```sql
PLSQL

Delimiter //

Create procedure myproc()

Begin

Declare vempno,vdeptno int;

Declare vename,vjob, vstatus varchar(20);

Declare vsal,vavg decimal(9,2);

Declare vset int default 0;

Declare empcur cursor for select empno,ename,job,sal,deptno from emp;

Declare continue handler for not found set vset=1;


Open empcur;

label1:loop

Fetch empcur into vempno,vename,vjob,vsal,vdeptno;

If vset=1 then

    Leave label1;

End if;

Select avg(sal) into vavg

From emp

Where deptno=vdeptno;

If vsal > vavg then

    Set vstatus ='greater';

Elseif vsal < vavg then

        Set vstatus ='lesser';

Else

     Set vstatus ='equal';

End if;

Select vempno,vename,vsal,vjob,vdeptno,vavg,vstatus;

End loop;

Close empcur;

End//
```

. 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

```
Create procedure updatesal()

Begin

Declare vempno,vdeptno int;

Declare vename varchar(20);

Declare vhiredate date;

Declare vsal decimal(9,2);

Declare vset int default 0;

Declare vexp int;

Declar vsal_update decimal(9,2)

Declare empcur cursor for select empno,ename,hiredate,sal,deptno from emp;

Declare continue handler for not found set vset=1;

Open empcur;

Label1:loop

  Fetch empcur into vempno,vename,vhiredate,vsal,vdeptno;

  If vset=1 then

     Leave label1;

   End if;

  Set vexp=floor(datediff(curdate(),vhiredate))/365;

  If vexp<= 35 then

     Select 'no update';

  Elseif vexp <=38 then

     Update emp

     Set sal=vsal+0.20*vsal

     Where empno= vempno;

Else

     Update emp

     Set sal=vsal+0.25*vsal

     Where empno= vempno;

End if;
```

```
Select vempno,vename,vdeptno,vsal;

End loop;

Close emp;

End//


Call updatesal()


3. update vehicle price

<10000 increase by 10%

>=10000 < 200000 20%

Otherwise 30%


Create procedure updateprice()

Begin

Declare vvid int

Declare vvname varchar(20);

Declare vprice,vnewprice decimal(9,2);

Declare vehiclecur cursor for select vid,vname,price from vehicle;

Declare continue handler for not found set vset=1;

Open vehiclecur;

Label1:loop

   Fetch vehiclecur into vvid,vvname,vprice;

   If vset=1 then

      Leave lable1;

   End if;

   If vprice<100000 then

      Set vnewprice=vprice+0.10* vprice;

Elseif  vprice<200000 then

      Set vnewprice=vprice+0.20*vprice

Else

      Set vnewprice=vprice+0.30*vprice

End if;
```

Update vehicle

Set price=vnewprice

Where vid=vvid;

End loop;

Close vehiclecur;

End //

4. display all vehicle with price >buying_price and sold by salesman who stays In pune

Select v.vname,v.price,cv.buying_price,s.sname,s.address

From vehicle v,cust_vehicle cv, salesman s

Where v.vid=cv.vid and cv.sid=s.sid and s.adress='Pune' and v.price>cv.buying_price;

Create procedure displayvehicles()

Begin

 Declare vvname, vsname, vaddress varchar(30);

Declare vpr,vbpr decimal(9,2);

Declare vehiclecur cursor for Select v.vname,v.price,cv.buying_price,s.sname,s.address

From vehicle v,cust_vehicle cv, salesman s

Where v.vid=cv.vid and cv.sid=s.sid and s.adress='Pune' and v.price>cv.buying_price;

Declare continue handler for not found set vset=1;

Open vehiclecur;

Label1:loop

    Fetch vehiclecur into vvname,vpr,vbpr,vsname,vaddress;

    If vset=1 then

       Leave lable1;

    End if;

    Select vvname,vpr,vbpr,vsname,vaddress;

End loop;

Close vehiclecur;

End//

Functions in mysql

Create function <name of function>(parameters) returns <type>

Begin

Function definition;

End//

Before defining function in mysql

SET GLOBAL log_bin_trust_function_creators = 1;

1. Write a function to find number of years of experience
   Create function calcexp(hdate date) returns int
   Begin
      Declare vexp int;
      Set vexp=floor(datediff(curdate(),hdate)/365);
      Return vexp;
   End//

Select upper(ename),hiredate, Calcexp(hiredate)

From emp;

To find list of all procedures in test database

show procedure status where db="test";

To find code of a particular procedure

show create procedure myproc;

write a function generate email

email address for each customer is 1 st 3 characters of ename , last 2 characters of job followed by @mycompany.com

create function generate_email(pnm varchar(20),pjob varchar(20)) returns varchar(20)

begin

    declare vemail varchar(20);

    set vemail=concat(substr(ename,1,3),substr(job,length(job)-1,2),"@mycompany.com");

    return vemail;

end//

procedure can give data as o/p using out parameter

procedures	functions
procedure can give data as o/p using out parameter, and it can use any number of out parameters	Function can return only one parameter as o/p
Procedure can be called using call statement	Functions can be called in select statements or can be called inside a procedure

Create procedure updatesal()

Begin

Declare vempno,vdeptno int;

Declare vename varchar(20);

Declare vhiredate date;

Declare vsal decimal(9,2);

Declare vset int default 0;

Declare vexp int;

Declar vsal_update decimal(9,2)

Declare empcur cursor for select empno,ename,hiredate,sal,deptno from emp;

Declare continue handler for not found set vset=1;

Open empcur;

Label1:loop

  Fetch empcur into vempno,vename,vhiredate,vsal,vdeptno;

  If vset=1 then

    Leave label1;

  End if;

  Set vexp=calcexp(vhiredate);

  If vexp<= 35 then

    Select 'no update';

 Elseif vexp <=38 then

    Update emp

    Set sal=vsal+0.20*vsal

    Where empno= vempno;

Else

   Update emp

   Set sal=vsal+0.25*vsal

   Where empno= vempno;

End if;

Select vempno,vename,vdeptno,vsal;

End loop;

Close emp;

End//

5. write a function to calculate netsalary

Netsal=sal+0.10*sal+0.15*sal-0.08*sal+comm

Create function calculatenetsal(psal decimal(9,2),pcomm decimal(9,2)) returns decimal(9,2)

Begin

   Return psal+psal*0.10+psal*0.15-psal*0.08+ifnull(pcomm,0);

End//

Exception handling

```
1 DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SELECT 'Error occured';
2 DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET IsError=1;
3 DECLARE EXIT HANDLER FOR SQLEXCEPTION SET IsError=1;
4 DECLARE EXIT HANDLER FOR SQLSTATE '1418' SET IsError = 1;
   create procedure ins_faculty(pfid int,pfname varchar(20),pexp int)

   begin

      declare continue handler for 1062 select "duplicate entry pls check";

      declare continue handler for SQLEXCEPTION select "error occured";

      insert into faculty values(pfid,pfname,pexp);

    select * from faculty;

   end//
```