

## Re: INSERT Query problem with Quotes & Apostrophes

**Source:**

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.modulesdaovba/2004-10/0538.html>

---

**From:** Douglas J. Steele (*NOSPAM\_djsteele\_at\_NOSPAM\_canada.com*)

**Date:** 10/08/04

Date: Fri, 8 Oct 2004 08:52:34 -0400

Sorry: I should have read my code closer! It was from an Access 97 database, and Access 97 doesn't have the Replace function in it, so I had to write my own. Change MyReplace to Replace everywhere for Access 2000 and newer.

I've got front-ends that go against both Jet and SQL Server databases. You just have to be careful that you use the correct delimiters in your queries, so I use a slightly different version of CorrectDate, where I pass not only the date field to format, but also a flag as to whether it's for Jet or SQL Server. I've got a sample in my September 2003 Smart Access column (downloadable at <http://members.rogers.com/douglas.j.steele/SmartAccess.html>)

```
--
Doug Steele, Microsoft Access MVP
http://I.Am/DougSteele
(no e-mails, please!)
"Pete" <Pete@discussions.microsoft.com> wrote in message
news:7F296B08-66C1-487E-86EE-E0107662AEFF@microsoft.com...
> Thanks for this. Is your MyReplace function significantly different from
the
> standard replace function?
>
> I was interested about your comment concerning SQL Server. I am about to
> rewrite another Access 2003 application with the intention that it will be
> SQL Server compatible so that, if we want to, we can move from an Access
> backend to SQL Server Express when it is released. I don't know whether it
is
> possible to write a front end that is mutually compatible with both. I
have
> had a small amount of experience with SQL Server and not had too many
> problems. I note that there is an option to make queries ANSI 92
compatible
> in Access and I am aware of the differences with yes/no fields and text
> fields.
>
> "Douglas J. Steele" wrote:
>
> > I typically add a function like the following to all of my applications:
> >
> > Public Function CorrectText( _
> >     InputText As String, _
```

## microsoft.public.access.modulesdaovba: Re: INSERT Query problem with Quotes & Apostrophes

```
> > Optional RemoveNulls As Boolean = True, _
> > Optional Delimiter As String = "" _
> > ) As String
> > ' This code was originally written by
> > ' Doug Steele, MVP
> > ' http://I.Am/DougSteele
> > ' You are free to use it in any application
> > ' provided the copyright notice is left unchanged.
> > '
> > ' Description: Given a text string, this function converts the
> > ' text so that it will work with SQL statements.
> > ' It does this by replacing any occurrence of the
> > ' delimiter character in the text by two occurrences
> > ' of the delimiter character. The default delimiter
> > ' character is a single quote ('), which will be
> > ' replaced by two single quotes ('', not to be confused
with
> > ")
> > '
> > ' Inputs: InputText String Expression
> > '
> > ' Returns: InputText, with any occurrence of ' replaced by ''
> > ' with delimiters at the beginning and end of the string.
> > ' (i.e. Doug's will be returned as 'Doug's')
> >
> > On Error GoTo Err_CorrectText
> >
> > Dim strTemp As String
> >
> > strTemp = Delimiter
> > If RemoveNulls = True Then
> > strTemp = strTemp & MyReplace(MyReplace(InputText, Chr$(0), ""),
> > Delimiter, Delimiter & Delimiter)
> > Else
> > strTemp = strTemp & MyReplace(InputText, Delimiter, Delimiter &
> > Delimiter)
> > End If
> > strTemp = strTemp & Delimiter
> >
> > End_CorrectText:
> > CorrectText = strTemp
> > Exit Function
> >
> > Err_CorrectText:
> > Err.Raise Err.Number, "CorrectText", Err.Description
> > strTemp = vbNullString
> > Resume End_CorrectText
> >
> > End Function
> >
> > I can then simply do:
> >
> > .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > cChange) VALUES (" & CorrectText(CurrentUser) & ", #" & Date() & "#, "
&
> > CorrectText(RecordChanged) & ")"
> >
> > and not have to worry about quotes at all. You could also add a
CorrectDate
> > function, and have:
> >
```

