DBMS ASSIGNMENT

Name- Himanshu Tiwari
MSC-CSA SEM-1
Class Roll number - 10

1. (Exercise on retrieving records from the table)
EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id)

(a) Find out the employee id, names, salaries of all the employees
(b) List out the employees who works under manager 100
(c) Find the names of the employees who have a salary greater than or equal to 4800
(d) List out the employees whose last name is 'AUSTIN'
(e) Find the names of the employees who works in departments 60,70 and 80
(f) Display the unique Manager_Id.

```
CREATE TABLE EMPLOYEES (
  Employee_Id INT PRIMARY KEY,
  First_Name VARCHAR(255),
  Last_Name VARCHAR(255),
  Email VARCHAR(255),
  Phone_Number VARCHAR(255),
  Hire_Date DATE,
  Job_Id VARCHAR(255),
  Salary DECIMAL(10, 2),
  Commission_Pct DECIMAL(3, 2),
  Manager_Id INT,
  Department_Id INT
);
```

```
mysql> INSERT INTO EMPLOYEES
    -> (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary,
Commission_Pct, Manager_Id, Department_Id)
    -> VALUES
    -> (100, 'Steven', 'King', 'steven.king@example.com', '515.123.4567', '2003-06-17', 'AD_PRES',
24000, NULL, NULL, 90),
    -> (101, 'Neena', 'Kochhar', 'neena.kochhar@example.com', '515.123.4568', '2005-09-21', 'AD_VP',
17000, NULL, 100, 90),
    -> (102, 'Lex', 'De Haan', 'lex.dehaan@example.com', '515.123.4569', '2001-01-13', 'AD_VP',
17000, NULL, 100, 90),
    -> (103, 'Alexander', 'Hunold', 'alexander.hunold@example.com', '590.423.4567', '2006-01-03',
'IT_PROG', 9000, NULL, 102, 60),
    -> (104, 'Bruce', 'Ernst', 'bruce.ernst@example.com', '590.423.4568', '2007-05-21', 'IT_PROG',
6000, NULL, 103, 60),
    -> (105, 'David', 'Austin', 'david.austin@example.com', '590.423.4569', '2005-06-25', 'IT_PROG',
4800, NULL, 103, 60),
    -> (106, 'Valli', 'Pataballa', 'valli.pataballa@example.com', '590.423.4560', '2006-02-05', 'IT_PROG',
4800, NULL, 103, 60),
```

-> (107, 'Diana', 'Lorentz', 'diana.lorentz@example.com', '590.423.5567', '2007-02-07', 'IT_PROG', 4200, NULL, 103, 60),
-> (108, 'Nancy', 'Greenberg', 'nancy.greenberg@example.com', '515.124.4569', '2002-08-17', 'FI_MGR', 12000, NULL, 101, 100),
-> (109, 'Daniel', 'Faviet', 'daniel.faviet@example.com', '515.124.4169', '2002-08-16', 'FI_ACCOUNT', 9000, NULL, 108, 100),
-> (110, 'John', 'Chen', 'john.chen@example.com', '515.124.4269', '2005-09-28', 'FI_ACCOUNT', 8200, NULL, 108, 100),
-> (111, 'Ismael', 'Sciarra', 'ismael.sciarra@example.com', '515.124.4369', '2005-09-30', 'FI_ACCOUNT', 7700, NULL, 108, 100),
-> (112, 'Jose Manuel', 'Urman', 'jose.manuel.urman@example.com', '515.124.4469', '2006-03-07', 'FI_ACCOUNT', 7800, NULL, 108, 100);

```
mysql> select * from employees;
+-------------+------------+-----------+------------------------------+--------------+------------+------------+----------+----------------+------
------+--------------+
| Employee_Id | First_Name | Last_Name | Email                        | Phone_Number | Hire_Date  | Job_Id     | Salary   | Commission_Pct | Manag
er_Id | Department_Id |
+-------------+------------+-----------+------------------------------+--------------+------------+------------+----------+----------------+------
------+--------------+
|         100 | Steven     | King      | steven.king@example.com      | 515.123.4567 | 2003-06-17 | AD_PRES    | 24000.00 |           NULL |
 NULL |           90 |
|         101 | Neena      | Kochhar   | neena.kochhar@example.com    | 515.123.4568 | 2005-09-21 | AD_VP      | 17000.00 |           NULL |
  100 |           90 |
|         102 | Lex        | De Haan   | lex.dehaan@example.com       | 515.123.4569 | 2001-01-13 | AD_VP      | 17000.00 |           NULL |
  100 |           90 |
|         103 | Alexander  | Hunold    | alexander.hunold@example.com | 590.423.4567 | 2006-01-03 | IT_PROG    |  9000.00 |           NULL |
  102 |           60 |
|         104 | Bruce      | Ernst     | bruce.ernst@example.com      | 590.423.4568 | 2007-05-21 | IT_PROG    |  6000.00 |           NULL |
  103 |           60 |
|         105 | David      | Austin    | david.austin@example.com     | 590.423.4569 | 2005-06-25 | IT_PROG    |  4800.00 |           NULL |
  103 |           60 |
|         106 | Valli      | Pataballa | valli.pataballa@example.com  | 590.423.4560 | 2006-02-05 | IT_PROG    |  4800.00 |           NULL |
  103 |           60 |
|         107 | Diana      | Lorentz   | diana.lorentz@example.com    | 590.423.5567 | 2007-02-07 | IT_PROG    |  4200.00 |           NULL |
  103 |           60 |
|         108 | Nancy      | Greenberg | nancy.greenberg@example.com  | 515.124.4569 | 2002-08-17 | FI_MGR     | 12000.00 |           NULL |
  101 |          100 |
|         109 | Daniel     | Faviet    | daniel.faviet@example.com    | 515.124.4169 | 2002-08-16 | FI_ACCOUNT |  9000.00 |           NULL |
  108 |          100 |
|         110 | John       | Chen      | john.chen@example.com        | 515.124.4269 | 2005-09-28 | FI_ACCOUNT |  8200.00 |           NULL |
  108 |          100 |
|         111 | Ismael     | Sciarra   | ismael.sciarra@example.com   | 515.124.4369 | 2005-09-30 | FI_ACCOUNT |  7700.00 |           NULL |
  108 |          100 |
|         112 | Jose Manuel | Urman    | jose.manuel.urman@example.com | 515.124.4469 | 2006-03-07 | FI_ACCOUNT |  7800.00 |           NULL |
  108 |          100 |
```

