Oppgavesett 1

Oppgave1

a) 10011011 = 155
b) 537 = 1000011001
c) DATABASE = 01000100 01000001 01010100 01000001 01000010 01000001 01010011 01000101 00001010 00001010

Oppgave2

<u>SQL SELECT</u>

Oppgavetekst	Svar
1) Insert the missing statement to get all the columns from the `Customers` table.	SELECT * FROM Customers;
2) Write a statement that will select the `City` column from the `Customers` table.	SELECT CITY FROM Customers;
3) Select all the *different* values from the `Country` column in the `Customers` table.	SELECT DISTINCT Country FROM Customers;

<u>SQL WHERE</u>

Oppgavetekst	Svar
1) Select all records where the `City` column has the value "Berlin".	WHERE City = 'Berlin';
2) Use the `NOT` keyword to select all records where `City` is NOT "Berlin".	WHERE NOT City = 'Berlin';
3) Select all records where the `CustomerID` column has the value 32.	WHERE CustomerID = 32;

4) Select all records where the `City` column has the value 'Berlin' *and* the `PostalCode` column has the value 12209.	SELECT * FROM Customers
	WHERE City = 'Berlin'
	AND Postalcode = 12209;

5) Select all records where the `City` column has the value 'Berlin' or 'London'.	SELECT * FROM Customers
	WHERE City = 'Berlin'
	OR City = 'London';

SQL ORDER BY

Oppgavetekst	Svar

1) Select all records from the Customers table, sort the result *reversed* alphabetically by the column City.	SELECT * FROM Customers
	ORDER BY City;

2) Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.	SELECT * FROM Customers
	ORDER BY City DESC;

3) Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.	SELECT * FROM Customers
	ORDER BY Country, City;

SQL INSERT

Oppgavetekst	Svar
1) Insert a new record in the Customers table.	INSERT INTO Customers(CustomerName, Adress, City, PostalCode, Country) VALUES (‘Hekkan burger’ ‘Gatevei 15’ ‘Sandes’ ‘4306’ ‘Norway’);

SQL NULL

Oppgavetekst	Svar

1) Select all records from the Customers where the PostalCode column is empty.	SELECT * FROM Customers
	WHERE PostalCode IS NULL;

2) Select all records from the Customers where the PostalCode column is NOT empty.	SELECT * FROM Customers
	WHERE PostalCode IS NOT NULL;

SQL UPDATE

Oppgavetekst	Svar

1) Update the City column of all records in the Customers table.	UPDATE Customers
	SET City = ‘Oslo’;

2) Set the value of the City columns to 'Oslo', but only the ones where the Country column has the value "Norway".	UPDATE Customers
	SET City = ‘Oslo’
	WHERE Country = ‘Norway’;

3) Update the City value *and* the Country value.	UPDATE Customers
	SET City = ‘Oslo’,
	Country = ‘Norway’
	WHERE CustomerID = 32;

SQL DELETE

Oppgavetekst	Svar

1) Delete all the records from the Customers table where the Country value is 'Norway'.	DELETE FROM Custromers
	WHERE Country = ‘Norway’;

2) Delete all the records from the Customers table.	DELETE FROM Custromers;

SQL FUNCTIONS

Oppgavetekst	Svar

1) Use the MIN function to select the record with the smallest value of the Price column.	SELECT MIN(PRICE)
	FROM Products;

2) Use an SQL function to select the record with the highest value of the Price column.	SELECT MAX(Price)
	FROM Products;

3) Use the correct function to return the number of records that have the Price value set to 18.	SELECT COUNT(*)
	FROM Products
	WHERE Price = 18;

4) Use an SQL function to calculate the average price of all products.	SELECT AVG(Price)
	FROM Products;

5) Use an SQL function to calculate the sum of all the Price column values in the Products table.	SELECT SUM(Price)
	FROM Products;

SQL LIKE

Oppgavetekst	Svar

1) Select all records where the value of the City column starts with the letter "a".	SELECT * FROM Customers
	WHERE City LIKE 'a%';

2) Select all records where the value of the City column *ends* with the letter "a".	SELECT * FROM Customers
	WHERE City LIKE '%a';

3) Select all records where the value of the City column contains the letter "a".	SELECT * FROM Customers
	WHERE City LIKE '%a%';

4) Select all records where the value of the City column starts with letter "a" and ends with the letter "b".	SELECT * FROM Customers
	WHERE City LIKE 'a%b';

5) Select all records where the value of the City column does NOT start with the letter "a".	SELECT * FROM Customers
	WHERE City NOT LIKE 'a%';

Oppgave 3

a) *Hva er en database?*

Svar: En database er en samling av data, som ofte er samlet elektronisk.

Kilde: https://no.wikipedia.org/wiki/Database

b) *Hva er en realsjonsdatabase?*

Svar: En realsjonsdatabase er en forbunding mellom tabeller. Ofte med noe tilfelles.

Kilde: https://no.wikipedia.org/wiki/Relasjonsdatabase

2.8 Filmoppgave

Oppgavetekst	Svar

a) All informasjon om filmer produsert i 1988	SELECT * FROM Film
	WHERE År = 1988;

b) Tittel på amerikanske filmer produsert på 1980-tallet	SELECT FNr, Tittel FROM Film
	WHERE Land = 'USA',
	AND År BETWEEN 1980 AND 1989;

c) Komedier med aldersgrense under 10 år og spilletid under 130 minutter	SELECT * FROM Film
	WHERE Sjanger = 'Komedie',
	AND Alder < 10 AND Tid < 130;

d) Tittel på alle action- og westernfilmer	SELECT FNr, Tittel FROM Film
	WHERE Sjanger = 'Action',
	OR Sjanger = 'Western';

e) Alle produksjonsland, sortert og uten gjentakelser	SELECT DISTINCT Land FROM Film
	ORDER BY Land;

f) Korteste og lengste spilletid innen hver sjanger	SELECT Sjanger, MIN(Tid) AS Korteste,
	MAX(Tid) AS Lengste
	FROM Film
	GROUP BY Sjanger;

g) Antall filmer som ikke er til salg	SELECT COUNT(*) AS SelgesIkke
	FROM Film
	WHERE Pris IS NULL;

h) Antall filmer under 100kr	SELECT COUNT(*) AS Under100
	FROM Film
	WHERE Pris < 100;

i) Filmer med tittel som slutter på "now"	SELECT * FROM Film
	WHERE Tittel LIKE '%now';

j) Gjennomsnittspris for sjangre med flere enn 2 filmer	SELECT Sjanger, AVG(Pris)
	AS SnittPrisSjanger FROM Film
	GROUP BY Sjanger
	HAVING COUNT(*) < 2;

k) Differansen mellom dyreste og billigste film innen hver sjanger	SELECT Sjanger, MAX(Pris)-MIN(Pris) AS
	Billigst FROM Film
	GROUP BY Sjanger

l) Total antall filmer og filmer til salgs, fordelt på produksjonsland	SELECT Land, COUNT(*) AS FordeltProduk
	COUNT(Pris) AS AntallTilSalgs
	FROM Film
	GROUP BY Land

m) Antall år siden utgivelse for filmer eldre enn 60 år	SELECT FNr, YEAR(CURDATE())-År AS
	AntallÅr FROM Film
	WHERE YEAR(CURDATE())-År > 60;

Oppgavesett 2

Oppgave 1

SQL CREATE TABLE

Oppgavetekst	Svar
1) Write the correct SQL statement to create a new table called Persons.	CREATE TABLE Person (PersonID int, LastName varchar(255), FirstName varchar(255), Address varchar(255), City varchar(255));

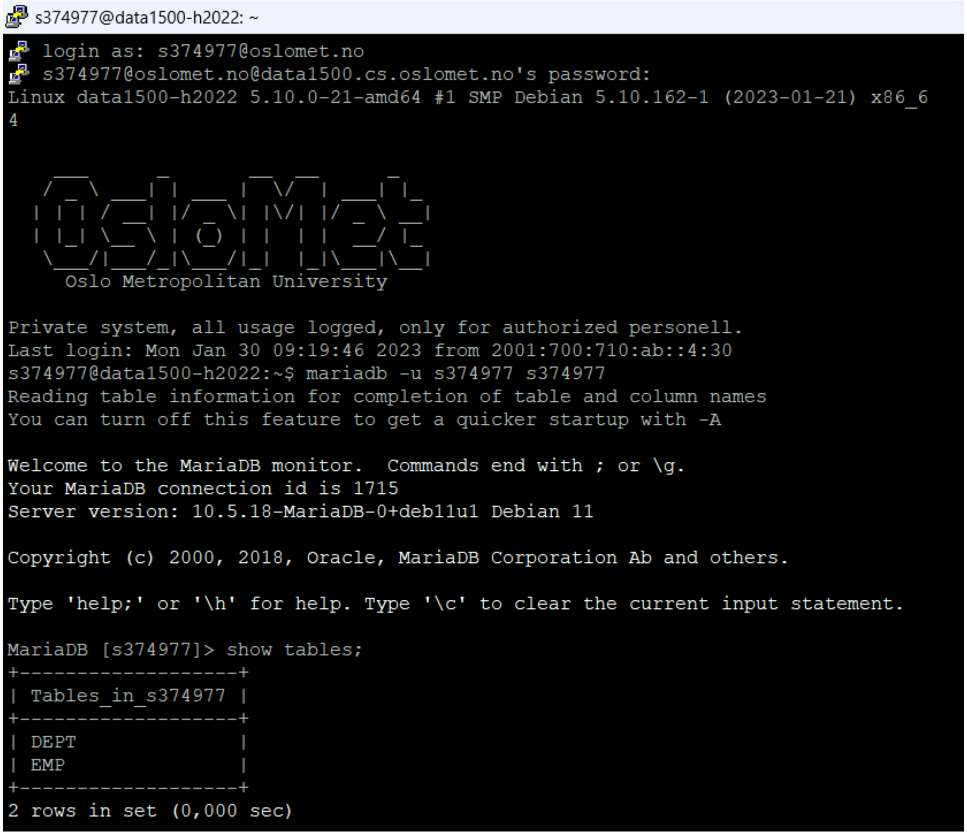
SQL DROP TABLE

Oppgavetekst	Svar
1) Write the correct SQL statement to delete a table called Persons.	DROP TABLE Persons;

