

# Prácticas de INDEX BY TABLES

## PL/SQL 12c-18c avanzado

**NOTA:** Aunque siempre pongo las soluciones, os recomiendo que primero intentéis hacer el ejemplo por vosotros mismos y luego lo comparéis con el mío. ¡¡¡En muchas ocasiones, el mismo proceso se puede hacer de varias formas!!!!

### INDEX BY TABLES

1. Crear un array asociativo que tenga los siguientes valores. Visualizarlos mediante un bucle

Clave	Valor
1	INFORMATICA
2	MATEMATICAS
3	BIOLOGIA
4	LITERATURA

Ejemplo:

```
SET SERVEROUTPUT ON
DECLARE
  TYPE ARRAY1 IS TABLE OF VARCHAR2(50) INDEX BY PLS_INTEGER;
  DATOS ARRAY1;
BEGIN
  DATOS(1):='INFORMATICA';
  DATOS(2):='MATEMÁTICAS';
  DATOS(3):='BIOLOGÍA';
  DATOS(4):='LITERATURA';
  FOR I IN 1..4 LOOP
    DBMS_OUTPUT.PUT_LINE(DATOS(I));
  END LOOP;
END;
/
```

2. Crear un procedimiento donde vamos a cargar en un array asociativo los datos de los empleados de la tabla employees que pertenezcan a un determinado departamento.

- El departamento lo pasamos como argumento del procedimiento.
- La clave será un número desde el 1 hasta el total de empleados cargados.
- Por último visualizamos el nombre y apellidos de los empleados.

#### Ejemplo

```
CREATE OR REPLACE PROCEDURE PRUEBA_ARRAY1(CODIGO NUMBER)
IS
  --CREAMOS TL TIPO INDEX BY TABLE
  TYPE V1 IS TABLE OF EMPLOYEES%ROWTYPE INDEX BY PLS_INTEGER;
  --CREAMOS UNA VARIABLE DE ESE TIPO PARA ALBERGAR LOS DATOS DE
  LOS EMPLEADOS
  TABLA V1;
  --INDICE PARA RECORRER EL ARRAY
  INDICE NUMBER:=1;
  --CURSOR PARA CARGAR LOS DATOS
  CURSOR DATOS IS SELECT * FROM EMPLOYEES WHERE
  DEPARTMENT_ID=CODIGO;
BEGIN
  --RECORREMOS EL CURSOR
  FOR I IN DATOS LOOP
    -- POR CADA EMPLEADO GUARDAMOS SUS DATOS EN EL ARRAY
    TABLA(INDICE):=I;
    --AUMENTAMOS EN 1 EL ÍNDICE
    INDICE:=INDICE+1;
  END LOOP;

  --RECORREMOS EL ARRAY VISUALIZANDO LOS DATOS DE LOS EMPLEADOS
  FOR X IN 1..INDICE-1 LOOP
    DBMS_OUTPUT.PUT_LINE(TABLA(X).FIRST_NAME||
    '||TABLA(X).LAST_NAME);
  END LOOP;
END;
/

EXECUTE PRUEBA_ARRAY1(100);
```

3. Crear una copia del procedimiento anterior, pero en este caso:
  - Hacemos un BULK COLLECT a la hora de cargar los datos
  - Visualizamos también el nombre y apellido de cada empleado para comprobar que está OK. Debemos usar la función COUNT para saber el tamaño del array

#### Ejemplo

```
CREATE OR REPLACE PROCEDURE PRUEBA_ARRAY2(CODIGO NUMBER)
```

```

IS
--CREAMOS TL TIPO INDEX BY TABLE
TYPE V1 IS TABLE OF EMPLOYEES%ROWTYPE INDEX BY PLS_INTEGER;
--CREAMOS UNA VARIABLE DE ESE TIPO PARA ALBERGAR LOS DATOS DE
LOS EMPLEADOS
TABLA V1;
--INDICE PARA RECORRER EL ARRAY

BEGIN
--HACEMOS EL BULK COLLECT
SELECT * BULK COLLECT INTO TABLA FROM EMPLOYEES WHERE
DEPARTMENT_ID=CODIGO;

--RECORREMOS EL ARRAY VISUALIZANDO LOS DATOS DE LOS EMPLEADOS
FOR X IN 1..TABLA.COUNT LOOP
    DBMS_OUTPUT.PUT_LINE(TABLA(X).FIRST_NAME||
' || TABLA(X).LAST_NAME);
END LOOP;
END;
/

EXECUTE PRUEBA_ARRAY2(100);
    
```

