>>>> SET @localsortby=N'dbo.CommissionPayments.CommissionAmountPaid '
>>>> else if @SortBy = 'Check Amount'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckAmount '
>>>>
>>>> SET @sql = @sql+ N' ORDER BY ' + @localsortby
>>>> END
>>>> if NOT(@SortBy is null) AND NOT(@SortOrder is null) --for DESC order
>>>> BEGIN
>>>> if @SortBy = 'Partial Number'
>>>> SET @localsortby = N'dbo.[Partial].PartialNumber DESC'
>>>> else if @SortBy = 'Check Date'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckDate DESC'
>>>> else if @SortBy = 'Check Number'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckNumber DESC'
>>>> else if @SortBy = 'Salesman'
>>>> SET @localsortby=N'dbo.Salesman.Salesman DESC'
>>>> else if @SortBy = 'Comm. Amt. Paid'
>>>> SET @localsortby=N'dbo.CommissionPayments.CommissionAmountPaid
DESC'
>>>> else if @SortBy = 'Check Amount'
>>>> SET @localsortby=N'dbo.CommissionPayments.CheckAmount DESC'
>>>>
>>>> SET @sql = @sql+ N' ORDER BY ' + @localsortby
>>>> END
>>>>
>>>> EXEC sp_ExecuteSQL @sql

```

>>>> *GO*
>>>>
>>>> *When I'm executing it with let's say @DateMin=01/01/2003 and*
>>>> *@DateMax=05/03/2004, it returns records with CheckDate=01/01/1900*
>> *(there*
>>>> *are some wrong entered data in a table), which is not between the*
values
>> *I*
>>>> *passed with parameters. It doesn't return records with CheckDate*
which
>> *are*
>>>> *really between entered dates.*
>>>> *What I wrote wrong in the dynamic part for selection?*
>>>>
>>>> *Thank you*
>>>> *Vlad*
>>>>
>>>>
>>>>
>>>
>>>
>>
>>
>>
>
>