

IN2090 – Databaser og datamodellering

05 – Nestede spørringer

Leif Harald Karlsen
leifhka@ifi.uio.no



Universitetet i Oslo

Delspørringer

- ◆ Husk at tingene i en **FROM**-klausul er tabeller
- ◆ Husk også at resultatet av en **SELECT**-spørring er en tabell
- ◆ Så, vi kan putte en **SELECT**-spørring i **FROM**-klausulen som en tabell!
- ◆ Altså

```
SELECT <columns>
  FROM (SELECT <columns>
        FROM <tables>
        WHERE <condition>
       ) AS subquery
WHERE <condition>
```

Ekempel-delspøringer

- ◆ F.eks., for å finne antall unike kombinasjoner av land og by for alle kunder:

```
SELECT count(*)
  FROM (SELECT DISTINCT country, city FROM customers) AS d
```

- ◆ Følgende spørring finner antall solgte drikkevarer med delspørring

```
SELECT sum(d.quantity)
FROM (
  SELECT p.product_id
  FROM products AS p INNER JOIN categories AS c
    ON (p.category_id = c.category_id)
  WHERE c.category_name = 'Beverages'
) AS beverages
INNER JOIN
order_details AS d
  ON (beverages.product_id = d.product_id)
```

- ◆ Merk: Alle delspøringer som tabeller må gis et navn

Delspøringer som verdier

- ◆ En aggregatfunksjon over en kolonne returnerer én enkelt verdi
- ◆ Vi kan derfor bruke den som en verdi i `WHERE`-klausulen
- ◆ Så for å finne alle produkter som koster mer enn gjennomsnittet kan vi skrive:

```
SELECT product_name
  FROM products
 WHERE unit_price > (SELECT avg(unit_price)
                    FROM products)
```

- ◆ Merk at én enkel verdi og en tabell med kun én verdi behandles likt av SQL

Delspøringer som mengder

- ◆ Dersom vi ønsker å begrense én verdi (eller et tuppel av verdier) til svarene av en annen spørring i **WHERE**-klausulen, kan vi bruke nøkkelorder **IN**
- ◆ Kan ofte brukes i stedet for joins
- ◆ F.eks. for å finne navnet på alle produkter med en "supplier" fra Tyskland:

```
SELECT product_name
  FROM products
 WHERE supplier_id IN (SELECT supplier_id
                       FROM suppliers
                       WHERE country = 'Germany')
```

Eksempel: Finn navn og pris på alle produktet med lavest pris (1)

Ved `min`-aggregering og delspørring som tabell

```
SELECT p.product_name, p.unit_price
FROM (
    SELECT min(unit_price) AS minprice
    FROM products
) AS h
INNER JOIN products AS p
ON (p.unit_price = h.minprice)
```

Eksempel: Finn navn og pris på alle produktet med lavest pris (2)

Ved `min`-aggregering og delspørring som verdi

```
SELECT product_name, unit_price
FROM products
WHERE unit_price = (SELECT min(unit_price)
                    FROM products)
```

Hva er den største differansen mellom prisen på laptopper?

```
SELECT max(l1.Price - l2.Price) AS diff
FROM (SELECT Price FROM products WHERE Name LIKE '%Laptop%') AS l1,
     (SELECT Price FROM products WHERE Name LIKE '%Laptop%') AS l2
```

- ◆ Dersom vi ønsker å bruke den samme delspørringen om igjen kan man navngi den først med WITH, f.eks.:

WITH

```
laptops AS (SELECT Price FROM products WHERE Name LIKE '%Laptop%')
SELECT max(l1.Price - l2.Price) AS diff
FROM laptops AS l1, laptops AS l2
```

- ◆ Dette er både enklere å lese, lettere å vedlikeholde, og mer effektivt (slipper å kjøre laptops-spørringen to ganger)
- ◆ WITH er også nyttig for lesbarhet dersom man har mange delspørringer

Takk for nå!

Neste uke vil vi lære hvordan SQL kan brukes for datamanipulering.