PRACTICAL – 06

**NAME : Aryan Kashikar  
Roll n0: D2 32**

**Introduction to PL/SQL and execution of PL/SQL blocks using oracle 11g**

**AIM: To write PL/SQL blocks using oracle 11g**

Problem Statements:

1. Write a PL/SQL program for swapping 2 numbers.

DECLARE

   num1 NUMBER := &num1;

   num2 NUMBER := &num2;

   temp NUMBER;

BEGIN

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

   temp := num1;

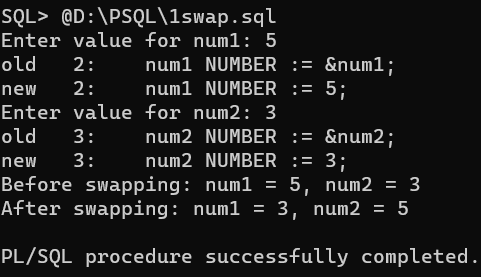
   num1 := num2;

   num2 := temp;

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

END;

/



2. Write a PL/SQL block to find the maximum number from given three

numbers.

DECLARE

   num1 NUMBER := &num1;

   num2 NUMBER := &num2;

   num3 NUMBER := &num3;

   max\_num NUMBER;

BEGIN

   IF num1 >= num2 AND num1 >= num3 THEN

      max\_num := num1;

   ELSIF num2 >= num1 AND num2 >= num3 THEN

      max\_num := num2;

   ELSE

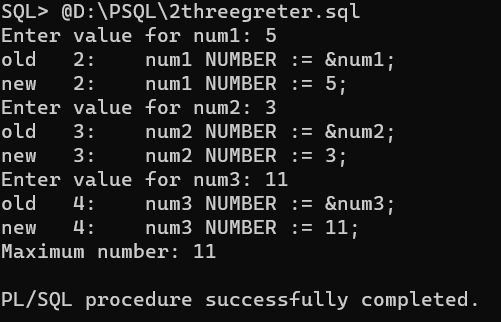
      max\_num := num3;

   END IF;

   DBMS\_OUTPUT.PUT\_LINE('Maximum number: ' || max\_num);

END;

/



3. Write a PL/SQL program to input marks of 4 subjects. find the total

and Percentage of 4 subjects and display the grade. [ Grade “fail “ if

any of subject got percentage less than 40,

Otherwise grade A (>85),B(less than 85 and >=75),C(less than 75 and

>=65), D(less than 65)

DECLARE

   subject1\_marks NUMBER := &subject1\_marks;

   subject2\_marks NUMBER := &subject2\_marks;

   subject3\_marks NUMBER := &subject3\_marks;

   subject4\_marks NUMBER := &subject4\_marks;

   total\_marks NUMBER;

   percentage NUMBER;

   grade VARCHAR2(10);

BEGIN

   total\_marks := subject1\_marks + subject2\_marks + subject3\_marks + subject4\_marks;

   percentage := total\_marks / 4;

   IF subject1\_marks < 40 OR subject2\_marks < 40 OR subject3\_marks < 40 OR subject4\_marks < 40 THEN

      grade := 'Fail';

   ELSIF percentage > 85 THEN

      grade := 'A';

   ELSIF percentage >= 75 THEN

      grade := 'B';

   ELSIF percentage >= 65 THEN

      grade := 'C';

   ELSE

      grade := 'D';

   END IF;

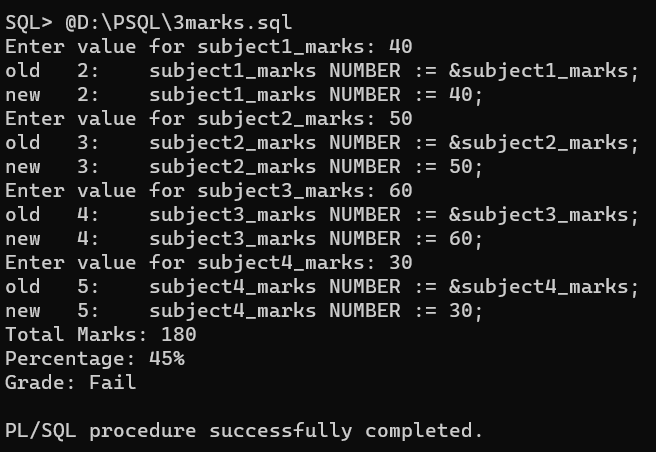
   DBMS\_OUTPUT.PUT\_LINE('Total Marks: ' || total\_marks);