SELECT Employee_Id, First_Name, Last_Name, Salary
FROM EMPLOYEES;

Employee_Id	First_Name	Last_Name	Salary
100	Steven	King	24000.00
101	Neena	Kochhar	17000.00
102	Lex	De Haan	17000.00
103	Alexander	Hunold	9000.00
104	Bruce	Ernst	6000.00
105	David	Austin	4800.00
106	Valli	Pataballa	4800.00
107	Diana	Lorentz	4200.00
108	Nancy	Greenberg	12000.00
109	Daniel	Faviet	9000.00
110	John	Chen	8200.00
111	Ismael	Sciarra	7700.00
112	Jose Manuel	Urman	7800.00

```
SELECT Employee_Id, First_Name, Last_Name, Manager_Id
FROM EMPLOYEES
WHERE Manager_Id = 100;
```

Employee_Id	First_Name	Last_Name	Manager_Id
101	Neena	Kochhar	100
102	Lex	De Haan	100

```
SELECT First_Name, Last_Name
FROM EMPLOYEES
WHERE Salary >= 4800;
```

First_Name	Last_Name
Steven	King
Neena	Kochhar
Lex	De Haan
Alexander	Hunold
Bruce	Ernst
David	Austin
Valli	Pataballa
Nancy	Greenberg
Daniel	Faviet
John	Chen
Ismael	Sciarra
Jose Manuel	Urman

```
SELECT Employee_Id, First_Name, Last_Name
FROM EMPLOYEES
WHERE Last_Name = 'AUSTIN';
```

Employee_Id	First_Name	Last_Name
105	David	Austin

```
SELECT First_Name, Last_Name
FROM EMPLOYEES
WHERE Department_Id IN (60, 70, 80);
```

First_Name	Last_Name
Alexander	Hunold
Bruce	Ernst
David	Austin
Valli	Pataballa
Diana	Lorentz

```
SELECT DISTINCT Manager_Id
FROM EMPLOYEES;
```

Manager_Id
NULL
100
102
103
101
108

2. (Exercise on updating records in table)

Create Client_master with the following fields(ClientNO, Name, Address, City, State, bal_due)

(a) Insert five records

(b) Find the names of clients whose bal_due> 5000 .

(c) Change the bal_due of ClientNO " C123" to Rs. 5100

(d) Change the name of Client_master to Client12 .

(e) Display the bal_due heading as "BALANCE"

```
mysql> create table Client_master(
    -> clientno int,
    -> name char(200),
    -> address varchar(200),
    -> city char(200),
    -> state char(200),
    -> bal_due int);
```

```
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| clientno | int         | YES  |     | NULL    |       |
| name     | char(200)   | YES  |     | NULL    |       |
| address  | varchar(200)| YES  |     | NULL    |       |
| city     | char(200)   | YES  |     | NULL    |       |
| state    | char(200)   | YES  |     | NULL    |       |
| bal_due  | int         | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
```

```
mysql> INSERT INTO Client_master (clientno, name, address, city, state, bal_due)
    -> VALUES (1, 'John Doe', '123 Main St', 'Anytown', 'CA', 100),
    ->        (2, 'Jane Smith', '456 Oak Ave', 'Sometown', 'NY', 50),
    ->        (3, 'Bob Johnson', '789 Elm St', 'Othertown', 'TX', 200),
    ->        (4, 'Alice Lee', '321 Pine St', 'Newtown', 'MA', 0),
    ->        (5, 'Samuel Chen', '987 Cedar St', 'Somewhere', 'IL', 75);
```

```
+----------+------------+-------------+-----------+-------+---------+
| clientno | name       | address     | city      | state | bal_due |
+----------+------------+-------------+-----------+-------+---------+
|        1 | John Doe   | 123 Main St | Anytown   | CA    |     100 |
|        2 | Jane Smith | 456 Oak Ave | Sometown  | NY    |      50 |
|        3 | Bob Johnson| 789 Elm St  | Othertown | TX    |     200 |
|        4 | Alice Lee  | 321 Pine St | Newtown   | MA    |       0 |
|        5 | Samuel Chen| 987 Cedar St| Somewhere | IL    |      75 |
+----------+------------+-------------+-----------+-------+---------+
```

```
mysql> Select name from Client_master where bal_due>5000;
Empty set (0.00 sec)
```

```
mysql> update Client_master set bal_due=5100 where clientno="1";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> alter table Client_master rename to Client12;
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> select bal_due as BALANCE from Client12;
```

BALANCE
5100
50
200
0
75

3. Rollback and Commit commands

Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary)

(a) Insert five records

(b) Give Increment of 25% salary for Mathematics Department .

