1.CREATE OR REPLACE PROCEDURE employer\_details

IS

CURSOR emp\_cur IS

SELECT first\_name, last\_name, esal FROM emp167;

emp\_rec emp\_cur%rowtype;

BEGIN

FOR emp\_rec in emp\_cur

LOOP

dbms\_output.put\_line(emp\_rec.first\_name || ' ' ||emp\_rec.last\_name

|| ' ' ||emp\_rec.esal);

END LOOP;

END;

2. CREATE OR REPLACE PROCEDURE emp\_salary\_increase1

(emp\_id IN emp16.eno%type)

IS

tmp\_sal number;

BEGIN

SELECT esal

INTO tmp\_sal

FROM emp16

WHERE eno = emp\_id;

dbms\_output.put\_line('the salary is :' || tmp\_sal);

END;

/

3.CREATE OR REPLACE PROCEDURE employee\_sal (id IN NUMBER, esalary OUT NUMBER)

IS

BEGIN

SELECT esal INTO esalary

FROM emp166 WHERE eno = id;

END;

##for executing the above::

DECLARE

esalary number;

CURSOR id\_cur is SELECT eno FROM emp166;

BEGIN

FOR emp\_rec in id\_cur

LOOP

employee\_sal(emp\_rec.eno, esalary);

dbms\_output.put\_line('The employee ' || emp\_rec.eno || ' has salary ' || esalary);

END LOOP;

END;

2. CREATE OR REPLACE PROCEDURE emp\_salary\_increase

(emp\_id IN emp16.eno%type)

IS

tmp\_sal number;

BEGIN

SELECT esal

INTO tmp\_sal

FROM emp16

WHERE eno = emp\_id;

IF tmp\_sal between 10000 and 20000 THEN

dbms\_output.put\_line('incremented salary is :' || tmp\_sal \* 1.2);

ELSIF tmp\_sal between 20000 and 30000 THEN

dbms\_output.put\_line('incremented salary is : ' || tmp\_sal \* 1.3);

ELSIF tmp\_sal > 30000 THEN

dbms\_output.put\_line('incremented salary is ' || tmp\_sal \* 1.4);

END IF;

END;

3. CREATE OR REPLACE PROCEDURE emp\_salary\_increase

(emp\_id IN emp16.eno%type, salary\_inc IN OUT emp16.esal%type)

IS

tmp\_sal number;

BEGIN

SELECT esal

INTO tmp\_sal

FROM emp16

WHERE eno = emp\_id;

IF tmp\_sal between 10000 and 20000 THEN

salary\_inc := tmp\_sal \* 1.2;

ELSIF tmp\_sal between 20000 and 30000 THEN

salary\_inc := tmp\_sal \* 1.3;

ELSIF tmp\_sal > 30000 THEN

salary\_inc := tmp\_sal \* 1.4;

END IF;

dbms\_output.put\_line('the incremented salary is :' || salary\_inc);

END;

##for executing this above :

DECLARE

CURSOR updated\_sal is

SELECT eno,esal

FROM emp16;

pre\_sal number;

BEGIN

FOR emp\_rec IN updated\_sal

LOOP

pre\_sal := emp\_rec.esal;

emp\_salary\_increase(emp\_rec.eno, emp\_rec.esal);

dbms\_output.put\_line('The salary of ' || emp\_rec.eno ||

' increased from '|| pre\_sal || ' to '||emp\_rec.esal);

END LOOP;

END;

declare

l\_val varchar2(30) := 'hello world';

procedure myproc (p\_val in out varchar2) is

begin

dbms\_output.put\_line('p\_val was ' || p\_val);

p\_val := 'something else';

end;

begin

myproc(l\_val);

dbms\_output.put\_line('l\_val is now ' || l\_val);

end;/

##execute emp\_salary\_increase('e002');

##the function takes only one parameter emp ID and then assigns the grade such a high salary, medium salary ,low salary depending on the employee

Create or replace function empl\_id return number

Is

Salary number;

Begin

Select esal into salary

From emp where eno=’e001’;

Return salary;

End;

Function created.

declare

ss number;

v\_msg varchar2(10);

begin

ss:=empl\_id();

dbms\_output.put\_line('the salary is ' || ss);

case

when ss >=45000 then

v\_msg :='high';

when ss >2000 and ss <=45000 then

v\_msg :='medium';

when ss < 20000 then

v\_msg :='low';

end case;

dbms\_output.put\_line(v\_msg);

end;

/