   DBMS\_OUTPUT.PUT\_LINE('Percentage: ' || percentage || '%');

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

END;

/



4. Write a program to accept a number and find the sum of its digits.

declare

    num int :=0;

    i int;

    s int :=0;

    r int;

begin

    num:=&num;

    while num > 0 loop

        r:= MOD(num, 10);

        s := s + r;

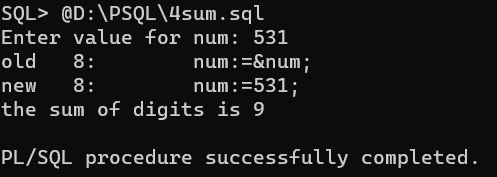
        num:=floor(num/10);

    end loop;

    dbms\_output.put\_line('  the sum of digits is '||s );

end;

/



5. Write a PL/SQL program to find the factorial of a given number.

DECLARE

   num NUMBER := &num;

   factorial NUMBER := 1;

BEGIN

   IF num < 0 THEN

      DBMS\_OUTPUT.PUT\_LINE('Factorial is not defined for negative numbers.');

   ELSE

      FOR i IN 1..num LOOP

         factorial := factorial \* i;

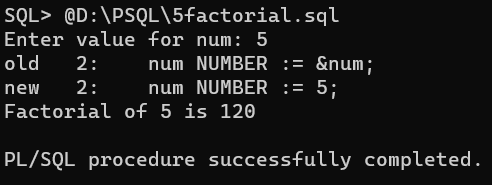
      END LOOP;

      DBMS\_OUTPUT.PUT\_LINE('Factorial of ' || num || ' is ' || factorial);

   END IF;

END;

/



6. Write a program to accept a number/string and check it is

palindrome or not.

DECLARE

   input\_str VARCHAR2(100) := '&input\_str';

   reversed\_str VARCHAR2(100) := '';

BEGIN

   FOR i IN REVERSE 1..LENGTH(input\_str) LOOP

      reversed\_str := reversed\_str || SUBSTR(input\_str, i, 1);

   END LOOP;

   IF input\_str = reversed\_str THEN

      DBMS\_OUTPUT.PUT\_LINE(input\_str || ' is a palindrome.');

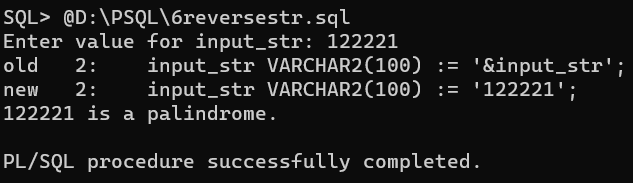
   ELSE

      DBMS\_OUTPUT.PUT\_LINE(input\_str || ' is not a palindrome.');

   END IF;

END;

/



7. Write a PL / SQL program to check whether the given number is

prime or not.

DECLARE

   num NUMBER := &num;

   is\_prime BOOLEAN := TRUE;

BEGIN

   IF num <= 1 THEN

      is\_prime := FALSE;

   ELSE

      FOR i IN 2..(num-1) LOOP

         IF MOD(num, i) = 0 THEN

            is\_prime := FALSE;

            EXIT;

         END IF;

      END LOOP;

   END IF;

   IF is\_prime THEN

      DBMS\_OUTPUT.PUT\_LINE(num || ' is a prime number.');

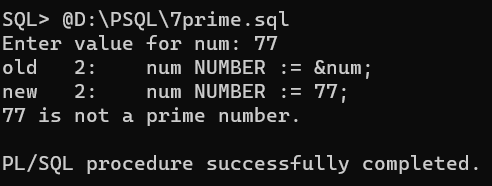
   ELSE

      DBMS\_OUTPUT.PUT\_LINE(num || ' is not a prime number.');

   END IF;

END;

/



8. Write a PL / SQL code to find sum of below series.

S=1/1!+2/2!+3/3!........n/n!

