Assignment 4

Solve the following

1. write a procedure to insert record into employee table.

the procedure should accept empno, ename, sal, job, hiredate as input parameter

write insert statement inside procedure insert_rec to add one record into table

create procedure insert_rec(peno int,pnm varchar(20),psal decimal(9,2),pjob

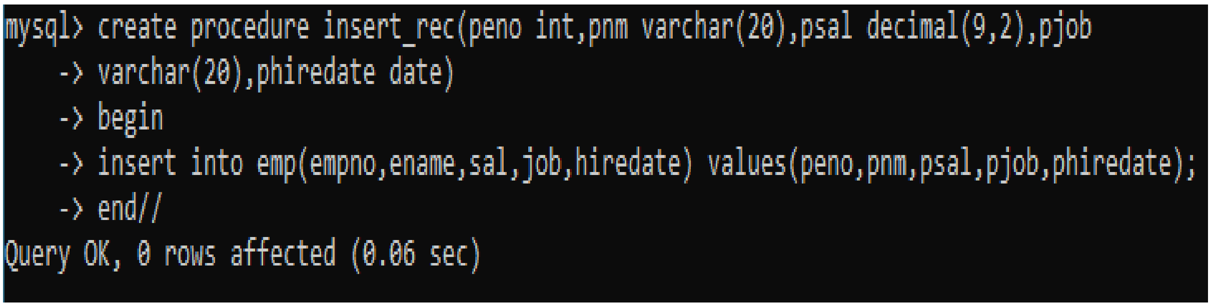
varchar(20),phiredate date)

begin

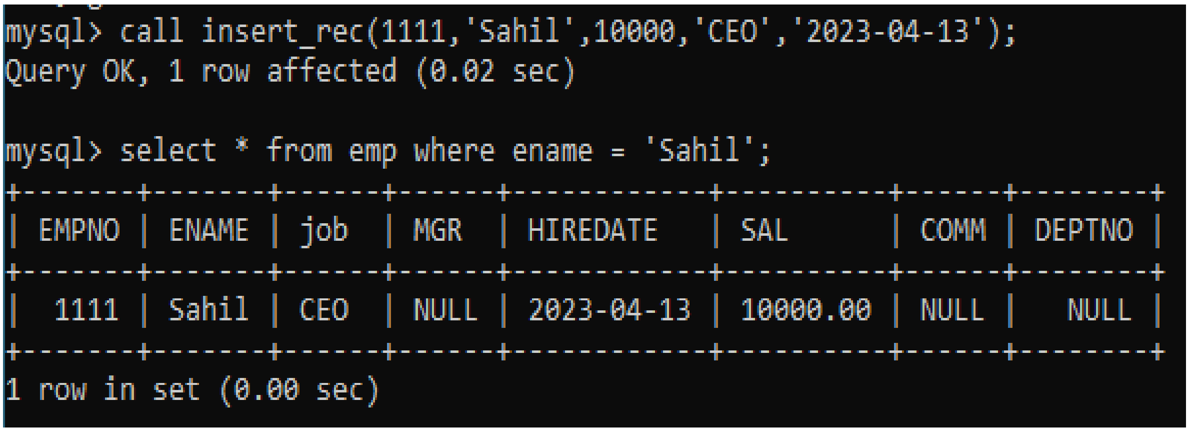
insert into emp(empno,ename,sal,job,hiredate)

values(peno,pnm,psal,pjob,phiredate)

end//

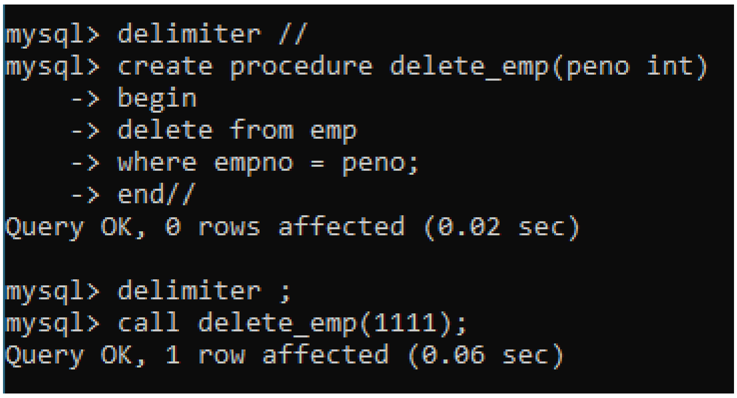


```
mysql> create procedure insert_rec(peno int,pnm varchar(20),psal decimal(9,2),pjob
    -> varchar(20),phiredate date)
    -> begin
    -> insert into emp(empno,ename,sal,job,hiredate) values(peno,pnm,psal,pjob,phiredate);
    -> end//
Query OK, 0 rows affected (0.06 sec)
```

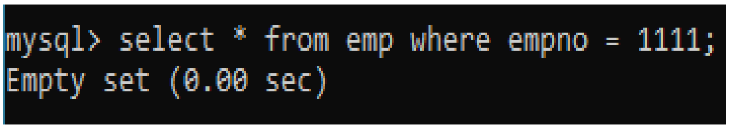


```
mysql> call insert_rec(1111,'Sahil',10000,'CEO','2023-04-13');
Query OK, 1 row affected (0.02 sec)

mysql> select * from emp where ename = 'Sahil';
+-------+-------+------+------+------------+----------+------+--------+
| EMPNO | ENAME | job  | MGR  | HIREDATE   | SAL      | COMM | DEPTNO |
+-------+-------+------+------+------------+----------+------+--------+
|  1111 | Sahil | CEO  | NULL | 2023-04-13 | 10000.00 | NULL |   NULL |
+-------+-------+------+------+------------+----------+------+--------+
1 row in set (0.00 sec)
```

2. write a procedure to delete record from employee table the procedure should accept empno as input parameter. write delete statement inside procedure delete_emp to delete one record from emp table.

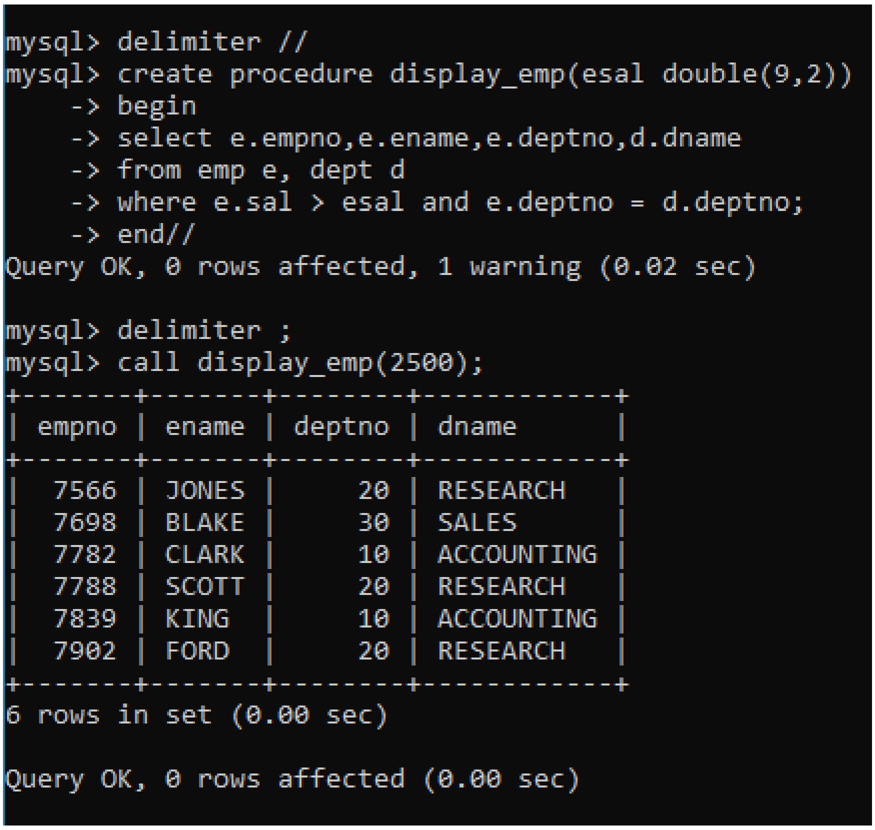


```
mysql> delimiter //
mysql> create procedure delete_emp(peno int)
    -> begin
    -> delete from emp
    -> where empno = peno;
    -> end//
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
mysql> call delete_emp(1111);
Query OK, 1 row affected (0.06 sec)
```



```
mysql> select * from emp where empno = 1111;
Empty set (0.00 sec)
```