(c) Perform Rollback command

(d) Give Increment of 15% salary for Commerce Department (e) Perform commit command

```
mysql> CREATE TABLE Teacher (
    ->   Name varchar(50),
    ->   DeptNo int,
    ->   JoinDate date,
    ->   DeptName varchar(50),
    ->   Location varchar(50),
    ->   Salary decimal(10,2)
    -> );
Query OK, 0 rows affected (0.17 sec)

mysql>
mysql> INSERT INTO Teacher (Name, DeptNo, JoinDate, DeptName, Location, Salary)
    -> VALUES
    ->   ('John Smith', 1, '2010-01-01', 'Mathematics', 'New York', 50000.00),
    ->   ('Jane Doe', 1, '2015-05-01', 'Mathematics', 'New York', 60000.00),
    ->   ('Bob Johnson', 2, '2012-06-01', 'Commerce', 'Los Angeles', 45000.00),
    ->   ('Alice Lee', 2, '2018-02-01', 'Commerce', 'Los Angeles', 55000.00),
    ->   ('Tom Wilson', 3, '2014-09-01', 'History', 'Chicago', 40000.00);
```

Name	DeptNo	JoinDate	DeptName	Location	Salary
John Smith	1	2010-01-01	Mathematics	New York	50000.00
Jane Doe	1	2015-05-01	Mathematics	New York	60000.00
Bob Johnson	2	2012-06-01	Commerce	Los Angeles	45000.00
Alice Lee	2	2018-02-01	Commerce	Los Angeles	55000.00
Tom Wilson	3	2014-09-01	History	Chicago	40000.00

```
UPDATE Teacher SET Salary = Salary * 1.25 WHERE DeptName = 'Mathematics';
```

```
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
mysql> select * from Teacher;
+-------------+--------+------------+-------------+-------------+----------+
| Name        | DeptNo | JoinDate   | DeptName    | Location    | Salary   |
+-------------+--------+------------+-------------+-------------+----------+
| John Smith  |      1 | 2010-01-01 | Mathematics | New York    | 62500.00 |
| Jane Doe    |      1 | 2015-05-01 | Mathematics | New York    | 75000.00 |
| Bob Johnson |      2 | 2012-06-01 | Commerce    | Los Angeles | 45000.00 |
| Alice Lee   |      2 | 2018-02-01 | Commerce    | Los Angeles | 55000.00 |
| Tom Wilson  |      3 | 2014-09-01 | History     | Chicago     | 40000.00 |
+-------------+--------+------------+-------------+-------------+----------+
```

```
mysql> ROLLBACK;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from Teacher;
+-------------+--------+------------+-------------+-------------+----------+
| Name        | DeptNo | JoinDate   | DeptName    | Location    | Salary   |
+-------------+--------+------------+-------------+-------------+----------+
| John Smith  |      1 | 2010-01-01 | Mathematics | New York    | 62500.00 |
| Jane Doe    |      1 | 2015-05-01 | Mathematics | New York    | 75000.00 |
| Bob Johnson |      2 | 2012-06-01 | Commerce    | Los Angeles | 45000.00 |
| Alice Lee   |      2 | 2018-02-01 | Commerce    | Los Angeles | 55000.00 |
| Tom Wilson  |      3 | 2014-09-01 | History     | Chicago     | 40000.00 |
+-------------+--------+------------+-------------+-------------+----------+
```

UPDATE Teacher SET Salary = Salary * 1.15 WHERE DeptName = 'Commerce';

```
mysql> UPDATE Teacher SET Salary = Salary * 1.15 WHERE DeptName = 'Commerce';
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from Teacher;
+-------------+--------+------------+-------------+-------------+----------+
| Name        | DeptNo | JoinDate   | DeptName    | Location    | Salary   |
+-------------+--------+------------+-------------+-------------+----------+
| John Smith  |      1 | 2010-01-01 | Mathematics | New York    | 62500.00 |
| Jane Doe    |      1 | 2015-05-01 | Mathematics | New York    | 75000.00 |
| Bob Johnson |      2 | 2012-06-01 | Commerce    | Los Angeles | 51750.00 |
| Alice Lee   |      2 | 2018-02-01 | Commerce    | Los Angeles | 63250.00 |
| Tom Wilson  |      3 | 2014-09-01 | History     | Chicago     | 40000.00 |
+-------------+--------+------------+-------------+-------------+----------+
```

COMMIT;

4 . (Exercise on order by and group by clauses)

Create Sales table with the following fields(Sales No, Salesname, Branch, Salesamount, DOB)

(a) Insert five records

(b) Calculate total salesamount in each branch

(c) Calculate average salesamount in each branch .

(d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09

(e) Display the name and DOB of salesman in alphabetical order of the month.

```
mysql> CREATE TABLE Sales (
    ->     SalesNo INT PRIMARY KEY,
    ->     Salesname VARCHAR(50),
    ->     Branch VARCHAR(50),
    ->     Salesamount DECIMAL(10,2),
    ->     DOB DATE
    -> );
```

```
mysql> desc Sales;
+-------------+---------------+------+-----+---------+-------+
| Field       | Type          | Null | Key | Default | Extra |
+-------------+---------------+------+-----+---------+-------+
| SalesNo     | int           | NO   | PRI | NULL    |       |
| Salesname   | varchar(50)   | YES  |     | NULL    |       |
| Branch      | varchar(50)   | YES  |     | NULL    |       |
| Salesamount | decimal(10,2) | YES  |     | NULL    |       |
| DOB         | date          | YES  |     | NULL    |       |
+-------------+---------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