SQL ALTER TABLE

Oppgavetekst	Svar
1) Add a column of type DATE called Birthday.	ALTER TABLE Persons ADD Birthday DATE;

Oppgave 2



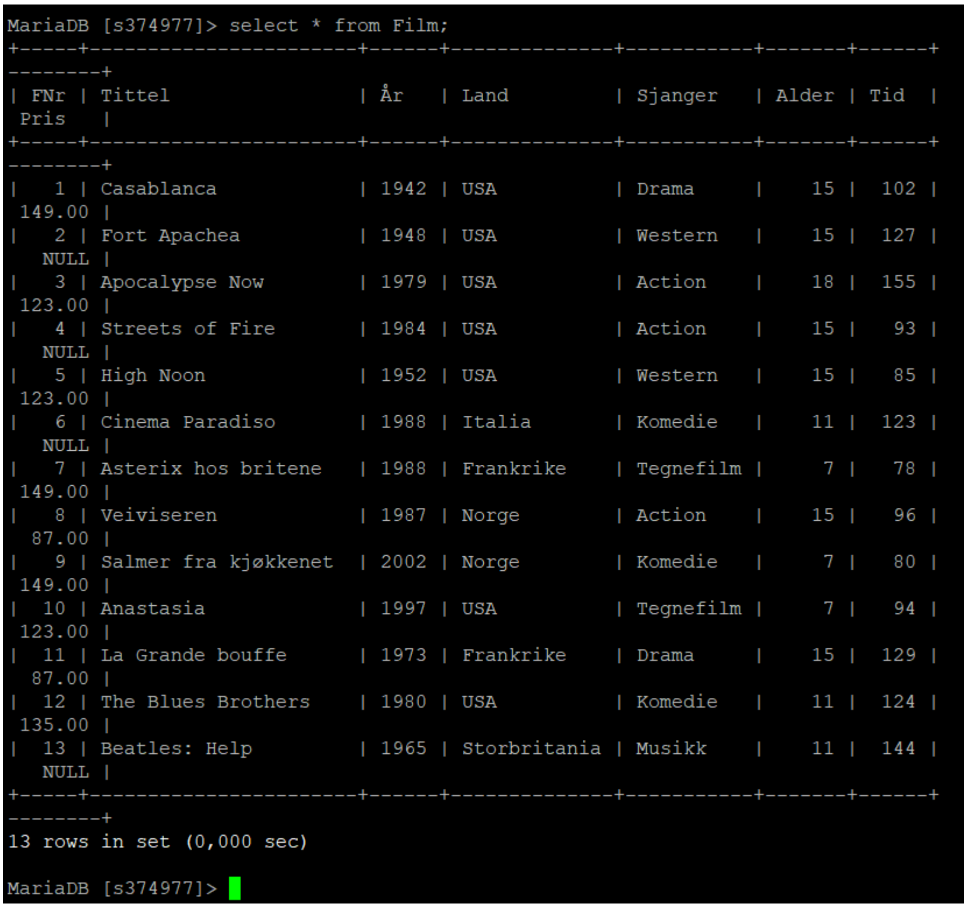
```
 login as: s374977@oslomet.no
 s374977@oslomet.no@data1500.cs.oslomet.no's password:
Linux data1500-h2022 5.10.0-21-amd64 #1 SMP Debian 5.10.162-1 (2023-01-21) x86_6
4


    ___      _       __  __      _
   / _ \ ___| | ___ |  \/  | ___| |_
  | | | / __| |/ _ \| |\/| |/ _ \ __|
  | |_| \__ \ | (_) | |  | |  __/ |_
   \___/|___/_|\___/|_|  |_|\___|\__|
     Oslo Metropolitan University


Private system, all usage logged, only for authorized personell.
Last login: Mon Jan 30 09:19:46 2023 from 2001:700:710:ab::4:30
s374977@data1500-h2022:~$ mariadb -u s374977 s374977
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 1715
Server version: 10.5.18-MariaDB-0+deb11u1 Debian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [s374977]> show tables;
+-------------------+
| Tables_in_s374977 |
+-------------------+
| DEPT              |
| EMP               |
+-------------------+
2 rows in set (0,000 sec)
```

Oppgave 3

Script film.txt

```
CREATE TABLE Film
(
   FNr     INTEGER NOT NULL,
   Tittel  VARCHAR(100),
   År      SMALLINT,
   Land    VARCHAR(50),
   Sjanger VARCHAR(50),
   Alder   SMALLINT,
   Tid     SMALLINT,
   Pris    DECIMAL(8, 2),
   CONSTRAINT FilmPN PRIMARY KEY (FNr)
);


INSERT INTO Film (FNr, Tittel, År, Land, Sjanger, Alder, Tid, Pris) VALUES
( 1, 'Casablanca',          1942, 'USA',         'Drama',     15, 102, '149.00'),
( 2, 'Fort Apachea',        1948, 'USA',         'Western',   15, 127,     NULL),
( 3, 'Apocalypse Now',      1979, 'USA',         'Action',    18, 155, '123.00'),
( 4, 'Streets of Fire',     1984, 'USA',         'Action',    15,  93,     NULL),
( 5, 'High Noon',           1952, 'USA',         'Western',   15,  85, '123.00'),
( 6, 'Cinema Paradiso',     1988, 'Italia',      'Komedie',   11, 123,     NULL),
( 7, 'Asterix hos britene', 1988, 'Frankrike',   'Tegnefilm',  7,  78, '149.00'),
( 8, 'Veiviseren',          1987, 'Norge',       'Action',    15,  96,  '87.00'),
( 9, 'Salmer fra kjøkkenet', 2002, 'Norge',      'Komedie',    7,  80, '149.00'),
(10, 'Anastasia',           1997, 'USA',         'Tegnefilm',  7,  94, '123.00'),
(11, 'La Grande bouffe',    1973, 'Frankrike',   'Drama',     15, 129, ' 87.00'),
(12, 'The Blues Brothers',  1980, 'USA',         'Komedie',   11, 124, '135.00'),
(13, 'Beatles: Help',       1965, 'Storbritania', 'Musikk',   11, 144,     NULL);
```

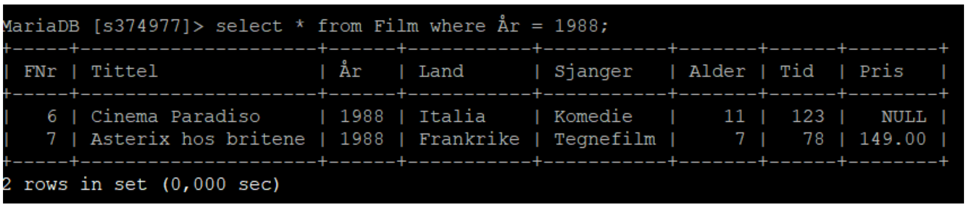
Screen av tabellen fra putty



```
MariaDB [s374977]> select * from Film;
+-----+-----------------------+------+--------------+-----------+-------+------+
--------+
| FNr | Tittel                | År   | Land         | Sjanger   | Alder | Tid  |
 Pris   |
+-----+-----------------------+------+--------------+-----------+-------+------+
--------+
|   1 | Casablanca            | 1942 | USA          | Drama     |    15 |  102 |
 149.00 |
|   2 | Fort Apachea          | 1948 | USA          | Western   |    15 |  127 |
   NULL |
|   3 | Apocalypse Now        | 1979 | USA          | Action    |    18 |  155 |
 123.00 |
|   4 | Streets of Fire       | 1984 | USA          | Action    |    15 |   93 |
   NULL |
|   5 | High Noon             | 1952 | USA          | Western   |    15 |   85 |
 123.00 |
|   6 | Cinema Paradiso       | 1988 | Italia       | Komedie   |    11 |  123 |
   NULL |
|   7 | Asterix hos britene   | 1988 | Frankrike    | Tegnefilm |     7 |   78 |
 149.00 |
|   8 | Veiviseren            | 1987 | Norge        | Action    |    15 |   96 |
  87.00 |
|   9 | Salmer fra kjøkkenet  | 2002 | Norge        | Komedie   |     7 |   80 |
 149.00 |
|  10 | Anastasia             | 1997 | USA          | Tegnefilm |     7 |   94 |
 123.00 |
|  11 | La Grande bouffe      | 1973 | Frankrike    | Drama     |    15 |  129 |
  87.00 |
|  12 | The Blues Brothers    | 1980 | USA          | Komedie   |    11 |  124 |
 135.00 |
|  13 | Beatles: Help         | 1965 | Storbritania | Musikk    |    11 |  144 |
   NULL |
+-----+-----------------------+------+--------------+-----------+-------+------+
--------+
13 rows in set (0,000 sec)

MariaDB [s374977]>
```

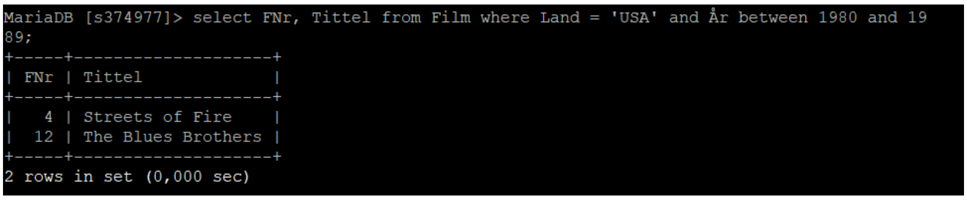
Oppgave 4

a)



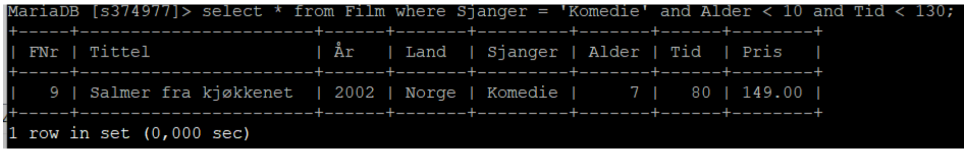
```
ariaDB [s374977]> select * from Film where År = 1988;
+-----+---------------------+------+-----------+-----------+-------+------+--------+
| FNr | Tittel              | År   | Land      | Sjanger   | Alder | Tid  | Pris   |
+-----+---------------------+------+-----------+-----------+-------+------+--------+
|   6 | Cinema Paradiso     | 1988 | Italia    | Komedie   |    11 |  123 |   NULL |
|   7 | Asterix hos britene | 1988 | Frankrike | Tegnefilm |     7 |   78 | 149.00 |
+-----+---------------------+------+-----------+-----------+-------+------+--------+
2 rows in set (0,000 sec)
```

b)



