##create or replace procedure display as

Begin

Dbms\_output.put\_line('Hello World');

End;

/

Procedure created.

OUTPUT:

SQL> Execute display;

Hello World

## create a procedure which shows information about emp table.

create or replace procedure displayemp1

is

cursor emp\_cur is

select eno,ename,esal,eadd from emp;

emp\_rec emp\_cur%rowtype;

begin

for emp\_rec in emp\_cur

loop

dbms\_output.put\_line(emp\_rec.eno ||' '||emp\_rec.ename ||' '||emp\_rec.esal ||' '||emp\_rec.eadd);

end loop;

end;

## create or replace procedure Swap1(a IN OUT number,b IN OUT number) is

c number;

Begin

c:=a;

a:=b;

b:=c;

End;

To execute the same above:

Declare

num1 number;

num2 number;

Begin

num1:=&num1;

num2:=&num2;

Dbms\_output.put\_line ('The numbers before swapping are '||to\_char (num1)||' '||to\_char(num2));

Swap1 (num1, num2);

Dbms\_output.put\_line ('the swapped numbers are '||to\_char (num1)||' '||to\_char(num2));

End;

//

1. ## **Write a PL/SQL block to define procedure to insert data in the Employee table.**

create table Employee1(emp\_id numeric,first\_name varchar2(30),last\_name varchar2(30),salary number,address varchar2(50),D\_O\_J date);

1 create or replace procedure InsertEmp1(id IN OUT number,fname IN OUT varchar2,lname IN OUT varchar2,salary IN OUT number,address IN OUT varchar,join IN OUT date)

2 Begin

3 insert into Employee1 values (id,fname,lname,salary,address,join);

4 end;

5 /

Procedure created

Declare

2 id number;

3 fname varchar2(30);

4 lname varchar2(30);

5 salary number;

6 address varchar2(50);

7 join date;

8 Begin

9 id:=&id;

10 fname:=&fname;

11 lname:=&lname;

12 salary:=&salary;

13 address:=&address;

14 join:=sysdate;

15 insertemp1(id,fname,lname,salary,address,join);

16 end;

17

1.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');

1. **Create a table lecturer. Write a PL/SQL block which Create a stored procedure and call it.**

create table lecturer ( id number, name varchar2(30), faculty varchar2(30), major\_subject varchar2(30), l\_date date )

Table created.

**PL/SQL BLOCK:**

1 create or replace procedure Lect(l\_id IN OUT number,subject OUT varchar2) as

2 begin

3 select major\_subject into subject from lecturer where id=l\_id;

4\* end;

5 /

Procedure created.

SQL> Declare

2 l\_id number;

3 sub varchar2(30);

4 Begin

5 l\_id:=&id;

6 Lect2(l\_id,sub);

7 Dbms\_output.put\_line('major subject '||sub);

8 End;

9 /

**OUTPUT:**

Enter value for id: 1

old 5: l\_id:=&id;

new 5: l\_id:=1;

Hi

major subject C#

PL/SQL procedure successfully completed.

1. **Write a PL\SQL block with number type parameter and insert the values in it.**

create table numbers(n number,username varchar2(30) )  
    
Table created.  
  
SQL> create or replace  
  2   procedure insert\_numbers( p\_num number ) authid definer as  
  3   begin  
  4     insert into numbers values ( p\_num, user );  
  5   end insert\_numbers;  
  6   /  
  
 Procedure created.  
  
 SQL> EXEC insert\_numbers(1);  
  
 PL/SQL procedure successfully completed.  
  
