

REALIZAR UN BLOQUE ANÓNIMO (SIN CURSORES) QUE MUESTRE EL AÑO EN QUE SE UNIÓ EL MÁXIMO NÚMERO DE EMPLEADOS Y MOSTRAR CUANTOS EMPLEADOS POR MES SE UNIERON DICHO AÑO

```
DECLARE
  V_MAX_YEAR  NUMBER;
BEGIN

  SELECT HIRE_YEAR
  INTO V_MAX_YEAR
  FROM (
    SELECT EXTRACT(YEAR FROM HIRE_DATE) AS HIRE_YEAR,
           COUNT(*) AS TOTAL
    FROM EMPLOYEES
    GROUP BY EXTRACT(YEAR FROM HIRE_DATE)
    ORDER BY TOTAL DESC
  )
  WHERE ROWNUM = 1;

  DBMS_OUTPUT.PUT_LINE('ANO CON MAS CONTRATACION : ' || V_MAX_YEAR);
  DBMS_OUTPUT.PUT_LINE('MES          NUMERO DE EMPLEADOS');
  DBMS_OUTPUT.PUT_LINE('-----');

  FOR R IN (
    SELECT EXTRACT(MONTH FROM HIRE_DATE) AS MES,
           COUNT(*) AS CANTIDAD
    FROM EMPLOYEES
    WHERE EXTRACT(YEAR FROM HIRE_DATE) = V_MAX_YEAR
    GROUP BY EXTRACT(MONTH FROM HIRE_DATE)
    ORDER BY MES
  ) LOOP
    DBMS_OUTPUT.PUT_LINE(
      LPAD(R.MES, 2) || CHR(9) || CHR(9) || R.CANTIDAD
    );
  END LOOP;
END;
```

)
WHERE ROWNUM = 1;

Results	Explain	Describe	Saved SQL	History
ANO CON MAS CONTRATACION : 1997				
MES	NUMERO DE EMPLEADOS			

1	3			
2	2			
3	6			
4	1			
6	2			
7	2			
8	4			
9	2			
10	3			
11	1			
12	2			

REALIZAR UN BLOQUE ANÓNIMO CON UN CURSOR EXPLICITO QUE MUESTRE LOS EMPLEADOS Y SUS MANAGERS. SI UN EMPLEADO NO TIENE MANAGER, MOSTRARÁ "SIN MANAGER"

DECLARE

```
CURSOR EMP_MGR_CURSOR IS
  SELECT E.FIRST_NAME AS EMPLEADO,
         M.FIRST_NAME AS MANAGER
  FROM EMPLOYEES E
  LEFT JOIN EMPLOYEES M ON E.MANAGER_ID = M.EMPLOYEE_ID;
```

```
V_EMPLEADO EMPLOYEES.FIRST_NAME%TYPE;
V_MANAGER EMPLOYEES.FIRST_NAME%TYPE;
```

BEGIN

```
DBMS_OUTPUT.PUT_LINE('EMPLEADO | MANAGER');
DBMS_OUTPUT.PUT_LINE('-----');
```

OPEN EMP_MGR_CURSOR;

LOOP

```
  FETCH EMP_MGR_CURSOR INTO V_EMPLEADO, V_MANAGER;
  EXIT WHEN EMP_MGR_CURSOR%NOTFOUND;
```

IF V_MANAGER IS NULL THEN

```
  DBMS_OUTPUT.PUT_LINE(RPAD(V_EMPLEADO, 13) || ' | SIN MANAGER');
```

ELSE

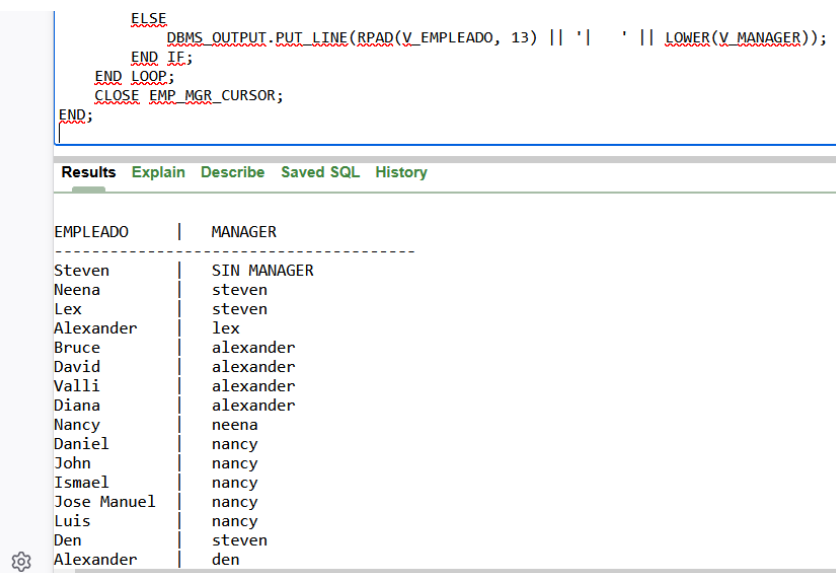
```
  DBMS_OUTPUT.PUT_LINE(RPAD(V_EMPLEADO, 13) || ' | ' || LOWER(V_MANAGER));
```

END IF;

END LOOP;

CLOSE EMP_MGR_CURSOR;

