

## Práctica 1

Vamos a crear la tabla de multiplicar del 1 al 10, con los tres tipos de bucles: LOOP, WHILE y FOR

```
-- Vamos a crear la tabla de multiplicar del 1 al 10, con los tres tipos de bucles: LOOP, WHILE y FOR
SET SERVEROUTPUT ON
DECLARE
CONT NUMBER := 1;
CONT2 NUMBER := 1;
RESULTADO NUMBER := 0;
BEGIN
    DBMS_OUTPUT.PUT_LINE('TABLAS DE MULTIPLICAR CON EL LOOP');
    LOOP
        DBMS_OUTPUT.PUT_LINE('TABLA DEL -----' || CONT );
        LOOP
            RESULTADO := CONT * CONT2;
            DBMS_OUTPUT.PUT_LINE(CONT || ' * ' || CONT2 || ' = ' || RESULTADO);
            CONT2 := CONT2 + 1;
            EXIT WHEN CONT2 > 10;
        END LOOP;
        CONT2 := 1;
        CONT := CONT + 1;
        EXIT WHEN CONT > 10;
    END LOOP;
```

```
END LOOP;
    DBMS_OUTPUT.PUT_LINE('TABLAS DE MULTIPLICAR CON EL WHILE');
    CONT:=1;
    CONT2 := 1;
    WHILE CONT <= 10 LOOP
        DBMS_OUTPUT.PUT_LINE('TABLA DEL -----' || CONT );
        WHILE CONT2 <= 10 LOOP
            DBMS_OUTPUT.PUT_LINE(CONT || ' * ' || CONT2 || ' = ' || RESULTADO);
            CONT2 := CONT2 + 1;
        END LOOP;
        CONT2:=1;
        CONT := CONT + 1;
    END LOOP;
```

```
END LOOP;  
DBMS_OUTPUT.PUT_LINE('TABLAS DE MULTIPLICAR CON EL FOR');  
CONT := 1;  
CONT2:= 1;  
FOR CONT IN 1..10 LOOP  
    DBMS_OUTPUT.PUT_LINE('TABLA DEL -----'|| CONT );  
    FOR CONT2 IN 1..10 LOOP  
        DBMS_OUTPUT.PUT_LINE(CONT || ' * ' || CONT2 || ' = ' || RESULTADO);  
    END LOOP;  
    CONT2:=1;  
END LOOP;  
END;
```

Salida de Script x

Tarea terminada en 0,057 segundos

```
TABLA DEL -----10  
10 * 1 = 100  
10 * 2 = 100  
10 * 3 = 100  
10 * 4 = 100  
10 * 5 = 100  
10 * 6 = 100  
10 * 7 = 100  
10 * 8 = 100  
10 * 9 = 100  
10 * 10 = 100
```

```
Salida de Script x
Tarea terminada en 0,074 segundos

TABLAS DE MULTIPLICAR CON EL LOOP
TABLA DEL -----1
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
1 * 4 = 4
1 * 5 = 5
1 * 6 = 6
1 * 7 = 7
1 * 8 = 8
1 * 9 = 9
1 * 10 = 10
TABLA DEL -----2
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
2 * 10 = 20
TABLA DEL -----3
3 * 1 = 3
```

```
10 * 2 = 20
10 * 3 = 30
10 * 4 = 40
10 * 5 = 50
10 * 6 = 60
10 * 7 = 70
10 * 8 = 80
10 * 9 = 90
10 * 10 = 100
TABLAS DE MULTIPLICAR CON EL WHILE
TABLA DEL -----1
1 * 1 = 100
1 * 2 = 100
1 * 3 = 100
1 * 4 = 100
1 * 5 = 100
1 * 6 = 100
1 * 7 = 100
1 * 8 = 100
1 * 9 = 100
1 * 10 = 100
TABLA DEL -----2
2 * 1 = 100
2 * 2 = 100
2 * 3 = 100
2 * 4 = 100
```

```
Salida de Script x
Tarea terminada en 0,074 segundos

10 * 8 = 100
10 * 9 = 100
10 * 10 = 100
TABLAS DE MULTIPLICAR CON EL FOR
TABLA DEL -----1
1 * 1 = 100
1 * 2 = 100
1 * 3 = 100
1 * 4 = 100
1 * 5 = 100
1 * 6 = 100
1 * 7 = 100
1 * 8 = 100
1 * 9 = 100
1 * 10 = 100
TABLA DEL -----2
2 * 1 = 100
2 * 2 = 100
2 * 3 = 100
2 * 4 = 100
2 * 5 = 100
2 * 6 = 100
2 * 7 = 100
2 * 8 = 100
2 * 9 = 100
```

## Práctica 2

Crear una variable llamada TEXTO de tipo VARCHAR2(100).