4. Modificar el procedimiento anterior para que visualice
- El nombre y salario del primer empleado
  - El nombre y salario del ultimo empleado
  - Comprobar si tenemos un empleado en el índice 200. Si no es así devolver un mensaje del tipo "Empleado inexistente"
  - Eliminar del ARRAY los empleados que ganen más de 5000 dolares. Visualizar el número de empleados antes y después del proceso.
  - Por ultimo,visualizar el array con los empleados que han quedado. Nota, hay que usar el método NEXT para que funcione

```

CREATE OR REPLACE PROCEDURE PRUEBA_ARRAY2(CODIGO NUMBER)
IS
--CREAMOS TL TIPO INDEX BY TABLE
TYPE V1 IS TABLE OF EMPLOYEES%ROWTYPE INDEX BY PLS_INTEGER;
--CREAMOS UNA VARIABLE DE ESE TIPO PARA ALBERGAR LOS DATOS DE
LOS EMPLEADOS
TABLA V1;
--INDICE PARA RECORRER EL ARRAY
INDICE NUMBER;
BEGIN
--HACEMOS EL BULK COLLECT
    
```

```

SELECT * BULK COLLECT INTO TABLA FROM EMPLOYEES WHERE
DEPARTMENT_ID=CODIGO;

--RECORREMOS EL ARRAY VISUALIZANDO LOS DATOS DE LOS EMPLEADOS
FOR X IN 1..TABLA.COUNT LOOP
    DBMS_OUTPUT.PUT_LINE(TABLA(X).FIRST_NAME||
'||TABLA(X).LAST_NAME);
END LOOP;

--NOMBRE Y SALARIO DEL PRIMER EMPLEADO
DBMS_OUTPUT.PUT_LINE('PRIMER EMPLEADO');
DBMS_OUTPUT.PUT_LINE(TABLA(TABLA.FIRST()).FIRST_NAME||
'||TABLA(TABLA.FIRST()).SALARY);

--NOMBRE Y SALARIO DEL ULTIMO EMPLEADO
DBMS_OUTPUT.PUT_LINE('ÚLTIMO EMPLEADO');
DBMS_OUTPUT.PUT_LINE(TABLA(TABLA.LAST()).FIRST_NAME||
'||TABLA(TABLA.LAST()).SALARY);

--COMPROBAR SI EXISTE EL EMPLEADO EN EL ÍNDICE 200
IF NOT TABLA.EXISTS(200) THEN
    DBMS_OUTPUT.PUT_LINE('EMPLEADO INEXISTENTE');
END IF;

--CONTAR EL NUMERO DE EMPLEADOS
DBMS_OUTPUT.PUT_LINE('ANTES DEL BORRADO HAY:'||TABLA.COUNT);

--BORRAR DEL ARRAY LOS QUE GANEN MÁS DE 5000
FOR X IN 1..TABLA.COUNT LOOP
    IF TABLA(X).SALARY> 5000 THEN
        TABLA.DELETE(X);
    END IF;
END LOOP;

--CONTAR DE NUEVO LOS EMPLEADOS
DBMS_OUTPUT.PUT_LINE('DESPUES DEL BORRADO HAY:'||TABLA.COUNT);

--VISUALIZARLOS. CUIDADO CON LOS HUECOS DEJADOS POR EL DELETE
FOR X IN TABLA.FIRST..TABLA.COUNT LOOP
    IF TABLA.EXISTS(X) THEN

```

```
        DBMS_OUTPUT.PUT_LINE(TABLA(X).FIRST_NAME||  
'||TABLA(X).LAST_NAME);  
    END IF;  
    END LOOP;  
END;  
/  
  
EXECUTE PRUEBA_ARRAY2(30);
```