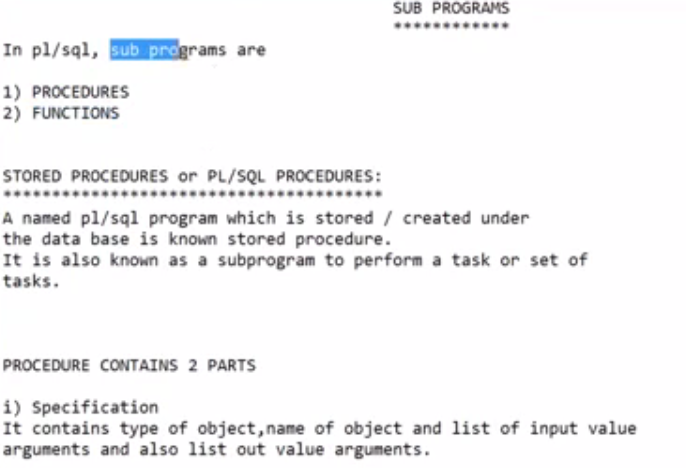
PROCEDURES : IS ALSO A PROG THAT WILL BE STORED IN DB

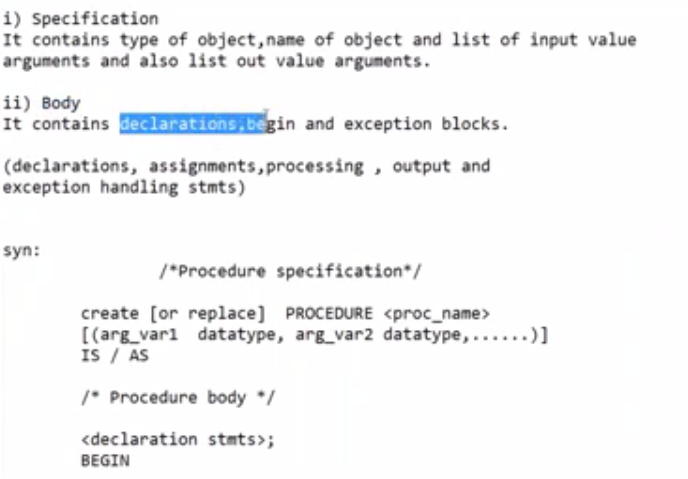
IN PROGRAMS / MEANS, COMPILES THE PROGRAM AND EXECUTES.

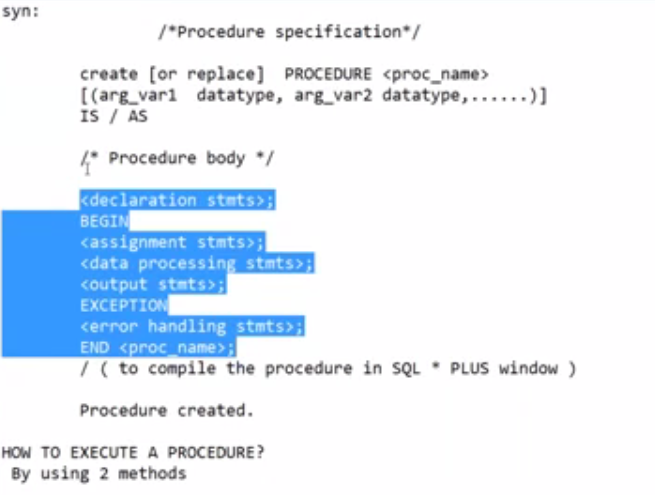
BUT IN PROCEDURE / MEANS JUST COMPILES THE PROCEDURE AND DOES NOT DISPLAY ANY OUTPUT.

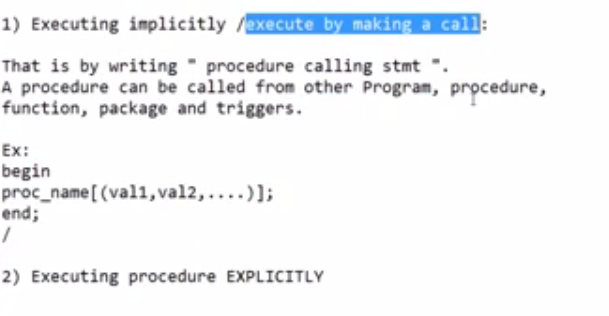
HERE WE GETTING SAME OUTPUT BECAUSE INPUT IS SAME, SO WE GO FOR ARGUMENTS IN PROCEDURE CALLED DYNAMIC.

