Denne guide er oprindeligt
udgivet på Eksperten.dk

COMPUTERWORLD

SQL for MySQL-begyndere

I denne artikel vil jeg prøve at beskrive MySQL på begynderniveau. SQL står for Structured Query Language, og er et sprog til at kommunikere med databaser - som fx MySQL, MSSQL, Access m.fl.

Skrevet den **02. Feb 2009** af **mikl-dk** I kategorien **Databaser / MySQL** | ★☆☆☆☆

Introduktion

I denne artikel vil jeg prøve at beskrive SQL (udtales "ess-que-el") på begynderniveau. SQL står for Structured Query Language, og er et sprog til at kommunikere med databaser - som fx MySQL, MSSQL, Access etc. Man kommunikerer via noget, der hedder forespørgsler - man bruger dog oftest det engelske ord, query: one query, a lot of queries.

Data ligger i noget, der hedder tabeller (tables). Tabeller (tables) ligger samlet i databaser (databases). Man kan altså have mange tabeller (tables) i en database. Data i tabellen (table) er organieret i kolonner (columns), der har faste navne (names) og datatyper (datatypes). De data man sætter ind og opdaterer i tabellen, kaldes rækker (rows eller records).

SQL er ikke casesensitive - det betyder, at det er lige meget, om du skriver SQL-kommandoen med stort eller småt. Jeg plejer - for læsevenlighedens skyld - at skrive reserverede ord med stort. Du kan se eksempler på dette i de kommende afsnit.

Jeg har ved nogle syntakser brugt skarpe parenteser [og] - det angiver, som ved alle andre ting, at det er valgfrit at angive det, der er i den skarpe parentes.

4 grundlæggende handlinger

Når man arbejder med databaser, er der 4 handlinger, man kan udføre. Disse er:
- SELECT (udtræk af data fra tabel)
- INSERT INTO (indsættelse af data i tabel)
- UPDATE (opdatér data i tabel)
- DELETE (slet data i tabel)

Det er disse 4 handlinger, jeg vil gennemgå overfladisk i denne artikel, så man kan lave de mest basale ting i forbindelse med databasekommunikation.

Kriterier

WHERE
Der er en ting, man bruger meget i SQL - nemlig WHERE. Dette er til at give kriterier til SQL-forespørgslerne. Fx kriterier, der er for de data, der skal:
- udtrækkes
- opdateres
- slettes

Som den opmærksomme læser har opdaget - det er dig! :) - svarer det til 3 af de førnævnte handlinger.

Man bruger altså næsten altid WHERE, medmindre, at man ønsker alle data i den angivne tabel manipulerede af den handling, man kører. Med WHERE kan man vælge, at bestemte kolonner i databasen skal være:
< (mindre end)
<= (mindre eller lig med)
= (lig med)
>= (større eller lig med)
> (større end)
<> (forskelligt fra)
LIKE (her kan man søge efter et mønster)
BETWEEN (mellem et inklusivt område - kaldet inclusive range)

LIMIT
LIMIT bruges også til at begrænse sit udtræk af data. Hvor WHERE bruges til at hente data, der opfylder kriterier, så bruges LIMIT til at beskære det udsnit af data, ens forespørgsel (query) ellers ville returnere. Hvis ens query ville resultere i, at man fik fx 10 rækker (rows) returneret, så kan man med LIMIT sige, at man kun fx ønsker de 5 af rækkerne.

Syntaksen for LIMIT er således:

```
LIMIT [start,] antal_rows
```

Det betyder, at hvis man angiver et kriterium, så er det antallet af rækker, man ønsker returneret. Hvis man giver to kriterier, med komma som separator, så er det første kriterium, hvilken række i de returnerede rækker, man starter ved, og nummer to, hvor mange rækker fra startrækken man ønsker at få med. Startrækken kaldes ofte offset. Man kan begrænse det førnævnte eksempel således:

```
LIMIT 5
```

ORDER BY
Dette er en kommando, man kan bruge til at vælge, hvordan rækker (kaldes ofte rows), man får returneret fra en query, skal sorteres.

Syntaksen for ORDER BY er således:

```
ORDER BY kollonne [DIRECTION]
```

Kolonne beskriver, hvilken kan ønsker, der primært skal sorteres efter.
[DIRECTION] er retningen på sorteringen. ASC betyder, at det sorteres ascending (på dansk: opadgående - også sagt på en anden måde: mindste er øverst og største nederst). DESC betyder det omvendte af ASC: her er det største øverst, og det mindste nederst. Hvis man ikke skriver noget, så bliver det sorteret ascending.

SELECT

Den simple syntaks for SELECT er:

```
SELECT * FROM tabel
```

Ved dette får man alle rækker og alle kolonner i tabel.

Man kan også vælge kun at tage nogle af kolonnerne, men stadig alle rækkerne, da der stadig ikke er nogen WHERE-clause på (betyder WHERE-betingelse):

```
SELECT navn, tlf, skonummer FROM tabel
```

Her vælges kun kolonnerne navn, tlf og skonummer.