3. write a procedure to display empno,ename,deptno,dname for all employees with sal> given salary. pass salary as a parameter to procedure



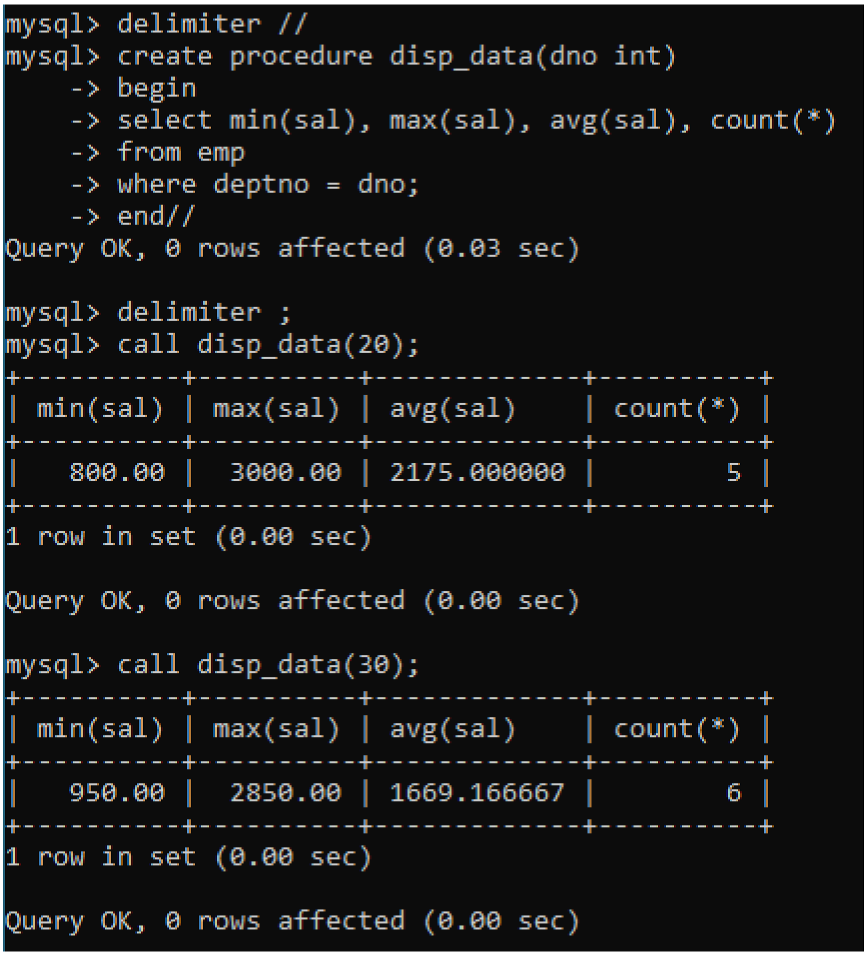
```
mysql> delimiter //
mysql> create procedure display_emp(esal double(9,2))
    -> begin
    -> select e.empno,e.ename,e.deptno,d.dname
    -> from emp e, dept d
    -> where e.sal > esal and e.deptno = d.deptno;
    -> end//
Query OK, 0 rows affected, 1 warning (0.02 sec)

mysql> delimiter ;
mysql> call display_emp(2500);
+-------+-------+--------+------------+
| empno | ename | deptno | dname      |
+-------+-------+--------+------------+
|  7566 | JONES |     20 | RESEARCH   |
|  7698 | BLAKE |     30 | SALES      |
|  7782 | CLARK |     10 | ACCOUNTING |
|  7788 | SCOTT |     20 | RESEARCH   |
|  7839 | KING  |     10 | ACCOUNTING |
|  7902 | FORD  |     20 | RESEARCH   |
+-------+-------+--------+------------+
6 rows in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

4. write a procedure to find min,max,avg of salary and number of employees in the given deptno.

deptno --→ in parameter

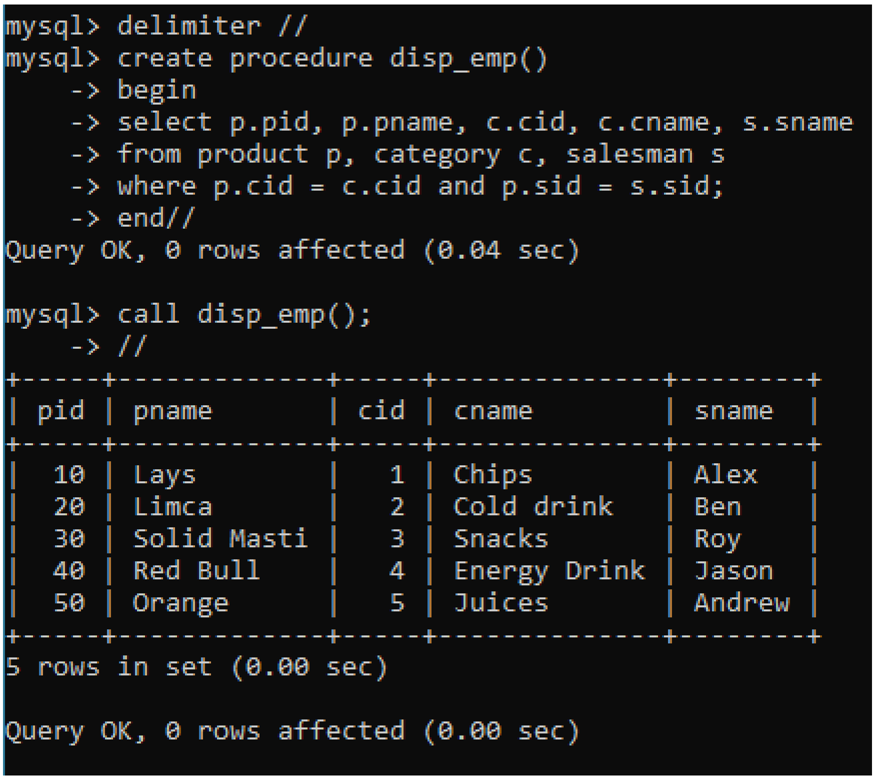
min,max,avg and count ---→ out type parameter

execute procedure and then display values min,max,avg and count



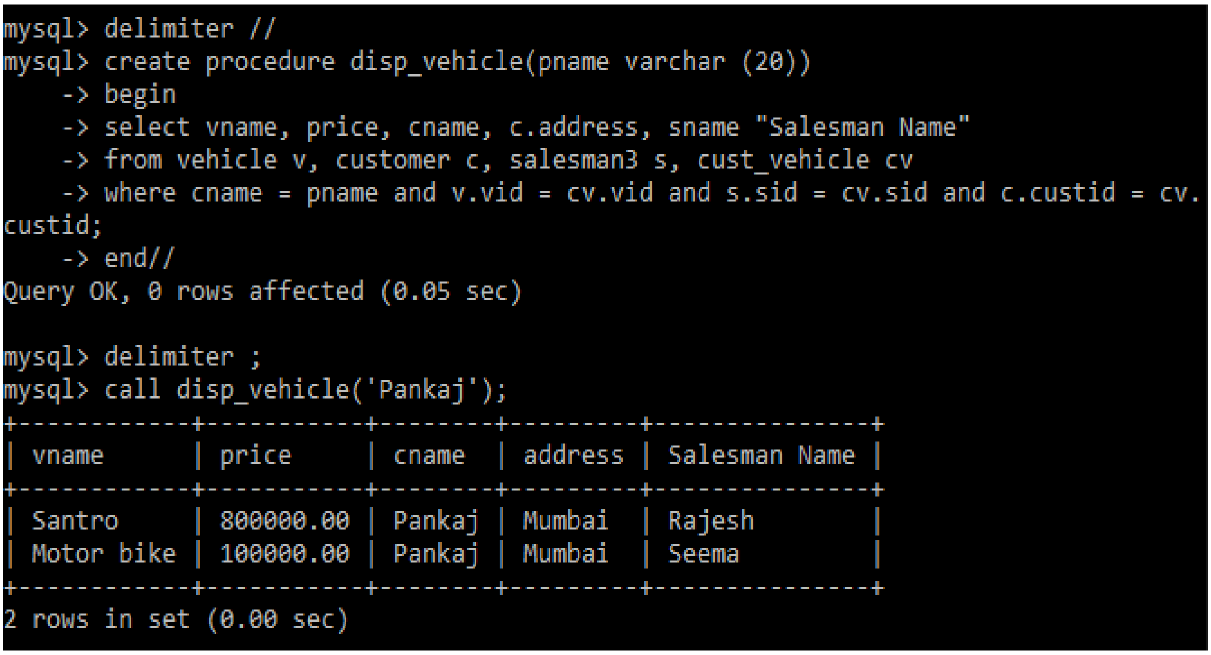
```
mysql> delimiter //
mysql> create procedure disp_data(dno int)
    -> begin
    -> select min(sal), max(sal), avg(sal), count(*)
    -> from emp
    -> where deptno = dno;
    -> end//
Query OK, 0 rows affected (0.03 sec)

mysql> delimiter ;
mysql> call disp_data(20);
+----------+----------+-------------+----------+
| min(sal) | max(sal) | avg(sal)    | count(*) |
+----------+----------+-------------+----------+
|   800.00 |  3000.00 | 2175.000000 |        5 |
+----------+----------+-------------+----------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> call disp_data(30);
+----------+----------+-------------+----------+
| min(sal) | max(sal) | avg(sal)    | count(*) |
+----------+----------+-------------+----------+
|   950.00 |  2850.00 | 1669.166667 |        6 |
+----------+----------+-------------+----------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