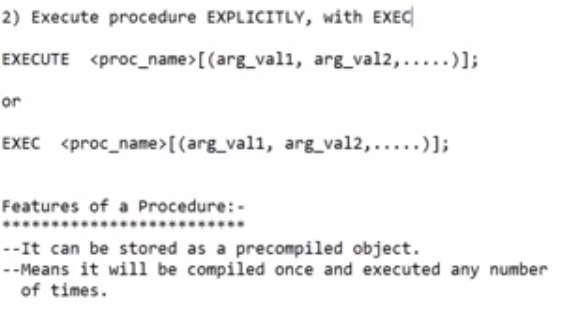
& DOES NOT WORK HERE

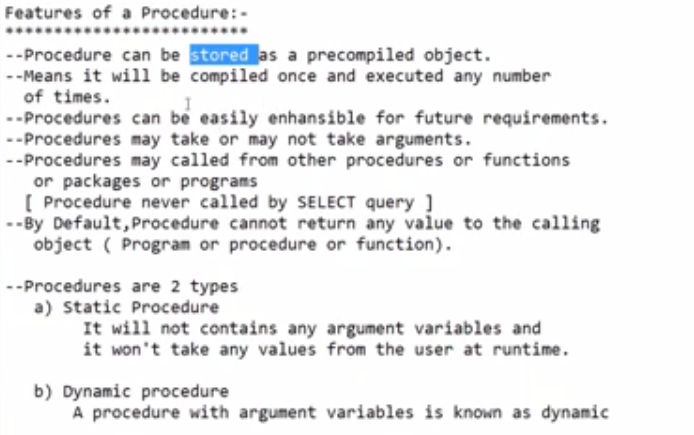


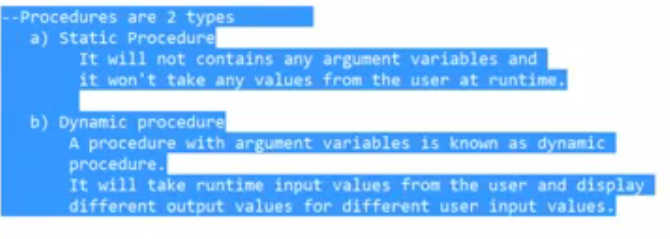




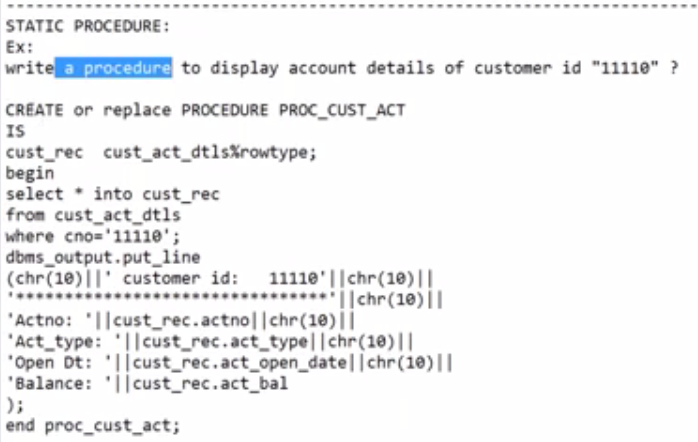


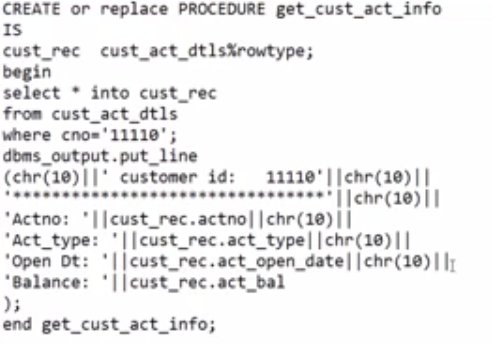


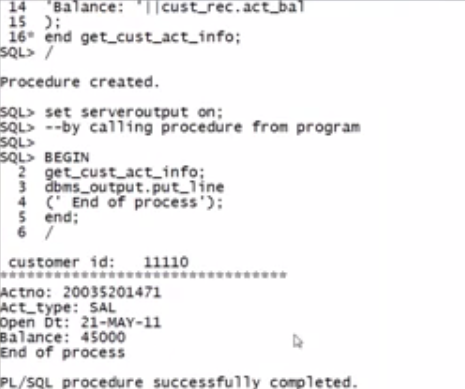


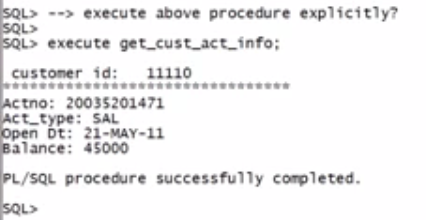






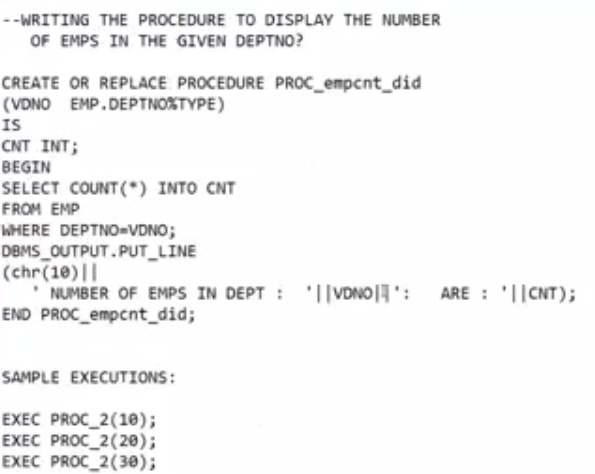


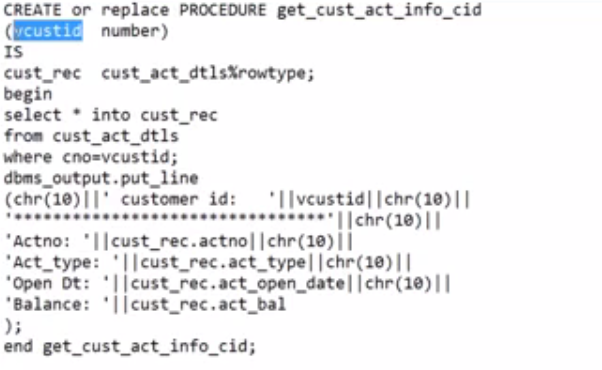


