**LAB 5**

**Ques 1 : Write a Pl/SQL function to find the factorial value of any number, the number should be passed as argument.**

CREATE OR REPLACE FUNCTION f(n IN NUMBER)

RETURN NUMBER

IS

BEGIN

IF n = 0 THEN

RETURN 1;

ELSE

RETURN n \* f(n - 1);

END IF;

END;

declare

x number;

ans number;

begin

x:=&x;

ans:=f(x);

DBMS\_output.put\_line(ans);

end;

**OUTPUT:**

old 15: x:=&x;

new 15: x:=5;

120

**Ques 2 : Create a Pl/SQL function to check whether the argument passed through function is prime or not.**

CREATE OR REPLACE PROCEDURE check\_prime(n IN NUMBER)

IS

prime BOOLEAN := TRUE;

BEGIN

IF n <= 1 THEN

prime := FALSE;

ELSE

FOR i IN 2..TRUNC(SQRT(n)) LOOP

IF MOD(n, i) = 0 THEN

prime := FALSE;

EXIT;

END IF;

END LOOP;

END IF;

IF prime THEN

DBMS\_OUTPUT.PUT\_LINE(n || ' is prime');

ELSE

DBMS\_OUTPUT.PUT\_LINE(n || ' is not prime');

END IF;

END;

declare

x number;

BEGIN

x:=&x;

check\_prime(x);

END;

**OUTPUT:**

old 4: x:=&x;

new 4: x:=5;

5 is prime

**Ques 3 : Write a procedure to upgrade the marks of a candidate whose roll number will be passed as argument.**

CREATE OR REPLACE PROCEDURE upgrade\_marks(p\_roll IN NUMBER)

IS

BEGIN

UPDATE student

SET marks = marks + 50

WHERE roll = p\_roll;

DBMS\_OUTPUT.PUT\_LINE('Marks upgraded for Roll ' || p\_roll);

END;

DECLARE

ROLL NUMBER;

BEGIN

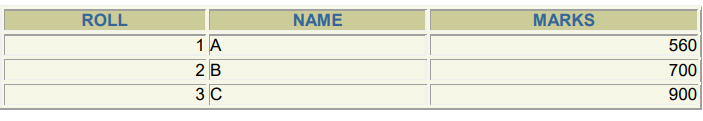
ROLL:=&ROLL;

upgrade\_marks(ROLL);

END;

**OUTPUT:**

**BEFORE**



**AFTER**

old 4: ROLL:=&ROLL;

new 4: ROLL:=2;

Marks upgraded for Roll 2

