

# Repetisjon og eksamenstips

Mathias Stang og Stein Michael Storleer

24. november 2017

# Agenda

- Om eksamen
  - Digital eksamen, dere bruker Inspera på Windows-maskiner
  - Hjelpemidler: Pensumboka, 4 håndskrevne A4-sider (=2 ark)
  - 6 timer eksamen
  - 2 deler: ORM og SQL
- SQL-repetisjon
  - Ikke-utfyllende liste over hva som er pensum i SQL
  - Eksempler på spørringer dere skal kunne løse
- Div. praktisk, info om prøveeksamen, spørsmål

# CREATE TABLE

- CREATE TABLE
- Datatyper: text/varchar, int, real
- PRIMARY KEY på én eller flere verdier
- NOT NULL
- UNIQUE (for kandidatnøkler)
- Fremmednøkler på én eller flere verdier
- CHECK

# INSERT INTO

- INSERT INTO <tabell> VALUES (....)
- INSERT INTO <tabell> (feltnavn...) VALUES (...)
- INSERT INTO <tabell> SELECT ... FROM ...

# UPDATE, DELETE

- UPDATE <tabell> SET kolonne1 = verdi, kolonne2 = v... WHERE ...
- DELETE FROM <tabell> WHERE ...

# SELECT

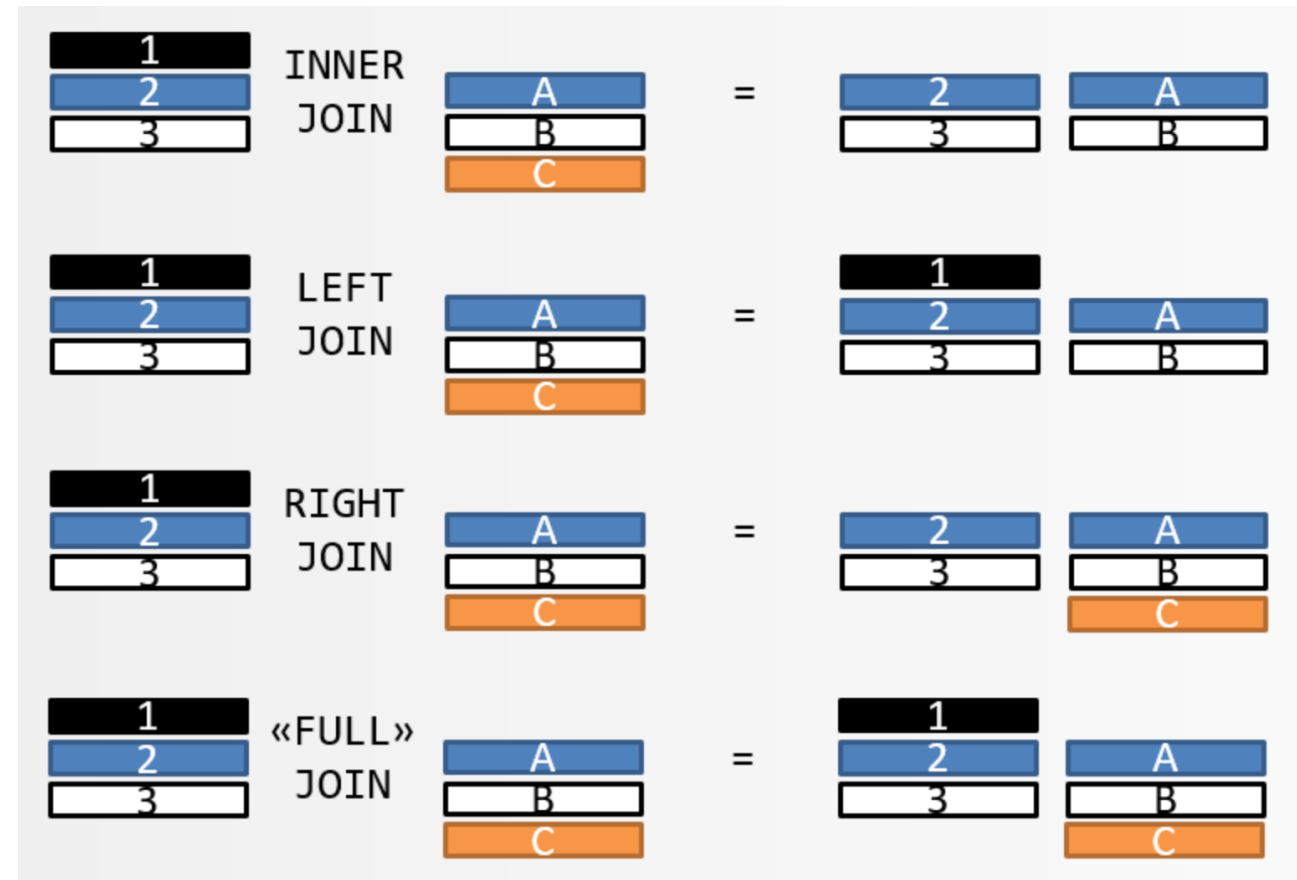
- SELECT [ DISTINCT ] [ \* | uttrykk [ [ AS ] alias ] ]
- FROM
- WHERE
- GROUP BY
- HAVING
- ORDER BY (ASC/DESC)
- LIMIT

# SELECT: Operatører

- > < >= <= != <>
- + \* - /
- ||
- AND, OR
- LIKE med wildcards % og \_
- IN (a, b, c)
- IN (SELECT ...)
- BETWEEN x AND y
- EXISTS (SELECT ...)