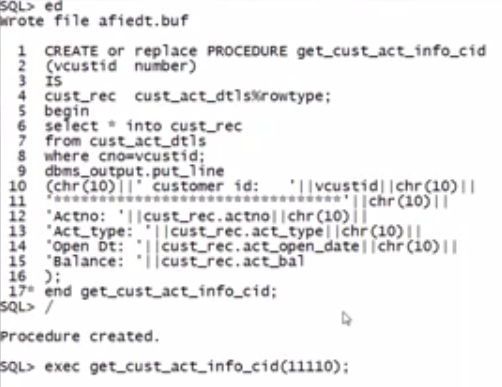


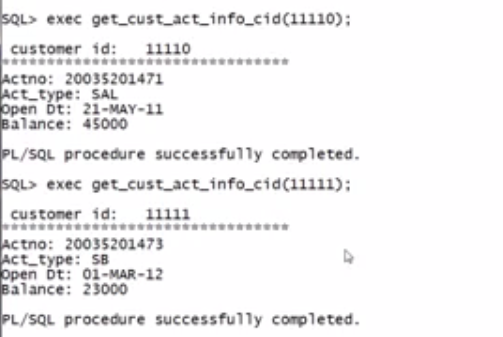
HERE WE GETTING SAME OUTPUT BECAUSE INPUT IS SAME, SO WE GO FOR ARGUMENTS IN PROCEDURE CALLED DYNAMIC.

& DOES NOT WORK HERE



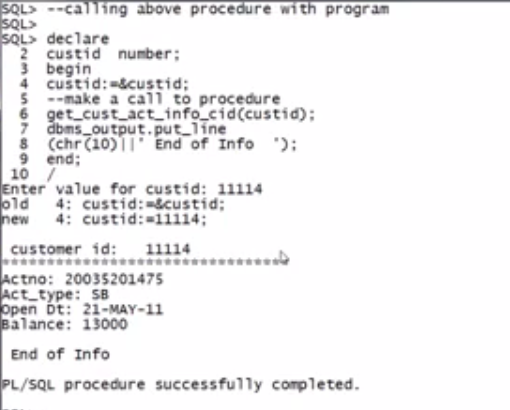






THERE ARE THOUSANDS OF TABLES, ITS NOT POSSIBLE TO REMEMBER ARGUMENTS FOR ALL PROC.

SO WE CAN DO AS FOLLOWING.