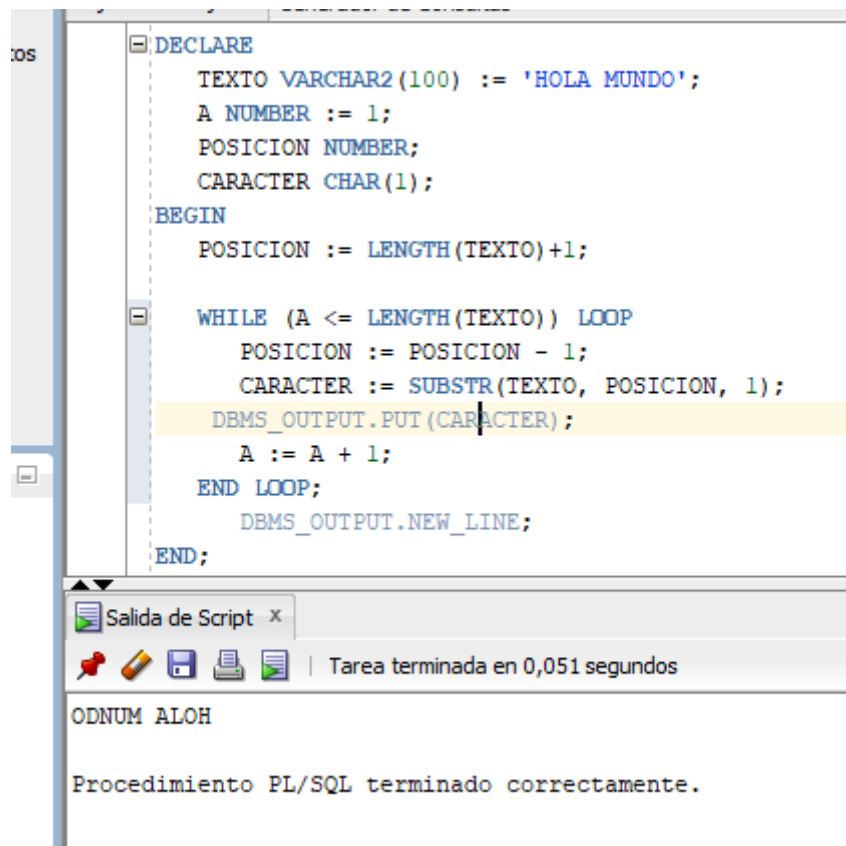
Poner alguna frase

Mediante un bucle, escribir la frase al revés, Usamos el bucle **WHILE**

### INVESTIGAR EL LENGTH

PABLO

OLBAP



```
DECLARE
    TEXTO VARCHAR2(100) := 'HOLA MUNDO';
    A NUMBER := 1;
    POSICION NUMBER;
    CARACTER CHAR(1);
BEGIN
    POSICION := LENGTH(TEXTO)+1;

    WHILE (A <= LENGTH(TEXTO)) LOOP
        POSICION := POSICION - 1;
        CARACTER := SUBSTR(TEXTO, POSICION, 1);
        DBMS_OUTPUT.PUT(CARACTER);
        A := A + 1;
    END LOOP;
    DBMS_OUTPUT.NEW_LINE;
END;
```

Salida de Script x

Tarea terminada en 0,051 segundos

ODNUM ALOH

Procedimiento PL/SQL terminado correctamente.