 SQL>  
 SQL> select \* from numbers;

**OUTPUT:**

  N  USERNAME  
 --------  ------------------  
     1  SYS

1. **Write a PL/SQL block with forward declaration of procedure.**

**PL/SQL BLOCK**

1 declare

2 counter number :=6;

3 procedure a(no1 IN OUT number);

4 procedure b(no2 IN OUT number) as

5 begin

6 if(no2 > 0) then

7 no2:=no2-1;

8 Dbms\_output.put\_line('counter= '||no2);

9 a(no2);

10 end if;

11 end;

12 procedure a(no1 IN OUT number) as

13 begin

14 no1:=no1-1;

15 Dbms\_output.put\_line('counter= '||no1);

16 b(no1);

17 end;

18 Begin

19 b(counter);

20\* end;

21 /

**OUTPUT:**

counter= 5

counter= 4

counter= 3

counter= 2

counter= 1

counter= 0

1. **write a plsql block for defining and calling simple function.**

**PL/SQL BLOCK**

CREATE OR REPLACE FUNCTION my\_first\_func (p\_name IN

VARCHAR2 := 'Lewis',

p\_address IN VARCHAR2 := '123 MockingbirdLn',

p\_an\_in\_out\_parameter IN OUT NUMBER,

p\_an\_out\_parameter OUT DATE )

RETURN VARCHAR2

AS

v\_a\_variable VARCHAR2(30);

BEGIN

IF p\_name = 'Lewis'

THEN

RETURN -1;

END IF;

v\_a\_variable := 99;

p\_an\_in\_out\_parameter := v\_a\_variable;

p\_an\_out\_parameter := SYSDATE;

RETURN v\_a\_variable;

END;

/

Function created.

DECLARE

v\_employee VARCHAR2(30) := 'BillyBob';

v\_number NUMBER := 22;

v\_date DATE;

v\_variable VARCHAR2(30);

BEGIN

v\_variable :=

my\_first\_func(

p\_name => v\_employee,

p\_an\_in\_out\_parameter => v\_number,

p\_an\_out\_parameter => v\_date );

DBMS\_OUTPUT.PUT\_LINE(

v\_variable || ', ' ||

v\_employee || ', ' ||

to\_Char(v\_number) || ', ' ||

to\_char(v\_date) );

v\_variable :=

my\_first\_func(

p\_an\_in\_out\_parameter => v\_number,

p\_an\_out\_parameter => v\_date );

DBMS\_OUTPUT.PUT\_LINE(

v\_variable || ', ' ||

v\_employee || ', ' ||

to\_Char(v\_number) || ', ' ||

to\_char(v\_date) );

END;

**OUTPUT:**

99, BillyBob, 99, 09-FEB-11

-1, BillyBob, 99,

Statement processed.

1. **Create a table Employee. Write a PL/SQL block which**

**acts as a Recursive function ‘listing’ which takes two parameters one is of type number & one is of type varchar, to display employee name.**

Creating table:

drop table Employee;

create table Employee

(

Eid number(3),

Ename varchar(15),

Esal number(8,2)

);

Table created.

insert into Employee values(1,'Tejashri',25000);

insert into Employee values(2,'Archana',30000);

insert into Employee values(3,'Arjun',50000);

**PL/SQL BLOCK**

SQL>

SQL> create or replace function listing(empid number)

2 return varchar2

3 as

4 name employee.Ename%Type;

5 total number(3);

6 begin

7 select count(\*) into total from employee

8 if(empid>total)then

9 return name;

10 end if;

11 select Ename into name from Employee where Eid=empid;

12 DBMS\_OUTPUT.PUT\_LINE(listing(empid+1));

13 return name;

14 End;

15 /

Function created

SQL> Declare

2 ename varchar(15)

3 begin

4 ename:=listing(1);

5 DBMS\_OUTPUT.PUT\_LINE(ename);

6 end;

7 /

**OUTPUT:**

Tejashri

Archana

Arjun

1. **Create a table customer. Insert 5 meaningful records in it. Write a function to count number of customers and return value back.**

**CODE:**

SQL> CREATE OR REPLACE TYPE mem\_type IS VARRAY(10)

of VARCHAR2(15)  
  2  /  
  
Type created.  
  
