

1. Write a procedure to update job for given employee and also display count for the given job

delimiter \$\$

```
create procedure empno_job(in peno int,in pjob varchar(20))
```

```
begin
```

```
-- update job in emp table
```

```
update emp
```

```
set job=pjob
```

```
where empno=peno;
```

```
-- display count from emp table
```

```
select count(*)
```

```
from emp
```

```
where job=pjob;
```

```
end$$
```

```
delimiter ;
```

2. Write a procedure to print the table of given number ex:5

5*1=5

5*2 =10

.

.

5*10=50

delimiter \$\$

```
create procedure displaytable(in pno int)
```

```
begin
```

```
declare x int default 1;
```

```
declare vno int;
```

```
while x<=10 do
```

```
set vno=pno*x;
```

```
select concat(pno,'*',x,'=',vno);
```

```
set x=x+1;
```

```
end while;
```

```
end$$
```

```
delimiter ;
```

```
call displaytable(in pno int);
```

3. Write procedure to display triangle

*
**

To delete procedure

Drop procedure <procedure name>

Functions in mysql

1. Functions always returns one value
2. To call a function use select statement

Delimiter \$\$

To allow writing functions to user

SET GLOBAL log_bin_trust_function_creators = 1;

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

Begin

Statements

End\$\$

Delimiter ;

Example

Write a function to calculate salary using following formula

$\text{Netsal} = \text{sal} + \text{ifnull}(\text{comm}, 0) + 0.10 * \text{sal} + 0.15 * \text{sal} - 0.06 * \text{sal}$

delimiter \$\$

create (psal decimal(9,2), pcomm decimal(9,2)) returns decimal(10,2)

begin

declare netsal decimal(10,2) default 0;

set netsal=psal+ifnull(pcomm,0)+0.10*psal+0.15*psal-0.06*psal;

return netsal;

end\$\$

delimiter ;

To call function

```
select empno,ename,sal,comm,calcsal(sal,comm)
```

```
-> from emp;
```

2. write a function generate_email to generate email for every employee using following rules

First 3 characters of ename, '.', first three characters of job, '@mycompany.com'

```
delimiter $$
```

```
create function generate_email(fename varchar(20),fjob varchar(20)) returns varchar(40)
```

```
begin
```

```
declare email varchar(40) default '';
```

```
set email=concat(substr(fename,1,3),'.',substr(fjob,1,3),'@mycompany.com');
```

```
return email;
```

```
end$$
```

```
delimiter ;
```

Using cursor in mysql

Smith----CLERK , JONES-----manager,blake-----analyst

In mysql if you want to read resultset of any select statement row by row and perform some tasks then use cursor

The cursors in mysql are

1. Read only ---- one cannot change the values inside the cursor
2. Non_scrollable --- can be read only once in forward direction, at a time only one row can be read, and can be moved to next row, cannot jump to nth row
3. Asensitive--- if anyone changes the data in the actual database then your cursor will not get affected

Step by step to use cursor

1. Declare cursor
Declare empcur cursor for select empno,ename from emp;
2. Declare NOT FOUND exception handler
Declare continue handler for NOT FOUND set v_stop=1;
3. Open the cursor
Open empcur;
4. Read a row
Fetch empcur into v_empno,v_ename;

5. Check the value of v_stop =1
 Stop the loop
 If v_stop=1
 Leave <loop lable>
 End if;
6. Write code for processing data and go to step 4
7. Once the loop is over close the cursor
 Close <cursor name>

```

delimiter $$
create procedure generate_user_list()
begin
declare str varchar(500) default '';
declare v_stop int default 0;
declare v_ename varchar(20);
declare v_job varchar(20);
declare empcur cursor for select ename,job from emp;
declare continue handler for NOT FOUND set v_stop=1;
open empcur; -- this will populate data in the cursor
label1: loop
    fetch empcur into v_ename,v_job;
    if v_stop=1 then
        leave label1;
    end if;
    set str=concat(str,v_ename,'-----',v_job,', ');
end loop;
select str;
close empcur;
end$$
delimiter ;

```

3. Write a procedure to update salary of managers by 10%, analyst by 20%,and clerk by 30%

```

delimiter $$
create procedure update_sal_jobwise()
begin
declare v_empno int;
declare v_ename varchar(20);
declare v_sal decimal(9,2);
declare v_job varchar(20);
declare v_stop int default 0;
-- declare cursor
declare empcur cursor for select empno,ename,sal,job from emp;
-- declare NOT FOUND continue handler
declare continue handler for NOT FOUND set v_stop=1;
-- open cursor
open empcur;