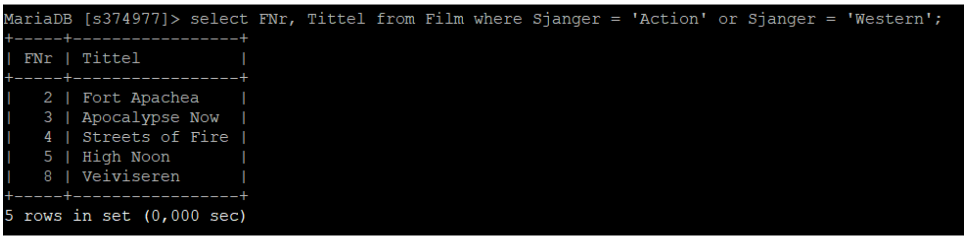
```
MariaDB [s374977]> select FNr, Tittel from Film where Land = 'USA' and År between 1980 and 19
89;
+-----+--------------------+
| FNr | Tittel             |
+-----+--------------------+
|   4 | Streets of Fire    |
|  12 | The Blues Brothers |
+-----+--------------------+
2 rows in set (0,000 sec)
```

c)



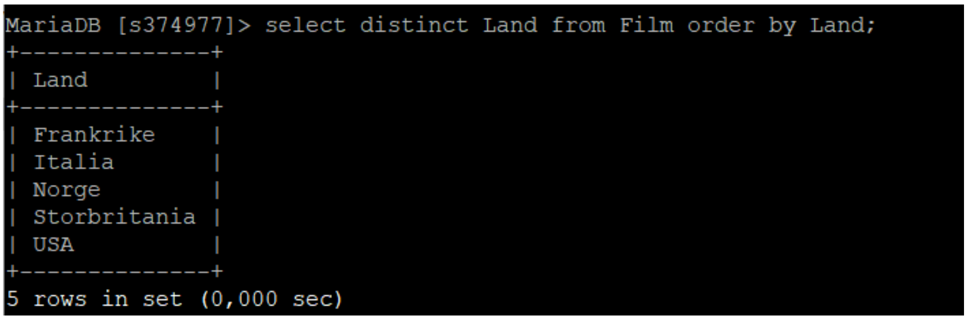
```
MariaDB [s374977]> select * from Film where Sjanger = 'Komedie' and Alder < 10 and Tid < 130;
+-----+-----------------------+------+-------+---------+-------+------+--------+
| FNr | Tittel                | År   | Land  | Sjanger | Alder | Tid  | Pris   |
+-----+-----------------------+------+-------+---------+-------+------+--------+
|   9 | Salmer fra kjøkkenet  | 2002 | Norge | Komedie |     7 |   80 | 149.00 |
+-----+-----------------------+------+-------+---------+-------+------+--------+
1 row in set (0,000 sec)
```

d)



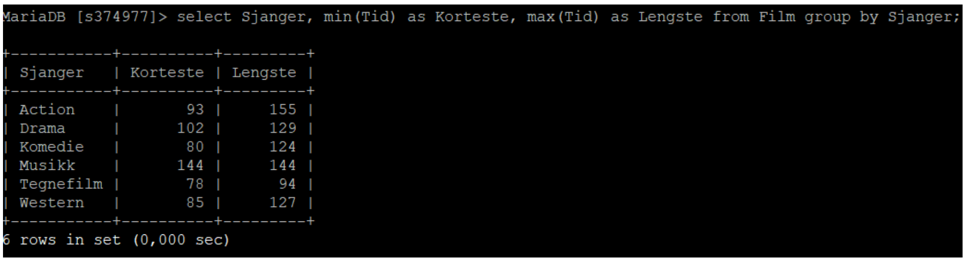
```
MariaDB [s374977]> select FNr, Tittel from Film where Sjanger = 'Action' or Sjanger = 'Western';
+-----+-----------------+
| FNr | Tittel          |
+-----+-----------------+
|   2 | Fort Apachea    |
|   3 | Apocalypse Now  |
|   4 | Streets of Fire |
|   5 | High Noon       |
|   8 | Veiviseren      |
+-----+-----------------+
5 rows in set (0,000 sec)
```

e)



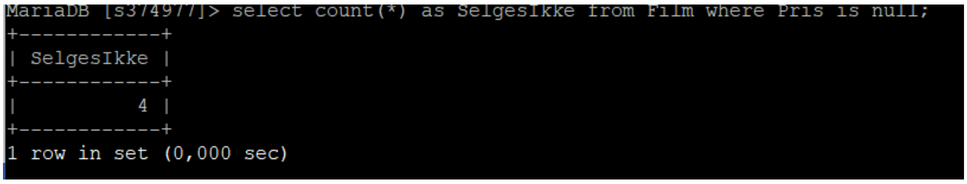
```
MariaDB [s374977]> select distinct Land from Film order by Land;
+---------------+
| Land          |
+---------------+
| Frankrike     |
| Italia        |
| Norge         |
| Storbritania  |
| USA           |
+---------------+
5 rows in set (0,000 sec)
```

f)



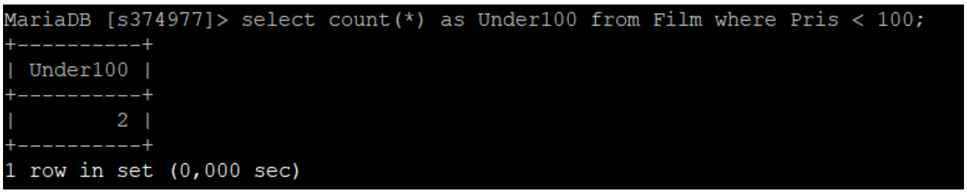
```
MariaDB [s374977]> select Sjanger, min(Tid) as Korteste, max(Tid) as Lengste from Film group by Sjanger;

+-----------+----------+---------+
| Sjanger   | Korteste | Lengste |
+-----------+----------+---------+
| Action    |       93 |     155 |
| Drama     |      102 |     129 |
| Komedie   |       80 |     124 |
| Musikk    |      144 |     144 |
| Tegnefilm |       78 |      94 |
| Western   |       85 |     127 |
+-----------+----------+---------+
6 rows in set (0,000 sec)
```

g)



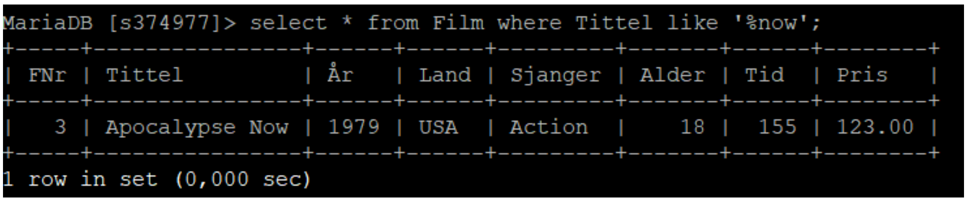
```
MariaDB [s374977]> select count(*) as SelgesIkke from Film where Pris is null;
+------------+
| SelgesIkke |
+------------+
|          4 |
+------------+
1 row in set (0,000 sec)
```

h)



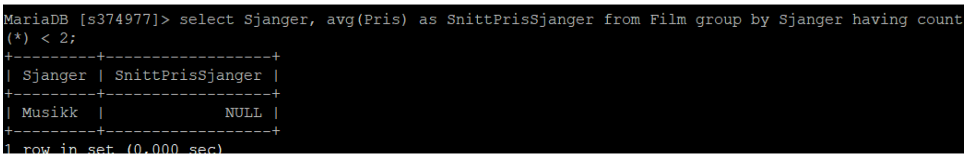
```
MariaDB [s374977]> select count(*) as Under100 from Film where Pris < 100;
+----------+
| Under100 |
+----------+
|        2 |
+----------+
1 row in set (0,000 sec)
```

i)



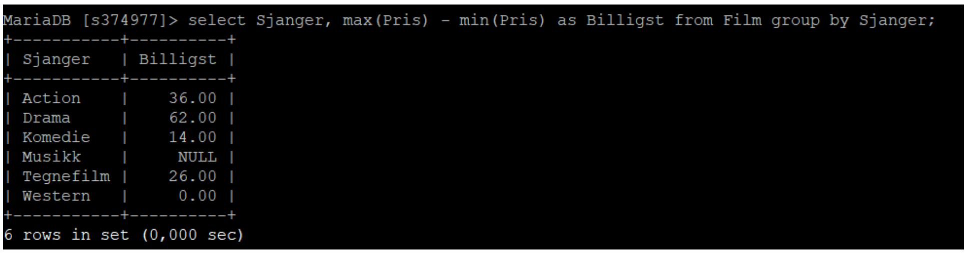
```
MariaDB [s374977]> select * from Film where Tittel like '%now';
+-----+----------------+------+------+---------+-------+------+--------+
| FNr | Tittel         | År   | Land | Sjanger | Alder | Tid  | Pris   |
+-----+----------------+------+------+---------+-------+------+--------+
|   3 | Apocalypse Now | 1979 | USA  | Action  |    18 |  155 | 123.00 |
+-----+----------------+------+------+---------+-------+------+--------+
1 row in set (0,000 sec)
```

j)



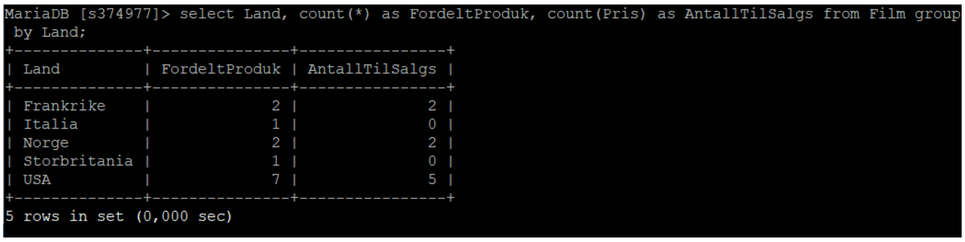
```
MariaDB [s374977]> select Sjanger, avg(Pris) as SnittPrisSjanger from Film group by Sjanger having count
(*) < 2;
+---------+------------------+
| Sjanger | SnittPrisSjanger |
+---------+------------------+
| Musikk  |             NULL |
+---------+------------------+
1 row in set (0,000 sec)
```

k)