Hvis man fx kun ønsker navnet og telefonnummeret på de personer, der bruger nummer 44 i sko, kan man gøre det således:

```
SELECT navn, tlf FROM tabel WHERE skonummer = 44
```

Her fås en liste over de personer, der bruger nummer 44. Hvis man ønsker dem, der bruger nummer 44 i sko og hvis navn starter med A, og de skal sorteres i alfabetisk rækkefølge efter navnet:

```
SELECT navn, tlf FROM tabel WHERE skonummer = 44 AND navn LIKE 'A%' ORDER BY navn
```

Her betyder navn LIKE 'A%', at navnet skal starte med A, og vi er ligeglade med, hvad der kommer bagefter - det er tegnet %, der markerer det. Grunden til, at der er ' omkring er, at det er en streng (string), vi arbejder med.

Nu vil jeg gerne have en liste med 5 personer, der har 55 i deres telefonnummer. Måden, vi vælger de 5 på er, at vi vil have dem, der har største fødder:

```
SELECT navn, tlf, skonummer FROM tabel WHERE tlf LIKE '%55%' ORDER BY skonummer DESC LIMIT 5
```

INSERT INTO

Denne handling bruges til at indsætte noget data i vores tabel. Hvis der ikke er noget i tabellen, så kan vi jo heller ikke hive noget ud med SELECT.

Syntaksen for INSERT INTO er således:

```
INSERT INTO tabel (kol_1, kol_2, ...) VALUES (vaerdi_1, vaerdi_2, ...)
```

Et par eksempler kan være, at jeg vil sætte nogle personer ind, med oplysninger om telfonnumre og skonumre:

```
INSERT INTO tabel (navn, tlf, skonummer) VALUES ('Anders And', '555', '44')
INSERT INTO tabel (navn, tlf, skonummer) VALUES ('mikl-dk', '40******', '46')
```

Jeg kan også sætte blot et navn ind, hvis jeg endnu ikke kender hans telefonnummer og skonummer:

```
INSERT INTO tabel (navn) VALUES ('Bonanza')
```

Husk igen at bruge ' ved strenge (strings).

Man behøver ikke nødvendigvis at angive kolonnerne, men jeg anbefaler stærkt, at man gør, som jeg har beskrevet.

UPDATE

Denne handling bruges til at opdatere data i vores tabel. Syntaksen er:

```
UPDATE tabel SET kol_1=vaerdi_1 [, kol_2=vaerdi_2[, ...]] [WHERE betingelse] [ORDER BY sortering] [LIMIT antalsbegrænsing]
```

Nu kan jeg opdatere Bonanzas record i tabellen:

```
UPDATE tabel SET tlf='158755', skonummer=21 WHERE navn = 'Bonanza'
```

Denne opdatering opdaterer alle, hvis navn er Bonanza. Man kan også sikre sig, at der kun sker fx en opdatering:

```
UPDATE tabel SET tlf='158755', skonummer=21 WHERE navn = 'Bonanza' LIMIT 1
```

Nu får jeg lige pludselig den idé, at de 5 med de største sko og med navnet sorteret alfabetisk, skal have et øgenavn (jeg har selv store fødder :)):

```
UPDATE tabel SET navn=CONCAT(navn, ' aka BIGFOOT') ORDER BY skonummer DESC LIMIT 5
```

Nu kommer de 5 med de største fødder til at hedde fx:
Jens aka BIGFOOT
Peter aka BIGFOOT
Michael aka BIGFOOT
Anne aka BIGFOOT
Kasper aka BIGFOOT

Dette er testet i MySQL, hvor CONCAT er en funktionen, der klistrer strenge sammen (her det originale navn med et statisk streng). Jeg ved ikke, om det virker i andre typer.

DELETE

Denne handling bruges til at slette records i tabellen med. Syntaksen for denne er:

```
DELETE FROM tabel [WHERE betingelse] [ORDER BY sortering] [LIMIT antalsbegrænsing]
```

Denne bruges på samme måde som fx UPDATE, hvor man blot sletter records i stedet for at opdatere dem. Man kan fx slette de 5, der har mindst fødder:

```
DELETE FROM tabel ORDER BY skonummer LIMIT 5
```

Man kan også slette dem med øgenavnet BIGFOOT:

```
DELETE FROM tabel WHERE navn LIKE %BIGFOOT%'
```

Læs mere

En god side, hvor man kan finde mere om SQL er http://www.mysql.com, der beskriver nærmere om MySQL's indbyggede muligheder.

