

Oppgave 1

a) Lag en SQL-spørring som utfører operasjonene seleksjon og projeksjon på tabellen Bok.

```
SELECT tittel,utgitt_aar FROM bok;
```

Eksempel-resultat:

```
MariaDB [oving2]> SELECT tittel,utgitt_aar FROM bok;
```

```
+-----+-----+
| tittel          | utgitt_aar |
+-----+-----+
| Tåpenes        | 1995      |
| Rebecca        | 1981      |
| Gutter er gutter | 1995      |
| Microserfs     | 1991      |
| Generation X    | 1995      |
...

```

b) Lag en SQL-spørring som utfører operasjonen produkt på tabellene Forlag og Bok. Beskriv resultatet med egne ord.

```
SELECT * FROM forlag,bok;
```

Eksempel-resultat:

```
MariaDB [oving2]> SELECT * FROM forlag,bok;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| forlag_id | forlag_navn      | adresse   | telefon  | bok_id | tittel          | utgitt_aar | forlag_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1         | Tapir           | Trondheim | 73590000 | 1      | Tåpenes        | 1995      | 7         |
| 2         | Gyldendal      | Oslo      | 22220000 | 1      | Tåpenes        | 1995      | 7         |
| 3         | Cappelen       | Oslo      | 22200000 | 1      | Tåpenes        | 1995      | 7         |
...

```

Her ser vi et produkt av forlag og bok, som et kryssprodukt. Resultatet er alle kombinasjoner/permutasjoner som er mulig. Som vi ser i tabellutdraget over gir ikke dette eksempelet så mye mening siden det viser alle bøker med alle forlag. Denne tabellen blir både lengre og bredere enn tabellene var til å begynne med, siden det må vise mange kombinasjoner med mange felter.

Om vi derimot bruker WHERE-selektorer, kan vi hente ut nyttig data med produktet.

c) Lag SQL-spørringer som utfører operasjonene likhetsforening (equijoin) og naturlig forening(natural join) på tabellene Forlag og Bok. Hva forteller resultatet?

Spesifisert equijoin:

```
SELECT * FROM forlag JOIN bok ON forlag.forlag_id = bok.forlag_id;
```

```
MariaDB [oving2]> SELECT * FROM forlag JOIN bok ON forlag.forlag_id = bok.forlag_id;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| forlag_id | forlag_navn      | adresse   | telefon  | bok_id | tittel          | utgitt_aar | forlag_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7         | Tiden           | Oslo      | 22232223 | 1      | Tåpenes        | 1995      | 7         |
| 3         | Cappelen       | Oslo      | 22200000 | 2      | Rebecca        | 1981      | 3         |
| 5         | Aschehaug      | Oslo      | 22000000 | 3      | Gutter er gutter | 1995      | 5         |
| 8         | Harper Collins | USA       | NULL     | 4      | Microserfs     | 1991      | 8         |
| 8         | Harper Collins | USA       | NULL     | 5      | Generation X    | 1995      | 8         |
...

```

Naturlig forening:

```
SELECT * FROM forlag NATURAL JOIN bok;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| forlag_id | forlag_navn      | adresse   | telefon  | bok_id | tittel          | utgitt_aar |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7         | Tiden           | Oslo      | 22232223 | 1      | Tåpenes        | 1995      |
| 3         | Cappelen       | Oslo      | 22200000 | 2      | Rebecca        | 1981      |
| 5         | Aschehaug      | Oslo      | 22000000 | 3      | Gutter er gutter | 1995      |
| 8         | Harper Collins | USA       | NULL     | 4      | Microserfs     | 1991      |

```

De to tabellene er like, med unntak av at NATURAL JOIN ikke tar med begge tilfeller av `forlag_id`, da disse to er like. NATURAL JOIN bruker navnet på feltene for å avgjøre likheten.

Alle våre bøker har en `forlag_id` (ingen er NULL), altså kommer alle med. Dersom vi hadde tomme felter i dette feltet ville de ikke blitt med i disse foreningene.

d) Finn eksempler på attributter eller kombinasjoner av attributter som er unionkompatible. Hvilke relasjonsoperasjoner krever at operandene er unionkompatible? Sett opp SQL-spørringer som utfører disse operasjonene, et eksempel på hver. Beskriv med egne ord hva spørringene gir deg svaret på.

Skjemaene i de to tabellene må være tilsvarende i lengde og datatyper. For å kombinere de to må kolonneordnes i samme rekkefølge.

Her slår vi sammen to varchar-attributter med union.

```
MariaDB [oving2]> SELECT nasjonalitet FROM forfatter UNION SELECT adresse FROM forlag;
```

```
+-----+
| nasjonalitet |
+-----+
| USA          |
| Britisk     |
| Portugisisk |
| Canadisk    |
| Norsk       |
| Svensk      |
| Trondheim   |
| Oslo        |
+-----+
```

```
8 rows in set (0.001 sec)
```

Resultatet viser oss alle nasjonaliteter og alle adresser som hører til henholdsvis en forfatter eller forlag.