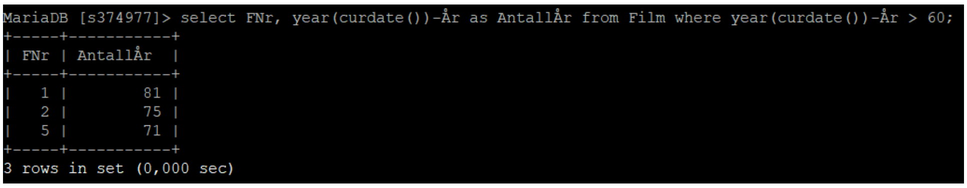
```
MariaDB [s374977]> select Sjanger, max(Pris) - min(Pris) as Billigst from Film group by Sjanger;
+-----------+----------+
| Sjanger   | Billigst |
+-----------+----------+
| Action    |    36.00 |
| Drama     |    62.00 |
| Komedie   |    14.00 |
| Musikk    |     NULL |
| Tegnefilm |    26.00 |
| Western   |     0.00 |
+-----------+----------+
6 rows in set (0,000 sec)
```

l)



```
MariaDB [s374977]> select Land, count(*) as FordeltProduk, count(Pris) as AntallTilSalgs from Film group
 by Land;
+--------------+---------------+----------------+
| Land         | FordeltProduk | AntallTilSalgs |
+--------------+---------------+----------------+
| Frankrike    |             2 |              2 |
| Italia       |             1 |              0 |
| Norge        |             2 |              2 |
| Storbritania |             1 |              0 |
| USA          |             7 |              5 |
+--------------+---------------+----------------+
5 rows in set (0,000 sec)
```

m)

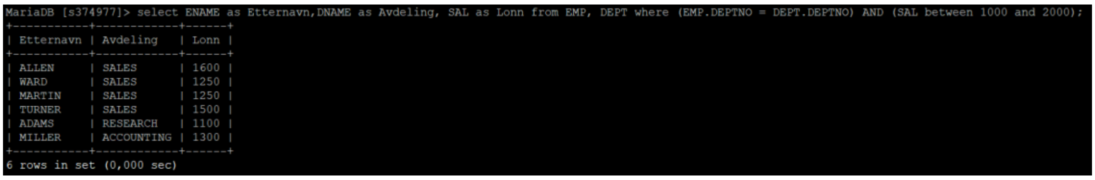


```
MariaDB [s374977]> select FNr, year(curdate())-År as AntallÅr from Film where year(curdate())-År > 60;
+-----+-----------+
| FNr | AntallÅr  |
+-----+-----------+
|   1 |        81 |
|   2 |        75 |
|   5 |        71 |
+-----+-----------+
3 rows in set (0,000 sec)
```

Oppgavesett 3

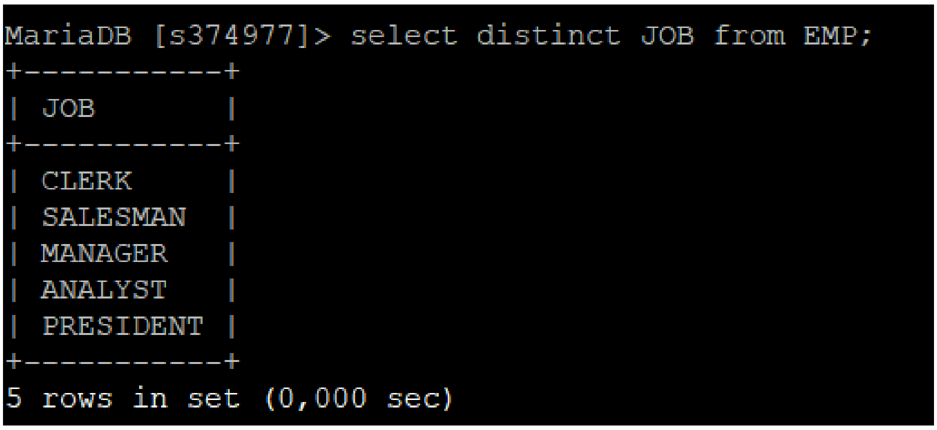
Oppgave1

1)



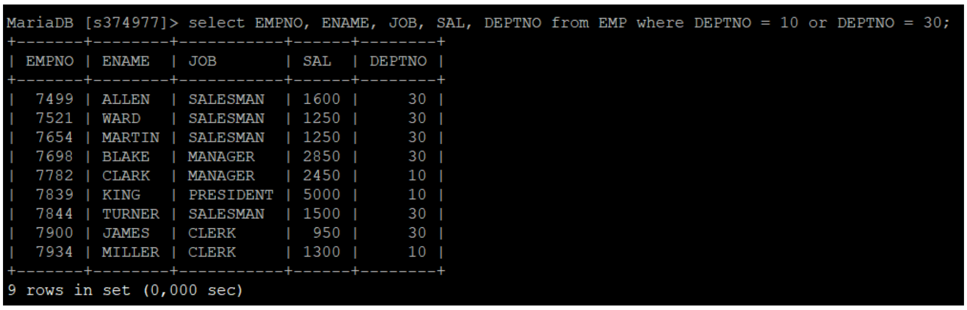
```
MariaDB [s374977]> select ENAME as Etternavn,DNAME as Avdeling, SAL as Lonn from EMP, DEPT where (EMP.DEPTNO = DEPT.DEPTNO) AND (SAL between 1000 and 2000);
+-----------+------------+------+
| Etternavn | Avdeling   | Lonn |
+-----------+------------+------+
| ALLEN     | SALES      | 1600 |
| WARD      | SALES      | 1250 |
| MARTIN    | SALES      | 1250 |
| TURNER    | SALES      | 1500 |
| ADAMS     | RESEARCH   | 1100 |
| MILLER    | ACCOUNTING | 1300 |
+-----------+------------+------+
6 rows in set (0,000 sec)
```

2)



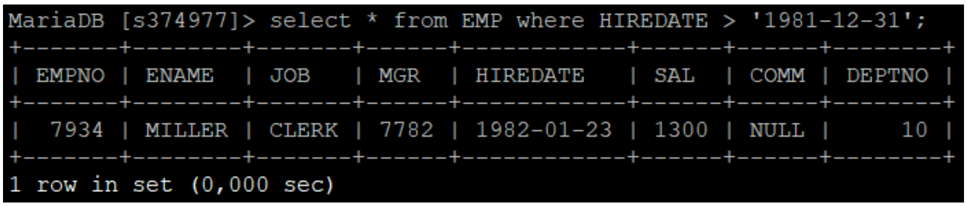
```
MariaDB [s374977]> select distinct JOB from EMP;
+-----------+
| JOB       |
+-----------+
| CLERK     |
| SALESMAN  |
| MANAGER   |
| ANALYST   |
| PRESIDENT |
+-----------+
5 rows in set (0,000 sec)
```

3)



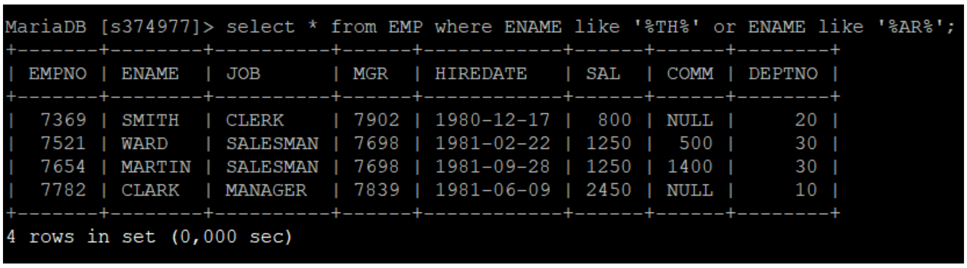
```
MariaDB [s374977]> select EMPNO, ENAME, JOB, SAL, DEPTNO from EMP where DEPTNO = 10 or DEPTNO = 30;
+-------+--------+-----------+------+--------+
| EMPNO | ENAME  | JOB       | SAL  | DEPTNO |
+-------+--------+-----------+------+--------+
|  7499 | ALLEN  | SALESMAN  | 1600 |     30 |
|  7521 | WARD   | SALESMAN  | 1250 |     30 |
|  7654 | MARTIN | SALESMAN  | 1250 |     30 |
|  7698 | BLAKE  | MANAGER   | 2850 |     30 |
|  7782 | CLARK  | MANAGER   | 2450 |     10 |
|  7839 | KING   | PRESIDENT | 5000 |     10 |
|  7844 | TURNER | SALESMAN  | 1500 |     30 |
|  7900 | JAMES  | CLERK     |  950 |     30 |
|  7934 | MILLER | CLERK     | 1300 |     10 |
+-------+--------+-----------+------+--------+
9 rows in set (0,000 sec)
```

4)



