



Få sin querystring til at fungere. (Nybegyndere)

Artikelen henvender sig til nybegyndere der har problemer med at få sin querystring til at fungere (Access/ASP).

Skrevet den **04. Feb 2009** af **michael_stim** | kategorien **Databaser / Access** | ★★☆☆☆☆

Jeg skriver denne artikel, fordi der er en hel del mennesker her på eksperten der har problemer med sine SQL-query's. Jeg tager ikke alle de forskellige udtryk med (ORDER BY, GROUP BY osv osv), da artikelen kun går ud på at få sin SQL-string til at være korrekt. Det er måske en idé til en senere artikel.

Jeg har taget udgangspunkt i Access og ASP, da jeg ser det er der, der er flest problemer med at få sin query til at fungere rigtigt.

1. Udskrive queryen

Når man skal eksekvere sin query og der er variabler indblandet, er det altid en god idé at udskrive den først (inden man eksekverer den). På den måde er det lettere at få et overblik over queryen, og se om man får alle sine variabler med.

Eksempel:

```
strID=request.querystring("id") 'Din variabel som evt. kommer fra en anden side
SQLstr="SELECT * FROM tabel WHERE id=" & strID
response.write SQLstr 'udskriver stringen og du kan se om du får dine variabler med
```

Og så skal man lige huske at afkommentere selve eksekveringen.

2. (dobbel)Appestrof eller ej

Hvornår skal man have apostrof om sine inputs/variabler? Man kan som tommefingerregel sige at når der er bogstaver indblandet så skal der apostrof om. Dvs tal og autonummering skal der ikke apostrof om.

Og så er der dato-feltet som er lidt specielt. Der skal der # om.

Eksempel:

```
ddato=request.form("dagensdato")'Henter variablerne fra en form (date-felt)
alder=request.form("alder")'(talfelt)
navn=request.form("navn")'(tekstfelt)
SQLstr="INSERT INTO tabel (ddato, alder, navn) VALUES(("#& ddato &","#, "& alder &","& navn &")"
```

Som i kan se, så er der både enkel og dobbel apostrof om navnet. Men man skal huske på, at den dobbelte kun er til for at udskille variablerne. Man kunne have klaret den med dobbelte apostroffer ("""& navn &"""), men personligt syntes jeg det ser bedre og mere overskueligt ud med enkelte apostroffer.

3. Reserverede ord

Access (og andre databaser for den sags skyld) har en hel del reserverede ord. Dvs ord der ikke må forekomme i dine tabel/databasenavne. Er det sådan at du har brugt et reserveret ord, kan det løses med at skrive [] rundt om det reserverede ord i din sql-sætning, men undgå helst at bruge reserverede ord. Er man i tvivl er det bare at sætte [] om, så er man på den sikre side.

Til sidst i artikelen er der en liste over reserverede ord i Access 2002.

Eksempel:

```
SQLstr="INSERT INTO tabel ([date], [name], adresse, alder) VALUES(#2004-03-14#, "Michael", "Sverige", 8)"
```

Dette er min første artikel, og er der noget der skulle være helt hen i skoven, må i gerne skrive det så det kan blive rettet. Men tror jeg har fået det med, hvor jeg oplever de fleste fejl bliver begået. Håber den kan hjælpe nogen.

Liste over reserverede ord i Access 2002:

```
-A
  ADD
  ALL
  Alphanumeric
  ALTER
  AND
  ANY
  Application
  AS
  ASC
  Assistant
  AUTOINCREMENT
  Avg
-B
  BETWEEN
  BINARY
  BIT
  BOOLEAN
  BY
  BYTE
-C
  CHAR, CHARACTER
  COLUMN
  CompactDatabase
  CONSTRAINT
  Container
  Count
  COUNTER
  CREATE
  CreateDatabase
  CreateField
  CreateGroup
  CreateIndex
  CreateObject
  CreateProperty
  CreateRelation
```

CreateTableDef
CreateUser
CreateWorkspace
CURRENCY
CurrentUser
-D
DATABASE
DATE
DATETIME
DELETE
DESC
Description
DISALLOW
DISTINCT
DISTINCTROW
Document
DOUBLE
DROP
-E
Echo
Else
End
Eqv
Error
EXISTS
Exit
-F
FALSE
Field, Fields
FillCache
FLOAT, FLOAT4, FLOAT8
FOREIGN
Form, Forms
FROM
Full
FUNCTION
-G
GENERAL
GetObject
GetOption
GotoPage
GROUP
GROUP BY
GUID
-H
HAVING
-I
Idle
IEEEDOUBLE, IEEESINGLE
If
IGNORE
Imp
IN
INDEX
Index, Indexes

INNER
INSERT
InsertText
INT, INTEGER, INTEGER1, INTEGER2, INTEGER4
INTO
IS
-J
JOIN
-K
KEY
-L
LastModified
LEFT
Level
Like
LOGICAL, LOGICAL1
LONG, LONGBINARY, LONGTEXT
-M
Macro
Match
Max, Min, Mod
MEMO
Module
MONEY
Move
-N
NAME
NewPassword
NO
Not
Note
NULL
NUMBER, NUMERIC
-O
Object
OLEOBJECT
OFF
ON
OpenRecordset
OPTION
OR
ORDER
Orientation
Outer
OWNERACCESS
-P
Parameter
PARAMETERS
Partial
PERCENT
PIVOT
PRIMARY
PROCEDURE
Property
-Q

Queries
Query
Quit
-R
REAL
Recalc
Recordset
REFERENCES
Refresh
RefreshLink
RegisterDatabase
Relation
Repaint
RepairDatabase
Report
Reports
Requery
RIGHT
-S
SCREEN
SECTION
SELECT
SET
SetFocus
SetOption
SHORT
SINGLE
SMALLINT
SOME
SQL
StDev, StDevP
STRING
Sum
-T
TABLE
TableDef, TableDefs
TableID
TEXT
TIME, TIMESTAMP
TOP
TRANSFORM
TRUE
Type
-U
UNION
UNIQUE
UPDATE
USER
-V
VALUE
VALUES
Var, VarP
VARBINARY, VARCHAR
-W
WHERE

WITH
Workspace
-X
Xor
-Y
Year
YES
YESNO

Følgende symboler må ikke bruges i felt/-objectnavne:

.
/
*
;
:
!

&
-
?
"
'
\$

Kommentar af thesurfer d. 28. Apr 2005 | 1

Det er ikke en god artikel.. men den er heller ikke 100% dårlig. Læs hellere: <http://html.dk/tutorials/asp/>

Kommentar af huset d. 29. Apr 2005 | 2

Okay arbejde

Kommentar af hyberpreprocessor d. 28. Apr 2005 | 3

God artikel, dejlig gratis. :D:D

Kommentar af mike1963 d. 25. Jun 2010 | 4

Du gør dig umage og så skal du have ros for det.

men en lille fælde ifbm dato og MS. Jeg opgav Access da følgende skete: 5 december blev til 12 maj

altså #05/12/....# #12/05/....# dur ikke og MS havde en fejl i deres ugenr DateAdd('W' ...) med uge 53 i 2007

Efter mange år med disse to gode rutiner DateAdd og DateDiff må man bare sige at når det fejler for MS så er der kun ens egen logik man kan sætte istedet - ærgeligt