-- 1 PL/SQL Program To Add Two Numbers

DECLARE

num1 NUMBER := 10;

num2 NUMBER := 20;

sum NUMBER;

BEGIN

sum := num1 + num2;

DBMS\_OUTPUT.PUT\_LINE('Sum: ' || sum);

END;

-- 2 PL/SQL Program for Prime Number

DECLARE

num number :=5;

ans number:=0;

i number;

BEGIN

for i IN 2..num-1 LOOP

if mod(num,i)=0 then

ans := 1;

EXIT;

end if;

END LOOP;

if (ans=1) then

dbms\_output.put\_line(num || ' is Not Prime ');

else

dbms\_output.put\_line(num || ' is Prime ');

end if;

END;

-- 3 PL/SQL Program to Find Factorial of a Number

DECLARE

num NUMBER := 5;

factorial NUMBER := 1;

BEGIN

FOR i IN 1..num LOOP

factorial := factorial \* i;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('Factorial: ' || factorial);

END;

-- 4 PL/SQL Program to Print Table of a Number

DECLARE

num number :=2;

BEGIN

for i IN 1..10 LOOP

dbms\_output.put\_line(num ||'\*'||i||'='|| num\*i);

end LOOP;

end;

-- 5 PL/SQL Program for Reverse of a Number

DECLARE

num NUMBER := 1234;

reverse\_num NUMBER := 0;

remainder NUMBER;

BEGIN

WHILE num > 0 LOOP

remainder := MOD(num, 10);

reverse\_num := (reverse\_num \* 10) + remainder;

num := TRUNC(num / 10);

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('Reversed Number: ' || reverse\_num);

END;

-- 6 PL/SQL Program for Fibonacci Series

DECLARE

n NUMBER := 10;

a NUMBER := 0;

b NUMBER := 1;

c NUMBER;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(a);

DBMS\_OUTPUT.PUT\_LINE(b);

FOR i IN 3..n LOOP

c := a + b;

DBMS\_OUTPUT.PUT\_LINE(c);

a := b;

b := c;

END LOOP;

END;

-- 7 PL/SQL Program to Check Number is Odd or Even

DECLARE

num NUMBER := 4;

BEGIN

IF MOD(num, 2) = 0 THEN

DBMS\_OUTPUT.PUT\_LINE(num || ' is even.');

ELSE

DBMS\_OUTPUT.PUT\_LINE(num || ' is odd.');

END IF;

END;

-- 8 PL/SQL Program to Reverse a String

DECLARE

str VARCHAR2(100) := 'HELLO';

rev\_str VARCHAR2(100) := '';

BEGIN

FOR i IN REVERSE 1..LENGTH(str) LOOP

rev\_str := rev\_str || SUBSTR(str, i, 1);

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('Reversed String: ' || rev\_str);

END;

-- 9 PL/SQL Program for Palindrome Number

DECLARE

num NUMBER := 121;

original\_num NUMBER := num;

reverse\_num NUMBER := 0;

remainder NUMBER;

BEGIN

WHILE num > 0 LOOP

remainder := MOD(num, 10);

reverse\_num := (reverse\_num \* 10) + remainder;

num := TRUNC(num / 10);

END LOOP;

IF original\_num = reverse\_num THEN

DBMS\_OUTPUT.PUT\_LINE(original\_num || ' is a palindrome.');

ELSE

DBMS\_OUTPUT.PUT\_LINE(original\_num || ' is not a palindrome.');

END IF;

END;

-- 10 PL/SQL Program to Swap two Numbers

DECLARE

num1 NUMBER := 10;

num2 NUMBER := 20;

BEGIN

num1 := num1 + num2;

num2 := num1 - num2;

num1 := num1 - num2;

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

END;