Øving 2: Relasjonsmodellen, del 2

# Oppgave 1

## Deloppgave a)

**Selection**: SELECT \* FROM bok WHERE forlag\_id = 3;

**Projection**: SELECT DISTINCT tittel, forlag\_id FROM bok;

## Deloppgave b)

**Cross**: SELECT \* FROM BOK, FORLAG;

Den nye tabellen vil ikke gi mening ettersom hvert forlag vil bli slått sammen med hver eneste bok uansett hvilket forlag boken tilhører.

## Deloppgave c)

**Equijoin**: SELECT \* FROM bok JOIN forlag ON bok.forlag\_id = forlag.forlag\_id;

**Natural join**: SELECT bok\*, forlag\_navn, adresse, telefon FROM bok NATURAL JOIN forlag

Ved natural join kan vi unngå å hente forlag\_id duplikat ganger

## Deloppgave d)

Fornavn og etternavn i forfatter og konsulent er *unionkompatible*.

**Union**: SELECT fornavn, etternavn FROM konsulent UNION SELECT fornavn, etternavn FROM forfatter;

# Oppgave 2

## Deloppgave a)

SELECT DISTINCT forlag\_navn FROM forlag;

Denne spørringen bruker projeksjon.

## Deloppgave b)

SELECT forlag.forlag\_id FROM forlag LEFT JOIN bok ON bok.forlag\_id = forlag.forlag\_id WHERE bok.forlag\_id IS NULL;

Denne spørringen bruker seleksjon og equijoin.

## Deloppgave c)

SELECT \* FROM forfatter WHERE fode\_aar = 1948

Denne spørringen bruker seleksjon.

## Deloppgave d)

SELECT forlag.forlag\_navn, forlag.adresse FROM forlag INNER JOIN bok ON forlag.forlag\_id = bok.forlag\_id AND bok.tittel = "Generation X"

Bruker seleksjon, join og projeksjon

## Deloppgave e)

SELECT bok.tittel FROM bok LEFT JOIN bok\_forfatter ON bok\_forfatter.bok\_id = bok.bok\_id LEFT JOIN forfatter ON bok\_forfatter.forfatter\_id = forfatter.forfatter\_id AND forfatter.forfatter\_id = 7;

Bruker seleksjon, join og projeksjon

## Deloppgave f)

SELECT bok.tittel, bok.utgitt\_aar, forlag.forlag\_navn, forlag.adresse, forlag.telefon FROM bok RIGHT JOIN forlag ON forlag.forlag\_id = bok.forlag\_id;

Bruker differanse