Experiment No.2

Name: Sumit Prabhakar Kamble.

Roll_no: 47

Div: B Batch: B2 Class: TY CSE

PRN: 21ST114282045

Title: Implement Procedures, Functions and Cursor in PL/SQL.

Problem statement

A. Implement procedures in PL/SQL.

1. Create a standalone procedure to display a simple message 'Hello'.

```
create or replace procedure msg
AS
BEGIN
        dbms_output.put_line('Hello');
END;
```

```
Procedure created.
```

2. Call the created procedure through a PL/SQL block.

```
begin
    msg;
end;
```

```
Statement processed.
Hello
```

3. Create a procedure to display a simple message 'Hello' inside PL/SQL block.

```
Declare
procedure display
is
Begin
```

```
Dbms_output.put_line('Hello';)
End;
Begin
Display;
End;
```

```
Statement processed.
Hello
```

4. Create a procedure to find square of a number using two different modes of parameter passing.

a. IN , OUT mode

```
Declare
Result number;
Procedure find_square(x in number, y out number)
Is
Begin
y:=x*x;
end;
begin
find_square(6,result);
dbms_output.put_line('square of a number is'||result);
end;
```

```
Statement processed.
square of a number is36
```

b. IN OUT mode.

```
Declare
Result number;
Procedure find_square(x in out number)
Is
Begin
x:=x*x;
end;
begin
result:=6;
find_square(result);
dbms_output.put_line('square of a number is'||result);
```

```
end;
```

```
Statement processed.
square of a number is36
```

5. Create table Student with attributes roll_no, name, address, contact_no.

```
Create table student
(
roll_no number,
name varchar(20),
address varchar(20),
contact_no number(10)
);
```

```
Table created.
```

6. Create a procedure to insert 4 values in Student table.

```
Declare
procedure insert_val(
s_roll student.roll_no %type,
s_name student.name %type,
s_addr student.address %type,
s_contact student.contact_no %type
)
Is
Begin
Insert into student values(s_roll, s_name, s_addr, s_contact);
End;

Begin
insert_val(1,'Sumit','Kamble', 7234567890);
End;
```

```
Statement processed.
```

7. Print the student table.

```
Select * from student;
```

ROLL_NO	NAME	ADDRESS	CONTACT_NO
1	Sumit	Kamble	7234567890

Download CSV

A. Implement Functions and Cursor in PL/SQL.

1. Create a standalone function to display a simple message 'Hello'.

```
CREATE OR REPLACE FUNCTION msgg
return varchar2
AS
str varchar2(20);
BEGIN
str:='HELLO';
return str;
END;
```

```
Function created.
```

```
DECLARE
str1 varchar2(20);
BEGIN
str1:=msgg;
dbms_output.put_line(str1);
END;
```

```
Statement processed.
HELLO
```

2. Create a function to add two numbers.

```
DECLARE
a number;
b number;
c number;
FUNCTION add(x IN number, y IN number)
RETURN number
IS
z number;
BEGIN
z:=x+y;
return z;
END;
BEGIN
a:=10;
b:=20;
c:=add(a,b);
dbms_output.put_line('Addition is:'||c);
END;
```

```
Statement processed.
Addition is:40
```

3. Create a table 'student' with attributes roll_no, name, address, contact.

```
CREATE TABLE student
(
roll_no number,
name varchar2(25),
address varchar2(50),
contact number(10)
```

);

```
Table created.
```

4. Create a function to insert values inside ‘student’ table.

5. Insert following values in ‘student’ table using created function.

```
DECLARE
a number;
function insert_val
(
s_id student.roll_no %type,
s_name student.name %type,
s_address student.address %type,
s_contact_no student.contact_no %type
)
return number
as
x number;
begin
x:=1;
insert into student values(s_id, s_name, s_address, s_contact_no);
return x;
end;
begin
a:=insert_val(1,'Sumit','pune',8746958608);
a:=insert_val(2,'Shubham','mumbai',8746958908);
a:=insert_val(3,'Yash','kolhapur',8946958608);
a:=insert_val(4,'Raj','karad',9946958608);
a:=insert_val(5,'pankaj','pune',8746058608);
```

end;

```
Statement processed.
```

6. Create a cursor to print all the values from ‘student’ table.

```
declare
s_no student1.roll_no %type;
s_name student1.name %type;
s_address student1.address %type;
s_phn_no student1.phn_no %type;
cursor c_stud IS select roll_no, name, address, phn_no from student1;
begin
open c_stud;
loop
fetch c_stud into s_no, s_name, s_address, s_phn_no;
EXIT when c_stud %notfound;
dbms_output.put_line(s_no||' '||s_name||' '||s_address||' '||s_phn_no||' ');
End loop;
close c_stud;
end;
```

```
Statement processed.
1 Sumit pune 8746958608
2 Shubham mumbai 8746958908
3 Yash kolhapur 8946958608
4 Raj karad 9946958608
5 pankaj pune 8746058608
```

7. Create a function to find the name of the student whose id is 2.

```
declare
str2 varchar(50);
function find_name(s_id student1.roll_no %type)
```

```
return varchar
IS
s_name student1.name % type;
BEGIN
select name into s_name from student1 where roll_no=s_id;
return s_name;
END;
BEGIN
str2:=find_name(2);
dbms_output.put_line(str2);
END;
```

```
Statement processed.
Shubham
```

8. Create a function to update the name of the student to ‘roma’ whose id is 4.

```
DECLARE
str3 varchar(50);
function update_rec(s_id student1.roll_no %type)
return varchar
IS
s_name student1.name % type;
BEGIN
update student1 SET name='roma' where roll_no=s_id;
return s_name;
END;
BEGIN
str3:=update_rec(4);
dbms_output.put_line(str3);
```

END;

select * from student1;

ROLL_NO	NAME	ADDRESS	PHN_NO
1	Sumit	pune	8746958608
2	Shubham	mumbai	8746958908
3	Yash	kolhapur	8946958608
4	roma	karad	9946958608
5	pankaj	pune	8746058608

Download CSV

5 rows selected.

9. Create a function to delete a record of a student whose id is 3.

```
DECLARE
str4 varchar(50);
function delete_rec(s_id student1.roll_no %type)
return varchar
IS
s_name student1.name %type;
BEGIN
delete from student1 where roll_no=s_id;
return s_name;
END;
BEGIN
str4:=delete_rec(3);
```

```
dbms_output.put_line(str4);
END;
```

```
Statement processed.
```

ROLL_NO	NAME	ADDRESS	PHN_NO
1	Sumit	pune	8746958608
2	Shubham	mumbai	8746958908
4	roma	karad	9946958608
5	pankaj	pune	8746058608

Download CSV

```
4 rows selected.
```