ACTUAL ARGUMENT IS VCUSTID BUT WE CAN DECLARE ANY VARIABLE IN PROG. AND THIS ARGUMENT CAN BE MENTIONED IN CALLING PROC. ARGUMENT AS INPUT.

SQL> ED

Wrote file afiedt.buf

1 CREATE PROCEDURE GET\_TOTSALAR\_AVGSALAR(DEPTNAME VARCHAR2)

2 AS

3 VTS NUMBER(10);

4 VAS NUMBER(10);

5 BEGIN

6 SELECT SUM(SAL), AVG(SAL) INTO VTS, VAS FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM DEPT WHERE DNAME=DEPTNAME);

7 DBMS\_OUTPUT.PUT\_LINE('TOTAL SALARY IS : '||VTS);

8 DBMS\_OUTPUT.PUT\_LINE('AVERAGE SALARY IS : '||VAS);

9\* END GET\_TOTSALAR\_AVGSALAR;

SQL> /

Procedure created.

SQL> BEGIN

2 GET\_TOTSALAR\_AVGSALAR('SALES');

3 END;

4 /

PL/SQL procedure successfully completed.

SQL> SET SERVEROUTPUT ON;

SQL> /

TOTAL SALARY IS : 9400

AVERAGE SALARY IS : 1567

PL/SQL procedure successfully completed.

SQL> /

DEPTNO DNAME LOC

---------- -------------- -------------

10 ACCOUNTING NEW YORK

20 RESEARCH DALLAS

30 SALES CHICAGO

40 OPERATIONS BOSTON

SQL> BEGIN

2 GET\_TOTSALAR\_AVGSALAR('SALES');

3 END;

4 /

TOTAL SALARY IS : 9400

AVERAGE SALARY IS : 1567

PL/SQL procedure successfully completed.

SQL> ED

Wrote file afiedt.buf

1 BEGIN

2 GET\_TOTSALAR\_AVGSALAR('ACCOUNTING');

3\* END;

SQL> /

TOTAL SALARY IS : 8750

AVERAGE SALARY IS : 2917

PL/SQL procedure successfully completed.

SQL> ED

Wrote file afiedt.buf

1 BEGIN

2 GET\_TOTSALAR\_AVGSALAR('RESEARCH');

3\* END;

SQL> /

TOTAL SALARY IS : 10875

AVERAGE SALARY IS : 2175

PL/SQL procedure successfully completed.

SQL> ED

Wrote file afiedt.buf

1 BEGIN

2 GET\_TOTSALAR\_AVGSALAR('ACCOUNTING');

3\* END;

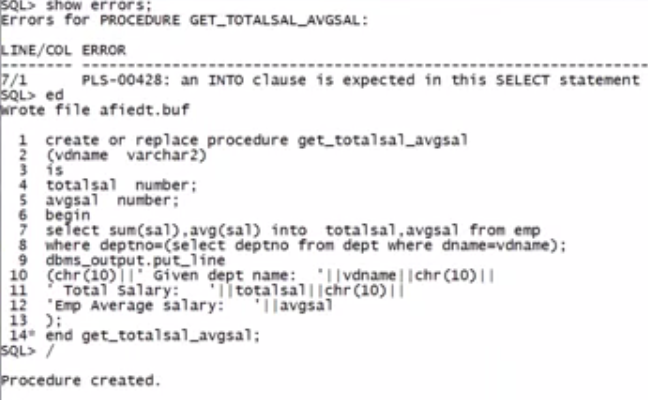
SQL> /

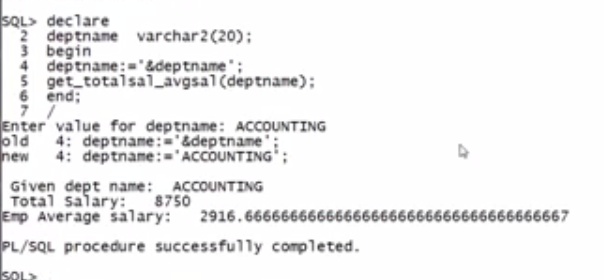
TOTAL SALARY IS : 8750

AVERAGE SALARY IS : 2917

PL/SQL procedure successfully completed.

SQL>







SUB PROGRAMS

\*\*\*\*\*\*\*\*\*\*\*\*

In pl/sql, sub programs are

1) PROCEDURES

2) FUNCTIONS

STORED PROCEDURES or PL/SQL PROCEDURES:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

A named pl/sql program which is stored / created under

the data base is known stored procedure.