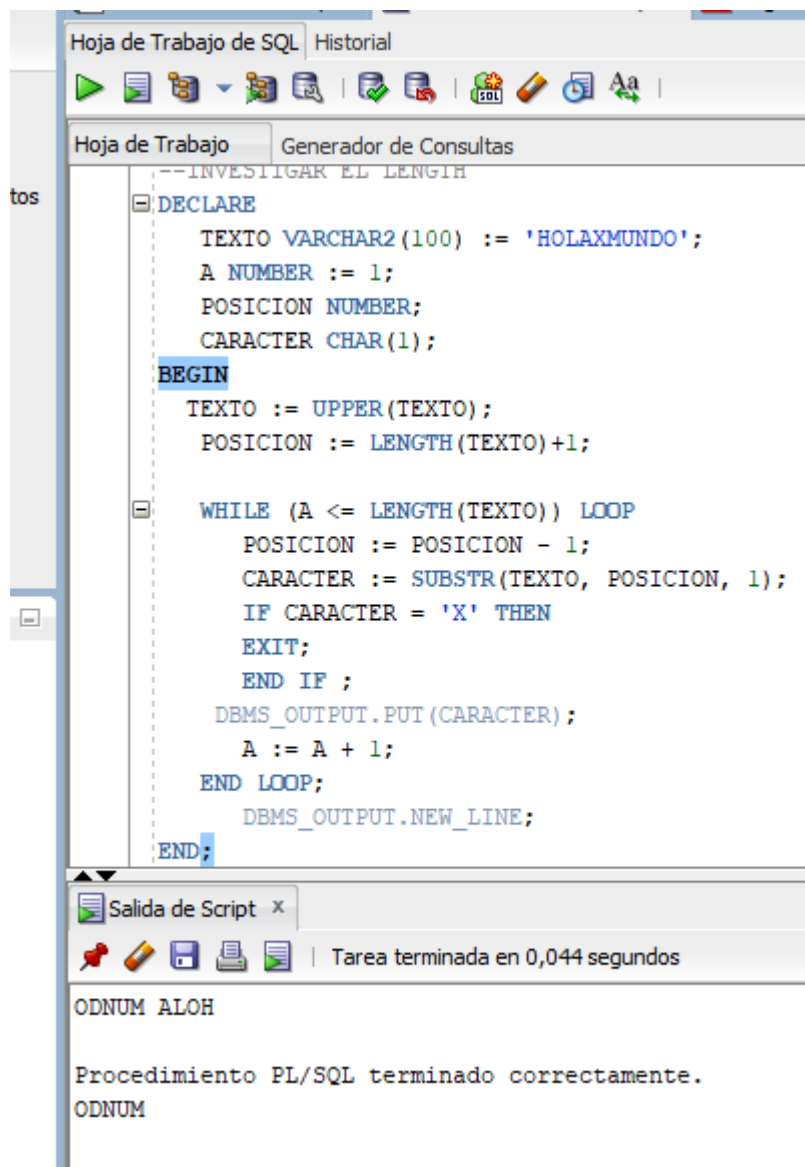
## Práctica 3

Usando la práctica anterior, si en el texto aparece el carácter "x" debe salir del bucle. Es igual en mayúsculas o minúsculas.

Debemos usar la cláusula EXIT.

PABLOXURRUTIA

AITURRUX



The screenshot shows the Oracle SQL Developer interface. The main window is titled 'Hoja de Trabajo de SQL' and contains a script titled '--INVESTIGAR EL LENGTH'. The script is as follows:

```
--INVESTIGAR EL LENGTH
DECLARE
  TEXTO VARCHAR2(100) := 'HOLAXMUNDO';
  A NUMBER := 1;
  POSICION NUMBER;
  CARACTER CHAR(1);
BEGIN
  TEXTO := UPPER(TEXTO);
  POSICION := LENGTH(TEXTO)+1;

  WHILE (A <= LENGTH(TEXTO)) LOOP
    POSICION := POSICION - 1;
    CARACTER := SUBSTR(TEXTO, POSICION, 1);
    IF CARACTER = 'X' THEN
      EXIT;
    END IF ;
    DBMS_OUTPUT.PUT(CARACTER);
    A := A + 1;
  END LOOP;
  DBMS_OUTPUT.NEW_LINE;
END;
```

Below the script editor, the 'Salida de Script' window shows the output of the execution:

```
ODNUM ALOH

Procedimiento PL/SQL terminado correctamente.
ODNUM
```

#### Práctica 4

Debemos crear una variable llamada NOMBRE

Debemos pintar tantos asteriscos como letras tenga el nombre. Usamos un bucle **FOR**

Por ejemplo, Alberto --> \*\*\*\*\*

**INVESTIGAR EL LENGTH**

```
tos
--POR EJEMPLO, ALBERTO --> *****
--INVESTIGAR EL LENGTH

SET SERVEROUTPUT ON

DECLARE
  NOMBRE VARCHAR2(10) := 'JUAN PEREZ';
BEGIN
  FOR A IN 1..LENGTH(NOMBRE) LOOP
    DBMS_OUTPUT.PUT('*');
  END LOOP;
  DBMS_OUTPUT.NEW_LINE;
END;
```

Salida de Script x

Tarea terminada en 0,062 segundos

\*\*\*\*\*

Procedimiento PL/SQL terminado correctamente.

## Práctica 5

Creamos dos variables numéricas, "inicio y fin"

Las inicializamos con algún valor:

Debemos sacar los números que sean múltiplos de 4 de ese rango





### INVESTIGAR EL MOD



```
PS  --INVESTIGAR EL MOD
    SET SERVEROUTPUT ON
    DECLARE
    INICIO NUMBER := 1;
    FIN NUMBER :=20;

    BEGIN
    FOR A IN INICIO..FIN LOOP
        IF MOD (A,4)=0 THEN
            DBMS_OUTPUT.PUT_LINE(A || ' ES MULTIPLO DE 4 ');
        END IF ;
    END LOOP;
    END;
```

Salida de Script x

 | Tarea terminada en 0,064 segundos

```
4 ES MULTIPLO DE 4
8 ES MULTIPLO DE 4
12 ES MULTIPLO DE 4
16 ES MULTIPLO DE 4
20 ES MULTIPLO DE 4

Procedimiento PL/SQL terminado correctamente.
```