```
mysql> INSERT INTO Sales VALUES (1, 'John Smith', 'Branch A', 1000.00, '1990-01-01');
mysql> INSERT INTO Sales VALUES (2, 'Jane Doe', 'Branch B', 1500.00, '1995-03-15');
mysql> INSERT INTO Sales VALUES (3, 'Bob Johnson', 'Branch A', 2000.00, '1992-12-10');
mysql> INSERT INTO Sales VALUES (4, 'Sarah Lee', 'Branch C', 500.00, '1998-06-20');
mysql> INSERT INTO Sales VALUES (5, 'Tom Brown', 'Branch B', 3000.00, '1991-12-31');
```

```
mysql> select * from Sales;
+---------+-------------+----------+-------------+------------+
| SalesNo | Salesname   | Branch   | Salesamount | DOB        |
+---------+-------------+----------+-------------+------------+
|       1 | John Smith  | Branch A |     1000.00 | 1990-01-01 |
|       2 | Jane Doe    | Branch B |     1500.00 | 1995-03-15 |
|       3 | Bob Johnson | Branch A |     2000.00 | 1992-12-10 |
|       4 | Sarah Lee   | Branch C |      500.00 | 1998-06-20 |
|       5 | Tom Brown   | Branch B |     3000.00 | 1991-12-31 |
+---------+-------------+----------+-------------+------------+
5 rows in set (0.00 sec)
```

```
mysql> SELECT Branch, SUM(Salesamount) AS TotalSales
    -> FROM Sales
```

```
   -> GROUP BY Branch;
```

```
+----------+------------+
| Branch   | TotalSales |
+----------+------------+
| Branch A |    3000.00 |
| Branch B |    4500.00 |
| Branch C |     500.00 |
+----------+------------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT Branch, AVG(Salesamount) AS AvgSales
   -> FROM Sales
   -> GROUP BY Branch;
```

```
+----------+-------------+
| Branch   | AvgSales    |
+----------+-------------+
| Branch A | 1500.000000 |
| Branch B | 2250.000000 |
| Branch C |  500.000000 |
+----------+-------------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT Salesname, DATE_FORMAT(DOB, '%d-%b-%y') AS DOB_char
   -> FROM Sales
   -> WHERE MONTH(DOB) = 12;
```

```
+-------------+-----------+
| Salesname   | DOB_char  |
+-------------+-----------+
| Bob Johnson | 10-Dec-92 |
| Tom Brown   | 31-Dec-91 |
+-------------+-----------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT Salesname, DATE_FORMAT(DOB, '%d-%b-%y') AS DOB_char
   -> FROM Sales
   -> ORDER BY MONTH(DOB), Salesname;
```

```
+-------------+-----------+
| Salesname   | DOB_char  |
+-------------+-----------+
| John Smith  | 01-Jan-90 |
| Jane Doe    | 15-Mar-95 |
| Sarah Lee   | 20-Jun-98 |
| Bob Johnson | 10-Dec-92 |
| Tom Brown   | 31-Dec-91 |
+-------------+-----------+
5 rows in set (0.00 sec)
```

5. Create an Emp table with the following fields:

(EmpNo, EmpName, Job,Basic, DA, HRA,PF, GrossPay, NetPay)

(Calculate DA as 30% of Basic and HRA as 40% of Basic)

(a) Insert Five Records and calculate GrossPay and NetPay.

(b) Display the employees whose Basic is lowest in each department .

(c) If NetPay is less than <Rs. 10,000 add Rs. 1200 as special allowances .

(d) Display the employees whose GrossPay lies between 10,000 & 20,000

(e) Display all the employees who earn maximum salary .

```
mysql> CREATE TABLE Emp (
    ->   EmpNo int,
    ->   EmpName varchar(50),
    ->   Job varchar(50),
    ->   Basic decimal(10,2),
    ->   DA decimal(10,2),
    ->   HRA decimal(10,2),
    ->   PF decimal(10,2),
    ->   GrossPay decimal(10,2),
    ->   NetPay decimal(10,2)
    -> );
Query OK, 0 rows affected (0.10 sec)

mysql>
mysql> INSERT INTO Emp (EmpNo, EmpName, Job, Basic)
    -> VALUES
    ->   (1, 'John Smith', 'Manager', 50000.00),
    ->   (2, 'Jane Doe', 'Analyst', 40000.00),
    ->   (3, 'Bob Johnson', 'Clerk', 30000.00),
    ->   (4, 'Alice Lee', 'Manager', 55000.00),
    ->   (5, 'Tom Wilson', 'Analyst', 45000.00);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql>
mysql> UPDATE Emp SET DA = Basic * 0.3, HRA = Basic * 0.4, PF = Basic * 0.1,
    ->   GrossPay = Basic + DA + HRA, NetPay = Basic + DA + HRA - PF;
Query OK, 5 rows affected (0.00 sec)
Rows matched: 5  Changed: 5  Warnings: 0
```

```
mysql> select * from Emp;
+-------+-------------+---------+----------+----------+----------+---------+----------+----------+
| EmpNo | EmpName     | Job     | Basic    | DA       | HRA      | PF      | GrossPay | NetPay   |
+-------+-------------+---------+----------+----------+----------+---------+----------+----------+
|     1 | John Smith  | Manager | 50000.00 | 15000.00 | 20000.00 | 5000.00 | 85000.00 | 80000.00 |
|     2 | Jane Doe    | Analyst | 40000.00 | 12000.00 | 16000.00 | 4000.00 | 68000.00 | 64000.00 |
|     3 | Bob Johnson | Clerk   | 30000.00 |  9000.00 | 12000.00 | 3000.00 | 51000.00 | 48000.00 |
|     4 | Alice Lee   | Manager | 55000.00 | 16500.00 | 22000.00 | 5500.00 | 93500.00 | 88000.00 |
|     5 | Tom Wilson  | Analyst | 45000.00 | 13500.00 | 18000.00 | 4500.00 | 76500.00 | 72000.00 |
+-------+-------------+---------+----------+----------+----------+---------+----------+----------+
```

mysql> select min(Basic) from emp group by Job;