It is also known as a subprogram to perform a task or set of

tasks.

PROCEDURE CONTAINS 2 PARTS

i) Specification

It contains type of object,name of object and list of input value

arguments and also list out value arguments.

ii) Body

It contains declarations,begin and exception blocks.

(declarations, assignments,processing , output and

exception handling stmts)

syn:

/\*Procedure specification\*/

create [or replace] PROCEDURE <proc\_name>

[(arg\_var1 datatype, arg\_var2 datatype,......)]

IS / AS

/\* Procedure body \*/

<declaration stmts>;

BEGIN

<assignment stmts>;

<data processing stmts>;

<output stmts>;

EXCEPTION

<error handling stmts>;

END <proc\_name>;

/ ( to compile the procedure in SQL \* PLUS window )

Procedure created.

HOW TO EXECUTE A PROCEDURE?

By using 2 methods

1) Execute procedure by making a call:

That is by writing " procedure calling stmt ".

A procedure can be called from other Program, procedure,

function, package and triggers.

Ex:

begin

proc\_name[(val1,val2,....)];

end;

/

2) Execute procedure EXPLICITLY, with EXECUTE command.

EXECUTE <proc\_name>[(arg\_val1, arg\_val2,.....)];

or

EXEC <proc\_name>[(arg\_val1, arg\_val2,.....)];

Features of a Procedure:-

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--Procedure can be stored as a precompiled object.

--Means it will be compiled once and executed any number

of times.

--Procedures can be easily enhansible for future requirements.

--Procedures may take or may not take arguments.

--Procedures may called from other procedures or functions

or packages or programs

[ Procedure never called by SELECT query ]

--By Default,Procedure cannot return any value to the calling

object ( Program or procedure or function).

-----------------------------------------------------------------------------------------

Ex:

write a procedure to display account details of customer id "11110" ?

CREATE or replace PROCEDURE PROC\_CUST\_ACT

IS

cust\_rec cust\_act\_dtls%rowtype;

begin

select \* into cust\_rec

from cust\_act\_dtls

where cno='11110';

dbms\_output.put\_line

(chr(10)||' customer id: 11110'||chr(10)||

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'||chr(10)||

'Actno: '||cust\_rec.actno||chr(10)||

'Act\_type: '||cust\_rec.act\_type||chr(10)||

'Open Dt: '||cust\_rec.act\_open\_date||chr(10)||

'Balance: '||cust\_rec.act\_bal

);

end proc\_cust\_act;

Ex:

Write a procedure to find number of customers from the city

"Delhi"?

create or replace procedure proc\_cust\_cnt\_delhi

is

vcnt int;

begin

select count(\*) into vcnt

from cust\_dtls where city='Delhi';

dbms\_output.put\_line

(chr(10)||

'City: Delhi'||chr(10)||

'Customer Count: '||vcnt

);

end;

Ex:

write a procedure which is finding number of emps working

under dept 30?

create procedure proc\_empcount\_30

is

e\_count int;

begin

select count(\*) into e\_count

from emp

where deptno=30;

dbms\_output.put\_line

(' Number of emps under deptno 30 are: '||e\_count);

end proc\_empcount\_30;

Ex: execute proc\_ecount\_30; /\* Executing procedure explicitly \*/

How do i execute a procedure implicitly?

By making a call to the procedure we can implicitly execute the procedure.

--Calling a procedure from any program

BEGIN

proc\_ecount\_30; --Procedure calling stmt

dbms\_output.put\_line(' calling procedure is finished');

dbms\_output.put\_line(' Program Execution is finished');

end;

--WRITING THE PROCEDURE TO DISPLAY THE NUMBER

OF EMPS IN THE GIVEN DEPTNO?

CREATE OR REPLACE PROCEDURE PROC\_empcnt\_did

(VDNO EMP.DEPTNO%TYPE)

IS

CNT INT;

BEGIN

SELECT COUNT(\*) INTO CNT

FROM EMP

WHERE DEPTNO=VDNO;

DBMS\_OUTPUT.PUT\_LINE

(chr(10)||

' NUMBER OF EMPS IN DEPT : '||VDNO||': ARE : '||CNT);

END PROC\_empcnt\_did;

SAMPLE EXECUTIONS:

EXEC PROC\_2(10);

EXEC PROC\_2(20);

