**CSE2007 DBMS LAB**

**SLOT: L39+L40**

**NAME - AMAN SAHU**

**REG. NO – 22BCE7224**

**EXERICE NO.-10**

1.Develop PL/SQL procedures to find the following:

i. Is given number Palindrome or not.(The procedure should return True or False)

declare

n number;

m number;

temp number:=0;

rem number;

begin

n:=5432112345;

m:=n;

while n>0

loop

rem:=mod(n,10);

temp:=(temp\*10)+rem;

n:=trunc(n/10);

end loop;

if m = temp

then

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

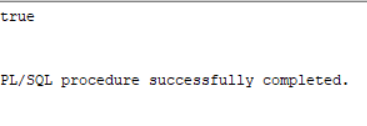
else

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

end if;

end;

/



ii. Sum of N natural numbers.(The Procedure should return sum)

DECLARE

sumVal NUMBER;

n NUMBER;

i NUMBER;

FUNCTION Findmax(n IN NUMBER)

RETURN NUMBER

IS

sums NUMBER := 0;

BEGIN

FOR i IN 1..n

LOOP

sums := sums + i\*(i+1)/2;

END LOOP;

RETURN sums;

END;

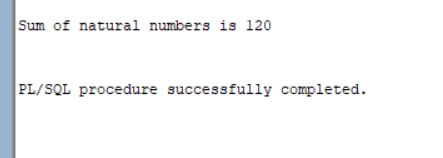
BEGIN

n := 8;

sumVal := findmax(n);

dbms\_output.Put\_line('Sum of natural numbers is ' || sumVal);

END;



iii. Fibonacci Series.(The procedure should return last number in the series)

declare

first number := 0;

second number := 1;

temp number;

n number := 5;

i number;

begin

dbms\_output.put\_line('Series:');

dbms\_output.put\_line(first);

dbms\_output.put\_line(second);

for i in 2..n

loop

temp:=first+second;

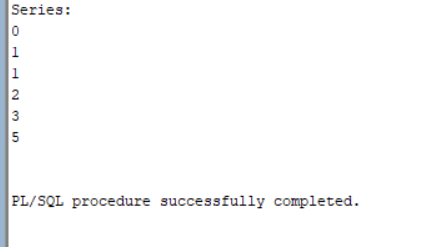
first := second;

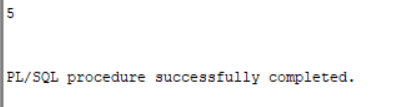
second := temp;

dbms\_output.put\_line(temp);

end loop;

end;





iv. Is a number perfect or not.(The procedure should return True or False)

SET SERVEROUTPUT ON

DECLARE

n NUMBER := &input\_number;

tot NUMBER := 0;

BEGIN

FOR i IN 1..n/2 LOOP

IF MOD(n, i) = 0 THEN

tot := tot + i;

END IF;

END LOOP;

IF tot = n THEN

DBMS\_OUTPUT.PUT\_LINE('True');

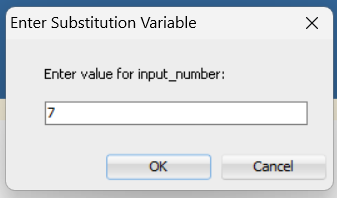
ELSE

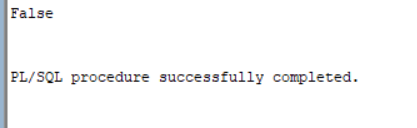
DBMS\_OUTPUT.PUT\_LINE('False');

END IF;

END;

/





v. Calculate the net salary and year salary if DA is 30% of basic, HRA is 10% of basic and PF is 7% if basic salary is less than 8000, PF is 10% if basic salary between 8000 to 160000.Basic salary is available in employee table.

DECLARE

v\_da NUMBER(7,2);

v\_hra NUMBER(7,2);

v\_pf NUMBER(7,2);

v\_net\_salary NUMBER(7,2);

v\_year\_salary NUMBER(10,2);

v\_basic\_salary NUMBER(7,2);

BEGIN

FOR rec IN (SELECT SAL FROM emp)

LOOP

v\_basic\_salary := rec.sal;

v\_da := v\_basic\_salary \* 0.3;

v\_hra := v\_basic\_salary \* 0.1;

IF v\_basic\_salary < 8000 THEN

v\_pf := v\_basic\_salary \* 0.07;

ELSE

v\_pf := v\_basic\_salary \* 0.1;

END IF;

v\_net\_salary := v\_basic\_salary + v\_da + v\_hra - v\_pf;

v\_year\_salary := v\_net\_salary \* 12;

DBMS\_OUTPUT.PUT\_LINE('Employee Number: ' || rec.sal);

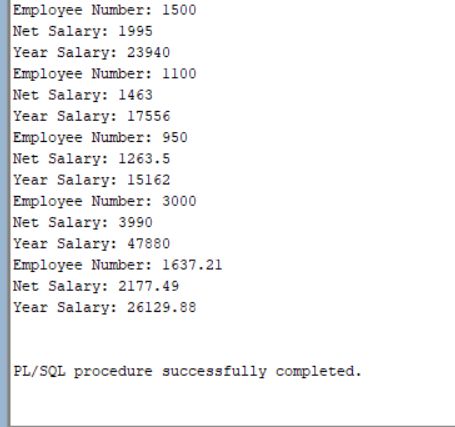
DBMS\_OUTPUT.PUT\_LINE('Net Salary: ' || v\_net\_salary);

DBMS\_OUTPUT.PUT\_LINE('Year Salary: ' || v\_year\_salary);

END LOOP;

END;

/



2.Develop PL/SQL functions to do the following:

i. Factorial of a Number.

