

DB1102, oppgaver til økt 03

Oppgave 1 - begreper & terminologi

Forklar **kort/enkelt** følgende terminologi og databasebegreper: (se gjerne læreboka for hjelp)

- Fremmednøkkel
- Primærnøkkel
- Nullmerke ("NULL")
- Løpenummer ("AUTO_INCREMENT")

TIPS: Gjør det skriftlig i word: (e.l.)

Oppgave 2 - SQL

For de første 11 oppgavene, opprett en tom MySQL eksempeldatabase ("create schema"). Kall den **test** e.l. Sett denne som "default schema".

Skriv SQL mot **test** databasen (schema) som utfører følgende:

1. Opprett tabellen `person`. Tabellen skal inneholde:

`ID`, `int`, autogenerated.

`Personnummer`, `char(11)`, skal ikke kunne være null og skal være unik.

`Navn`, `varchar(100)`, skal ha defaultverdi 'Ukjent'.

`EPost`, `varchar(100)`, skal være unik.

Som primærnøkkel skal kolonnen `ID` angis.

Sjekk at tabellen er opprettet ved å refresh'e object browser'en til venstre i MySQL workbench, deretter klikke test og Tables. Eller ved å skrive SQLen: `DESC person`.

2. Legg inn en rad i tabellen `person`. Spesifiser følgende data:

Personnummer: '12345678901 '

Navn: 'Ola Nordmann '

EPost: 'ola@nordmann.no '

Skriv en passende spørring for å se at dataene ligger i tabellen.

3. Du finner ut at det er kjekt å registrere fødselsdato også. Modifiser den eksisterende `person` tabellen, så den også inneholder kolonnen:

`Foedselsdato`, `date`.

Sjekk at tabellen har fått med seg endringene. (Tilsvarende fremgangsmåte som i oppg 1.)

4. Legg inn en ny rad i tabellen `person`. Spesifiser følgende data:

personnummer: '98765432109 '

Skriv en passende spørring for å se at data ligger i tabellen som forventet.

5. Oppdater raden med personnummeret '98765432109', slik at Foedselsdato til denne blir: '1990-01-01'. **Merk:** Datofelt formateres internt i MySQL med år, så måned, så dag.

6. Legg inn deg selv i person tabellen! Fyll inn alle felter (trenger ikke bruke ekte info da).

7. Vi innser at databaseløsningen vår er litt snevert designet: Hver person kan bare ha 1 epost adresse. Opprett en egen tabell epost med følgende kriterier:

Adresse, varchar(75), skal ikke kunne være null og skal være unik.

Type, varchar(50).

Person_ID, int, skal ikke kunne være null.

Som primærnøkkel skal kolonnen Adresse angis.

Som fremmednøkkel Person_ID som refererer tabellen person sin kolonne ID.

Skriv en passende spørring for å se at tabellen er opprettet som forventet.

8. **Hint:** Kap. 3.3.1 i pensumboka.

Skriv en SQL som kopierer epostadresser og IDer fra person tabellen over til Adresse og Person_ID i epost tabellen.

9. (Hvis du har klart nummer 8)

Nå som vi har flyttet epost ut i en egen tabell, så kan vi slette epostkolonnen fra person-tabellen. Vi ønsker å fjerne hele kolonnen, ikke bare dataene som ligger der.

10. Lag en tabell tilsvarende den i punkt 1 som heter personCopy. (Du trenger ikke kopiere data inn i den.) Sjekk at denne er opprettet, før du sletter den igjen. (Poenget er å få testet kommandoen for å slette tabell, uten å ødelegge de to fine tabellene du har jobbet med en stund nå.)

11. **VANSKELIG:** Hvis vi henter ut navn og fødselsdato fra person tabellen, kommer dato på MySQL sitt for oss "baklengsformat". Omformater output så dato har formatet:

31-01-2001. (<-- Eksempel på dato med rett format.)

Tips: Les om (w3schools eller google) MySQL funksjonen: DATE_FORMAT()

Legg også inn dato for Ola Nordmann. Han skal ha fødselsdatoen 20.02.1912 (gammel mann!). Skriv den inn på dette formatet, men bruk STR_TO_DATE() så MySQL skjønner hvilket format datoen kommer på.

Sjekk at tabellen er slettet.

Benytt MySQL eksempeldatabasen ("schema") som heter **world** for de siste oppgavene. (Sett **world** som "default schema".)

Skriv SQL mot **world** databasen (schema) som utfører følgende:

- 12 Benytt max()-funksjonen til å hente ut den største ID-en i world.city.
SELECT * fra information_schema.tables for world-databasen (WHERE table_schema = 'world'). Kan du finne hvor mange rader du har i de ulike tabellene dine? Hva heter kolonnen som angir dette? Kan du finne kolonnen som gir oss neste verdi for city.ID?
 - 13 Slett raden som inneholder byen 'Bærum' fra city tabellen. (Dette er ingen by, og skal ikke ligge i city tabellen!)
- Skriv en passende spørring for å se at data er slettet som forventet.
- 14 Undersøk om lærebok har rett når den skriver (s.79): «MySQL har fra versjon 8 støtte for CHECK-regler. Eldre versjoner godtok at man skrev CHECK, men sjekket ikke regelen.»
Hvordan kan du undersøke om utsagnet er riktig (for din versjon)? Del svaret ditt (og hvordan du kom fram til det) i Mattermost!

Tips: Ved siden av å spørre veilederne, foreleser og å benytte forumet på Canvas, se på egenhånd på <http://www.w3schools.com/sql/default.asp>. Her finner du svar på mange av dine SQL spm. :-) Husk også at læreboka har mange fine ressurser, inklusive svar på spørsmål fra boka. Du vil lære mye av å løse oppgavene i boka også!

Du kan også bruke tid på å utforske andre databaser fra læreboka, eller i Workbench/på nett.
(Eks: SAKILA databasen.)