EXEC PROC\_2(30);

anonymous block completed

NUMBER OF EMPS IN DEPT 10 ARE 2

anonymous block completed

NUMBER OF EMPS IN DEPT 20 ARE 5

anonymous block completed

NUMBER OF EMPS IN DEPT 30 ARE 6

ex:

declare

vdno int;

begin

vdno:=&vdno;

proc\_ecount\_did(vdno); --Procedure calling statement

dbms\_output.put\_line(' Exec Finished');

end;

WRITING PROCEDURE CALLING STATEMENTS IN A PROGRAM:-

For the following program we need to create

2 procedures called p\_add and p\_multi.

declare

x int;

y int;

begin

x:=&x;

y:=&y;

dbms\_output.put\_line(' adding '||x ||' and '||y);

Dbms\_output.put\_line(' Making a call to P\_ADD procedure');

p\_add(x,y);

dbms\_output.put\_line(' Multiplication of '||x||' and '||y);

dbms\_output.put\_line(' make a call to P\_MULTI procedure');

p\_multi(x,y);

dbms\_output.put\_line(' 2 tasks successfully completed');

end;

Ex:

wap to display customer name,city,account name,actno and bal

from given customer id?

create or replace procedure proc\_cust\_info

(vcid cust\_dtls.cno%type)

as

type cust\_type IS RECORD

(

cname cust\_dtls.cname%type,

city cust\_dtls.city%type,

act\_name act\_types.act\_name%type,

actno cust\_act\_dtls.actno%type,

actbal cust\_act\_dtls.act\_bal%type

);

Vcustrec cust\_type;

begin

select cd.cname,cd.city,at.act\_name,cad.actno,cad.act\_bal

into

vcustrec

from cust\_dtls cd, cust\_act\_dtls cad, act\_types at

where cd.cno=vcid

and

(

cd.cno=cad.cno

and

cad.act\_type=at.act\_type

);

dbms\_output.put\_line

(chr(10)||' Given customer Id: '||vcid||chr(10)||

'-----------------------------------------------'||chr(10)||

'Cust name: '||vcustrec.cname||chr(10)||

'City: '||vcustrec.city||chr(10)||

'Account Name: '||vcustrec.act\_name||chr(10)||

'Account Number: '||vcustrec.actno||chr(10)||

'Balance: '||vcustrec.actbal

);

end proc\_cust\_info;

/

Ex:

write a procedure to display the customer details and

his account details for the custid "cust-5"?

create or replace procedure proc\_CUST\_ACT\_DTLS

is

v\_cname cust\_dtls.cname%type;

v\_gender cust\_dtls.gender%type;

v\_city cust\_dtls.city%type;

v\_actno cust\_act\_dtls.actno%type;

v\_act\_type cust\_act\_dtls.act\_type%type;

v\_act\_bal cust\_act\_dtls.act\_bal%type;

begin

select c.cname,c.gender,c.city,ca.actno,ca.act\_type,ca.act\_bal INTO

V\_CNAME,v\_gender,v\_city,v\_actno,v\_act\_type,v\_act\_bal

from cust\_dtls c , cust\_act\_dtls ca

where c.cno='cust-5'

and

c.cno=ca.cust\_code;

dbms\_output.put\_line(' Details of customer 5');

dbms\_output.put\_line(v\_cname||' '||v\_gender||' '||v\_city||' '||v\_actno||' '||v\_act\_type||' '||v\_act\_bal);

end proc\_CUST\_ACT\_DTLS;

Ex:

write a procedure to display the customer details and his account

details for the GIVEN custid ?

create or replace procedure proc\_3

(VCID cust\_dtls.cno%type)

is

v\_cname cust\_dtls.cname%type;

v\_gender cust\_dtls.gender%type;

v\_city cust\_dtls.city%type;

v\_actno cust\_act\_dtls.actno%type;

v\_act\_type cust\_act\_dtls.act\_type%type;

v\_act\_bal cust\_act\_dtls.act\_bal%type;

begin

select c.cname,c.gender,c.city,ca.actno,ca.act\_type,ca.act\_bal INTO

V\_CNAME,v\_gender,v\_city,v\_actno,v\_act\_type,v\_act\_bal

from cust\_dtls c , cust\_act\_dtls ca

where c.cno=VCID

and

c.cno=ca.cust\_code;

dbms\_output.put\_line(' Details of customer '||VCID);

dbms\_output.put\_line(v\_cname||' '||v\_gender||' '||v\_city||' '||v\_actno||' '||v\_act\_type||' '||v\_act\_bal);

end proc\_3;

Ex:

Ex:

write a procedure to display the customer details and his account

details for the GIVEN custid ?