```
+------------+
| min(Basic) |
+------------+
|   50000.00 |
|   40000.00 |
|   30000.00 |
+------------+
```

```
mysql> UPDATE Emp SET NetPay = NetPay + 1200 WHERE NetPay < 10000;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0
```

```
mysql> SELECT EmpName, Job, GrossPay
    -> FROM Emp
    -> WHERE GrossPay BETWEEN 10000 AND 20000;
Empty set (0.03 sec)
```

mysql> SELECT EmpName, Job, GrossPay
-> FROM Emp
-> WHERE GrossPay = (SELECT MAX(GrossPay) FROM Emp);

```
+-----------+---------+----------+
| EmpName   | Job     | GrossPay |
+-----------+---------+----------+
| Alice Lee | Manager | 93500.00 |
+-----------+---------+----------+
```

6. Employee Database

An Enterprise wishes to maintain a database to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas

Dept (deptno, dname, loc)
Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno)

a) Update the employee salary by 15%, whose experience is greater than 10 years.
b) Delete the employees, who completed 30 years of service.
c) Display the manager who is having maximum number of employees working under him?
d) Create a view, which contain employee names and their manager

mysql> CREATE TABLE Dept (
-> deptno INT PRIMARY KEY,
-> dname VARCHAR(50),
-> loc VARCHAR(50)
->);
Query OK, 0 rows affected (0.09 sec)

mysql>

```
mysql> CREATE TABLE Emp (
    ->   empno INT PRIMARY KEY,
    ->   ename VARCHAR(50),
    ->   job VARCHAR(50),
    ->   mgr INT,
    ->   hiredate DATE,
    ->   sal FLOAT,
    ->   comm FLOAT,
    ->   deptno INT,
    ->   FOREIGN KEY (deptno) REFERENCES Dept(deptno)
    -> );
Query OK, 0 rows affected (0.07 sec)

mysql> INSERT INTO Dept VALUES
    ->   (10, 'Research', 'New York'),
    ->   (20, 'Sales', 'Dallas'),
    ->   (30, 'Operations', 'Chicago');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql>
mysql> INSERT INTO Emp VALUES
    ->   (7369, 'SMITH', 'CLERK', 7902, '1980-12-17', 800, NULL, 20),
    ->   (7499, 'ALLEN', 'SALESMAN', 7698, '1981-02-20', 1600, 300, 30),
    ->   (7521, 'WARD', 'SALESMAN', 7698, '1981-02-22', 1250, 500, 30),
    ->   (7566, 'JONES', 'MANAGER', 7839, '1981-04-02', 2975, NULL, 20),
    ->   (7654, 'MARTIN', 'SALESMAN', 7698, '1981-09-28', 1250, 1400, 30),
    ->   (7698, 'BLAKE', 'MANAGER', 7839, '1981-05-01', 2850, NULL, 30),
    ->   (7782, 'CLARK', 'MANAGER', 7839, '1981-06-09', 2450, NULL, 10),
    ->   (7788, 'SCOTT', 'ANALYST', 7566, '1982-12-09', 3000, NULL, 20),
    ->   (7839, 'KING', 'PRESIDENT', NULL, '1981-11-17', 5000, NULL, 10),
    ->   (7844, 'TURNER', 'SALESMAN', 7698, '1981-09-08', 1500, 0, 30),
    ->   (7876, 'ADAMS', 'CLERK', 7788, '1983-01-12', 1100, NULL, 20),
    ->   (7900, 'JAMES', 'CLERK', 7698, '1981-12-03', 950, NULL, 30),
    ->   (7902, 'FORD', 'ANALYST', 7566, '1981-12-03', 3000, NULL, 20),
    ->   (7934, 'MILLER', 'CLERK', 7782, '1982-01-23', 1300, NULL, 10);
```

```
mysql> select * from Dept;
+--------+------------+----------+
| deptno | dname      | loc      |
+--------+------------+----------+
|     10 | Research   | New York |
|     20 | Sales      | Dallas   |
|     30 | Operations | Chicago  |
+--------+------------+----------+
3 rows in set (0.00 sec)
```

```
mysql> select * from Emp;
+-------+--------+-----------+------+------------+------+------+--------+
| empno | ename  | job       | mgr  | hiredate   | sal  | comm | deptno |
+-------+--------+-----------+------+------------+------+------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |  800 | NULL |     20 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 | 1600 |  300 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 | 1250 |  500 |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 2975 | NULL |     20 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 | 1250 | 1400 |     30 |
|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 | 2850 | NULL |     30 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 | 2450 | NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 | 3000 | NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 | 5000 | NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 | 1500 |    0 |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 | 1100 | NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |  950 | NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 | 3000 | NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 | 1300 | NULL |     10 |
+-------+--------+-----------+------+------------+------+------+--------+
14 rows in set (0.00 sec)
```