DECLARE

   n NUMBER := &n;

   s NUMBER := 0;

   factorial NUMBER := 1;

BEGIN

   FOR i IN 1..n LOOP

      factorial := factorial \* i;

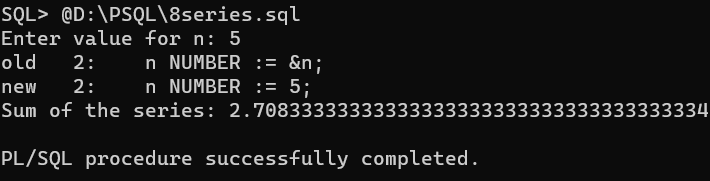
      s := s + (i / factorial);

   END LOOP;

   DBMS\_OUTPUT.PUT\_LINE('Sum of the series: ' || s);

END;

/



9. Calculate the area of a triangle for a value of base varying from 3 to 9

and for each base value height varying from 3 to 9. Store the base,

height and the corresponding values of calculated area in table

areatri. Consisting of three columns ht, bs and area.

CREATE TABLE areatri (

   ht NUMBER,

   bs NUMBER,

   area NUMBER

);

DECLARE

   base\_start NUMBER := 3;

   base\_end NUMBER := 9;

   height\_start NUMBER := 3;

   height\_end NUMBER := 9;

   area NUMBER;

BEGIN

   FOR base IN base\_start..base\_end LOOP

      FOR height IN height\_start..height\_end LOOP

         area := (base \* height) / 2;

         INSERT INTO areatri (ht, bs, area) VALUES (height, base, area);

      END LOOP;

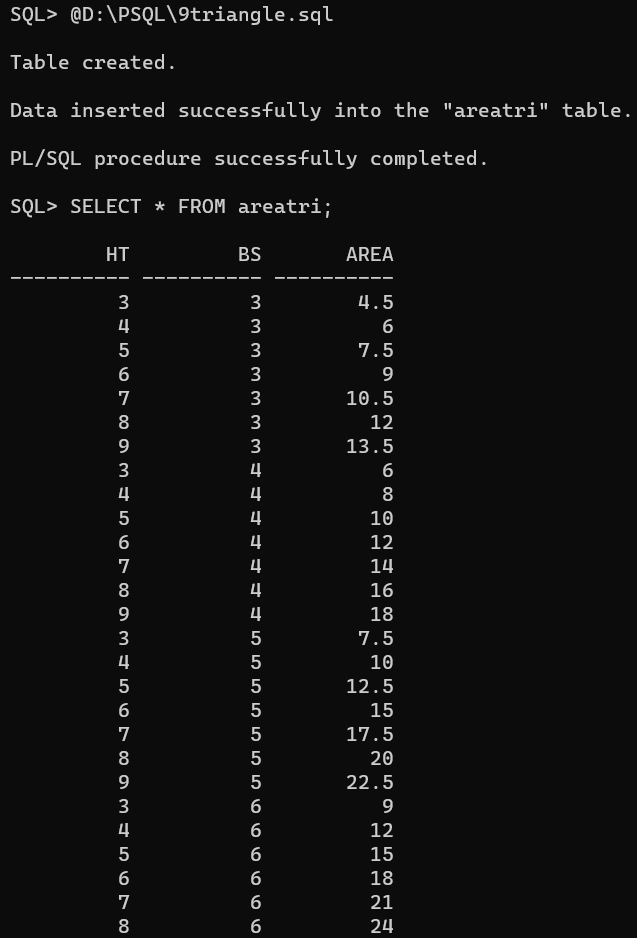
   END LOOP;

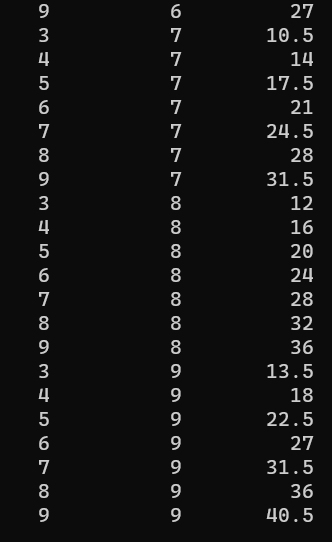
   COMMIT;

   DBMS\_OUTPUT.PUT\_LINE('Data inserted successfully into the "areatri" table.');

END;

/





10. Print below pattern

55555

4444

333

22

1

DECLARE

           n INTEGER;

           i INTEGER;

           j INTEGER;

          BEGIN

           n:=&n;

        FOR i IN 1..n LOOP

        FOR j IN 1..n-i+1 LOOP

        dbms\_output.put((n-i+1));

        END LOOP;

          DBMS\_OUTPUT.NEW\_LINE;

         END LOOP;

         END;

         /