create or replace procedure proc\_3

(VCID cust\_dtls.cno%type)

is

type cust\_type is record

(

custname cust\_dtls.cname%type,

custcity cust\_dtls.city%type,

gender cust\_dtls.gender%type,

actno cust\_act\_dtls.actno%type,

act\_type cust\_act\_dtls.act\_type%type,

act\_bal cust\_act\_dtls.act\_bal%type

);

vactdtls cust\_type;

begin

select c.cname,c.city,c.gender,

cad.actno,cad.act\_type,cad.act\_bal

INTO

vactdtls

from cust\_dtls c , cust\_act\_dtls cad

where c.cno=VCID

and

c.cno=cad.cno;

dbms\_output.put\_line

(' Details of customer: '||VCID||CHR(10)||

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line

(

chr(10)||

'Name: '||vactdtls.custname||chr(10)||

'City: '||vactdtls.custcity||chr(10)||

'gender: '||vactdtls.gender||chr(10)||

'A/c No: '||vactdtls.actno||chr(10)||

'A/c Type: '||vactdtls.act\_type||chr(10)||

'Balance: '||vactdtls.act\_bal

);

end proc\_3;

Ex: write a procedure to display the employee details and dept details for the empno 7902?

create or replace procedure proc\_4

is

v\_ename emp.ename%type;

v\_job emp.job%type;

v\_sal emp.sal%type;

v\_hiredate emp.hiredate%type;

v\_dno emp.deptno%type;

v\_dname dept.dname%type;

v\_loc dept.loc%type;

begin

select e.ename,e.sal,e.job,e.hiredate,e.deptno,d.dname,d.loc INTO

v\_ename,v\_sal,v\_job,v\_hiredate,v\_dno,v\_dname,v\_loc

from emp e , dept d

where e.empno=7902 and e.deptno=d.deptno;

dbms\_output.put\_line('Details of empno= 7902');

dbms\_output.put\_line(v\_ename||' '||v\_job||' '||v\_sal||' '||v\_hiredate||' '||v\_dno||' '||v\_dname||' '||v\_loc);

end proc\_4;

DYNAMIC PROCEDURES:

EX: write a procedure to display the details of employee for the given employee id?

create or replace procedure proc\_5 ( vempid emp.empno%type)

is

emp\_rec emp%rowtype;

begin

select \* into emp\_rec from emp where empno=vempid;

dbms\_output.put\_line(' employee details of '||vempid);

dbms\_output.put\_line(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

dbms\_output.put\_line(emp\_rec.ename||','||emp\_rec.sal||','||emp\_rec.job||','||emp\_rec.hiredate);

end proc\_5;

sample output:

employee details of 7902

FORD,3000,ANALYST,03-DEC-81

anonymous block completed

employee details of 7788

SCOTT,3000,ANALYST,19-APR-87

anonymous block completed

employee details of 7654

MARTIN,1250,SALESMAN,28-SEP-81

Ex: write a procedure to display the customer mobile number and name of the given custid if he is male and from the city delhi?

desc cust\_dtls

Name Null Type

---------- ---- ------------

CUSTID CHAR(4)

CUSTNAME VARCHAR2(20)

CUSTCITY VARCHAR2(10)

CUSTGENDER CHAR(1)

CUSTMOBILE NUMBER(10)

create or replace procedure proc\_5(vcid cust\_dtls.cno%type) is

--vmobile cust\_dtls.mobile%type;

vcustname cust\_dtls.cname%type;

begin

select cname into vcustname

from cust\_dtls

where cno=vcid and gender='M' and city='Delhi';

dbms\_output.put\_line(' give customer id='||vcid);

dbms\_output.put\_line(' he is male and he is from delhi');

--dbms\_output.put\_line('mobile number is -'||vmobile);

dbms\_output.put\_line(' His name is '||vcustname);

end proc\_5;

Ex: write a procedure to display number of emps working under given department name?

Ex: write a procedure to display the number of customers from the given city with given gender?

Ex: write a procedure to display the number of accounts of given account type?