5. write a procedure to display all pid,pname,cid,cname and salesman name(use product,category and salesman table)



```
mysql> delimiter //
mysql> create procedure disp_emp()
    -> begin
    -> select p.pid, p.pname, c.cid, c.cname, s.sname
    -> from product p, category c, salesman s
    -> where p.cid = c.cid and p.sid = s.sid;
    -> end//
Query OK, 0 rows affected (0.04 sec)

mysql> call disp_emp();
    -> //
+-----+-------------+-----+---------------+--------+
| pid | pname       | cid | cname         | sname  |
+-----+-------------+-----+---------------+--------+
|  10 | Lays        |   1 | Chips         | Alex   |
|  20 | Limca       |   2 | Cold drink    | Ben    |
|  30 | Solid Masti |   3 | Snacks        | Roy    |
|  40 | Red Bull    |   4 | Energy Drink  | Jason  |
|  50 | Orange      |   5 | Juices        | Andrew |
+-----+-------------+-----+---------------+--------+
5 rows in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

6. write a procedure to display all vehicles bought by a customer. pass customer name as a parameter.(use vehicle,salesman,customer and relation table)

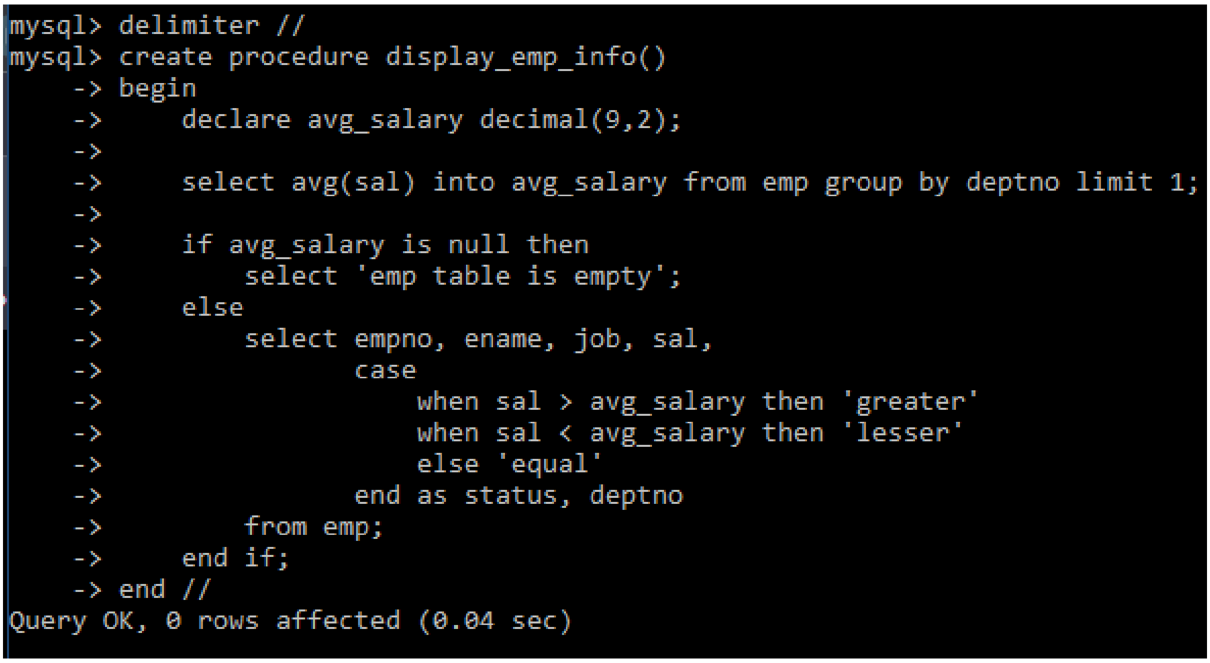


```
mysql> delimiter //
mysql> create procedure disp_vehicle(pname varchar (20))
    -> begin
    -> select vname, price, cname, c.address, sname "Salesman Name"
    -> from vehicle v, customer c, salesman3 s, cust_vehicle cv
    -> where cname = pname and v.vid = cv.vid and s.sid = cv.sid and c.custid = cv.
custid;
    -> end//
Query OK, 0 rows affected (0.05 sec)

mysql> delimiter ;
mysql> call disp_vehicle('Pankaj');
+------------+-----------+--------+---------+---------------+
| vname      | price     | cname  | address | Salesman Name |
+------------+-----------+--------+---------+---------------+
| Santro     | 800000.00 | Pankaj | Mumbai  | Rajesh        |
| Motor bike | 100000.00 | Pankaj | Mumbai  | Seema         |
+------------+-----------+--------+---------+---------------+
2 rows in set (0.00 sec)
```

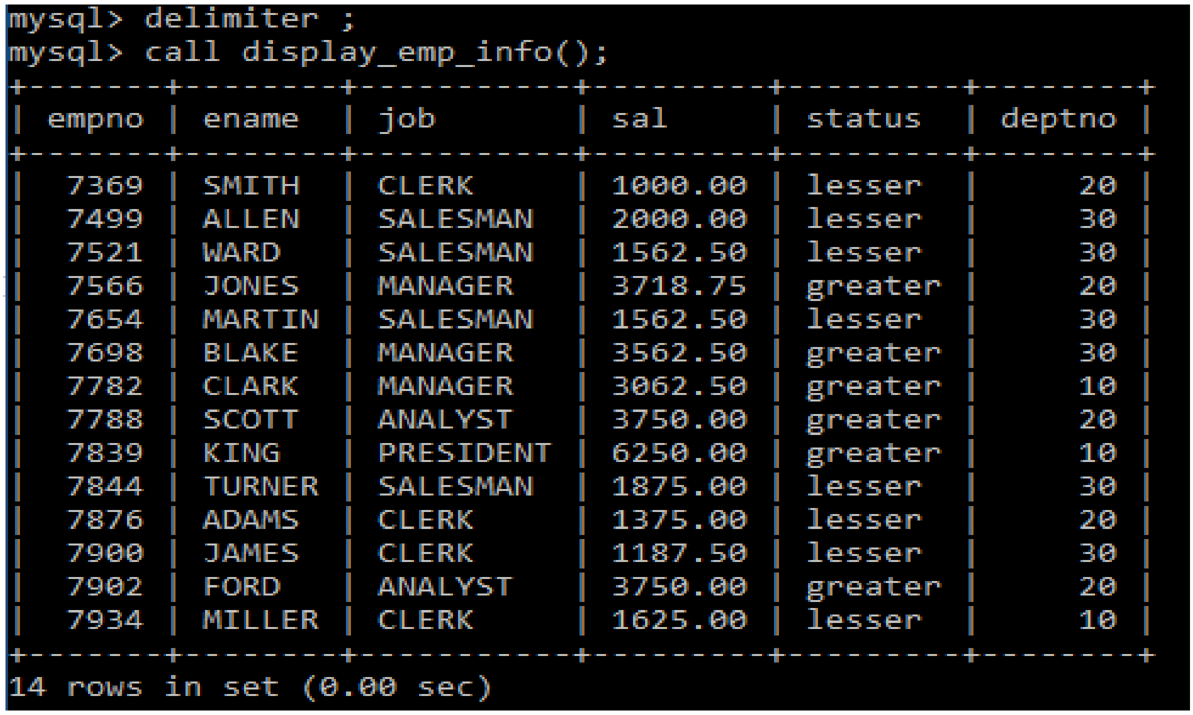
7. Write a procedure that displays the following information of all emp

Empno,Name,job,Salary,Status,deptno

Note: - Status will be (Greater, Lesser or Equal) respective to average salary of their own department. Display an error message Emp table is empty if there is no matching record.



