

PL/SQL Procedures

Ex.no: 10

Aim:

To write a program on PL/SQL procedures on

sample exercises

Program:

Program to add 2 numbers

CREATE OR REPLACE PROCEDURE add AS

num1 NUMBER;

num2 NUMBER;

sum NUMBER;

BEGIN

DBMS_OUTPUT.PUT('Enter the 1st number');

num1 := &num1;

DBMS_OUTPUT.PUT('Enter the 2nd number');

num2 := &num2;

sum := num1 + num2;

DBMS_OUTPUT.PUT_LINE ('The sum of || num1 || and
num2 || is || sum);

END;

Output:

Enter the 1st number : 5

Enter the 2nd number : 10

The sum of 5 and 10 is 15

Result.

Thus a program on PL/SQL procedures was verified & executed successfully.

Ex. no: 11

PL/SQL Functions

Aim:

To write a program on PL/SQL functions

Program:

CREATE OR REPLACE FUNCTION add(num1 NUMBER,

 num2 NUMBER)

RETURN NUMBER

IS

 sum NUMBER;

BEGIN

 sum := num1 + num2;

 RETURN sum;

END;

DECLARE

 num1 NUMBER := &num1;

 num2 NUMBER := &num2;

 result NUMBER;

BEGIN

 result := add (num1, num2);

DBMS_OUTPUT.PUT_LINE ('The sum of ' || num1 ||

' and ' || num2 || ' is ' || result);

END;

Output:

5

10

The sum of 5 and 10 is 15

PL/SQL block successfully created.

Result:

Thus a program on PL/SQL functions was verified & executed successfully.

Aim:

To write a program on PL/SQL cursors

Program:

```

Declare
CURSOR C_employees IS
SELECT employee_id, first_name, last_name, salary
FROM employees;

employee_id employees.employee_id%Type;
salary employees.salary%Type;

BEGIN
OPEN C_employees;
FETCH C_employees INTO employee_id, first_name, salary;
WHILE C_employees.%FOUND LOOP
DBMS_OUTPUT.PUT_LINE ('Employee ID: ' || employee_id);
DBMS_OUTPUT.PUT_LINE ('Salary' || salary);
FETCH C_employees INTO employee_id, salary;
END LOOP;
close C_employees;
END;

```

Result:

This a program on PL/SQL cursor was verified & executed successfully.

PL/SQL exception handling

Ex-no: 13

Aim: To write a program on PL/SQL exception handling

Program:

DECLARE

V-num1 Number := &Enter-Fir1-num;

V-num2 Number := &Enter-second-num;

V-result number;

BEGIN

V-result := V-num1/V-num2;

DBMS-OUTPUT.PUT-LINE ('The result is '||V-result);

Exceptions

WHEN ZERO-DIVIDE THEN

DBMS-OUTPUT.PUT-LINE ('Error: cannot divide by 0');

END;

Exception ops

Error: cannot divide by 0.

Result:

Thus a program on PL/SQL exception handling was verified & executed successfully.

Ex.no: 14

PL/SQL Trigger

Aim:

To write a program on PL/SQL Triggers

Program:

CREATE OR REPLACE Trigger tr

Before insert or update on employees

for each row

Declare

v-old-sal employees .sal % type;

Begin

if inserting then

DBMS-output.PUT-LINE ('Inserting new employee
with salary : ' || :new.sal);

else if updating then

select sal into v-old-sal from employees where
employee.id = old.employee_id;

if :new.sal > v-old-sal then

DBMS-output.PUT-LINE ('Employee with id ' || old.
employee_id || ' got a sal raise to ' || new.sal);

end if;

end if;

end;

Result:

Thus a program on PL/SQL Triggers was verified &
executed successfully.