**SQL - Grundlæggende DML**

**Kilde:** [**https://it.systime.dk/**](https://it.systime.dk/)

* [Select](https://it.systime.dk/index.php?id=995#c3535)
* [Insert](https://it.systime.dk/index.php?id=995#c3536)
* [Update](https://it.systime.dk/index.php?id=995#c3538)
* [Delete](https://it.systime.dk/index.php?id=995#c3541)

I denne tekst gennemgås kortfattet klassisk SQL syntaks for de 4 grundlæggende DML-operationer: Læsning af data (SELECT), indsættelse af data (INSERT), opdatering af data (UPDATE), og sletning (DELETE). Til formålet antager vi, at vi har en tabel Person(PersonId, PersonNavn) med følgende indhold til at starte med:

| **PersonId** | **PersonNavn** |
| --- | --- |
| 1 | Herman Borgkvist |
| 2 | Pernille Rosenlund |
| 3 | Hubert Svendsen |
| 4 | Svend Hubertsen |
| 5 | Silja Agresso Mygdahl |

**Select Id**

For at læse indholdet af tabellen, benytter vi følgende forespørgsel:

SELECT PersonId, PersonNavn

FROM Person

Forespørgslen udskriver tabelindholdet som vist ovenfor. Er man i tvivl om kolonnenavne i en tabel, kan man bruge den hurtige form SELECT \* FROM <tabelnavn> for at hente alle kolonnerne.

**Insert Id**

Vil vi indsætte en ny række med person-data i tabellen, benytter vi følgende forespørgsel:

INSERT INTO Person (PersonId, PersonNavn)

VALUES (6, 'Gertrud Hörnqvist')

Resultatet af at afvikle forespørgslen er typisk en besked om, at indsættelsen er udført (hvis altså ikke noget gik galt undervejs – såsom brud på databegrænsninger). En sådan besked kan fx se ud som følger:

(1 row(s) affected)

Vil man se tabelindholdet på ny, kan man bruge SELECT-statementet fra ovenfor, og denne gang vil resultatet af at læse fra tabellen være:

| **PersonId** | **PersonNavn** |
| --- | --- |
| 1 | Herman Borgkvist |
| 2 | Pernille Rosenlund |
| 3 | Hubert Svendsen |
| 4 | Svend Hubertsen |
| 5 | Silja Agresso Mygdahl |
| 6 | Gertrud Hörnqvist |

**Update Id**

Nu ønsker vi at opdatere alle personer til at hedde Hansen til efternavn. Det gøres således:

UPDATE Person

SET PersonNavn = PersonNavn + ' Hansen'

Output af at køre forespørgslen vil ligne nedenstående

(6 row(s) affected)

SELECT’er vi nu tabellens indhold, ser vi, at opdateringen er gået igennem:

| **PersonId** | **PersonNavn** |
| --- | --- |
| 1 | Herman Borgkvist Hansen |
| 2 | Pernille Rosenlund Hansen |
| 3 | Hubert Svendsen Hansen |
| 4 | Svend Hubertsen Hansen |
| 5 | Silja Agresso Mygdahl Hansen |
| 6 | Gertrud Hörnqvist Hansen |

**Delete Id**

Nu ønsker vi at slette alle personer, hvis fornavn starter med ’H’. Det kan være en god ide at starte med at SELECTE disse personer, for at sikre sig, at vi har formuleret udvælgelseskriteriet korrekt:

SELECT PersonId, PersonNavn FROM Person

WHERE PersonNavn LIKE 'H%'

Ja, den forespørgsel var god nok. Så er vi klar til at skrive et DELETE statement, som er en let omskrivning:

DELETE FROM Person

WHERE PersonNavn LIKE 'H%'

Bemærk, at DELETE-statementet, som det eneste af de 4 grundlæggende DML-statements, ikke omtaler specifikke kolonne-navne. Der skal blot slettes hele rækker, der opfylder visse betingelser (angivet i WHERE-klausulen).

Tjekker vi tabelindholdet nu, vil vi se, at det er reduceret til:

| **PersonId** | **PersonNavn** |
| --- | --- |
| 2 | Pernille Rosenlund Hansen |
| 4 | Svend Hubertsen Hansen |
| 5 | Silja Agresso Mygdahl Hansen |
| 6 | Gertrud Hörnqvist Hansen |

ISBN: 9788761635716. Copyright forfatterne og Systime A/S 2020