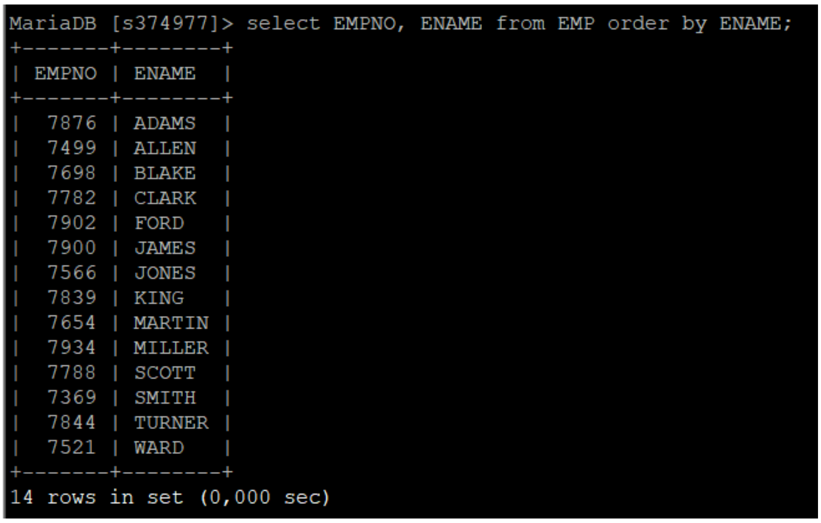
```
MariaDB [s374977]> select * from EMP where HIREDATE > '1981-12-31';
+-------+--------+-------+------+------------+------+------+--------+
| EMPNO | ENAME  | JOB   | MGR  | HIREDATE   | SAL  | COMM | DEPTNO |
+-------+--------+-------+------+------------+------+------+--------+
|  7934 | MILLER | CLERK | 7782 | 1982-01-23 | 1300 | NULL |     10 |
+-------+--------+-------+------+------------+------+------+--------+
1 row in set (0,000 sec)
```

5)



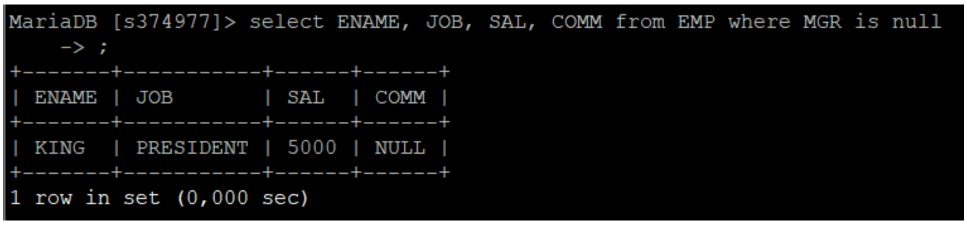
```
MariaDB [s374977]> select * from EMP where ENAME like '%TH%' or ENAME like '%AR%';
+-------+--------+----------+------+------------+------+------+--------+
| EMPNO | ENAME  | JOB      | MGR  | HIREDATE   | SAL  | COMM | DEPTNO |
+-------+--------+----------+------+------------+------+------+--------+
|  7369 | SMITH  | CLERK    | 7902 | 1980-12-17 |  800 | NULL |     20 |
|  7521 | WARD   | SALESMAN | 7698 | 1981-02-22 | 1250 |  500 |     30 |
|  7654 | MARTIN | SALESMAN | 7698 | 1981-09-28 | 1250 | 1400 |     30 |
|  7782 | CLARK  | MANAGER  | 7839 | 1981-06-09 | 2450 | NULL |     10 |
+-------+--------+----------+------+------------+------+------+--------+
4 rows in set (0,000 sec)
```

6)



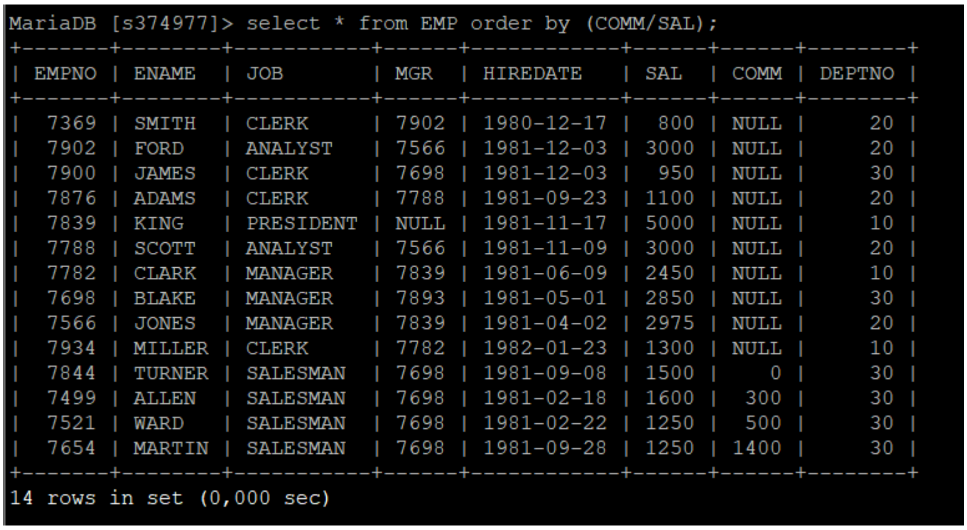
```
MariaDB [s374977]> select EMPNO, ENAME from EMP order by ENAME;
+-------+--------+
| EMPNO | ENAME  |
+-------+--------+
|  7876 | ADAMS  |
|  7499 | ALLEN  |
|  7698 | BLAKE  |
|  7782 | CLARK  |
|  7902 | FORD   |
|  7900 | JAMES  |
|  7566 | JONES  |
|  7839 | KING   |
|  7654 | MARTIN |
|  7934 | MILLER |
|  7788 | SCOTT  |
|  7369 | SMITH  |
|  7844 | TURNER |
|  7521 | WARD   |
+-------+--------+
14 rows in set (0,000 sec)
```

7)



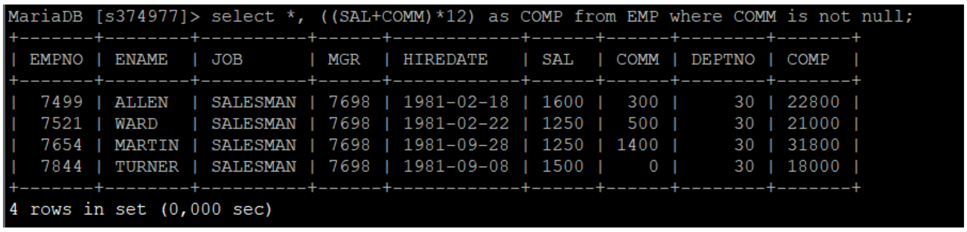
```
MariaDB [s374977]> select ENAME, JOB, SAL, COMM from EMP where MGR is null
    -> ;
+-------+-----------+------+------+
| ENAME | JOB       | SAL  | COMM |
+-------+-----------+------+------+
| KING  | PRESIDENT | 5000 | NULL |
+-------+-----------+------+------+
1 row in set (0,000 sec)
```

8)



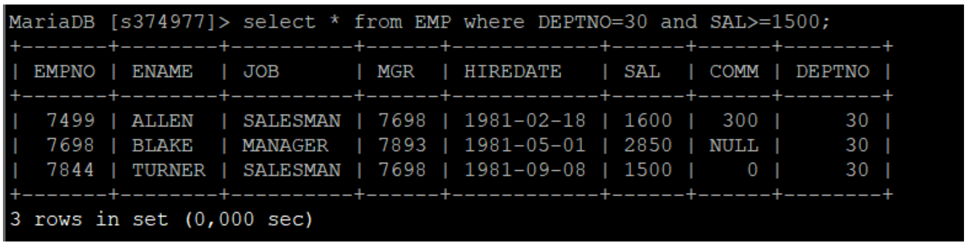
```
MariaDB [s374977]> select * from EMP order by (COMM/SAL);
+-------+--------+-----------+------+------------+------+------+--------+
| EMPNO | ENAME  | JOB       | MGR  | HIREDATE   | SAL  | COMM | DEPTNO |
+-------+--------+-----------+------+------------+------+------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |  800 | NULL |     20 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 | 3000 | NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |  950 | NULL |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1981-09-23 | 1100 | NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 | 5000 | NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1981-11-09 | 3000 | NULL |     20 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 | 2450 | NULL |     10 |
|  7698 | BLAKE  | MANAGER   | 7893 | 1981-05-01 | 2850 | NULL |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 2975 | NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 | 1300 | NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 | 1500 |    0 |     30 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-18 | 1600 |  300 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 | 1250 |  500 |     30 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 | 1250 | 1400 |     30 |
+-------+--------+-----------+------+------------+------+------+--------+
14 rows in set (0,000 sec)
```

9)



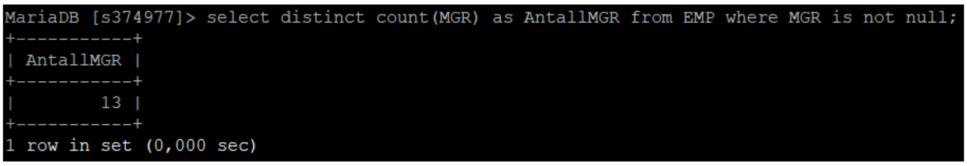
```
MariaDB [s374977]> select *, ((SAL+COMM)*12) as COMP from EMP where COMM is not null;
+-------+--------+----------+------+------------+------+------+--------+-------+
| EMPNO | ENAME  | JOB      | MGR  | HIREDATE   | SAL  | COMM | DEPTNO | COMP  |
+-------+--------+----------+------+------------+------+------+--------+-------+
|  7499 | ALLEN  | SALESMAN | 7698 | 1981-02-18 | 1600 |  300 |     30 | 22800 |
|  7521 | WARD   | SALESMAN | 7698 | 1981-02-22 | 1250 |  500 |     30 | 21000 |
|  7654 | MARTIN | SALESMAN | 7698 | 1981-09-28 | 1250 | 1400 |     30 | 31800 |
|  7844 | TURNER | SALESMAN | 7698 | 1981-09-08 | 1500 |    0 |     30 | 18000 |
+-------+--------+----------+------+------------+------+------+--------+-------+
4 rows in set (0,000 sec)
```

10)



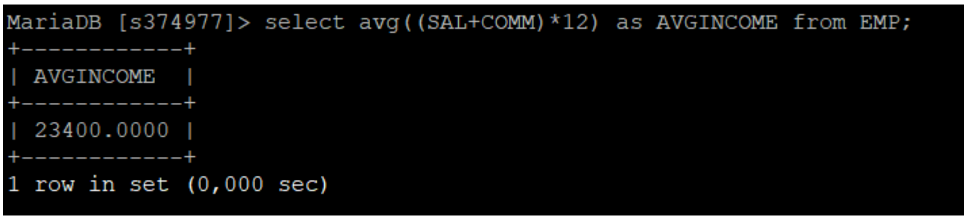
```
MariaDB [s374977]> select * from EMP where DEPTNO=30 and SAL>=1500;
+-------+--------+----------+------+------------+------+------+--------+
| EMPNO | ENAME  | JOB      | MGR  | HIREDATE   | SAL  | COMM | DEPTNO |
+-------+--------+----------+------+------------+------+------+--------+
|  7499 | ALLEN  | SALESMAN | 7698 | 1981-02-18 | 1600 |  300 |     30 |
|  7698 | BLAKE  | MANAGER  | 7893 | 1981-05-01 | 2850 | NULL |     30 |
|  7844 | TURNER | SALESMAN | 7698 | 1981-09-08 | 1500 |    0 |     30 |
+-------+--------+----------+------+------------+------+------+--------+
3 rows in set (0,000 sec)
```