```
mysql> delimiter //
mysql> create procedure display_emp_info()
    -> begin
    ->     declare avg_salary decimal(9,2);
    ->
    ->     select avg(sal) into avg_salary from emp group by deptno limit 1;
    ->
    ->     if avg_salary is null then
    ->         select 'emp table is empty';
    ->     else
    ->         select empno, ename, job, sal,
    ->                case
    ->                    when sal > avg_salary then 'greater'
    ->                    when sal < avg_salary then 'lesser'
    ->                    else 'equal'
    ->                end as status, deptno
    ->         from emp;
    ->     end if;
    -> end //
Query OK, 0 rows affected (0.04 sec)
```



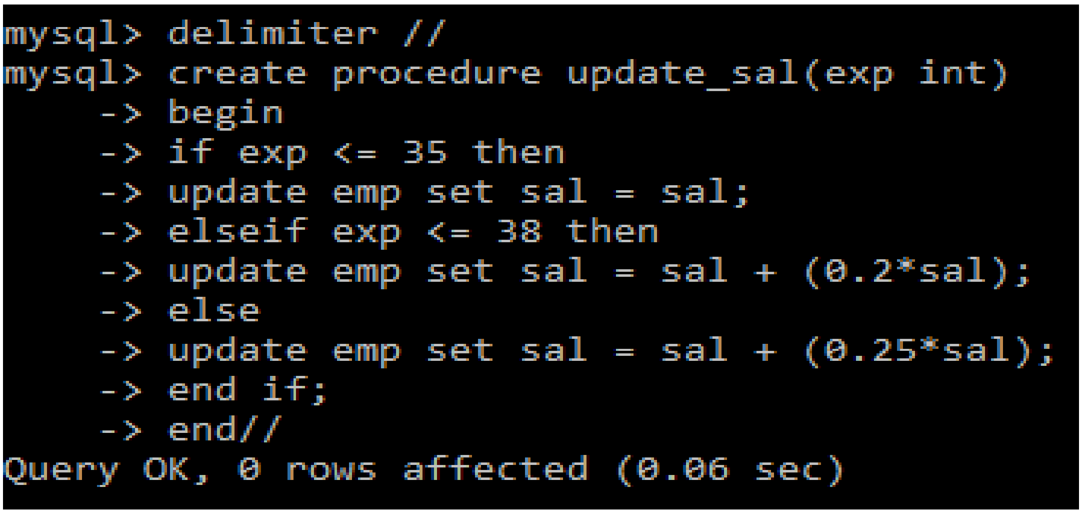
```
mysql> delimiter ;
mysql> call display_emp_info();
+-------+--------+-----------+---------+---------+--------+
| empno | ename  | job       | sal     | status  | deptno |
+-------+--------+-----------+---------+---------+--------+
|  7369 | SMITH  | CLERK     | 1000.00 | lesser  |     20 |
|  7499 | ALLEN  | SALESMAN  | 2000.00 | lesser  |     30 |
|  7521 | WARD   | SALESMAN  | 1562.50 | lesser  |     30 |
|  7566 | JONES  | MANAGER   | 3718.75 | greater |     20 |
|  7654 | MARTIN | SALESMAN  | 1562.50 | lesser  |     30 |
|  7698 | BLAKE  | MANAGER   | 3562.50 | greater |     30 |
|  7782 | CLARK  | MANAGER   | 3062.50 | greater |     10 |
|  7788 | SCOTT  | ANALYST   | 3750.00 | greater |     20 |
|  7839 | KING   | PRESIDENT | 6250.00 | greater |     10 |
|  7844 | TURNER | SALESMAN  | 1875.00 | lesser  |     30 |
|  7876 | ADAMS  | CLERK     | 1375.00 | lesser  |     20 |
|  7900 | JAMES  | CLERK     | 1187.50 | lesser  |     30 |
|  7902 | FORD   | ANALYST   | 3750.00 | greater |     20 |
|  7934 | MILLER | CLERK     | 1625.00 | lesser  |     10 |
+-------+--------+-----------+---------+---------+--------+
14 rows in set (0.00 sec)
```

8. Write a procedure to update salary in emp table based on following rules.

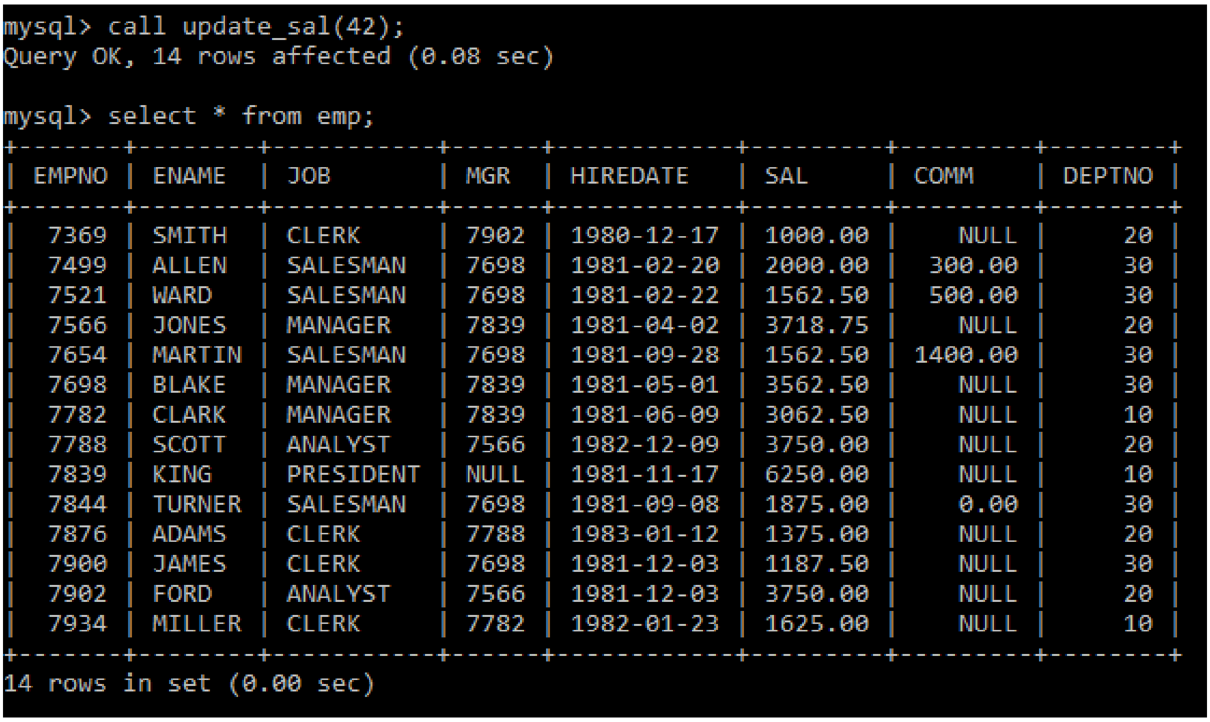
Exp< =35 then no Update

Exp> 35 and <=38 then 20% of salary

Exp> 38 then 25% of salary



```
mysql> delimiter //
mysql> create procedure update_sal(exp int)
    -> begin
    -> if exp <= 35 then
    -> update emp set sal = sal;
    -> elseif exp <= 38 then
    -> update emp set sal = sal + (0.2*sal);
    -> else
    -> update emp set sal = sal + (0.25*sal);
    -> end if;
    -> end//
Query OK, 0 rows affected (0.06 sec)
```