END;



```
ELSE
  DBMS_OUTPUT.PUT_LINE(RPAD(V_EMPLEADO, 13) || ' | ' || LOWER(V_MANAGER));
END IF;
END LOOP;
CLOSE EMP_MGR_CURSOR;
END;
```

EMPLEADO	MANAGER
Steven	SIN MANAGER
Neena	steven
Lex	steven
Alexander	lex
Bruce	alexander
David	alexander
Valli	alexander
Diana	alexander
Nancy	neena
Daniel	nancy
John	nancy
Ismael	nancy
Jose Manuel	nancy
Luis	nancy
Den	steven
Alexander	den

REALIZAR UN BLOQUE ANÓNIMO CON UN CURSOR IMPLÍCITO PARA AVERIGUAR CUANTOS EMPLEADOS RESIDEN EN CADA CIUDAD, POR CADA OFICIO. LAS CIUDADES SON: SEATTLE, LONDON OXFORD, TORONTO Y MUNICH

```
BEGIN
  DBMS_OUTPUT.PUT_LINE(RPAD('CIUDAD', 20) || RPAD('OFICIO', 40) || 'TOTAL DE EMPLEADOS');
  DBMS_OUTPUT.PUT_LINE(RPAD('-', 90, '-'));

  FOR REC IN (
    SELECT
      UPPER(L.CITY) AS CIUDAD,
      INITCAP(J.JOB_TITLE) AS OFICIO,
      COUNT(*) AS TOTAL_EMPLEADOS
    FROM EMPLOYEES E
    JOIN DEPARTMENTS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
    JOIN LOCATIONS L ON D.LOCATION_ID = L.LOCATION_ID
    JOIN JOBS J ON E.JOB_ID = J.JOB_ID
    WHERE UPPER(L.CITY) IN ('SEATTLE', 'LONDON', 'OXFORD', 'TORONTO', 'MUNICH')
    GROUP BY L.CITY, J.JOB_TITLE
    ORDER BY L.CITY, J.JOB_TITLE
  ) LOOP
    DBMS_OUTPUT.PUT_LINE(
      RPAD(REC.CIUDAD, 20) || RPAD(REC.OFICIO, 40) || REC.TOTAL_EMPLEADOS
    );
  END LOOP;
```

<pre> COUNT(*) AS total_empleados FROM employees e JOIN departments d ON e.department_id = d.department_id JOIN locations l ON d.location_id = l.location_id JOIN jobs j ON e.job_id = j.job_id WHERE UPPER(l.city) IN ('SEATTLE', 'LONDON', 'OXFORD', 'TORONTO', 'MUNICH') GROUP BY l.city, j.job_title </pre>		
Results	Explain	Describe
Saved SQL History		
CIUDAD	OFICIO	TOTAL DE EMPLEADOS
LONDON	Human Resources Representative	1
MUNICH	Public Relations Representative	1
OXFORD	Sales Manager	5
OXFORD	Sales Representative	30
SEATTLE	Accountant	5
SEATTLE	Accounting Manager	1
SEATTLE	Administration Assistant	1
SEATTLE	Administration Vice President	2
SEATTLE	Finance Manager	1
SEATTLE	President	1
SEATTLE	Public Accountant	1
SEATTLE	Purchasing Clerk	5
SEATTLE	Purchasing Manager	1

REALIZAR UNA FUNCIÓN QUE MUESTRE EL TIEMPO DE TRABAJO QUE TIENE UN EMPLEADO, ESTE DEBE SER MOSTRADO EN AÑOS, MESES Y DÍAS.

```
CREATE OR REPLACE FUNCTION TIEMPO_DE_TRABAJO(P_EMP_ID IN EMPLOYEES.EMPLOYEE_ID%TYPE)
RETURN VARCHAR2
IS
    V_HIRE_DATE EMPLOYEES.HIRE_DATE%TYPE;
    V_ANOS      NUMBER;
    V_MESES     NUMBER;
    V_DIAS      NUMBER;
    V_RESULTADO VARCHAR2(100);
BEGIN

    SELECT HIRE_DATE
    INTO V_HIRE_DATE
    FROM EMPLOYEES
    WHERE EMPLOYEE_ID = P_EMP_ID;
    V_ANOS := TRUNC(MONTHS_BETWEEN(SYSDATE, V_HIRE_DATE) / 12);
    V_MESES := TRUNC(MOD(MONTHS_BETWEEN(SYSDATE, V_HIRE_DATE), 12));
    V_DIAS := TRUNC(SYSDATE - ADD_MONTHS(V_HIRE_DATE, V_ANOS * 12 + V_MESES));

    V_RESULTADO := V_ANOS || ' AÑOS, ' || V_MESES || ' MESES, ' || V_DIAS || ' DÍAS';

    RETURN V_RESULTADO;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 'EMPLEADO NO ENCONTRADO';
    WHEN OTHERS THEN
        RETURN 'ERROR: ' || SQLERRM;
END;

-- USO DEL METODO

SELECT TIEMPO_DE_TRABAJO(101) AS TIEMPO_DE_TRABAJO FROM DUAL;
```

Results		Explain	Describe	Saved SQL	History
TIEMPO_DE_TRABAJO					
35 años, 8 meses, 12 días					
1 rows returned in 0,00 seconds		CSV Export			