Man kan også læse andre tutorials som fx W3Schools', der er at finde på http://www.w3schools.com/sql/default.asp

Og til sidst, men slet ikke mindst, er Google (http://www.google.dk) jo stadig din ven ;)

Kommentar af skwat d. 14. Apr 2004 | 1

Du skulle nok sætte noget ligende det du har skrevet under 4 grundlæggende handlinger i brødteksten

Kommentar af iss d. 28. Jun 2006 | 2

Kommentar af alexander_j d. 04. Feb 2008 | 3

Lidt for meget begynder - savner noget om JOIN af forskellig art.

Kommentar af human d. 14. Apr 2004 | 4

Synes det er en fin artikel til begyndere, som den jo også er lavet til.

Pas dog på med hvordan du behandler din tekst. Det er meget forvirrende og irriterende at se på alle de paranteser i din indledning. Især når de enlig ikke er nødvendige.
Synes du skal tænke på det en anden gang.

Kommentar af per-d d. 11. Feb 2005 | 5

--Når man arbejder med databaser, er der 4 handlinger, man kan udføre. Disse er:--

Øhh er det bare mig men skriver du ikke her det samme som at der kun findes disse 4 handlinger hvilket er en direkteløgn

Kommentar af baso d. 15. Apr 2004 | 6

Jer er begynder, og artiklen skærer nogle ting ud i pap omkring SQL. Jeg fik noget ud af artiklen, og derfor må den være god.

Kommentar af hl3358 d. 13. Dec 2004 | 7

Fin intro til sql, det var bare ikke det jeg var på jagt efter..

Kommentar af ger2001 d. 03. Jun 2004 | 8

Kommentar af miffe d. 03. Nov 2004 | 9

God til mig, som begynder...
Nu mangler jeg bare noget at bygge videre på.. :)

Kommentar af qtax87 (nedlagt brugerprofil) d. 25. May 2006 | 10

Synes ikke så godt om denne artikel da den ikke giver det vilde indblik.
Men vil nærmere hellere forslå bruger at kigge på: www.sql-tutorial.net

Kommentar af pildal d. 21. Oct 2004 | 11

God intro

Kommentar af larsen45 d. 17. Jan 2005 | 12

Blev nået skuffet. Havde regnet med mere information.

Kommentar af u6e96b22 d. 03. Aug 2005 | 13

du mangler at skrive hvordan man opsætter en SQL server og hvordan du laver en DB, faktisk det vigtigste, da man ikke kan bruge det du har skrevet til en **** uden at vide hvordan man laver en SQL server

Kommentar af benhur d. 27. Apr 2005 | 14

Udmærket nybegynder artikkel.

Kommentar af janjacobsen d. 15. Apr 2004 | 15

Man kan selvfølgelig finde de oplysninger mange andre steder, men sjældent er informationerne samlet så kompakt og godt.
Super ;-)

Kommentar af kongbak d. 28. Dec 2004 | 16

ikke hvad jeg søgte desværre

Kommentar af swn d. 16. Apr 2004 | 17

Godt sted for en nybegynder at starte.
Meget kompakt oversigt og eksempler.

Kommentar af thomas-a d. 21. Jul 2004 | 18

kan ikke bruge den til noget når jeg skal opsætte en mqsql server

Kommentar af iioleii d. 23. May 2004 | 19

Kommentar af uniqe d. 12. Nov 2004 | 20

blev sq ikk meget bedre til MySQL efter at ha læst dette.... fatter stadig NADA ! :S

Kommentar af venchil d. 14. Feb 2005 | 21

Jeg er selv nybegynder til MySQL, og jah, den fik mig til at forstå MySQL... Så nu er denne artikel grundlaget for min indtil videre meget simple programmering i MySQL. Tak for det :).

Kommentar af anderdehn d. 13. Dec 2004 | 22

har ikke læst den endnu, men bare kopieret den. Har haft svært ved at komme igang med MySQL, håber den hjælper

Kommentar af ohmygod d. 17. Dec 2004 | 23

Lever ikke helt op til hvad jeg troede, i særdelshed håbede på da jeg har et mega bøvl med opsætningen som jeg har været igennem flere gange nu... Men stoffet fejler ikke noget da =)

[Gid der var nogen der gad lave en guide til installation af MySQL hvor man kunne kontrollere at det hele var gjort rigtigt step by step, så man vidste HVOR det gik galt hvis, eller i mit tilfælde, når det går galt... eller blot hjælpe mig igennem (Apache+PHP installed and working.. SQL.. lol... i WISH! ;(..)

Kommentar af shemah d. 25. Jan 2005 | 24

Kommentar af lazeric d. 16. Mar 2006 | 25

Meget basal information. God for nybegyndere, men jeg synes ikke der står hvordan man sætter database connection?

Kommentar af superfisker d. 09. Jul 2008 | 26

Havde helt klart regnet med noget mere information !

Kommentar af rix17172 d. 26. Jan 2010 | 27

nu ved jeg godt at MySQL er meget stort men når men skriver MySQL for begynder trode jeg at der ville stå hvordan man forbinder MySQL med det man ville have den til men ellers en meget fin atikel som fortæler meget