11)



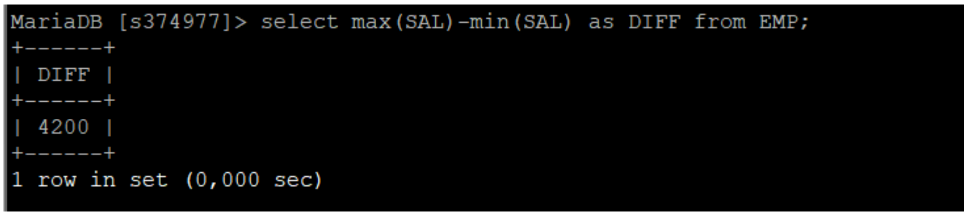
```
MariaDB [s374977]> select distinct count(MGR) as AntallMGR from EMP where MGR is not null;
+-----------+
| AntallMGR |
+-----------+
|        13 |
+-----------+
1 row in set (0,000 sec)
```

12)



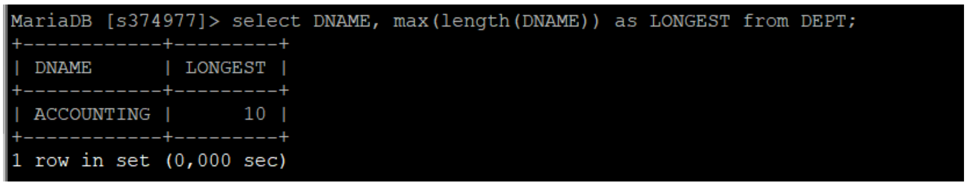
```
MariaDB [s374977]> select avg((SAL+COMM)*12) as AVGINCOME from EMP;
+------------+
| AVGINCOME  |
+------------+
| 23400.0000 |
+------------+
1 row in set (0,000 sec)
```

13)



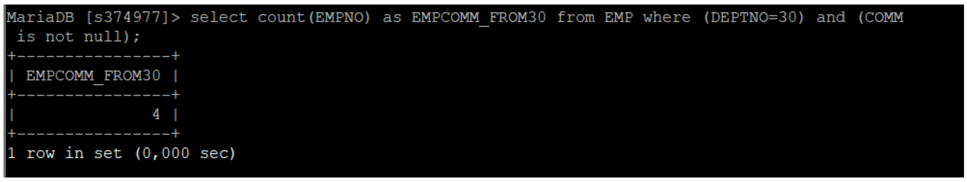
```
MariaDB [s374977]> select max(SAL)-min(SAL) as DIFF from EMP;
+------+
| DIFF |
+------+
| 4200 |
+------+
1 row in set (0,000 sec)
```

14)



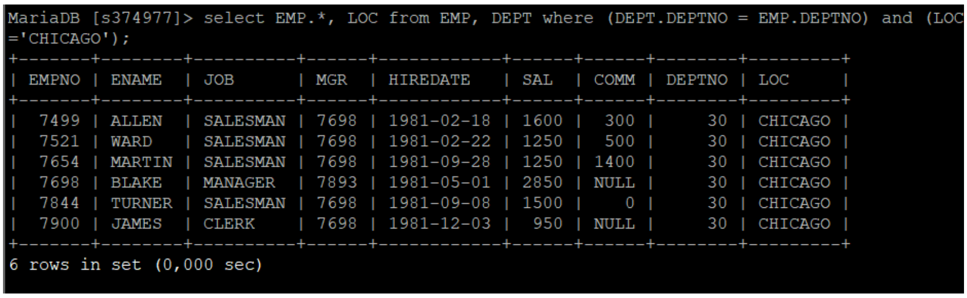
```
MariaDB [s374977]> select DNAME, max(length(DNAME)) as LONGEST from DEPT;
+------------+---------+
| DNAME      | LONGEST |
+------------+---------+
| ACCOUNTING |      10 |
+------------+---------+
1 row in set (0,000 sec)
```

15)



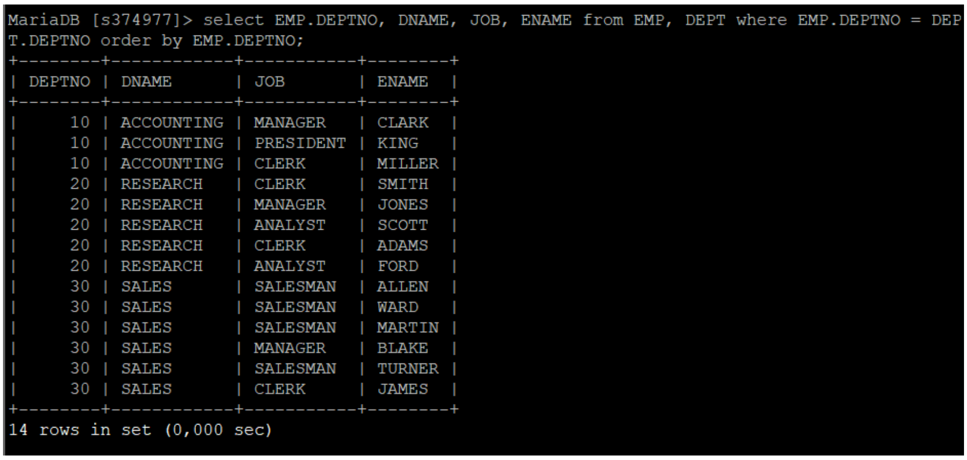
```
MariaDB [s374977]> select count(EMPNO) as EMPCOMM_FROM30 from EMP where (DEPTNO=30) and (COMM
 is not null);
+----------------+
| EMPCOMM_FROM30 |
+----------------+
|              4 |
+----------------+
1 row in set (0,000 sec)
```

16)



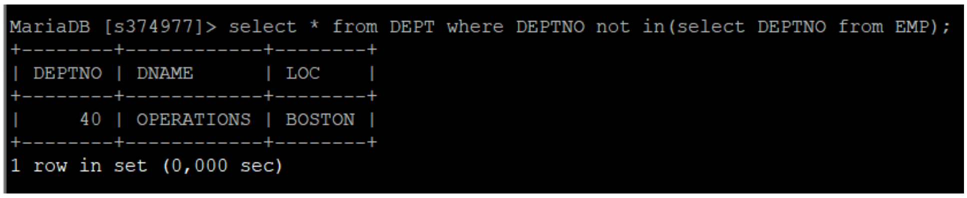
```
MariaDB [s374977]> select EMP.*, LOC from EMP, DEPT where (DEPT.DEPTNO = EMP.DEPTNO) and (LOC
='CHICAGO');
+-------+--------+----------+------+------------+------+------+--------+---------+
| EMPNO | ENAME  | JOB      | MGR  | HIREDATE   | SAL  | COMM | DEPTNO | LOC     |
+-------+--------+----------+------+------------+------+------+--------+---------+
|  7499 | ALLEN  | SALESMAN | 7698 | 1981-02-18 | 1600 |  300 |     30 | CHICAGO |
|  7521 | WARD   | SALESMAN | 7698 | 1981-02-22 | 1250 |  500 |     30 | CHICAGO |
|  7654 | MARTIN | SALESMAN | 7698 | 1981-09-28 | 1250 | 1400 |     30 | CHICAGO |
|  7698 | BLAKE  | MANAGER  | 7893 | 1981-05-01 | 2850 | NULL |     30 | CHICAGO |
|  7844 | TURNER | SALESMAN | 7698 | 1981-09-08 | 1500 |    0 |     30 | CHICAGO |
|  7900 | JAMES  | CLERK    | 7698 | 1981-12-03 |  950 | NULL |     30 | CHICAGO |
+-------+--------+----------+------+------------+------+------+--------+---------+
6 rows in set (0,000 sec)
```

17)



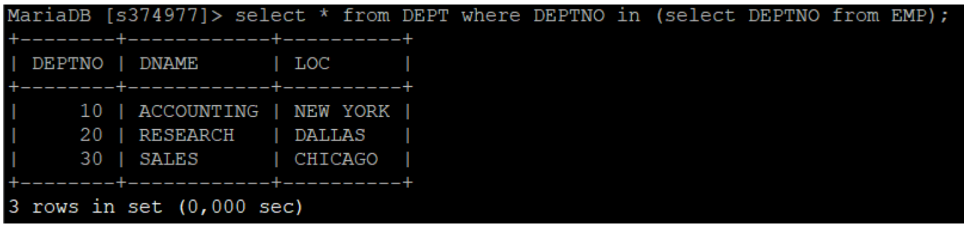
```
MariaDB [s374977]> select EMP.DEPTNO, DNAME, JOB, ENAME from EMP, DEPT where EMP.DEPTNO = DEP
T.DEPTNO order by EMP.DEPTNO;
+--------+------------+-----------+--------+
| DEPTNO | DNAME      | JOB       | ENAME  |
+--------+------------+-----------+--------+
|     10 | ACCOUNTING | MANAGER   | CLARK  |
|     10 | ACCOUNTING | PRESIDENT | KING   |
|     10 | ACCOUNTING | CLERK     | MILLER |
|     20 | RESEARCH   | CLERK     | SMITH  |
|     20 | RESEARCH   | MANAGER   | JONES  |
|     20 | RESEARCH   | ANALYST   | SCOTT  |
|     20 | RESEARCH   | CLERK     | ADAMS  |
|     20 | RESEARCH   | ANALYST   | FORD   |
|     30 | SALES      | SALESMAN  | ALLEN  |
|     30 | SALES      | SALESMAN  | WARD   |
|     30 | SALES      | SALESMAN  | MARTIN |
|     30 | SALES      | MANAGER   | BLAKE  |
|     30 | SALES      | SALESMAN  | TURNER |
|     30 | SALES      | CLERK     | JAMES  |
+--------+------------+-----------+--------+
14 rows in set (0,000 sec)
```