CREATE TABLE cusomer (Name VARCHAR2(10),Address VARCHAR2(20),City VARCHAR2(20), Phone VARCHAR2(8), Members mem\_type)  
  
Table created.  
  
SQL> INSERT INTO customer VALUES ('AL','111 First St.','Mobile',2  '222-2222', mem\_type('Brenda','Richard'));  
  
SQL> INSERT INTO club VALUES ('FL','222 Second St.','Orlando',  
  2  '333-3333', mem\_type('Gen','John','Steph','JJ'));

SQL> INSERT INTO club VALUES ('CL','333 Third St.','Pluto',  
  2  '444-4444', mem\_type('Peand'));

SQL> INSERT INTO club VALUES ('DL','444 Second St.','Miechal',  
  2  '555-5555', mem\_type('Leanda','John','Josef','JJ',’Jacson’));

SQL> INSERT INTO club VALUES ('FL','222 Second St.','Tom',  
  2  '333-3333', mem\_type('XYZ','Aryan'));

**PL/SQL BLOCK:**

SQL> CREATE OR REPLACE PROCEDURE vartest IS  
  2   CURSOR fcur IS  
  3    SELECT city, members FROM club;  
  4   BEGIN  
  5    FOR j IN fcur LOOP  
  6       IF j.members.exists(1) THEN  
  7         dbms\_output.put\_line(j.City||' has '|| j.members.count||' members');  
  8  
  9         END IF;  
 10      END LOOP;   /\* end for j in fcur loop \*/  
 11  END vartest;  
 12  /  
  
Procedure created.

SQL> exec vartest;  
Mobile has 2 members  
Orlando has 4 members

Pluto has 1 members

Miechal has 5 members

Tom has 2 members  
  
PL/SQL procedure successfully completed.

1. **Create table book. Insert 5 meaningful records in it. Write a function to count numbers of books and store the return value to a variable.**

create table Book(B\_id varchar2(8),Author varchar2(30),Edition varchar2(20),Category varchar2(30),Cost number(6,2))

insert into Book values('B001','Forouzen','3rd','DCN','650');

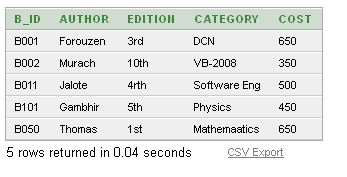
insert into Book values('B002','Murach','10th','VB-2008','350');

insert into Book values('B011','Jalote','4rth','Software Eng','500');

insert into Book values('B101','Gambhir','5th','Physics','450');

insert into Book values('B050','Thomas','1st','Mathemaatics','650');

select \* from Book;



To count the number of books

**PL/SQL CODE:**

SQL> create or replace function Book

2 return number as

3 cnt number(3);

4 Begin

5 select count(\*) into cnt from Book

6 return cnt;

7 End;

8 /

Function created.

**PL/SQL BLOCK:**

SQL> Declare

2 Bcnt numbers(3);

3 Begin

4 Bcnt:=bookcnt();

5 DBMS\_OUTPU.PUT\_LINE('Count of book is :' ||Bcnt);

6 End;

7 /

**OUTPUT:**

Count of books is :5

PL/SQL procedure successfully completed.

1. **write a function and call it in dbms.output.putline.**

**PL/SQL BLOCK:**

SQL> set serveroutput on  
SQL> set echo on  
SQL>  
SQL> CREATE OR REPLACE FUNCTION squareme(thenum number)  
  2       RETURN NUMBER IS  
  3   BEGIN  
  4       RETURN thenum \* thenum;  
  5   END squareme;  
  6   /  
  
Function created.  
  
SQL>  
SQL> BEGIN  
  2       DBMS\_OUTPUT.PUT\_LINE('9 squared is ' || squareme(9) );  
  3   END;  
  4   /

**OUTPUT:**

9 squared is 81  
  
PL/SQL procedure successfully completed.