declare

fac number :=1;

n number := &1;

begin

while n > 0 loop

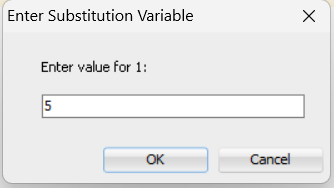
fac:=n\*fac;

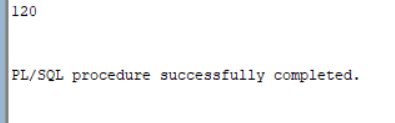
n:=n-1;

end loop;

dbms\_output.put\_line(fac);

end;





ii. n C r

CREATE OR REPLACE FUNCTION combination(n IN NUMBER, r IN NUMBER)

RETURN NUMBER IS

fact\_n NUMBER := 1;

fact\_r NUMBER := 1;

fact\_nr NUMBER := 1;

result NUMBER;

BEGIN

FOR i IN 1..n LOOP

fact\_n := fact\_n \* i;

END LOOP;

FOR i IN 1..r LOOP

fact\_r := fact\_r \* i;

END LOOP;

FOR i IN 1..(n-r) LOOP

fact\_nr := fact\_nr \* i;

END LOOP;

result := fact\_n / (fact\_r \* fact\_nr);

RETURN result;

END;

/

DECLARE

n NUMBER := 5;

r NUMBER := 2;

comb NUMBER;

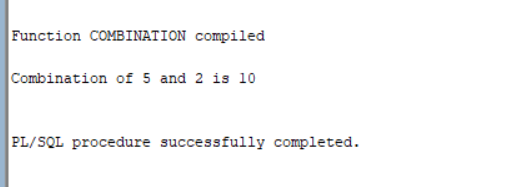
BEGIN

comb := combination(n, r);

DBMS\_OUTPUT.PUT\_LINE('Combination of ' || n || ' and ' || r || ' is ' || comb);

END;

/



iii. GCD of two numbers.

DECLARE

num1 INTEGER;

num2 INTEGER;

t INTEGER;

BEGIN

num1 := 8;

num2 := 48;

WHILE MOD(num2, num1) != 0 LOOP

t := MOD(num2, num1);

num2 := num1;

num1 := t;

END LOOP;

dbms\_output.Put\_line('GCD of '

||num1

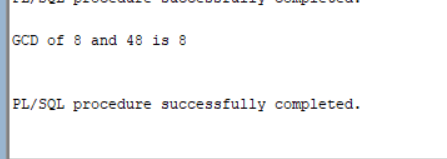
||' and '

||num2

||' is '

||num1);

END;



iv. Accept date of birth as ‘dd-mm-yyyy’ and sum all digits till you get a single digit number to show it as user’s lucky number.

DECLARE

l\_input VARCHAR2(20) := '31/01/1978';

l\_output INT;

BEGIN

LOOP

DBMS\_OUTPUT.PUT\_LINE('------------------');

DBMS\_OUTPUT.PUT\_LINE('l\_input=' || l\_input);

l\_output := 0;

FOR i IN 1 .. LENGTH(l\_input)

LOOP

IF SUBSTR(l\_input, i, 1) BETWEEN '0' AND '9' THEN

l\_output := l\_output + TO\_NUMBER(SUBSTR(l\_input, i, 1));

END IF;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('l\_output=' || l\_output);

EXIT WHEN l\_output < 10;

l\_input := TO\_CHAR(l\_output);

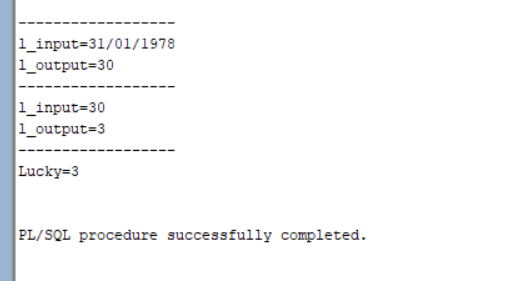
END LOOP;

DBMS\_OUTPUT.PUT\_LINE('------------------');

DBMS\_OUTPUT.PUT\_LINE('Lucky=' || l\_output);

END;

/



3.Develop math package to include all mathematical procedures and functions implemented in the above two questions and utilize them in other blocks and queries.

CREATE OR REPLACE PACKAGE math\_pkg AS

-- Procedure to check if a number is palindrome

PROCEDURE is\_palindrome (p\_num IN NUMBER, p\_result OUT BOOLEAN);

-- Function to find the sum of N natural numbers

FUNCTION sum\_natural\_numbers (p\_n IN NUMBER) RETURN NUMBER;

-- Procedure to generate Fibonacci series

PROCEDURE fibonacci\_series (p\_n IN NUMBER);

-- Procedure to check if a number is perfect

PROCEDURE is\_perfect\_number (p\_num IN NUMBER, p\_result OUT BOOLEAN);

-- Function to calculate net salary and year salary

FUNCTION calculate\_salary (p\_salary IN NUMBER) RETURN NUMBER;

-- Function to calculate the factorial of a number

FUNCTION factorial (p\_num IN NUMBER) RETURN NUMBER;

-- Function to calculate nCr

FUNCTION combination (p\_n IN NUMBER, p\_r IN NUMBER) RETURN NUMBER;

-- Function to calculate the GCD of two numbers

FUNCTION gcd (p\_num1 IN INTEGER, p\_num2 IN INTEGER) RETURN INTEGER;

-- Procedure to calculate the lucky number

PROCEDURE calculate\_lucky\_number (p\_dob IN DATE, p\_lucky\_number OUT NUMBER);

END math\_pkg;

/