# JOINS

- INNER JOIN (=JOIN), NATURAL JOIN og forskjellen på disse to
- Implisitt join
- OUTER JOIN:
  - LEFT JOIN (=LEFT OUTER JOIN)
  - RIGHT JOIN (=LEFT OUTER JOIN)
  - FULL OUTER JOIN



# GROUP BY og aggregeringsfunksjoner

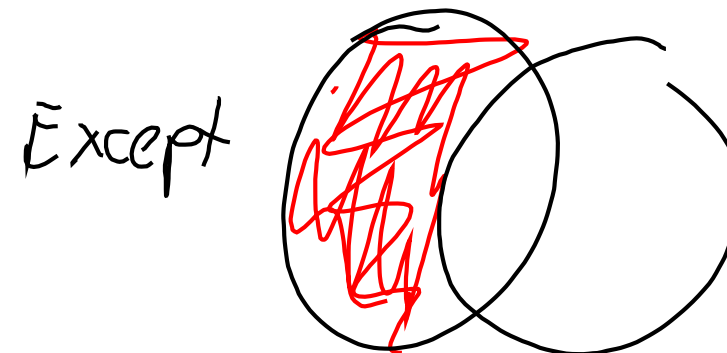
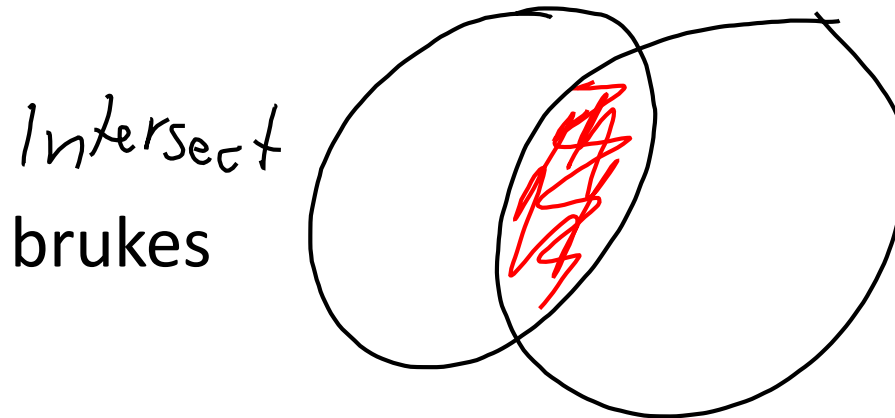
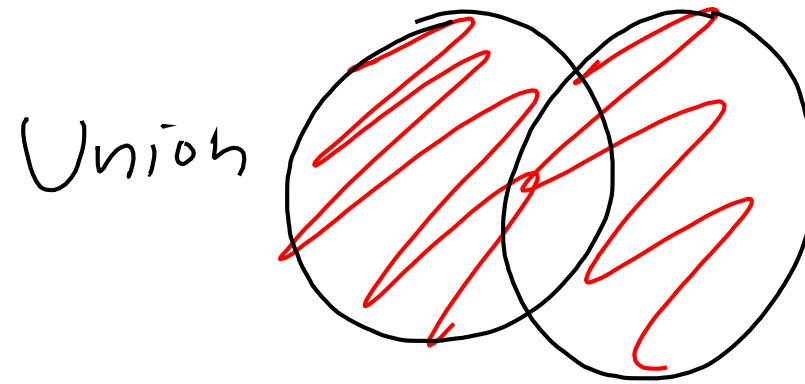
- GROUP BY på ett eller flere attributer
- Bruk av HAVING
- count, min, max, sum, avg



# Mengdeoperatorer

- UNION
- INTERSECT
- EXCEPT

+ betingelser for når disse kan brukes



# Andre funksjoner

- Dere trenger ikke kunne funksjoner som SUBSTR, LOWER, UPPER
- **Men:** Dere må kunne bruke dem om funksjonen blir definert i oppgaveteksten
- **Eksempeloppgave:** Skriv ut alle **firstname** i tabellen **person** med store bokstaver  
**Tilleggsinformasjon:** Funksjonen **upper(string)** gjør alle bokstaver i en string om til store bokstaver (f.eks. gir **upper('Ola')** resultatet **OLA**)
- Da må dere kunne skrive løsningen:  

```
SELECT upper(firstname) FROM person;
```
- Unntak: Aggregeringsfunksjoner og stringkonkatenering ||

# CREATE VIEW

- CREATE VIEW navn AS  
SELECT ...

# Eksempel fra filmdatabasen

- Finn alle filmer fra regissør med personid 582198 som har høyere rating enn gjennomsnittsratingen til denne regissøren

# Eksempel fra oblig 4

Finn filmer som både er med i sjangeren Film-Noir og Comedy

Trenger vi subspørring her?

Mage måter vi kan løse denne på:

- Mengder (INTERSECT)
- Where ... in (...)
- Where exists
- Natural join/inner join
- Gruppering og count