```
mysql> UPDATE Emp
    -> SET sal = sal * 1.15
    -> WHERE DATEDIFF(CURDATE(), hiredate) > 3650;
```

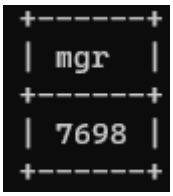
```
mysql> select * from Emp;
+-------+--------+-----------+------+------------+---------+------+--------+
| empno | ename  | job       | mgr  | hiredate   | sal     | comm | deptno |
+-------+--------+-----------+------+------------+---------+------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |     920 | NULL |     20 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 |    1840 |  300 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 |  1437.5 |  500 |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 3421.25 | NULL |     20 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 |  1437.5 | 1400 |     30 |
|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 |  3277.5 | NULL |     30 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 |  2817.5 | NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 |    3450 | NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 |    5750 | NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 |    1725 |    0 |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 |    1265 | NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |  1092.5 | NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 |    3450 | NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 |    1495 | NULL |     10 |
+-------+--------+-----------+------+------------+---------+------+--------+
```

```
DELETE FROM Emp
WHERE DATEDIFF(CURDATE(), hiredate) > 10950;
```

```
mysql> select * from Emp;
Empty set (0.03 sec)
```

```
mysql> create view mgrcount as select mgr, count(empno) total from emp group by mgr;
Query OK, 0 rows affected (0.04 sec)
```

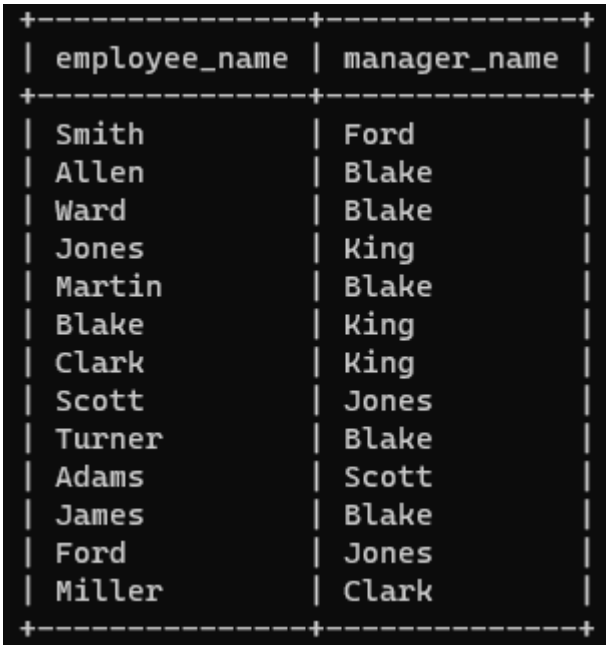
mysql> select mgr from mgrcount where total in (select max(total) from mgrcount);



```
+------+
| mgr  |
+------+
| 7698 |
+------+
```

mysql> CREATE VIEW EmpManager AS
    -> SELECT e.ename AS employee_name, m.ename AS manager_name
    -> FROM Emp e
    -> JOIN Emp m ON e.mgr = m.empno;
Query OK, 0 rows affected (0.05 sec)

mysql> SELECT * FROM EmpManager;



```
+---------------+--------------+
| employee_name | manager_name |
+---------------+--------------+
| Smith         | Ford         |
| Allen         | Blake        |
| Ward          | Blake        |
| Jones         | King         |
| Martin        | Blake        |
| Blake         | King         |
| Clark         | King         |
| Scott         | Jones        |
| Turner        | Blake        |
| Adams         | Scott        |
| James         | Blake        |
| Ford          | Jones        |
| Miller        | Clark        |
+---------------+--------------+
```

7. Using Employee Database perform the following queries

a) Determine the names of employee, who earn more than their managers.
b) Determine the names of employees, who take highest salary in their departments.
c) Determine the employees, who are located at the same place.
d) Determine the employees, whose total salary is like the minimum Salary of any department.
e) Determine the department which does not contain any employees.

mysql> CREATE TABLE Dept (
    -> deptno INT PRIMARY KEY,
    -> dname VARCHAR(50),

```
    ->   loc VARCHAR(50)
    -> );
```

```
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| deptno | int         | NO   | PRI | NULL    |       |
| dname  | varchar(50) | YES  |     | NULL    |       |
| loc    | varchar(50) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
```

```
mysql> CREATE TABLE Emp (
    ->   empno INT PRIMARY KEY,
    ->   ename VARCHAR(50),
    ->   job VARCHAR(50),
    ->   mgr INT,
    ->   hiredate DATE,
    ->   sal DECIMAL(10,2),
    ->   comm DECIMAL(10,2),
    ->   deptno INT,
    ->   FOREIGN KEY (deptno) REFERENCES Dept(deptno)
    -> );
```

```
+----------+---------------+------+-----+---------+-------+
| Field    | Type          | Null | Key | Default | Extra |
+----------+---------------+------+-----+---------+-------+
| empno    | int           | NO   | PRI | NULL    |       |
| ename    | varchar(50)   | YES  |     | NULL    |       |
| job      | varchar(50)   | YES  |     | NULL    |       |
| mgr      | int           | YES  |     | NULL    |       |
| hiredate | date          | YES  |     | NULL    |       |
| sal      | decimal(10,2) | YES  |     | NULL    |       |
| comm     | decimal(10,2) | YES  |     | NULL    |       |
| deptno   | int           | YES  | MUL | NULL    |       |
+----------+---------------+------+-----+---------+-------+
```

```
mysql> INSERT INTO Dept (deptno, dname, loc) VALUES
    ->   (10, 'ACCOUNTING', 'NEW YORK'),
    ->   (20, 'RESEARCH', 'DALLAS'),
    ->   (30, 'SALES', 'CHICAGO'),
    ->   (40, 'OPERATIONS', 'BOSTON');
```

