

# Oblig 2 Del 1

March 5, 2025

Lagde databasen ved å skrive SQL-koden nedenfor:

```
[1]: for i in range(10):  
      print(i)
```

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

```
[ ]: -- Lagde Database kalt Oblig2  
      CREATE DATABASE Oblig2;
```

Kjørte SQL koden fra hobbyhuset.sql filen.

## 1 Oppgave 1

1.1 Utfør hobbyhuset.sql scriptet og gjør følgende oppgaver ved hjelp av SQL-spørringer mot databaseserveren:

1.1.1 1. Finn Etternavn, Fornavn og beløpet på den kunden som har kjøpt mest totalt.

```
[ ]: SELECT K.Etternavn, K.Fornavn, SUM(OL.Antall * OL.PrisPrEnhet) AS TotalBelop  
      FROM Kunde AS K  
      JOIN Ordre AS O ON K.KNr = O.KNr  
      JOIN Ordrelinje AS OL ON O.OrdreNr = OL.OrdreNr  
      GROUP BY K.KNr, K.Etternavn, K.Fornavn  
  
      -- Filtreerer for å få kunden med det største TotalBelop  
      HAVING TotalBelop = (  
      -- bruker subquery-en MAX() for å finne det høyeste totale beløpet blant alle  
      -- kunder.
```

```

SELECT MAX(X.Total)
FROM (
    SELECT K2.KNr, SUM(OL2.Antall * OL2.PrisPrEnhet) AS Total
    FROM Kunde AS K2
    JOIN Ordre AS O2 ON K2.KNr = O2.KNr
    JOIN Ordrelinje AS OL2 ON O2.OrdreNr = OL2.OrdreNr
    GROUP BY K2.KNr
) AS X
);

-- Til slutt filtreres resultatet slik at bare de kundene med et totalbeløp lik
-- dette maksimumet blir vist (det kan bli flere om flere kunder har brukt like
-- mye).

```

Output:

Etternavn	Fornavn	TotalBelop
Langaas	Evelyn	133711.00

### 1.1.2 2. Finn Betegnelse(-ne) til vare(-r) som aldri har vært solgt.

```

[ ]: SELECT V.Betegnelse
FROM Vare AS V
LEFT JOIN Ordrelinje AS O ON V.VNr = O.VNr
WHERE O.OrdreNr IS NULL;

```

Output:

Betegnelse
"No Data"

**\*\*Det ser ut som om alle varer har blitt solgt minst 1 gang\***

### 1.1.3 3. Finn den mest populære varen og hvor mange ordrer denne har inngått i.

```

[ ]: SELECT V.Betegnelse, COUNT(O.OrdreNr) AS AntallOrdrer
FROM Vare V
JOIN Ordrelinje O ON V.VNr = O.VNr
GROUP BY V.Betegnelse
ORDER BY AntallOrdrer DESC
LIMIT 1;

```

Output:

Betegnelse	AntallOrdre
Boule	66

#### 1.1.4 4. Finn den varen som var mest populær i september 2019.

```
[ ]: SELECT V.Betegnelse, COUNT(O.OrdreNr) AS AntallOrdre, P.Dato AS Måned
FROM Vare V
JOIN Ordrelinje O ON V.VNr = O.VNr
JOIN Prishistorikk P ON V.VNr = P.VNr
WHERE MONTHNAME(P.Dato) = 'September'
GROUP BY V.Betegnelse, P.Dato
ORDER BY AntallOrdre
LIMIT 1;
```

Output:

Betegnelse	AntallOrdre	Måned
Hobbyleire, hvit 1 kg	29	2019-09-29

#### 1.1.5 5. Finn Etternavn, Fornavn til kunder som har mer enn 10 ordre.

```
[ ]: SELECT K.Etternavn, K.Fornavn, COUNT(O.OrdreNr) AS AntallOrdre
FROM Kunde K
JOIN Ordre O ON K.KNr = O.KNr
GROUP BY K.KNr, K.Etternavn, K.Fornavn
HAVING AntallOrdre > 10;
```

Output:

Etternavn	Fornavn	AntallOrdre
Hansson	Gjermund	18
Astrup	David	15
Dirdal	Mimmi	12
Lian	Villy	12
Akselsen	Alex	12
Morken	Ingmar	12
Mundal	Jakob	12
Alexandersen	Amina	12
Furu	Ismail	12
Bjorke	Goril	12
Berre	Ian	12
Ekeberg	Abdi	12
Hognestad	Oddlaug	12
Hestad	Adam	12

### 1.1.6 6. Finn hvilken kunde som har flest ordrer.

```
[ ]: SELECT K.Etternavn, K.Fornavn, COUNT(O.OrdreNr) AS AntallOrdre
FROM Kunde K
JOIN Ordre O ON K.KNr = O.KNr
GROUP BY K.Etternavn, K.Fornavn
HAVING AntallOrdre = (
    SELECT MAX(X.Total)
    FROM(
        SELECT K2.KNr, COUNT(O2.OrdreNr) AS Total
        FROM Kunde K2
        JOIN Ordre O2 ON K2.KNr = O2.KNr
        GROUP BY K2.KNr
    ) AS X
)
```

Output:

Etternavn	Fornavn	AntallOrdre
Hansson	Gjermund	18
Dammen	Sven	18
Bremseth	Mohamed	18

### 1.1.7 7. Finn hvilken kunde som har færrest ordre.

```
[ ]: SELECT K.Etternavn, K.Fornavn, COUNT(O.OrdreNr) AS AntallOrdre
FROM Kunde K
JOIN Ordre O ON K.KNr = O.KNr
GROUP BY K.Etternavn, K.Fornavn
HAVING AntallOrdre = (
    SELECT MIN(X.Total)
    FROM (
        SELECT K2.KNr, COUNT(O2.OrdreNr) AS Total
        FROM Kunde K2
        JOIN Ordre O2 ON K2.KNr = O2.KNr
        GROUP BY K2.KNr
    ) AS X
);
```

Output:

Etternavn	Fornavn	AntallOrdre
Reinertsen	Arne	1

### 1.1.8 8. List alfabetisk kunder som kommer fra Hamar.

```
[ ]: SELECT K.Etternavn, K.Fornavn, PS.Poststed AS Poststed
FROM Kunde K
JOIN Poststed PS ON K.PostNr = PS.PostNr
WHERE Poststed = "Hamar"
ORDER BY K.Etternavn;
```

Output:

Etternavn	Fornavn	Poststed
Astrup	David	HAMAR
Bergstrom	Hilmar	HAMAR
Bjorke	Goril	HAMAR
Bjorke	Goril	HAMAR
Bjorke	Goril	HAMAR
Hauan	Thore	HAMAR
Holte	Brian	HAMAR
Holte	Brian	HAMAR
Mundal	Jakob	HAMAR

### 1.1.9 9. Finn hvilke poststeder som ingen kunder eller ansatte bor i.

```
[ ]: SELECT PS.Poststed, COUNT(K.KNr) AS AntallKunder, COUNT(A.AnsNr) AS
    ↪AntallAnsatte
FROM Poststed PS
LEFT JOIN Kunde K ON PS.PostNr = K.PostNr
LEFT JOIN Ansatt A ON PS.PostNr = A.PostNr
GROUP BY Poststed
HAVING AntallKunder = 0
    AND AntallAnsatte = 0

-- !!!Begrenser til 10 stk for å gjøre det lettere å vise i PDF-en
LIMIT 10;
```

Poststed	AntallKunder	AntallAnsatte
SANDVIKA	0	0
HASLUM	0	0
FORNEBU	0	0
JAR	0	0
RUD	0	0
HoVIKODDEN	0	0
SLEPENDEN	0	0
VoYENENGA	0	0
EIKSMARKA	0	0
BAERUMS VERK	0	0

#### 1.1.10 10. Finn dato og antall salg på den beste salgsdagen.

```
[ ]: SELECT OrdreDato, COUNT(OrdreNr) AS AntallOrdrer
FROM Ordre
GROUP BY OrdreDato
HAVING AntallOrdrer = (
    SELECT MAX(X.Total)
    FROM (
        SELECT O2.OrdreDato, COUNT(O2.OrdreNr) AS Total
        FROM Ordre O2
        GROUP BY O2.OrdreDato
    ) AS X
);
```