```
mysql> call update_sal(42);
Query OK, 14 rows affected (0.08 sec)

mysql> select * from emp;
+-------+--------+-----------+------+------------+---------+---------+--------+
| EMPNO | ENAME  | JOB       | MGR  | HIREDATE   | SAL     | COMM    | DEPTNO |
+-------+--------+-----------+------+------------+---------+---------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 | 1000.00 |    NULL |     20 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 | 2000.00 |  300.00 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 | 1562.50 |  500.00 |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 3718.75 |    NULL |     20 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 | 1562.50 | 1400.00 |     30 |
|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 | 3562.50 |    NULL |     30 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 | 3062.50 |    NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 | 3750.00 |    NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 | 6250.00 |    NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 | 1875.00 |    0.00 |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 | 1375.00 |    NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 | 1187.50 |    NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 | 3750.00 |    NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 | 1625.00 |    NULL |     10 |
+-------+--------+-----------+------+------------+---------+---------+--------+
14 rows in set (0.00 sec)
```

9. Write a procedure and a function.

Function: write a function to calculate number of years of experience of employee.(note: pass hiredate as a parameter)

Procedure: Capture the value returned by the above function to calculate the additional allowance for the emp based on the experience.

Additional Allowance = Year of experience x 3000

Calculate the additional allowance

and store Empno, ename,Date of Joining, and Experience in

years and additional allowance in Emp_Allowance table.

create table emp_allowance(

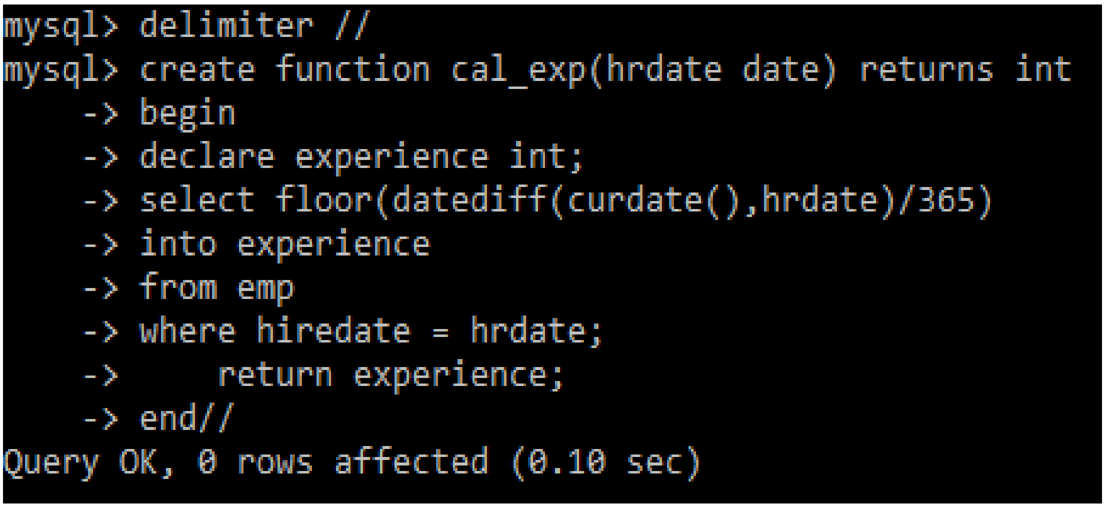
empno int,

ename varchar(20),

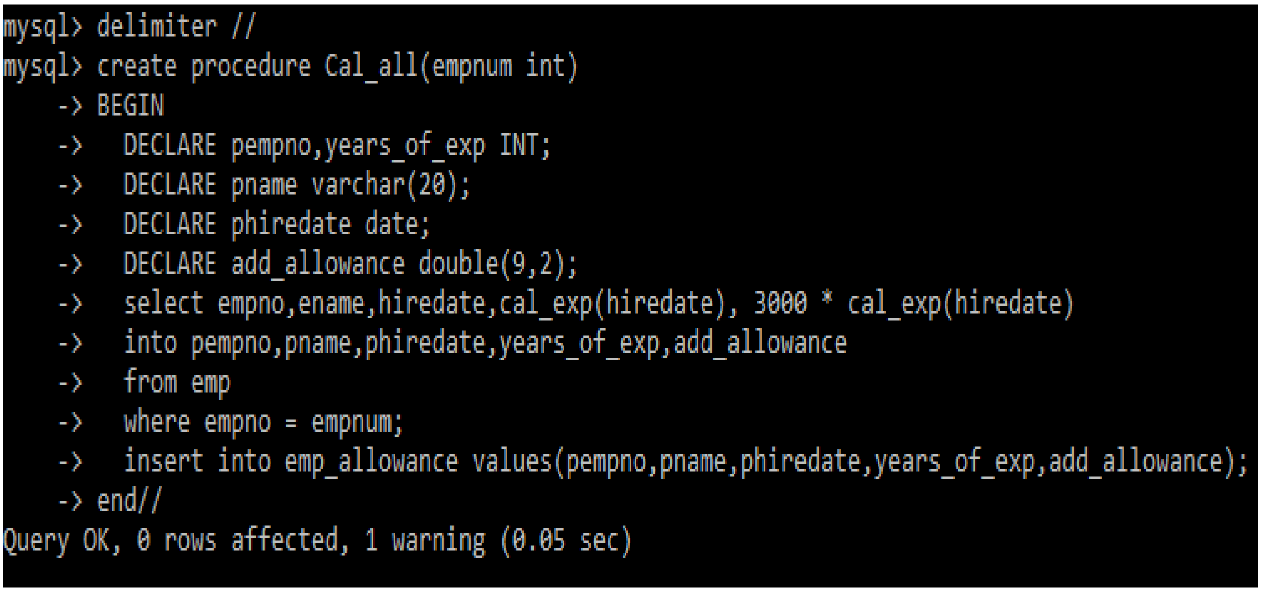
hiredate date,

experience int,

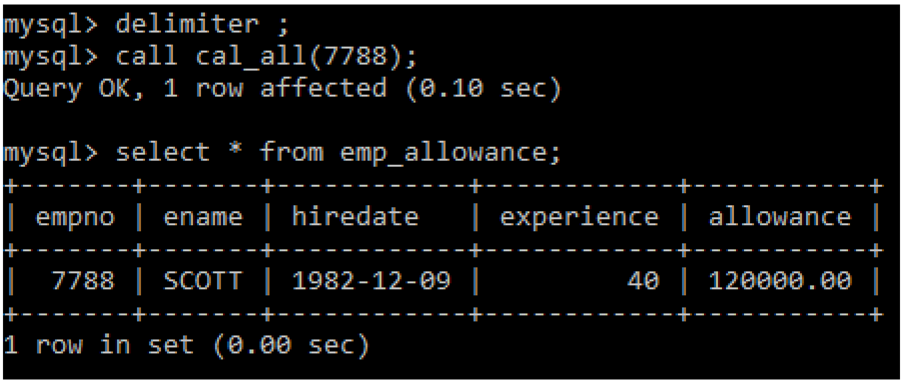
allowance decimal(9,2));



```
mysql> delimiter //
mysql> create function cal_exp(hrdate date) returns int
    -> begin
    -> declare experience int;
    -> select floor(datediff(curdate(),hrdate)/365)
    -> into experience
    -> from emp
    -> where hiredate = hrdate;
    ->     return experience;
    -> end//
Query OK, 0 rows affected (0.10 sec)
```



```
mysql> delimiter //
mysql> create procedure Cal_all(empnum int)
    -> BEGIN
    ->   DECLARE pempno,years_of_exp INT;
    ->   DECLARE pname varchar(20);
    ->   DECLARE phiredate date;
    ->   DECLARE add_allowance double(9,2);
    ->   select empno,ename,hiredate,cal_exp(hiredate), 3000 * cal_exp(hiredate)
    ->   into pempno,pname,phiredate,years_of_exp,add_allowance
    ->   from emp
    ->   where empno = empnum;
    ->   insert into emp_allowance values(pempno,pname,phiredate,years_of_exp,add_allowance);
    -> end//
Query OK, 0 rows affected, 1 warning (0.05 sec)
```