```
mysql> select * from Dept;
+--------+------------+----------+
| deptno | dname      | loc      |
+--------+------------+----------+
|     10 | ACCOUNTING | NEW YORK |
|     20 | RESEARCH   | DALLAS   |
|     30 | SALES      | CHICAGO  |
|     40 | OPERATIONS | BOSTON   |
+--------+------------+----------+
```

```
mysql> INSERT INTO Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno) VALUES
    ->   (7369, 'SMITH', 'CLERK', 7902, '1980-12-17', 800, NULL, 20),
    ->   (7499, 'ALLEN', 'SALESMAN', 7698, '1981-02-20', 1600, 300, 30),
    ->   (7521, 'WARD', 'SALESMAN', 7698, '1981-02-22', 1250, 500, 30),
    ->   (7566, 'JONES', 'MANAGER', 7839, '1981-04-02', 2975, NULL, 20),
    ->   (7654, 'MARTIN', 'SALESMAN', 7698, '1981-09-28', 1250, 1400, 30),
    ->   (7698, 'BLAKE', 'MANAGER', 7839, '1981-05-01', 2850, NULL, 30),
    ->   (7782, 'CLARK', 'MANAGER', 7839, '1981-06-09', 2450, NULL, 10),
    ->   (7788, 'SCOTT', 'ANALYST', 7566, '1982-12-09', 3000, NULL, 20),
    ->   (7839, 'KING', 'PRESIDENT', NULL, '1981-11-17', 5000, NULL, 10),
    ->   (7844, 'TURNER', 'SALESMAN', 7698, '1981-09-08', 1500, 0, 30),
    ->   (7876, 'ADAMS', 'CLERK', 7788, '1983-01-12', 1100, NULL, 20),
    ->   (7900, 'JAMES', 'CLERK', 7698, '1981-12-03', 950, NULL, 30),
    ->   (7902, 'FORD', 'ANALYST', 7566, '1981-12-03', 3000, NULL, 20),
    ->   (7934, 'MILLER', 'CLERK', 7782, '1982-01-23', 1300, NULL, 10);
```

```
mysql> select * from Emp;
+-------+--------+-----------+------+------------+---------+---------+--------+
| empno | ename  | job       | mgr  | hiredate   | sal     | comm    | deptno |
+-------+--------+-----------+------+------------+---------+---------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |  800.00 |    NULL |     20 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 | 1600.00 |  300.00 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 | 1250.00 |  500.00 |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 2975.00 |    NULL |     20 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 | 1250.00 | 1400.00 |     30 |
|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 | 2850.00 |    NULL |     30 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 | 2450.00 |    NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 | 3000.00 |    NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 | 5000.00 |    NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 | 1500.00 |    0.00 |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 | 1100.00 |    NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |  950.00 |    NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 | 3000.00 |    NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 | 1300.00 |    NULL |     10 |
+-------+--------+-----------+------+------------+---------+---------+--------+
```

```
mysql> SELECT e.ename
    -> FROM Emp e
    -> INNER JOIN Emp m ON e.mgr = m.empno
    -> WHERE e.sal > m.sal;
```

```
+-------+
| ename |
+-------+
| SCOTT |
| FORD  |
+-------+
```

```
mysql> SELECT e.ename
    -> FROM Emp e
    -> WHERE e.sal = (
    ->   SELECT MAX(sal)
    ->   FROM Emp
```

```
    ->   WHERE deptno = e.deptno
    -> );
```

ename
BLAKE
SCOTT
KING
FORD

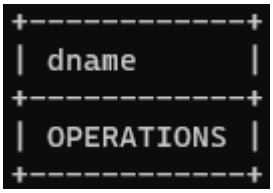
```
mysql> select ename,dname from emp , dept where emp.deptno=dept.deptno order by dname;
```

ename	dname
CLARK	ACCOUNTING
KING	ACCOUNTING
MILLER	ACCOUNTING
SMITH	RESEARCH
JONES	RESEARCH
SCOTT	RESEARCH
ADAMS	RESEARCH
FORD	RESEARCH
ALLEN	SALES
WARD	SALES
MARTIN	SALES
BLAKE	SALES
TURNER	SALES
JAMES	SALES

```
mysql> select empno, ename, sal from emp where sal in(select min(sal) from emp group by
    -> deptno);
```

empno	ename	sal
7369	SMITH	800.00
7900	JAMES	950.00
7934	MILLER	1300.00

```
mysql> select dname from dept where deptno not in(select deptno from emp);
```



dname
OPERATIONS

8. Consider the following tables namely "DEPARTMENTS" and "EMPLOYEES" Their schemas are as follows,

Departments (dept _no , dept_ name , dept_location);

Employees (emp_id , emp_name , emp_salary,dept_no);

a) Develop a query to grant all privileges of employees table into departments table
b) Develop a query to grant some privileges of employees table into departments table
c) Develop a query to revoke all privileges of employees table from departments table
d) Develop a query to revoke some privileges of employees table from departments table
e) Write a query to implement the save point.

