**CSE2007 DBMS LAB**

**SLOT: L39+L40**

**NAME - AMAN SAHU**

**REG. NO – 22BCE7224**

**EXPERIMENT NO.-8**

1.

a. Write a PL/SQL program to swap two numbers.

set serveroutput on

DECLARE

num1 NUMBER := 5;

num2 NUMBER := 7;

temp NUMBER;

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Before swapping: num1 = ' || num1 || ', num2 = ' || num2);

temp := num1;

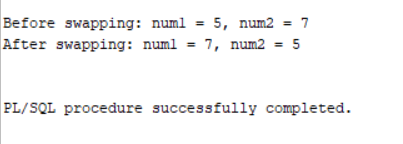
num1 := num2;

num2 := temp;

DBMS\_OUTPUT.PUT\_LINE('After swapping: num1 = ' || num1 || ', num2 = ' || num2);

END;

/



b. Write a PL/SQL program to find the largest of three numbers.

DECLARE

num1 NUMBER := 5;

num2 NUMBER := 100;

num3 NUMBER := 50;

largest NUMBER;

BEGIN

largest := num1;

IF num2 > largest THEN

largest := num2;

END IF;

IF num3 > largest THEN

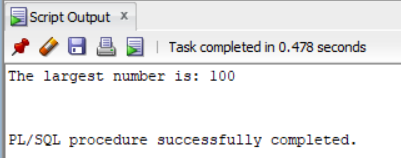
largest := num3;

END IF;

DBMS\_OUTPUT.PUT\_LINE('The largest number is: ' || largest);

END;

/



2.

a.Write a PL/SQL program to find the total and average of 6 subjects and display the grade.

DECLARE

s1 number :=89;

s2 number :=85;

s3 number :=95;

s4 number :=99;

s5 number :=81;

s6 number :=94;

total NUMBER;

average NUMBER;

BEGIN

total := s1+s2+s3+s4+s5+s6;

average := total / 6;

DBMS\_OUTPUT.PUT\_LINE('Total marks: ' || total);

DBMS\_OUTPUT.PUT\_LINE('Average marks: ' || average);

IF average >= 90 THEN

DBMS\_OUTPUT.PUT\_LINE('Grade: S');

ELSIF average >= 80 THEN

DBMS\_OUTPUT.PUT\_LINE('Grade: A');

ELSIF average >= 70 THEN

DBMS\_OUTPUT.PUT\_LINE('Grade: B');

ELSIF average >= 60 THEN

DBMS\_OUTPUT.PUT\_LINE('Grade: C');

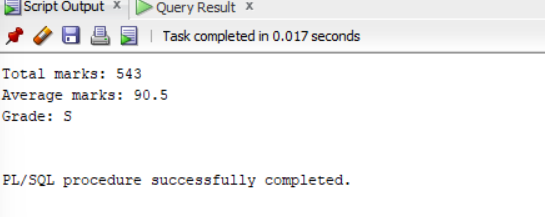
ELSE

DBMS\_OUTPUT.PUT\_LINE('Grade: D');

END IF;

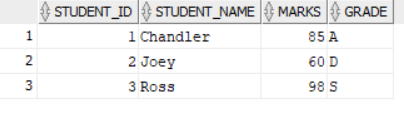
END;

/



b. Write a PL/SQL program to print a student information along with his marks and grade and also update the grade in the student database. Input to be fetched from a student table.

Initial Table-



SET SERVEROUTPUT ON;

DECLARE

v\_student\_id student1.student\_id%TYPE;

v\_student\_name student1.student\_name%TYPE;

v\_marks student1.marks%TYPE;

v\_grade student1.grade%TYPE;

BEGIN

FOR student\_rec IN (SELECT \* FROM student1) LOOP

v\_student\_id := student\_rec.student\_id;

v\_student\_name := student\_rec.student\_name;

v\_marks := student\_rec.marks;

v\_grade := student\_rec.grade;

DBMS\_OUTPUT.PUT\_LINE('Student ID: ' || v\_student\_id || ', Student Name: ' || v\_student\_name || ', Marks: ' || v\_marks || ', Grade: ' || v\_grade)

IF v\_marks >= 90 THEN

v\_grade := 'S'; -- Outstanding

ELSIF v\_marks >= 80 THEN

v\_grade := 'A'; -- Excellent

ELSIF v\_marks >= 70 THEN

v\_grade := 'B'; -- Good

ELSIF v\_marks >= 60 THEN

v\_grade := 'C'; -- Satisfactory

ELSE

v\_grade := 'D'; -- Needs improvement

END IF;

UPDATE student1

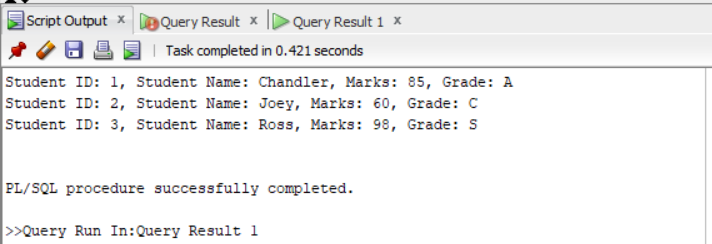
SET grade = v\_grade

WHERE student\_id = v\_student\_id;

END LOOP;

END;

/



3.

a. Write a PL/SQL program to take two numbers and a choice(1/2/3/4). Based on the choice the program to be performed addition/subtraction/multiplication/division of given two numbers.