```
mysql> delimiter ;
mysql> call cal_all(7788);
Query OK, 1 row affected (0.10 sec)

mysql> select * from emp_allowance;
+-------+-------+------------+------------+-----------+
| empno | ename | hiredate   | experience | allowance |
+-------+-------+------------+------------+-----------+
|  7788 | SCOTT | 1982-12-09 |         40 | 120000.00 |
+-------+-------+------------+------------+-----------+
1 row in set (0.00 sec)
```

10. Write a function to compute the following. Function should take sal and hiredate as i/p and return the cost to company.

DA = 15% Salary, HRA= 20% of Salary, TA= 8% of Salary.

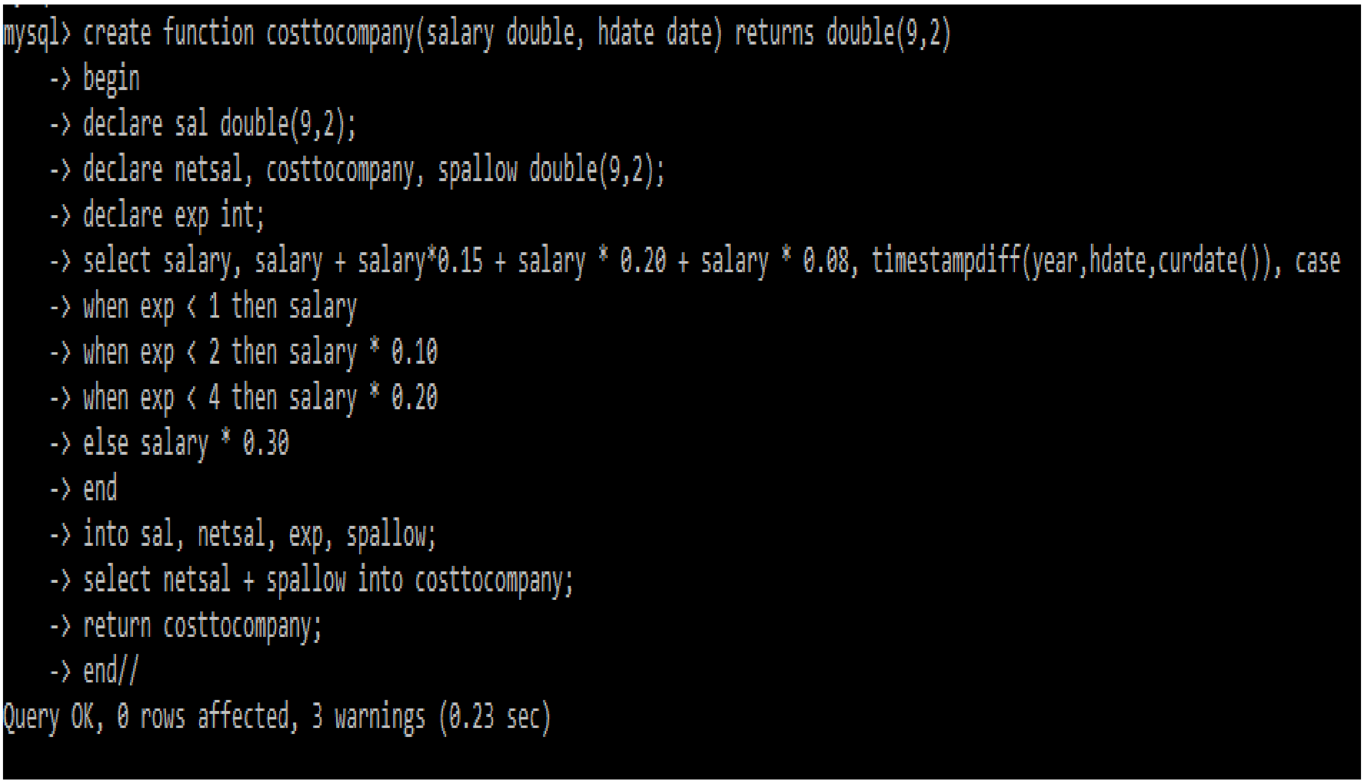
Special Allowance will be decided based on the service in the company.

< 1 Year Nil

>=1 Year< 2 Year 10% of Salary

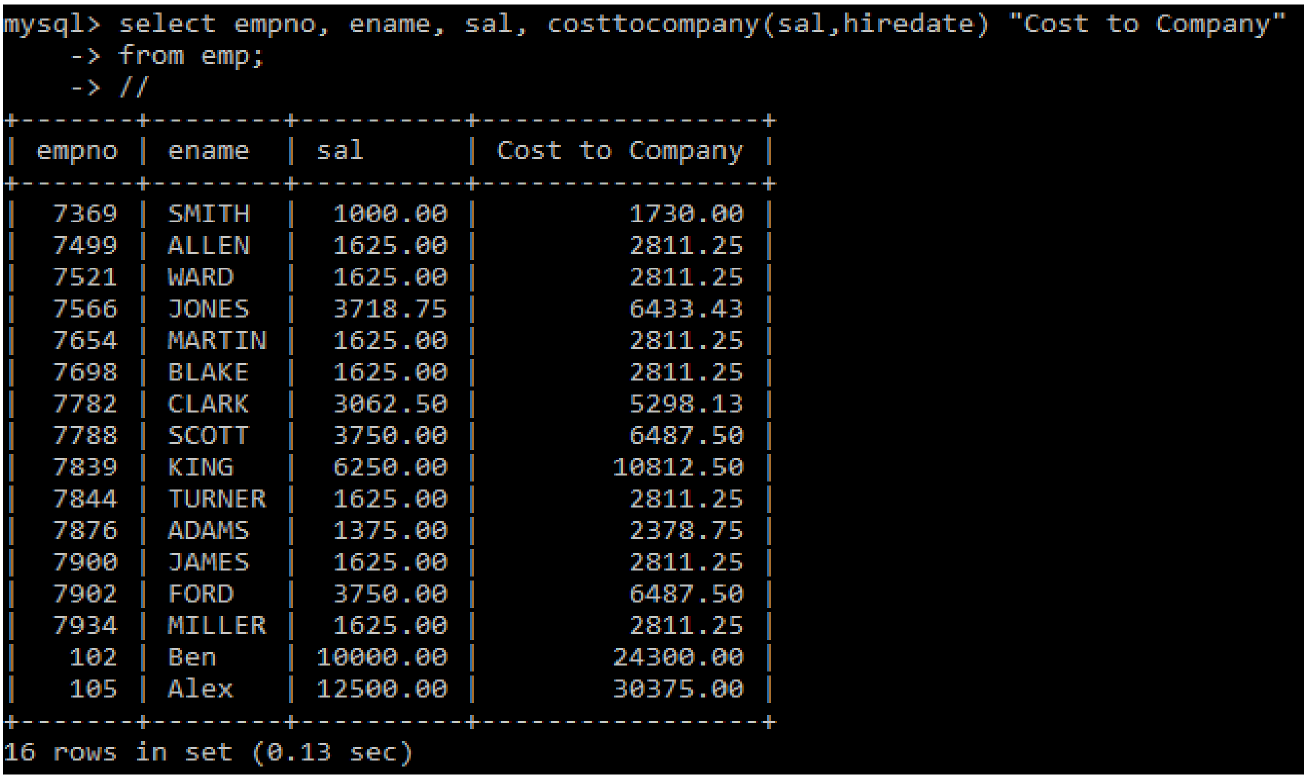
>=2 Year< 4 Year 20% of Salary

>4 Year 30% of Salary



```
mysql> create function costtocompany(salary double, hdate date) returns double(9,2)
    -> begin
    -> declare sal double(9,2);
    -> declare netsal, costtocompany, spallow double(9,2);
    -> declare exp int;
    -> select salary, salary + salary*0.15 + salary * 0.20 + salary * 0.08, timestampdiff(year,hdate,curdate()), case
    -> when exp < 1 then salary
    -> when exp < 2 then salary * 0.10
    -> when exp < 4 then salary * 0.20
    -> else salary * 0.30
    -> end
    -> into sal, netsal, exp, spallow;
    -> select netsal + spallow into costtocompany;
    -> return costtocompany;
    -> end//
Query OK, 0 rows affected, 3 warnings (0.23 sec)
```

11. Write query to display empno,ename,sal,cost to company for all employees(note: use function written in question 10)



