AN_Oct28

```
set serveroutput on
set verify off
set feedback off
```

Question 1: Write a function to calculate the hcf of two numbers, and the lcm of two numbers.

```
--hcf_lcm_1.sql: Write a program in PL/SQL to calculate

--(i) hcf of 2 numbers,

--(ii) Use two functions



DECLARE

    A  NUMBER :=&A;

    B  NUMBER :=&B;

    H  NUMBER; --h = hc of 2 numbers

    LC NUMBER; --lc = lcm of 2 numbers

    FUNCTION HCF(

        A IN NUMBER,

        B IN NUMBER

    ) --'IN' means input-parameter

    RETURN NUMBER IS --`IS`means available in the main function

        R NUMBER;

        X NUMBER;

        Y NUMBER;

        H NUMBER;

    BEGIN

        X :=A;
```

```
        Y :=B;
        R := MOD(X, Y);
        WHILE(R != 0) LOOP
            X := Y;
            Y := R;
            R := MOD(X, Y);
        END LOOP;
        H := Y;
        RETURN H;
    END;
    FUNCTION LCM(
        A IN NUMBER,
        B IN NUMBER
    ) RETURN NUMBER IS
        LC NUMBER;
        X  NUMBER;
        Y  NUMBER;
    BEGIN
        X := A;
        Y := B;
        LC := A*B/HCF(A, B);
        RETURN LC;
    END;
 --main program starts
BEGIN
    H := HCF(A, B);
```

```
    LC := LCM(A, B);
    DBMS_OUTPUT.PUT_LINE('HCF('
        ||A
        ||','
        ||B
        ||')= '
        ||H);
    DBMS_OUTPUT.PUT_LINE('LCM('
        ||A
        ||','
        ||B
        ||') ='
        ||LC);
END;
/
```

The same thing can be done with creating 2 functions and calling them in a 3rd one as well. Create a function defining the HCF calculation.

```
CREATE OR REPLACE FUNCTION HCF(
    A IN NUMBER,
    B IN NUMBER
) RETURN NUMBER AS
    R NUMBER;
    X NUMBER;
    Y NUMBER;
    H NUMBER;
BEGIN
    X :=A;
```

```
    Y :=B;

    R := MOD(X, Y);

    WHILE(R != 0) LOOP

        X := Y;

        Y := R;

        R := MOD(X, Y);

    END LOOP;

    H := Y;

    RETURN H;

END;
/
```

Now write another function defining the LCM calculation.

```
--lcm_function_1.sql: Write a function to calculate lcm of 2 numbers.



CREATE OR REPLACE FUNCTION LCM(

    A IN NUMBER,

    B IN NUMBER

) RETURN NUMBER AS

    LC NUMBER;

BEGIN

    LC := A*B/HCF(A, B);

    RETURN LC;

END;
/
```

Now write the main function calling both the avove functions.

```
--hcf_lcm_main_1.sql: Write a program in PL/SQL

--to call 2 user defined functions to calculate HCF, LCM of 2 numbers.
```

```
DECLARE

    A  NUMBER := &A;

    B  NUMBER := &B;

    H  NUMBER;

    LC NUMBER;

BEGIN

    H := HCF(A, B);

    LC := LCM(A, B);

    DBMS_OUTPUT.PUT_LINE('HCF('

        ||A

        ||','

        ||B

        ||')= '

        ||H);

    DBMS_OUTPUT.PUT_LINE('LCM('

        ||A

        ||','

        ||B

        ||')= '

        ||LC);

END;
/
```

Question 2: Write a function to test whether a number is prime/not prime. `FUNCTION NAME IS-PRIME(X IN NUMBER)`, `RETURN VALUE = 1/0`.

The following function is the function for calculating prime or not.

```
--hcf_lcm_1.sql: Write a program in PL/SQL to calculate

--(i) hcf of 2 numbers,

--(ii) Use two functions



DECLARE

    A  NUMBER :=&A;

    B  NUMBER :=&B;

    H  NUMBER; --h = hc of 2 numbers

    LC NUMBER; --lc = lcm of 2 numbers

    FUNCTION HCF(

        A IN NUMBER,

        B IN NUMBER

    ) --'IN' means input-parameter

    RETURN NUMBER IS --`IS`means available in the main function

        R NUMBER;

        X NUMBER;

        Y NUMBER;

        H NUMBER;

    BEGIN

        X :=A;

        Y :=B;

        R := MOD(X, Y);

        WHILE(R != 0) LOOP

            X := Y;

            Y := R;

            R := MOD(X, Y);
```

```
        END LOOP;
        H := Y;
        RETURN H;
    END;
    FUNCTION LCM(
        A IN NUMBER,
        B IN NUMBER
    ) RETURN NUMBER IS
        LC NUMBER;
        X  NUMBER;
        Y  NUMBER;
    BEGIN
        X := A;
        Y := B;
        LC := A*B/HCF(A, B);
        RETURN LC;
    END;
 --main program starts
BEGIN
    H := HCF(A, B);
    LC := LCM(A, B);
    DBMS_OUTPUT.PUT_LINE('HCF('
        ||A
        ||','
        ||B
        ||')= '
        ||H);
```

```
    DBMS_OUTPUT.PUT_LINE('LCM('
        ||A
        ||','
        ||B
        ||') ='
        ||LC);
END;
/
```

Now, write a main function and call the function above to get output.

```
```