set serveroutput on

DECLARE

num1 NUMBER;

num2 NUMBER;

choice NUMBER;

result NUMBER;

BEGIN

num1 := 100;

num2 := 5;

DBMS\_OUTPUT.PUT\_LINE('Enter your choice: 1) Addition, 2) Subtraction, 3) Multiplication, 4) Division');

DBMS\_OUTPUT.PUT\_LINE('Your choice: ');

choice := &choice;

IF choice = 1 THEN

result := num1 + num2;

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 2 THEN

result := num1 - num2;

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 3 THEN

result := num1 \* num2;

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 4 THEN

IF num2 = 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Division by zero is not allowed');

ELSE

result := num1 / num2;

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

END IF;

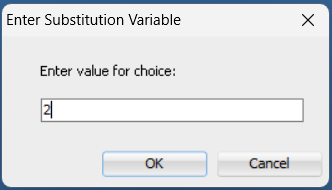
ELSE

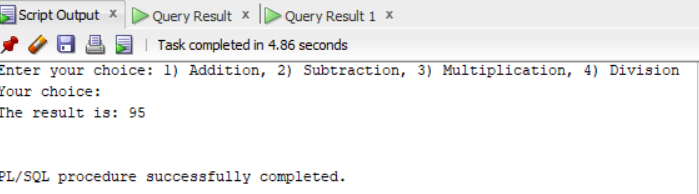
DBMS\_OUTPUT.PUT\_LINE('Invalid choice');

END IF;

END;

/





b. Write a PL/SQL program to take two strings and a choice. Based on the choice, you need to perform different string operations on given strings.(Ex: LTRIM, RTRIM, LENGTH, SUBSTR, LPAD, RPAD, etc.,)

DECLARE

str1 VARCHAR2(100) := ' Hello, World! ';

str2 VARCHAR2(100) := 'Database Management System';

choice NUMBER;

result VARCHAR2(100);

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Enter your choice: 1) LTRIM, 2) RTRIM, 3) LENGTH, 4) SUBSTR, 5) LPAD, 6) RPAD');

DBMS\_OUTPUT.PUT\_LINE('Your choice: ');

choice := &choice;

IF choice = 1 THEN

result := LTRIM(str1);

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 2 THEN

result := RTRIM(str1);

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 3 THEN

result := TO\_CHAR(LENGTH(str1));

DBMS\_OUTPUT.PUT\_LINE('The length of the string is: ' || result);

ELSIF choice = 4 THEN

result := SUBSTR(str1, 1, 5);

DBMS\_OUTPUT.PUT\_LINE('The substring is: ' || result);

ELSIF choice = 5 THEN

result := LPAD(str1, 15, '\*');

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

ELSIF choice = 6 THEN

result := RPAD(str1, 15, '\*');

DBMS\_OUTPUT.PUT\_LINE('The result is: ' || result);

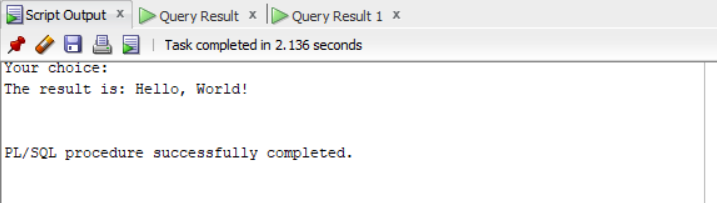
ELSE

DBMS\_OUTPUT.PUT\_LINE('Invalid choice');

END IF;

END;

/



4.

a.Write a PL/SQL code block to calculate the area of a circle for a value of radius given by the user. Store the radius and the corresponding values of calculated area in a table named areas, consisting of two columns radius and area.

CREATE TABLE areas (

radius NUMBER(5,2),

area NUMBER(10,4)

);

DECLARE

radius\_input NUMBER;

area\_calc NUMBER;

BEGIN

radius\_input := &radius;

area\_calc := 3.14 \* radius\_input \* radius\_input;

INSERT INTO areas (radius, area) VALUES (radius\_input, area\_calc);

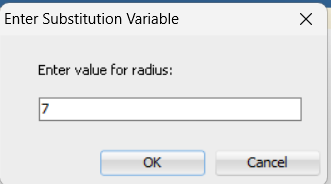
DBMS\_OUTPUT.PUT\_LINE('Area calculated and stored successfully for radius: ' || radius\_input);

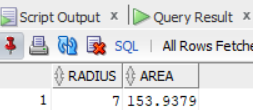
COMMIT;

END;

/

Select \* from areas;





b.Write a PL/SQL code to update the salary and commission of employee as per the input given by user.

DECLARE

v\_salary NUMBER(8,2);

v\_commission NUMBER(8,2);

v\_employee\_id NUMBER(6);

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Enter the employee ID: ');

v\_employee\_id := &input\_employee\_id;

DBMS\_OUTPUT.PUT\_LINE('Enter the new salary: ');

v\_salary := &input\_salary;

DBMS\_OUTPUT.PUT\_LINE('Enter the new commission: ');

v\_commission := &input\_commission;

UPDATE emp

SET sal = v\_salary,

comm = v\_commission

WHERE empno = v\_employee\_id;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Salary and commission updated for employee ID: ' || v\_employee\_id);

END;

/

SELECT \* FROM emp;

