

ESTRUCTURAS DE CONTROL

IF

1. PRÁCTICA 1

- Debemos hacer un bloque PL/SQL anónimo, donde declaramos una variable NUMBER y la ponemos algún valor.
- Debe indicar si el número es PAR o IMPAR. Es decir debemos usar IF..... ELSE para hacer el ejercicio
- Como pista, recuerda que hay una función en SQL denominada MOD, que permite averiguar el resto de una división.
- Por ejemplo MOD(10,4) nos devuelve el resto de dividir 10 por 4.

```
DECLARE
numerito number(5):=99998;
BEGIN
IF MOD(numerito,2)=0 then
DBMS_OUTPUT.PUT_LINE('PAR');
else
DBMS_OUTPUT.PUT_LINE('impar');
end if;
END;
```

impar

Procedimiento PL/SQL terminado correctamente.

CASE

Práctica comando CASE

- Vamos a crear una variable denominada "usuario", de tipo VARCHAR2(40)
- Vamos a poner dentro el nombre del usuario con el que nos hemos conectado. Para saberlo, usamos la función USER de Oracle que devuelve el nombre del usuario con el que estamos conectados (Recuerda que en Oracle no hace falta poner paréntesis si una función no tiene argumentos) usuario:=user
- Luego hacemos un CASE para que nos pinte un mensaje del estilo: o Si el usuario es SYS ponemos el mensaje "Eres superadministrador" o Si el usuario es SYSTEM ponemos el mensaje "Eres un administrador normal" o Si el usuario es HR ponemos el mensaje "Eres de Recursos humanos" o Cualquier otro usuario ponemos "usuario no autorizado"

```
DECLARE
USUARIO varchar2(40):=user;
BEGIN
case USUARIO
when 'SYS' then DBMS_OUTPUT.PUT_LINE('Eres superadministrador');
when 'SYSTEM' then DBMS_OUTPUT.PUT_LINE('Eres un administrador norma');
when 'HR' then DBMS_OUTPUT.PUT_LINE('Eres de Recursos humanos');
else DBMS_OUTPUT.PUT_LINE('Eressuperadministrador');
END CASE;
END;
```

Eres de Recursos humanos

Procedimiento PL/SQL terminado correctamente.

BUCLES

Práctica 1

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

LOOP:

```
DECLARE
i number(3):=1;
j number(3):=1;
BEGIN
LOOP
DBMS_OUTPUT.PUT_LINE('TABLA DEL ' || i || ':');
j:=1;
LOOP
DBMS_OUTPUT.PUT_LINE('> ' || i || ' x ' || j || ' = ' || j*i);
j:=j+1;
EXIT WHEN j=11;
END LOOP;
i:=i+1;
EXIT WHEN i = 11;
DBMS_OUTPUT.PUT_LINE('');
END LOOP;
END;
```

WHILE

```

/***** WHILE *****/
DECLARE
i number(3):=1;
j number(3):=1;
BEGIN
WHILE i<=10 LOOP
DBMS_OUTPUT.PUT_LINE('TABLA DEL ' || i || ':');
j:=1;
WHILE j<=10 LOOP
DBMS_OUTPUT.PUT_LINE('> ' || i || ' x ' || j || ' = ' || j*i);
j:=j+1;
END LOOP;
i:=i+1;
DBMS_OUTPUT.PUT_LINE('');
END LOOP;
END;
```

FOR:

```

/***** FOR *****/
DECLARE
i number(3):=1;
j number(3):=1;
BEGIN
FOR i in 1..10 LOOP
DBMS_OUTPUT.PUT_LINE('TABLA DEL ' || i || ':');
j:=1;
FOR j in 1..10 LOOP
DBMS_OUTPUT.PUT_LINE('> ' || i || ' x ' || j || ' = ' || j*i);
END LOOP;
DBMS_OUTPUT.PUT_LINE('');
END LOOP;
END;
```

```
TABLA DEL 1:
> 1 x 1 = 1
> 1 x 2 = 2
> 1 x 3 = 3
> 1 x 4 = 4
> 1 x 5 = 5
> 1 x 6 = 6
> 1 x 7 = 7
> 1 x 8 = 8
> 1 x 9 = 9
> 1 x 10 = 10
```

```
TABLA DEL 2:
> 2 x 1 = 2
> 2 x 2 = 4
> 2 x 3 = 6
> 2 x 4 = 8
> 2 x 5 = 10
> 2 x 6 = 12
> 2 x 7 = 14
> 2 x 8 = 16
> 2 x 9 = 18
> 2 x 10 = 20
```

```
TABLA DEL 3:
> 3 x 1 = 3
> 3 x 2 = 6
> 3 x 3 = 9
> 3 x 4 = 12
> 3 x 5 = 15
> 3 x 6 = 18
> 3 x 7 = 21
> 3 x 8 = 24
> 3 x 9 = 27
> 3 x 10 = 30
```

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

```
DECLARE
texto varchar(99):='parangaracutirimicuaro espiralidoso';
invertido varchar(99):='';
i integer:= length(texto);
BEGIN
while i>0 LOOP
invertido:=invertido || SUBSTR(TEXTO,i,1);
i:=i-1;
END LOOP;
DBMS_OUTPUT.PUT_LINE(invertido);
END;
```

```
osodilaripse oraucimiritucaragnarap

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.

```
DECLARE
texto varchar(99):='la bahia de Xomichimilco';
invertido varchar(99):='';
i integer:= length(texto);
BEGIN
while i>0 LOOP
invertido:=invertido || SUBSTR(TEXTTO,i,1);
if lower(SUBSTR(TEXTTO,i,1))='x' then
exit;
end if;
i:=i-1;
END LOOP;
DBMS_OUTPUT.PUT_LINE(invertido);
END;

oclimihcimoX

Procedimiento PL/SQL terminado correctamente.
```

Práctica 4

- Debemos crear una variable llamada NOMBRE
- Debemos pintar tantos asteriscos como letras tenga el nombre.

Usamos un bucle FOR

- Por ejemplo Alberto → *****
- O por ejemplo Pedro → *****

```
DECLARE
nombre varchar(99):='Hubert Blaine Wolfeschlegelsteinhausenbergerdorff Sr.';
asterisco varchar(99):='';
i integer:= length(nombre);
BEGIN
while i>0 LOOP
asterisco:=asterisco|| '*';
i:=i-1;
END LOOP;
DBMS_OUTPUT.PUT_LINE(asterisco);
END;
```

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*

```
DECLARE
inicio number(20):=77;
fin number(20):=974;
BEGIN
DBMS_OUTPUT.PUT_LINE('MULTIPLOS DE 4 ENTRE ' || inicio || ' y ' || fin);
while inicio<=fin LOOP
if MOD(inicio,4)=0 then
DBMS_OUTPUT.PUT_LINE('> '||inicio);
end if;
inicio := inicio +1;
END LOOP;
END;
```

> 80
> 84
> 88
> 92
> 96
> 100
> 104
> 108
> 112
> 116
> 120
> 124
> 128