18)



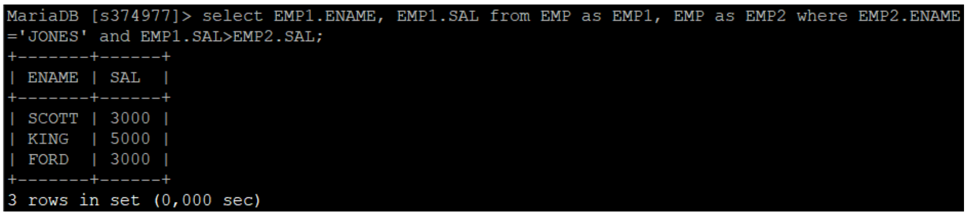
```
MariaDB [s374977]> select * from DEPT where DEPTNO not in(select DEPTNO from EMP);
+--------+------------+--------+
| DEPTNO | DNAME      | LOC    |
+--------+------------+--------+
|     40 | OPERATIONS | BOSTON |
+--------+------------+--------+
1 row in set (0,000 sec)
```

19)



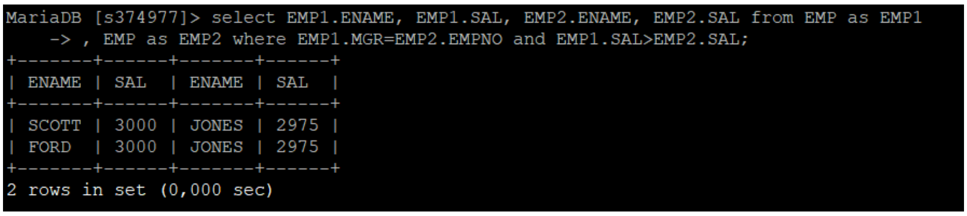
```
MariaDB [s374977]> select * from DEPT where DEPTNO in (select DEPTNO from EMP);
+--------+------------+----------+
| DEPTNO | DNAME      | LOC      |
+--------+------------+----------+
|     10 | ACCOUNTING | NEW YORK |
|     20 | RESEARCH   | DALLAS   |
|     30 | SALES      | CHICAGO  |
+--------+------------+----------+
3 rows in set (0,000 sec)
```

20)



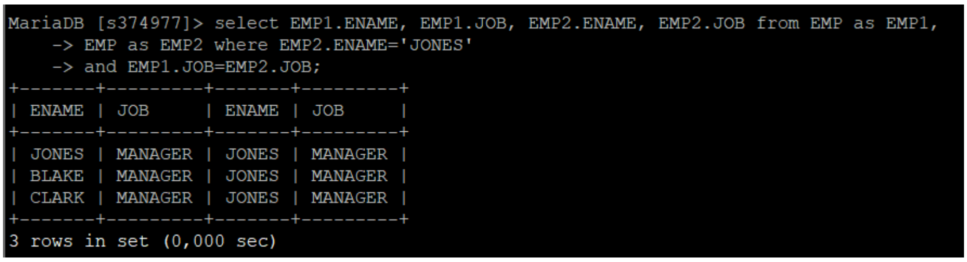
```
MariaDB [s374977]> select EMP1.ENAME, EMP1.SAL from EMP as EMP1, EMP as EMP2 where EMP2.ENAME
='JONES' and EMP1.SAL>EMP2.SAL;
+-------+------+
| ENAME | SAL  |
+-------+------+
| SCOTT | 3000 |
| KING  | 5000 |
| FORD  | 3000 |
+-------+------+
3 rows in set (0,000 sec)
```

21)



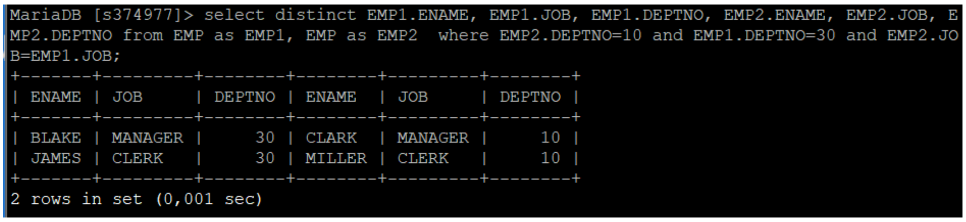
```
MariaDB [s374977]> select EMP1.ENAME, EMP1.SAL, EMP2.ENAME, EMP2.SAL from EMP as EMP1
    -> , EMP as EMP2 where EMP1.MGR=EMP2.EMPNO and EMP1.SAL>EMP2.SAL;
+-------+------+-------+------+
| ENAME | SAL  | ENAME | SAL  |
+-------+------+-------+------+
| SCOTT | 3000 | JONES | 2975 |
| FORD  | 3000 | JONES | 2975 |
+-------+------+-------+------+
2 rows in set (0,000 sec)
```

22)



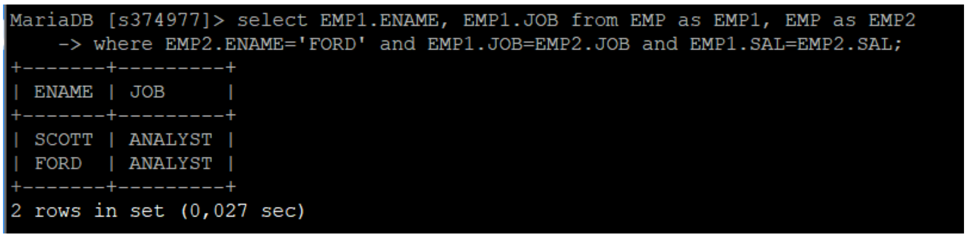
```
MariaDB [s374977]> select EMP1.ENAME, EMP1.JOB, EMP2.ENAME, EMP2.JOB from EMP as EMP1,
    -> EMP as EMP2 where EMP2.ENAME='JONES'
    -> and EMP1.JOB=EMP2.JOB;
+-------+---------+-------+---------+
| ENAME | JOB     | ENAME | JOB     |
+-------+---------+-------+---------+
| JONES | MANAGER | JONES | MANAGER |
| BLAKE | MANAGER | JONES | MANAGER |
| CLARK | MANAGER | JONES | MANAGER |
+-------+---------+-------+---------+
3 rows in set (0,000 sec)
```

23)



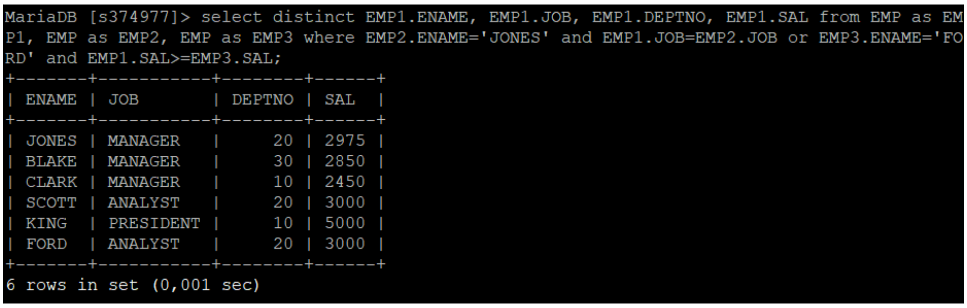
```
MariaDB [s374977]> select distinct EMP1.ENAME, EMP1.JOB, EMP1.DEPTNO, EMP2.ENAME, EMP2.JOB, E
MP2.DEPTNO from EMP as EMP1, EMP as EMP2  where EMP2.DEPTNO=10 and EMP1.DEPTNO=30 and EMP2.JO
B=EMP1.JOB;
+-------+---------+--------+--------+---------+--------+
| ENAME | JOB     | DEPTNO | ENAME  | JOB     | DEPTNO |
+-------+---------+--------+--------+---------+--------+
| BLAKE | MANAGER |     30 | CLARK  | MANAGER |     10 |
| JAMES | CLERK   |     30 | MILLER | CLERK   |     10 |
+-------+---------+--------+--------+---------+--------+
2 rows in set (0,001 sec)
```

24)



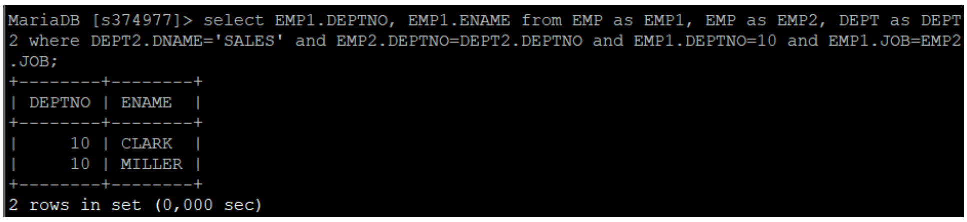
```
MariaDB [s374977]> select EMP1.ENAME, EMP1.JOB from EMP as EMP1, EMP as EMP2
    -> where EMP2.ENAME='FORD' and EMP1.JOB=EMP2.JOB and EMP1.SAL=EMP2.SAL;
+-------+---------+
| ENAME | JOB     |
+-------+---------+
| SCOTT | ANALYST |
| FORD  | ANALYST |
+-------+---------+
2 rows in set (0,027 sec)
```

25)



