

Øving 3: SQL del 1 (obligatorisk)

Skrevet av Felix Albrigtsen, felixalb@stud.ntnu.no

Innleveringsfrist: se Blackboard

Tidligste godkjenning: datoer blir annonsert

Løsningsforslag legges ut i etterkant.

Alle obligatoriske øvinger må være godkjente for å få karakter i emnet.

Oppgave 1 SQL

I denne oppgaven skal vi bruke en borettslag-databasen. Bruk følgende sql-script: byggOgBo_mysql.sql (tekstfil som finnes i BB). Scriptet inneholder data som oppgavene nedenfor spør etter. Sett opp SELECT-setninger som besvarer spørsmålene nedenfor. Kun én setning pr oppgave.

Databaseskjema:

```
MariaDB [idatt2103_ov3]> SELECT * FROM andelseier LIMIT 1;
```

```
+-----+-----+-----+-----+-----+-----+
| and_eier_nr | fornavn | etternavn | telefon | ansiennitet | bolag_navn |
+-----+-----+-----+-----+-----+-----+
|          1 | Even    | Trulsbo   | 56667743 |          3 | Tertitten |
+-----+-----+-----+-----+-----+-----+
```

```
MariaDB [idatt2103_ov3]> SELECT * FROM borettslag LIMIT 1;
```

```
+-----+-----+-----+-----+
| bolag_navn | bolag_adr | etabl_aar | postnr |
+-----+-----+-----+-----+
| Lerken     | Storgt 5 | 2000      | 6408   |
+-----+-----+-----+-----+
```

```
MariaDB [idatt2103_ov3]> SELECT * FROM bygning LIMIT 1;
```

```
+-----+-----+-----+-----+-----+
| bygn_id | bygn_adr      | ant_etasjer | bolag_navn | postnr |
+-----+-----+-----+-----+-----+
|        1 | Åsveien 100a |          3 | Tertitten | 7020   |
+-----+-----+-----+-----+-----+
```

```
MariaDB [idatt2103_ov3]> SELECT * FROM leilighet LIMIT 1;
```

```
+-----+-----+-----+-----+-----+-----+
| leil_nr | ant_rom | ant_kvm | etasje | bygn_id | and_eier_nr |
+-----+-----+-----+-----+-----+-----+
|        1 |        5 |       110 |        3 |        1 |          1 |
+-----+-----+-----+-----+-----+-----+
```

```
MariaDB [idatt2103_ov3]> SELECT * FROM poststed LIMIT 1;
```

```
+-----+-----+
| postnr | poststed |
+-----+-----+
|    2020 | Skedsmokorset |
+-----+-----+
```

Kommentar:

Flere steder i oppgavene har jeg måtte gjøre antagelser om spørsmålene, så alle løsningene er ikke deifnitive. For eksempel "Sortert etter antallet" i oppgave 10 kan være enten stigende eller avtagende.

I tillegg har jeg testet på eksempeldataen gitt i blackboard, men prøvd å gjøre setningene kompatible med alle verdier der det er mulig. Dette resulterer i noen ekstra sjekker i besvarelsen, som ikke har noen effekt på eksempeldataen, men for eksempel ville påvirket leiligheter uten noen andelseier.

Besvarelse:

1. Finn alle borettslag etablert i årene 1975-1985.

```
SELECT * FROM borettslag WHERE etabl_aar >= 1975 AND etabl_aar <= 1985;
```

2. Skriv ut en liste over andelseiere. Listen skal ha linjer som ser slik ut (tekster i kursiv er data fra databasen): “fornavn etternavn, ansiennitet: ansiennitet år”. Listen skal være sortert på ansiennitet, de med lengst ansiennitet øverst.

```
SELECT CONCAT(fornavn, ' ', etternavn, ', ansiennitet: ', ansiennitet) AS andelseier  
FROM andelseier ORDER BY ansiennitet DESC;
```

3. I hvilket år ble det eldste borettslaget etablert?

```
SELECT etabl_aar FROM borettslag ORDER BY etabl_aar ASC LIMIT 1;
```