```
mysql> CREATE TABLE Departments (
   ->   dept_no INT,
   ->   dept_name VARCHAR(50),
   ->   dept_location VARCHAR(50)
   -> );
Query OK, 0 rows affected (0.10 sec)

mysql>
mysql> CREATE TABLE Employees (
   ->   emp_id INT,
   ->   emp_name VARCHAR(50),
   ->   emp_salary INT,
   ->   dept_no INT
   -> );
Query OK, 0 rows affected (0.06 sec)

mysql>
mysql> INSERT INTO Departments VALUES
   -> (1, 'HR', 'New York'),
   -> (2, 'IT', 'San Francisco'),
   -> (3, 'Marketing', 'Chicago');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql>
mysql> INSERT INTO Employees VALUES
   -> (1, 'John', 50000, 1),
   -> (2, 'Jane', 60000, 2),
   -> (3, 'Mike', 55000, 3);
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> select * from Employees;
+--------+----------+------------+---------+
| emp_id | emp_name | emp_salary | dept_no |
+--------+----------+------------+---------+
|      1 | John     |      50000 |       1 |
|      2 | Jane     |      60000 |       2 |
|      3 | Mike     |      55000 |       3 |
+--------+----------+------------+---------+
3 rows in set (0.00 sec)

mysql> select * from Departments;
+---------+-----------+---------------+
| dept_no | dept_name | dept_location |
+---------+-----------+---------------+
|       1 | HR        | New York      |
|       2 | IT        | San Francisco |
|       3 | Marketing | Chicago       |
+---------+-----------+---------------+
3 rows in set (0.00 sec)
```

GRANT ALL PRIVILEGES ON Employees TO Departments;

GRANT SELECT, UPDATE(emp_name, emp_salary) ON Employees TO Departments;

REVOKE ALL PRIVILEGES ON Employees FROM Departments;

REVOKE SELECT, UPDATE(emp_name, emp_salary) ON Employees FROM Departments;

SAVEPOINT my_savepoint;

ROLLBACK TO my_savepoint;

9. Using the tables "DEPARTMENTS" and "EMPLOYEES" perform the following queries

a) Display the employee details, departments that the departments are same in both the emp and dept.
b) Display the employee name and Department name by implementing a left outer join.
c) Display the employee name and Department name by implementing a right outer join.
d) Display the details of those who draw the salary greater than the average salary.

```
mysql> CREATE TABLE Departments (
    -> dept_no INT PRIMARY KEY,
    -> dept_name VARCHAR(255),
    -> dept_location VARCHAR(255)
```

```
   -> );

mysql> INSERT INTO Departments (dept_no, dept_name, dept_location)
   -> VALUES (10, 'Sales', 'New York'),
   ->        (20, 'Marketing', 'Los Angeles'),
   ->        (30, 'Human Resources', 'Chicago');
```

```
+---------+-----------------+---------------+
| dept_no | dept_name       | dept_location |
+---------+-----------------+---------------+
|      10 | Sales           | New York      |
|      20 | Marketing       | Los Angeles   |
|      30 | Human Resources | Chicago       |
+---------+-----------------+---------------+
```

```
mysql> CREATE TABLE Employees (
   ->   emp_id INT PRIMARY KEY,
   ->   emp_name VARCHAR(255),
   ->   emp_salary DECIMAL(10, 2),
   ->   dept_no INT,
   ->   FOREIGN KEY (dept_no) REFERENCES Departments(dept_no)
   -> );

mysql> INSERT INTO Employees (emp_id, emp_name, emp_salary, dept_no)
   -> VALUES (1001, 'John Smith', 50000.00, 10),
   ->        (1002, 'Jane Doe', 55000.00, 20),
   ->        (1003, 'Bob Johnson', 60000.00, 20),
   ->        (1004, 'Mary Lee', 65000.00, 30),
   ->        (1005, 'Tom Chen', 70000.00, 30);
```

```
+--------+-------------+------------+---------+
| emp_id | emp_name    | emp_salary | dept_no |
+--------+-------------+------------+---------+
|   1001 | John Smith  |   50000.00 |      10 |
|   1002 | Jane Doe    |   55000.00 |      20 |
|   1003 | Bob Johnson |   60000.00 |      20 |
|   1004 | Mary Lee    |   65000.00 |      30 |
|   1005 | Tom Chen    |   70000.00 |      30 |
+--------+-------------+------------+---------+
```

```
mysql> select * from Departments INNER JOIN Employees ON
Employees.dept_no=Departments.dept_no;
```

```
mysql> select * from Departments INNER JOIN Employees ON Employees.dept_no=Departments.dept_no;
+---------+-----------------+---------------+--------+-------------+------------+---------+
| dept_no | dept_name       | dept_location | emp_id | emp_name    | emp_salary | dept_no |
+---------+-----------------+---------------+--------+-------------+------------+---------+
|      10 | Sales           | New York      |   1001 | John Smith  |   50000.00 |      10 |
|      20 | Marketing       | Los Angeles   |   1002 | Jane Doe    |   55000.00 |      20 |
|      20 | Marketing       | Los Angeles   |   1003 | Bob Johnson |   60000.00 |      20 |
|      30 | Human Resources | Chicago       |   1004 | Mary Lee    |   65000.00 |      30 |
|      30 | Human Resources | Chicago       |   1005 | Tom Chen    |   70000.00 |      30 |
+---------+-----------------+---------------+--------+-------------+------------+---------+
```

mysql> select Departments.dept_name,Employees.emp_name from Departments LEFT JOIN Employees ON Employees.dept_no=Departments.dept_no;

dept_name	emp_name
Sales	John Smith
Marketing	Jane Doe
Marketing	Bob Johnson
Human Resources	Mary Lee
Human Resources	Tom Chen

mysql> select Departments.dept_name,Employees.emp_name from Departments RIGHT JOIN Employees ON Employees.dept_no=Departments.dept_no;

dept_name	emp_name
Sales	John Smith
Marketing	Jane Doe
Marketing	Bob Johnson
Human Resources	Mary Lee
Human Resources	Tom Chen

mysql> select * from Employees where emp_salary> (select AVG(emp_salary) from Employees);

emp_id	emp_name	emp_salary	dept_no
1004	Mary Lee	65000.00	30
1005	Tom Chen	70000.00	30