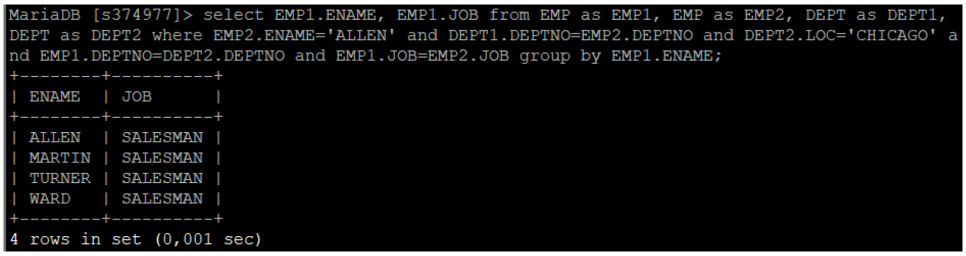
```
MariaDB [s374977]> select distinct EMP1.ENAME, EMP1.JOB, EMP1.DEPTNO, EMP1.SAL from EMP as EM
P1, EMP as EMP2, EMP as EMP3 where EMP2.ENAME='JONES' and EMP1.JOB=EMP2.JOB or EMP3.ENAME='FO
RD' and EMP1.SAL>=EMP3.SAL;
+-------+-----------+--------+------+
| ENAME | JOB       | DEPTNO | SAL  |
+-------+-----------+--------+------+
| JONES | MANAGER   |     20 | 2975 |
| BLAKE | MANAGER   |     30 | 2850 |
| CLARK | MANAGER   |     10 | 2450 |
| SCOTT | ANALYST   |     20 | 3000 |
| KING  | PRESIDENT |     10 | 5000 |
| FORD  | ANALYST   |     20 | 3000 |
+-------+-----------+--------+------+
6 rows in set (0,001 sec)
```

26)



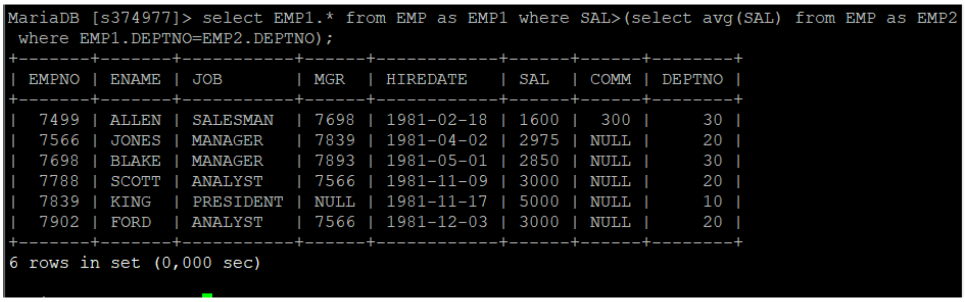
```
MariaDB [s374977]> select EMP1.DEPTNO, EMP1.ENAME from EMP as EMP1, EMP as EMP2, DEPT as DEPT
2 where DEPT2.DNAME='SALES' and EMP2.DEPTNO=DEPT2.DEPTNO and EMP1.DEPTNO=10 and EMP1.JOB=EMP2
.JOB;
+--------+--------+
| DEPTNO | ENAME  |
+--------+--------+
|     10 | CLARK  |
|     10 | MILLER |
+--------+--------+
2 rows in set (0,000 sec)
```

27)



```
MariaDB [s374977]> select EMP1.ENAME, EMP1.JOB from EMP as EMP1, EMP as EMP2, DEPT as DEPT1,
DEPT as DEPT2 where EMP2.ENAME='ALLEN' and DEPT1.DEPTNO=EMP2.DEPTNO and DEPT2.LOC='CHICAGO' a
nd EMP1.DEPTNO=DEPT2.DEPTNO and EMP1.JOB=EMP2.JOB group by EMP1.ENAME;
+--------+----------+
| ENAME  | JOB      |
+--------+----------+
| ALLEN  | SALESMAN |
| MARTIN | SALESMAN |
| TURNER | SALESMAN |
| WARD   | SALESMAN |
+--------+----------+
4 rows in set (0,001 sec)
```

28)



```
MariaDB [s374977]> select EMP1.* from EMP as EMP1 where SAL>(select avg(SAL) from EMP as EMP2
 where EMP1.DEPTNO=EMP2.DEPTNO);
+-------+-------+-----------+------+------------+------+------+--------+
| EMPNO | ENAME | JOB       | MGR  | HIREDATE   | SAL  | COMM | DEPTNO |
+-------+-------+-----------+------+------------+------+------+--------+
|  7499 | ALLEN | SALESMAN  | 7698 | 1981-02-18 | 1600 |  300 |     30 |
|  7566 | JONES | MANAGER   | 7839 | 1981-04-02 | 2975 | NULL |     20 |
|  7698 | BLAKE | MANAGER   | 7893 | 1981-05-01 | 2850 | NULL |     30 |
|  7788 | SCOTT | ANALYST   | 7566 | 1981-11-09 | 3000 | NULL |     20 |
|  7839 | KING  | PRESIDENT | NULL | 1981-11-17 | 5000 | NULL |     10 |
|  7902 | FORD  | ANALYST   | 7566 | 1981-12-03 | 3000 | NULL |     20 |
+-------+-------+-----------+------+------------+------+------+--------+
6 rows in set (0,000 sec)
```

Oppgave 2

1)

Oppgave	Svar
a) Anta at vi har 161 varer plassert i 21 kategorier. Hvor mange rader gir spørringen **SELECT** * **FROM** Vare, Kategori?	Det vil gi 161*21=3381 rader.
b) Hvor mange rader vil en likekobling av tabellene Vare og Kategori med hensyn på KatNr inneholde? Skriv SQL-koden. Hva skjer hvis noen av varene ikke er plassert i en kategori?	Vi vil da få 21 rader. SELECT DISTINCT * FROM Vare, Kategori WHERE Kategori.KatNr=Vare.KatNr; Varer som ikke er plassert i en kategori vil få et nullmerke i KatNr, og vises da ikke i tabellen.

c) Vis alle ordrelinjer påført varenavn og ordredato.	SELECT Ordrelinje.*, Vare.Betegnelse, Ordre.Ordredato
	FROM Ordrelinje, Ordre, Vare
	WHERE Ordrelinje.OrdreNr=Ordre.OrdreNr
	AND Ordrelinje.VNr=Vare.VNr;

d) Utvid SQL-koden fra oppgave 1c med en ny kolonne som viser totalbeløp for hver ordrelinje.	SELECT Ordrelinje.*, Vare.Betegnelse, Ordre.Ordredato, Ordrelinje.PrisPrEnhet*Ordrelinje.Antall AS TotalBeløp
	FROM Ordrelinje, Ordre, Vare
	WHERE Ordrelinje.OrdreNr=Ordre.OrdreNr
	AND Ordrelinje.VNr=Vare.VNr;

e) Vis samlet beløp hver kunde har handlet for.	SELECT Kunde.KNr, Kunde.Fornavn, Kunde.Etternavn, SUM(Ordrelinje.Antall*Ordrelinje.PrisPrEnhet)
	WHERE Ordrelinje.OrdreNr=Ordre.OrdreNr
	AND Kunde.KNr=O.KNr
	GROUP BY Kunde.KNr, Kunde.Fornavn, Kunde.Etternavn;

f) Prøv å utvide SQL-koden fra oppgave 1e med en ny kolonne som inneholder antall ordrer for hver kunde. Hva er problemet?	SELECT OrdrePrKunde.KNr, Beløp, AntallOrdre
	FROM BeløpPrKunde, OrdrePrKunde
	WHERE BeløpPrKunde.KNr= OrdrePrKunde.KNr;

g) Vis samlet beløp pr. ordre.	SELECT OrdreNr,

	SUM(Antall*PrisPrEnhet) AS TotalBelPrOrdre
	FROM Ordrelinje
	GROUP BY OrdreNr;

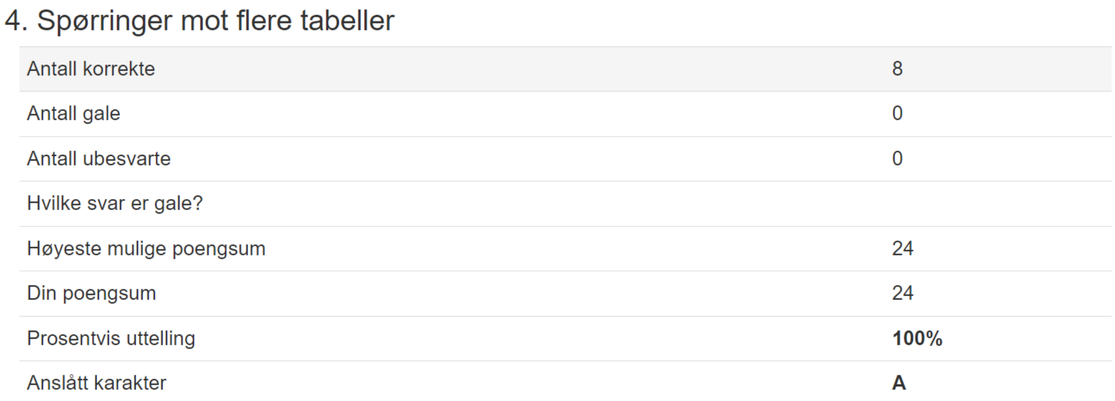
h) Lag en vareliste som for hver vare viser antall enheter på lager og samlet lagerverdi for denne varen.	SELECT VNr, Antall, Antall*Pris AS LagerVerdi
	FROM Vare;

i) Finn samlet verdi av varelageret.	SELECT SUM(Antall*Pris) AS PrisVareLager
	FROM Vare;

j) Finn ut hvor mye hver varekategori har solgt for. Lag en sortert liste med bestselgerne først, og få med navn på kategori i utskriften.	SELECT Kategori.KatNr, Kategori.Navn, SUM(Ordrelinje.Antall*PrisPrEnhet) AS TotalBeløp
	FROM Vare, Ordrelinje, Kategori
	WHERE Vare.VNr=Ordrelinje.VNr AND Vare.KatNr=Kategori.KatNr
	GROUP BY Kategori.KatNr, Kategori.Navn
	ORDER BY SUM(Ordrelinje.Antall*PrisPrEnhet) DESC;

k) Vis alle postnumre der det enten bor en ansatt eller en kunde. Hva med steder der det bor både en ansatt og en kunde? Hva kan du gjøre for å få med navn på poststedet?	SELECT DISTINCT Poststed.PostNr
	FROM Ansatt, Kunde, Poststed
	WHERE Ansatt.PostNr=Poststed.PostNr
	OR Kunde.PostNr=Poststed.PostNr;

2)



4. Spørringer mot flere tabeller

Antall korrekte	8
Antall gale	0
Antall ubesvarte	0
Hvilke svar er gale?	
Høyeste mulige poengsum	24
Din poengsum	24
Prosentvis uttelling	**100%**
Anslått karakter	**A**