4. Finn adressene til alle bygninger som inneholder leiligheter med minst tre rom.

```
SELECT bygn_adr FROM bygning b RIGHT JOIN (SELECT * FROM leilighet WHERE ant_rom >= 3) l  
ON b.bygn_id = l.bygn_id GROUP BY bygn_adr;
```

5. Finn antall bygninger i borettslaget “Tertitten”.

```
SELECT COUNT(*) FROM bygning WHERE bolag_navn = "Tertitten";
```

6. Lag en liste som viser antall bygninger i hvert enkelt borettslag. Listen skal være sortert på borettslagsnavn. Husk at det kan finnes borettslag uten bygninger - de skal også med.

```
SELECT COUNT(b.bygn_id) as "Antall Bygninger", bl.bolag_navn FROM borettslag bl LEFT  
JOIN bygning b ON b.bolag_navn = bl.bolag_navn GROUP BY bolag_navn ORDER BY bl.bolag_navn;
```

7. Finn antall leiligheter i borettslaget “Tertitten”.

```
SELECT COUNT(*) FROM leilighet l JOIN bygning b ON l.bygn_id = b.bygn_id  
WHERE b.bolag_navn = "Tertitten";
```

8. Hvor høyt kan du bo i borettslaget “Tertitten”?

```
SELECT ant_etasjer FROM bygning WHERE bolag_navn = "Tertitten" ORDER BY  
ant_etasjer DESC LIMIT 1;
```

9. Finn navn og nummer til andelseiere som ikke har leilighet.

```
SELECT fornavn, etternavn, telefon FROM andelseier a LEFT JOIN leilighet l
ON a.and_eier_nr = l.and_eier_nr WHERE leil_nr IS NULL;
```

10. Finn antall andelseiere pr borettslag, sortert etter antallet. Husk at det kan finnes borettslag uten andelseiere - de skal også med.

```
SELECT b.bolag_navn, COUNT(a.and_eier_nr) as antall_andelseiere FROM andelseier a
RIGHT JOIN borettslag b ON b.bolag_navn = a.bolag_navn GROUP BY b.bolag_navn
ORDER BY antall_andelseiere DESC;
```

11. Skriv ut en liste over alle andelseiere. For de som har leilighet, skal leilighetsnummeret skrives ut.

```
SELECT a.fornavn, a.etternavn, l.leil_nr FROM andelseier a LEFT JOIN leilighet l
ON a.and_eier_nr = l.and_eier_nr;
```

12. Hvilke borettslag har leiligheter med eksakt 4 rom?

```
SELECT bolag_navn FROM leilighet l JOIN bygning b ON l.bygn_id = b.bygn_id
WHERE l.ant_rom = 4 GROUP BY bolag_navn;
```

NB: tomt resultat, ingen leiligheter i eksempeldataen har 4 rom.

13. Skriv ut en liste over antall andelseiere pr postnr og poststed, begrenset til de som bor i leiligheter tilknyttet et borettslag. Husk at postnummeret til disse er postnummeret til bygningen de bor i, og ikke postnummeret til borettslaget. Du trenger ikke ta med poststeder med 0 andelseiere. (Ekstraoppgave: Hva hvis vi vil ha med poststeder med 0 andelseiere?)

```
SELECT COUNT(l.and_eier_nr) AS 'Antall andelseiere', b.poststed, b.postnr FROM
leilighet l LEFT JOIN (SELECT b.bygn_id,p.poststed,p.postnr FROM bygning b LEFT JOIN
poststed p ON b.postnr = p.postnr) b ON l.bygn_id = b.bygn_id WHERE
l.and_eier_nr IS NOT NULL GROUP BY b.poststed;
```

kortere løsning, men bruker natural join(ikke anbefalt)

```
SELECT l.and_eier_nr, p.poststed, p.postnr FROM leilighet l NATURAL JOIN bygning b
NATURAL JOIN poststed p WHERE l.and_eier_nr IS NOT NULL;
```