```

```

-- fetch data
label1: loop
    fetch empcur into v_empno,v_ename,v_sal,v_job;
-- check for end of cursor
    if v_stop=1 then
        leave label1;
    end if;
-- update sal
    if v_job='MANAGER' then

        update emp
        set sal=v_sal+0.10*v_sal
        where empno=v_empno;
    elseif v_job='ANALYST' then
        update emp
        set sal=v_sal+0.20*v_sal
        where empno=v_empno;
    elseif v_job='CLERK' then
        update emp
        set sal=v_sal+0.30*v_sal
        where empno=v_empno;
    end if;

end loop;
-- close the cursor
close empcur;

end$$
delimiter ;

```

4. Write a procedure to update vehicle prices by
 - 10% if registration city is Pune
 - 15% if registration city is Nashik
 - 20% if registration city is Mumbai
 - 5% otherwise
 Also update registration fees by
 - 1% if city Pune
 - 2% for 'Mumbai'
 - 1.5% otherwise

```

delimiter $$
create procedure update_vehicle()
begin
    declare v_vno int;
    declare v_vname varchar(20);
    declare v_price decimal(9,2);

```

```

declare v_city varchar(20);
declare v_regfee decimal(9,2);
declare v_stop int default 0;
-- declare cursor
declare vehcur cursor for select * from vehicle;

-- declare NOT FOUND continue handler
declare continue handler for NOT FOUND set v_stop=1;
-- open cursor
open vehcur;
-- fetch data
label1: loop
    fetch vehcur into v_vno,v_vname,v_price,v_city,v_regfee;
-- check for end of cursor
    if v_stop=1 then
        leave label1;
    end if;
-- update sal
    if v_city='PUNE' then
        update vehicle
        set price=1.10*v_price
        where vno=v_vno;
    elseif v_city='Mumbai' then
        update vehicle
        set price=1.20*v_price
        where vno=v_vno;
    elseif v_city='Nashik' then
        update vehicle
        set price=1.15*v_price
        where vno=v_vno;
    else
        update vehicle
        set price=1.05*v_price
        where vno=v_vno;
    end if;

end loop;
-- close the cursor
close vehcur;
set v_stop=0;
open vehcur;
-- fetch data
label2: loop
    fetch vehcur into v_vno,v_vname,v_price,v_city,v_regfee;
-- check for end of cursor
    if v_stop=1 then
        leave label2;
    end if;

```

```

-- update regfee
if v_city='PUNE' then
    update vehicle
    set regfee=v_price*0.01
    where vno=v_vno;
elseif v_city='Mumbai' then
    update vehicle
    set regfee=v_price*0.02
    where vno=v_vno;
else
    update vehicle
    set regfee=0.015*v_price
    where vno=v_vno;
end if;

end loop;
-- close the cursor
close vehcur;

```

```

end$$
delimiter ;

```

```

----- 2 nd approach
delimiter $$
create procedure update_vehicle()
begin
    declare v_vno int;
    declare v_vname varchar(20);
    declare v_price decimal(9,2);
    declare v_city varchar(20);
    declare v_regfee decimal(9,2);
    declare v_stop int default 0;
-- declare cursor
    declare vehcur cursor for select * from vehicle;

-- declare NOT FOUND continue handler
    declare continue handler for NOT FOUND set v_stop=1;
-- open cursor
    open vehcur;
-- fetch data
    label1: loop
        fetch vehcur into v_vno,v_vname,v_price,v_city,v_regfee;
-- check for end of cursor
        if v_stop=1 then
            leave label1;

```

```

        end if;
-- update price, regfee
if v_city='PUNE' then
    v_price=v_price*1.10
    v_regfee=v_price*0.01
elseif v_city='Mumbai' then
    v_price=v_price*1.20
    v_regfee=v_price*0.02
elseif v_city='Nashik' then
    v_price=v_price*1.15
    v_regfee=v_price*0.015
else
    v_price=v_price*1.05
    v_regfee=v_price*0.015
end if;
    update vehicle
    set price=v_price,regfee=v_regfee
    where vno=v_vno;
end loop;
-- close the cursor
close vehcur;
end$$
delimiter ;

```

Triggers in mysql

--- are procedures which gets executed automatically when any user performs any DML operations

The trigger will get executed either before or after DML statement execution

Usually the data gathered by triggers can be used for auditing or data analysis purpose

Create table emp_audit(

Empno int,

Ename varchar(20)

Old_sal decimal(9,2),

New_sal decimal(9,2),

Changedate date,

Uname varchar(30));