## microsoft.public.access.modulesdaovba: Re: INSERT Query problem with Quotes & Apostrophes

```
> > .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > cChange) VALUES (" & CorrectText(CurrentUser) & ", " &
CorrectDate(Date())
> > & ", " &
> > CorrectText(RecordChanged) & ")"
> >
> > This has an added advantage if you ever move to another platform, such
as
> > SQL Server, where the rules regarding quotes and dates are different.
> >
> >
> > --
> > Doug Steele, Microsoft Access MVP
> > http://I.Am/DougSteele
> > (No private e-mails, please)
> >
> >
> > "Pete" <Pete@discussions.microsoft.com> wrote in message
> > news:91301DF4-18DD-4217-93D8-AE198CE172CA@microsoft.com...
> > > Thanks very much. Works a treat and I note your comment about the date
> > > (the
> > > system will only ever be used in-hosue so this is not a problem).
> > >
> > > Using punctuation characters in strings always seems to cause
problems!
> > > Shame there isn't a special data type "StringContainingPunctuation" or
> > > something like that whereby Access does this for you, as the double
quotes
> > > solution is not very intuitive for novices and even trips up
exerperienced
> > > programmers like myself occasionally!! :-)
> > >
> > >
> > > "Douglas J. Steele" wrote:
> > >
> > > > What you have to do is decide whether you want to use single quotes
or
> > > > double quotes as your delimiter, and then double each occurrence of
that
> > > > delimiter symbol in your string.
> > > >
> > > > Assuming you're using Access 2000 or newer, the Replace function
will
> > > > help
> > > > here.
> > > >
> > > > In your example, you're using double quotes as the delimiter,
therefore
> > > > you
> > > > need to change each occurrence of " in your text to "&":
> > > >
> > > > .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > > > cChange) VALUES (' & CurrentUser & ', # & Date() & "#, " &
Chr(34) &
> > > > Replace(RecordChanged, Chr(34), Chr(34) & Chr(34)) & Chr(34) & ")"
> > > >
> > > > Don't worry that this will impact what you're inserting. It won't:
even
> > > > though you're putting two double quotes into the string, it will be
> > > > recognized as only being one for insertion purposes.
```

## microsoft.public.access.modulesdaovba: Re: INSERT Query problem with Quotes & Apostrophes

```
> > > >
> > > > Had you wanted to use a single quote as the delimiter, you'd use:
> > > >
> > > >         .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > > > cChange) VALUES (' & CurrentUser & "', #" & Date() & "#, ' " &
> > > > Replace(RecordChanged, "'", "'') & "')"
> > > >
> > > > Exaggerated for clarity, that's:
> > > >
> > > >         .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > > > cChange) VALUES (' & CurrentUser & "', #" & Date() & "#, ' " &
> > > > Replace(RecordChanged, " ' ", " ' ' ") & " ' ")
> > > >
> > > > BTW, if there's a chance that people in other countries will be
using
> > your
> > > application, you should be aware that putting Date() into the SQL
like
> > that
> > > > may not always work. If your user has the Short Date format set to
> > > > dd/mm/yyyy (as it is in much of the world), the SQL statement will
not
> > work
> > > > for the first 12 days of each month. My advice is to format the date
> > > > explicitly to avoid the possibility of problems:
> > > >
> > > >         .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
dDateTime,
> > > > cChange) VALUES (' & CurrentUser & "', " & Format(Date(),
> > > > "\#mm\dd\yyyy\#") & ", ' " & Replace(RecordChanged, "'", "'') &
" ' ")
> > > >
> > > >
> > > > --
> > > > Doug Steele, Microsoft Access MVP
> > > > http://I.Am/DougSteele
> > > > (No private e-mails, please)
> > > >
> > > >
> > > > "Pete" <Pete@discussions.microsoft.com> wrote in message
> > > > news:1A6EBE46-B2B0-4BDA-A144-D67899F75E48@microsoft.com...
> > > > > We have an application with an audit table to keep track of who
> > > > changed
> > > > what
> > > > > and when. This is updated using the following code:
> > > > > Dim cmd1 As ADODB.Command
> > > > > Set cmd1 = New ADODB.Command
> > > > > With cmd1
> > > > > .ActiveConnection = CurrentProject.Connection
> > > > > .CommandText = "INSERT INTO tblAuditTrail(cUserCode,
> > > > > dDateTime,
> > > > > cChange) VALUES (' & CurrentUser & "', #" & Date() & "#, " &
Chr(34)
> > > > > &
> > > > > > RecordChanged & Chr(34) & " )"
> > > > > .CommandType = adCmdText
> > > > > .Execute
> > > > > End With
> > > > > Set cmd1 = Nothing
```

microsoft.public.access.modulesdaovba: Re: INSERT Query problem with Quotes & Apostrophes

```
> > > > >
> > > > > The Chr(34)s were added as I ran into problem if the value of the
> > > > > RecordChanged string contained an apostrophe (').
> > > > >
> > > > > I now have a problem as a user wants to write a record containg
quotes
> > (")
> > > > > and an apostrophe ('). I have written a function that strips these
> > > > > characters
> > > > > out of the string, but ideally I would like to audit exactly what
the
> > user
> > > > > has entered. I have tried playing around with adding Chr(39) into
the
> > > > > above
> > > > > CommandText but not had any luck. Help please!
> > > > >
> > > > > --
> > > > > Peter Schmidt
> > > > > Ross-on-Wye, UK
> > > >
> > > >
> > > >
> >
> >
> >
```