```
mysql> select empno, ename, sal, costtocompany(sal,hiredate) "Cost to Company"
    -> from emp;
    -> //
+-------+--------+----------+-----------------+
| empno | ename  | sal      | Cost to Company |
+-------+--------+----------+-----------------+
|  7369 | SMITH  |  1000.00 |         1730.00 |
|  7499 | ALLEN  |  1625.00 |         2811.25 |
|  7521 | WARD   |  1625.00 |         2811.25 |
|  7566 | JONES  |  3718.75 |         6433.43 |
|  7654 | MARTIN |  1625.00 |         2811.25 |
|  7698 | BLAKE  |  1625.00 |         2811.25 |
|  7782 | CLARK  |  3062.50 |         5298.13 |
|  7788 | SCOTT  |  3750.00 |         6487.50 |
|  7839 | KING   |  6250.00 |        10812.50 |
|  7844 | TURNER |  1625.00 |         2811.25 |
|  7876 | ADAMS  |  1375.00 |         2378.75 |
|  7900 | JAMES  |  1625.00 |         2811.25 |
|  7902 | FORD   |  3750.00 |         6487.50 |
|  7934 | MILLER |  1625.00 |         2811.25 |
|   102 | Ben    | 10000.00 |        24300.00 |
|   105 | Alex   | 12500.00 |        30375.00 |
+-------+--------+----------+-----------------+
16 rows in set (0.13 sec)
```

Q2. Write trigger

1. Write a tigger to store the old salary details in Emp _Back (Emp _Back has the

same structure as emp table without any

constraint) table.

(note :create emp_back table before writing trigger)

----- to create emp_back table

create table emp_back(

empno int,

ename varchar(20),

oldsal decimal(9,2),

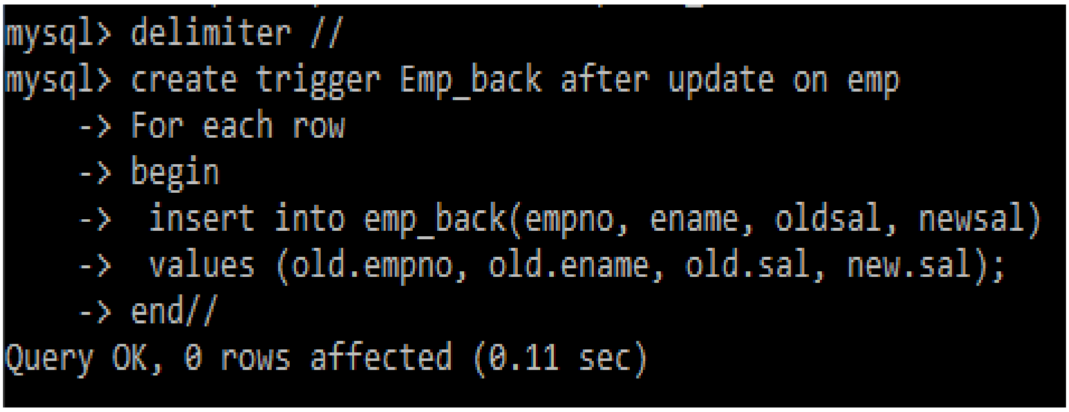
newsal decimal(9,2)

)

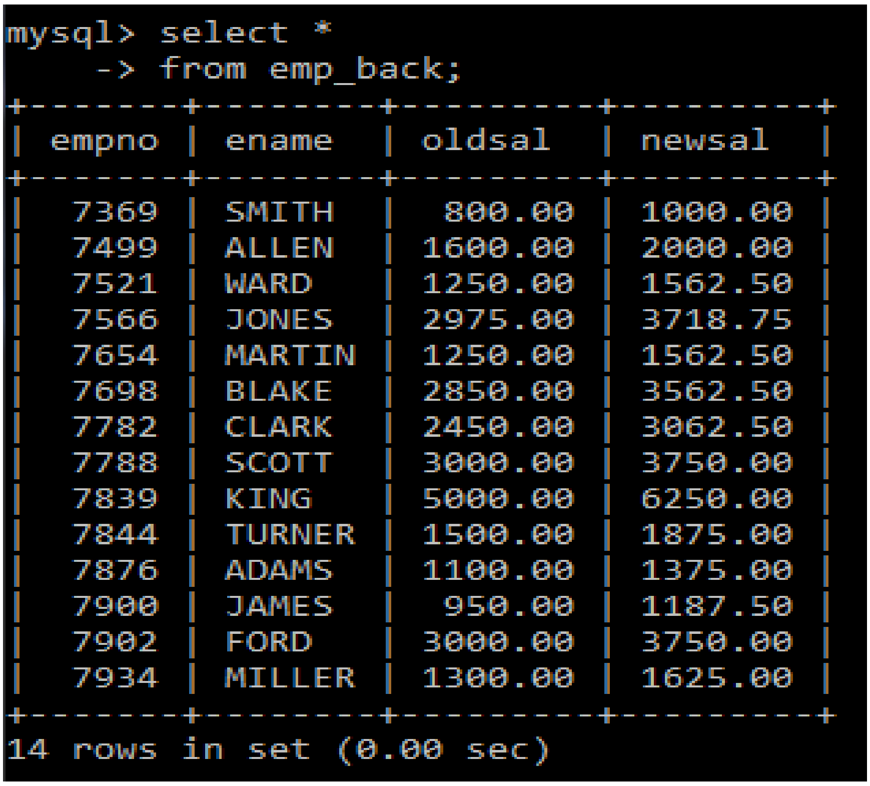
(note :

execute procedure written in Q8 and

check the entries in EMP_back table after execution of the procedure)



```
mysql> delimiter //
mysql> create trigger Emp_back after update on emp
    -> For each row
    -> begin
    ->  insert into emp_back(empno, ename, oldsal, newsal)
    ->  values (old.empno, old.ename, old.sal, new.sal);
    -> end//
Query OK, 0 rows affected (0.11 sec)
```



```
mysql> select *
    -> from emp_back;
+-------+--------+---------+---------+
| empno | ename  | oldsal  | newsal  |
+-------+--------+---------+---------+
|  7369 | SMITH  |  800.00 | 1000.00 |
|  7499 | ALLEN  | 1600.00 | 2000.00 |
|  7521 | WARD   | 1250.00 | 1562.50 |
|  7566 | JONES  | 2975.00 | 3718.75 |
|  7654 | MARTIN | 1250.00 | 1562.50 |
|  7698 | BLAKE  | 2850.00 | 3562.50 |
|  7782 | CLARK  | 2450.00 | 3062.50 |
|  7788 | SCOTT  | 3000.00 | 3750.00 |
|  7839 | KING   | 5000.00 | 6250.00 |
|  7844 | TURNER | 1500.00 | 1875.00 |
|  7876 | ADAMS  | 1100.00 | 1375.00 |
|  7900 | JAMES  |  950.00 | 1187.50 |
|  7902 | FORD   | 3000.00 | 3750.00 |
|  7934 | MILLER | 1300.00 | 1625.00 |
+-------+--------+---------+---------+
14 rows in set (0.00 sec)
```

2. Write a trigger which add entry in audit table when user tries to insert or delete records in employee table store empno,name,username and date on which operation performed and which action is done insert or delete. in emp_audit table.

create table before writing trigger.

create table empaudit(

empno int;

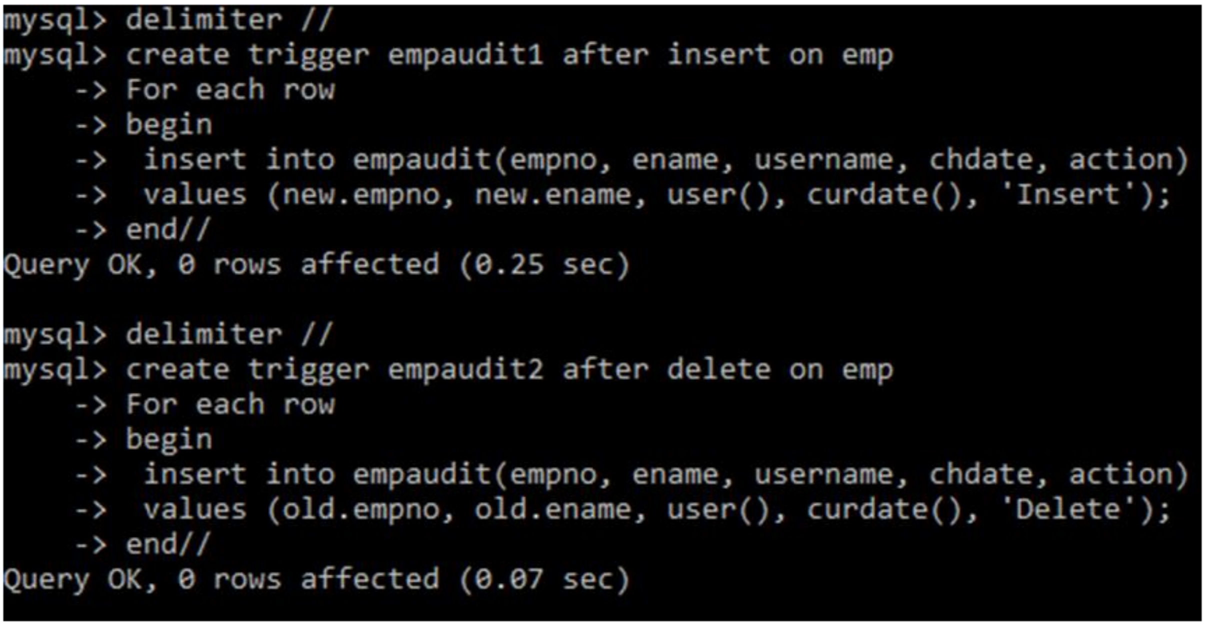
ename varchar(20),

username varchar(20);

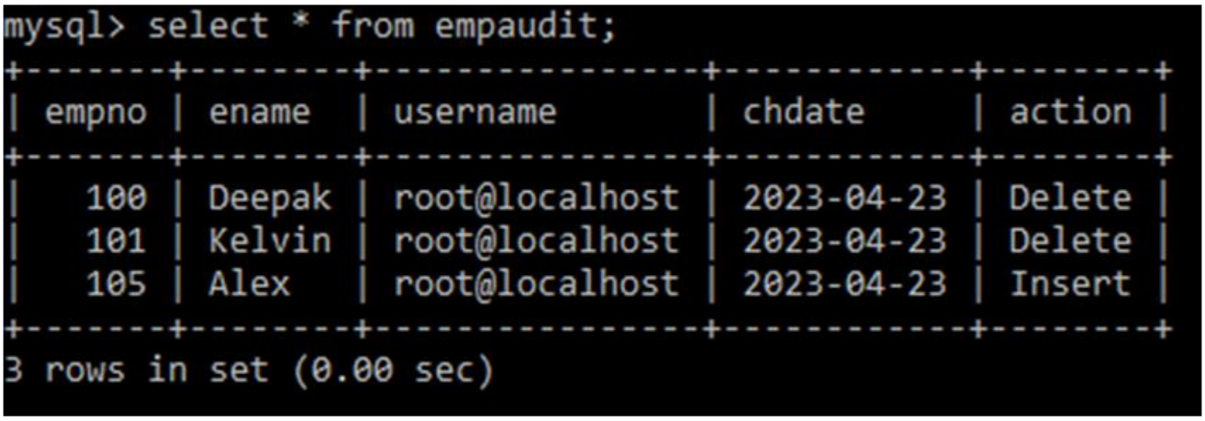
chdate date;

action varchar(20)

);