Her slår vi sammen det samme attributtet (`forlag_navn`) fra forlag med adresse Trondheim og fra forlag uten telefonnummer.

```
MariaDB [oving2]> SELECT forlag_navn FROM forlag WHERE telefon IS NULL UNION SELECT forlag_navn FROM forlag WHERE telefon IS NOT NULL;
```

```
+-----+
| forlag_navn |
+-----+
| Harper Collins |
| Tapir         |
+-----+
```

```
2 rows in set (0.001 sec)
```

Oppgave 2

a) Bruk SQL til å finne navnene til alle forlagene. Hvilken eller hvilke operasjoner fra relasjon-salgebraen brukte du?

```
MariaDB [oving2]> SELECT forlag_navn FROM forlag;
```

```
+-----+
| forlag_navn |
+-----+
| Tapir       |
| Gyldendal  |
| Cappelen   |
| Universitetsforlaget |
| Aschehaug  |
| Oktober    |
| Tiden      |
| Harper Collins |
+-----+
```

```
+-----+
8 rows in set (0.001 sec)
```

Brukt seleksjon.

b) Bruk SQL til å finne eventuelle forlag (forlag_id er nok) som ikke har gitt ut bøker. Hvilken eller hvilke operasjoner fra relasjonsalgebraen brukte du?

```
MariaDB [oving2]> SELECT forlag_id FROM forlag WHERE forlag_id NOT IN (SELECT forlag_id FROM bok);
```

```
+-----+
| forlag_id |
+-----+
|         1 |
|         4 |
|         6 |
+-----+
3 rows in set (0.014 sec)
```

Brukt seleksjon og projeksjon.

c) Bruk SQL til å finne forfattere som er født i 1948. Hvilken eller hvilke operasjoner fra relasjonsalgebraen brukte du?

```
MariaDB [oving2]> SELECT * FROM forfatter WHERE fode_aar = 1948;
```

```
+-----+-----+-----+-----+-----+
| forfatter_id | fornavn | etternavn | fode_aar | dod_aar | nasjonalitet |
+-----+-----+-----+-----+-----+
|             8 | Henning | Mankell   |    1948 |    NULL | Svensk       |
+-----+-----+-----+-----+-----+
```

```
1 row in set (0.001 sec)
```

Brukt seleksjon

d) Bruk SQL til å finne navn og adresse til forlaget som har gitt ut boka 'Generation X'. Hvilken eller hvilke operasjoner fra relasjonsalgebraen brukte du?

```
MariaDB [oving2]> SELECT forlag_navn,adresse FROM forlag NATURAL JOIN bok WHERE tittel = "Generation X";
```

```
+-----+-----+
| forlag_navn | adresse |
+-----+-----+
| Harper Collins | USA    |
+-----+-----+
1 row in set (0.001 sec)
```

Brukt seleksjon og naturlig forening.

e) Bruk SQL til å finne titlene på bøkene som Hamsun har skrevet. Hvilken eller hvilke operasjoner fra relasjonsalgebraen brukte du?

```
MariaDB [oving2]> SELECT b.tittel FROM bok b,bok_forfatter bf,forfatter f WHERE bf.forfatter_id = f.forfatter_id
```

```
+-----+
| tittel      |
+-----+
| Markens grøde |
| Victoria    |
| Sult        |
+-----+
3 rows in set (0.001 sec)
```

Brukt seleksjon og projeksjon

f) Bruk SQL til å finne informasjon om bøker og forlagene som har utgitt dem. Én linje i oversikten skal inneholde bokas tittel og utgivelsesår, samt forlagets navn, adresse og telefonnummer. Forlag som ikke har gitt ut noen bøker skal også med i listen. Hvilken eller hvilke operasjoner fra relasjonsalgebraen brukte du?

Formatting: SELECT b.tittel,b.utgitt_aar,f.forlag_navn,f.adresse FROM bok b,forlag f;

Selection: SELECT * FROM forlag f LEFT JOIN bok b ON b.forlag_id = f.forlag_id;

MariaDB [oving2]> SELECT b.tittel,b.utgitt_aar,f.forlag_navn,f.adresse,f.telefon FROM forlag f LEFT JOIN bok b ON b.forlag_id = f.forlag_id;

tittel	utgitt_aar	forlag_navn	adresse	telefon
Tåpenes	1995	Tiden	Oslo	22232223
Rebecca	1981	Cappelen	Oslo	22200000
Gutter er gutter	1995	Aschehaug	Oslo	22000000
Microserfs	1991	Harper Collins	USA	NULL
Generation X	1995	Harper Collins	USA	NULL
Klosterkrønike	1982	Cappelen	Oslo	22200000
Universet	1988	Cappelen	Oslo	22200000
Nålen	1978	Cappelen	Oslo	22200000
Markens grøde	1917	Gyldendal	Oslo	22220000
Victoria	1898	Gyldendal	Oslo	22220000
Sult	1890	Gyldendal	Oslo	22220000
Benoni	1908	Gyldendal	Oslo	22220000
Rosa	1908	Gyldendal	Oslo	22220000
Et skritt	1997	Gyldendal	Oslo	22220000
Den femte	1996	Gyldendal	Oslo	22220000
Villspor	1995	Gyldendal	Oslo	22220000
Silkeridderen	1994	Gyldendal	Oslo	22220000
Den hvite hingsten	1992	Gyldendal	Oslo	22220000
Hunder	1992	Gyldendal	Oslo	22220000
Bridget Jones	1995	Aschehaug	Oslo	22000000
Se terapeuten	1998	Cappelen	Oslo	22200000
Sa mor	1996	Cappelen	Oslo	22200000
Jubel	1995	Cappelen	Oslo	22200000
Tatt av kvinnen	1993	Cappelen	Oslo	22200000
Supernaiv	1996	Cappelen	Oslo	22200000
NULL	NULL	Tapir	Trondheim	73590000
NULL	NULL	Universitetsforlaget	Oslo	23230000
NULL	NULL	Oktober	Oslo	22002200

Brukt: seleksjon, produkt, outer join