1 create or replace procedure proc\_insert\_accounts

2 (argactno cust\_act\_dtls.actno%type,

3 argacttype cust\_act\_dtls.act\_type%type,

4 argopendt cust\_act\_dtls.act\_open\_date%type,

5 argactbal cust\_act\_dtls.act\_bal%type,

6 argcno cust\_act\_dtls.cno%type

7 )

8 is

9 reccnt int;

10 begin

11 select count(\*) into reccnt

12 from cust\_act\_dtls;

13 dbms\_output.put\_line

14 (chr(10)||' old Record count : '||reccnt);

15 insert into cust\_act\_dtls values

16 (argactno,argacttype,argopendt,argactbal,argcno);

17 dbms\_output.put\_line

18 (chr(10)||' record inserted successfully');

19 select count(\*) into reccnt

20 from cust\_act\_dtls;

21 dbms\_output.put\_line

22 (chr(10)||' New Record count : '||reccnt);

23\* end proc\_insert\_accounts;

SQL> /

Procedure created.

SQL> declare

2 vactno cust\_act\_dtls.actno%type;

3 vacttype cust\_act\_dtls.act\_type%type;

4 vactopendt date;

5 vactbal number;

6 vcno cust\_act\_dtls.cno%type;

7 begin

8 vactno:='&vactno';

9 vacttype:='&vacttype';

10 vactopendt:=sysdate;

11 vactbal:='&vactbal';

12 vcno:='&vcno';

13 proc\_insert\_accounts

14 (vactno,vacttype,vactopendt,vactbal,vcno);

15 end;

16 /

Enter value for vactno: 20035201489

old 8: vactno:='&vactno';

new 8: vactno:='20035201489';

Enter value for vacttype: CA

old 9: vacttype:='&vacttype';

new 9: vacttype:='CA';

Enter value for vactbal: 35000

old 11: vactbal:='&vactbal';

new 11: vactbal:='35000';

Enter value for vcno: cust-1

old 12: vcno:='&vcno';

new 12: vcno:='cust-1';

old Record count : 7

record inserted successfully

New Record count : 8

PL/SQL procedure successfully completed.

SQL>

Ex:

write a procedure to display number of customers from given

account type with min bal 20000?

SQL> create or replace procedure proc\_cust\_cnt

2 (vact\_type cust\_act\_dtls.act\_type%type)

3 is

4 vcnt int;

5 begin

6 select count(\*) into vcnt

7 from cust\_act\_dtls

8 where act\_type=vact\_type

9 and

10 act\_bal>=20000;

11 dbms\_output.put\_line

12 (chr(10)||

13 'Given account type: '||vact\_type||chr(10)||

14 'Number of customers: '||vcnt

15 );

16 end;

17 /

Ex: Write a procedure to display the name,sal,job,

hiredate,deptno for the given employee id?

create or replace procedure proc\_emp( veno emp.empno%type)

is

type erec is record(

name emp.ename%type,

esal emp.sal%type,

desg emp.job%type,

jdate emp.hiredate%type,

dno emp.deptno%type

);

rec erec;

begin

select ename,sal,job,hiredate,deptno into rec

from emp where empno=veno;

dbms\_output.put\_line(' Ename='||rec.name);

dbms\_output.put\_line(' EmpSal='||rec.esal);

dbms\_output.put\_line(' Emp Job Title='||rec.desg);

dbms\_output.put\_line(' Emp join date='||rec.jdate);

dbms\_output.put\_line(' Emp Dept No='||rec.dno);

end proc\_emp;

Advantages:

i) It is reducing the number of declarations of variables.

ii) It is reducing the amount of memory reserved for Record based variable.

Ex:

Write a procedure to display the name,cost and warrenty of the

product for the given product id?

Ex:

Write a procedure to display pname,cost,mfg,exp,companyName,Dealer

address,phonenumber for the given product id?

EX:

write procedure to display customer name,city,gender,account

name, balance from given account number?

create or replace procedure proc\_cust

(vactno IN cust\_act\_dtls.actno%type)

is

TYPE custtype IS RECORD

(

cname cust\_dtls.cname%type,

city cust\_dtls.city%type,

gender cust\_dtls.gender%type,

actname act\_types.act\_name%type,

actbal cust\_act\_dtls.act\_bal%type

);

vrec custtype;

begin

select a.cname,a.city,a.gender,b.act\_name,

c.act\_bal

into vrec

from cust\_dtls a inner join cust\_act\_dtls c

on

c.actno=vactno

and

(a.cno=c.cno

inner join act\_types b

on c.act\_type=b.act\_type);

dbms\_output.put\_line

(chr(10)||

'Account number: '||vactno||chr(10)||

'Customer name: '||vrec.cname||chr(10)||

'City: '||vrec.city||chr(10)||

'Gender: '||vrec.gender||chr(10)||

'Account Name: '||vrec.actname||chr(10)||

'Account Bal: '||vrec.actbal

);

end;

-----------------------------------------------------------------------------------------