```
mysql> delimiter //
mysql> create trigger empaudit1 after insert on emp
    -> For each row
    -> begin
    ->  insert into empaudit(empno, ename, username, chdate, action)
    ->  values (new.empno, new.ename, user(), curdate(), 'Insert');
    -> end//
Query OK, 0 rows affected (0.25 sec)

mysql> delimiter //
mysql> create trigger empaudit2 after delete on emp
    -> For each row
    -> begin
    ->  insert into empaudit(empno, ename, username, chdate, action)
    ->  values (old.empno, old.ename, user(), curdate(), 'Delete');
    -> end//
Query OK, 0 rows affected (0.07 sec)
```



```
mysql> select * from empaudit;
+-------+--------+----------------+------------+--------+
| empno | ename  | username       | chdate     | action |
+-------+--------+----------------+------------+--------+
|   100 | Deepak | root@localhost | 2023-04-23 | Delete |
|   101 | Kelvin | root@localhost | 2023-04-23 | Delete |
|   105 | Alex   | root@localhost | 2023-04-23 | Insert |
+-------+--------+----------------+------------+--------+
3 rows in set (0.00 sec)
```

3. Create table vehicle_history. Write a trigger to store old vehicle price and new vehicle price in history table before you update price in vehicle table

(note: use vehicle table).

create table vehicle_history(

vno int,

vname varchar(20),

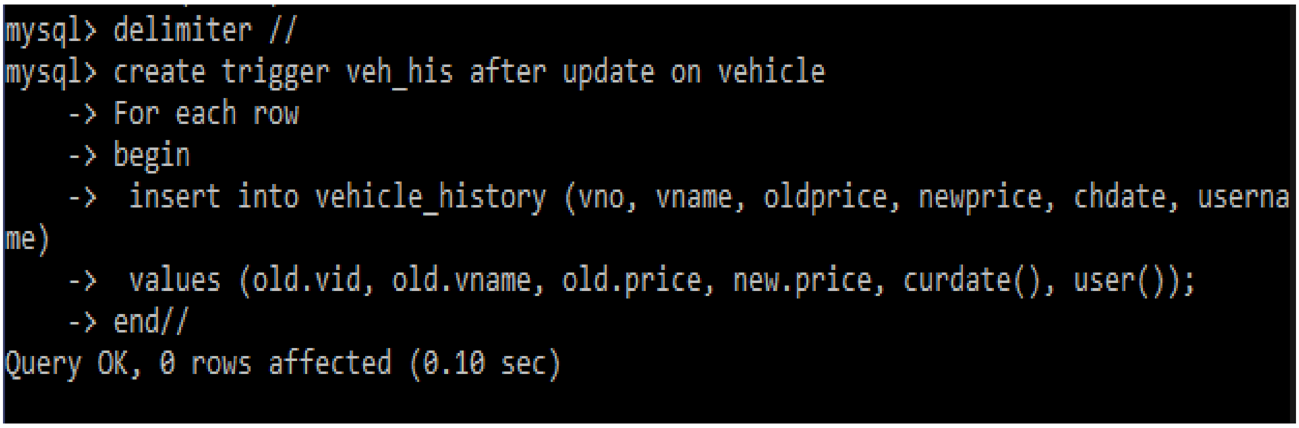
oldprice decimal(9,2),

newprice decimal(9,2),

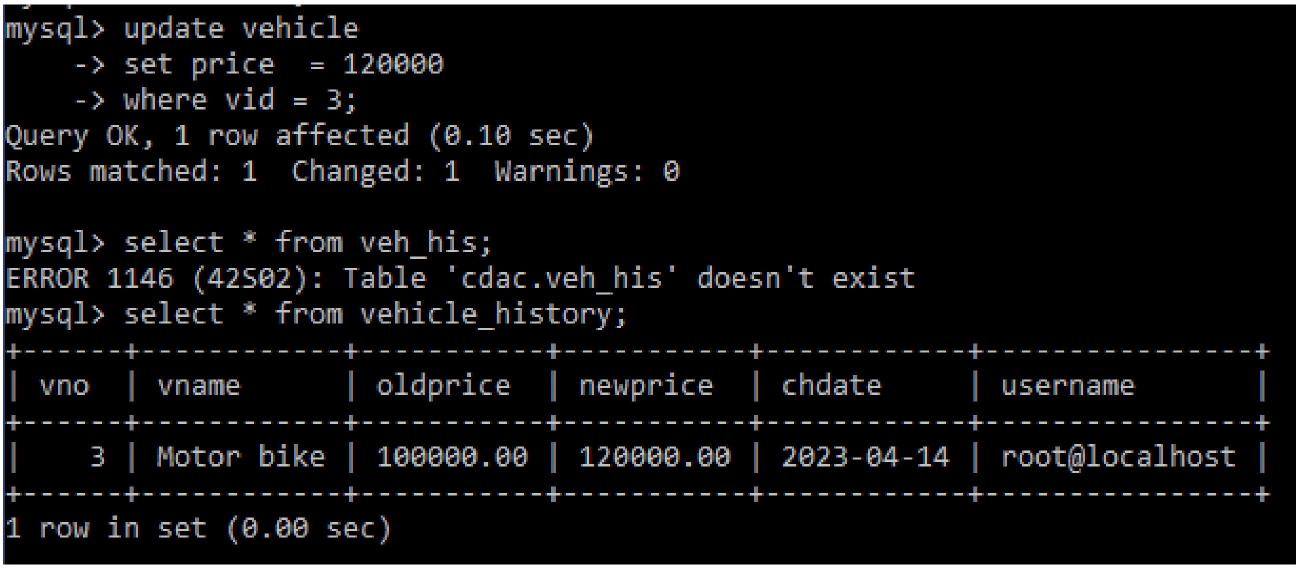
chdate date,

username varchar(20)

);



```
mysql> delimiter //
mysql> create trigger veh_his after update on vehicle
    -> For each row
    -> begin
    ->  insert into vehicle_history (vno, vname, oldprice, newprice, chdate, userna
me)
    ->  values (old.vid, old.vname, old.price, new.price, curdate(), user());
    -> end//
Query OK, 0 rows affected (0.10 sec)
```



```
mysql> update vehicle
    -> set price  = 120000
    -> where vid = 3;
Query OK, 1 row affected (0.10 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from veh_his;
ERROR 1146 (42S02): Table 'cdac.veh_his' doesn't exist
mysql> select * from vehicle_history;
+------+------------+-----------+-----------+------------+----------------+
| vno  | vname      | oldprice  | newprice  | chdate     | username       |
+------+------------+-----------+-----------+------------+----------------+
|    3 | Motor bike | 100000.00 | 120000.00 | 2023-04-14 | root@localhost |
+------+------------+-----------+-----------+------------+----------------+
1 row in set (0.00 sec)
```