

TAREA PARA BD05

Guiones y automatización de tareas

Índice

Ejercicio 1.....	3
Enunciado:.....	3
Solución:.....	4
Ejercicio 2.....	7
Enunciado:.....	7
Solución:.....	8
Ejercicio 3.....	12
Enunciado:.....	12
Solución:.....	13

Ejercicio 1

Enunciado:

Ejercicio 1: Función PL/SQL para obtener la siguiente letra del alfabeto

Debes crear una función en PL/SQL que reciba una letra mayúscula del alfabeto como parámetro y devuelva la siguiente letra en orden alfabético. Si la letra proporcionada es la última letra del alfabeto, como 'Z', la función debe devolver la primera letra, 'A'.

Requisitos:

- La función debe recibir como parámetro una letra mayúscula (de 'A' a 'Z').
- La función debe devolver la siguiente letra en el alfabeto. Si la letra es 'Z', debe devolver 'A'.
- Asegúrate de que la función sea robusta, manejando correctamente el caso de entrada 'Z'.

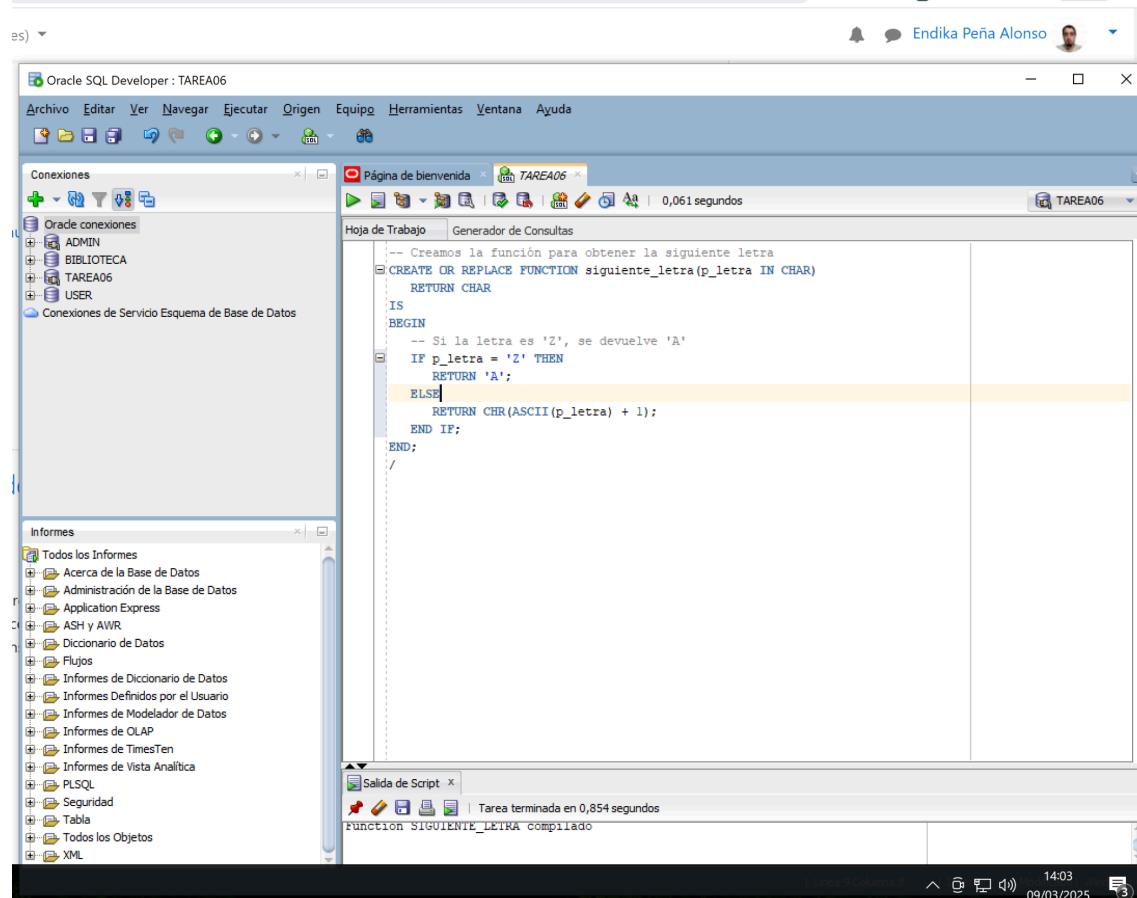
Sugerencia: Puedes utilizar la función ASCII y CHR de PL/SQL para trabajar con las letras. La función ASCII devuelve el valor numérico de un carácter, y CHR devuelve el carácter correspondiente a un valor numérico.

Solución:

Primero generamos una función que permite cambiar a la siguiente letra

Código:

```
-- Creamos la función para obtener la siguiente letra
CREATE OR REPLACE FUNCTION siguiente_letra(p_letra IN CHAR)
    RETURN CHAR
IS
BEGIN
    -- Si la letra es 'Z', se devuelve 'A'
    IF p_letra = 'Z' THEN
        RETURN 'A';
    ELSE
        RETURN CHR(ASCII(p_letra) + 1);
    END IF;
END;
/
```



Procedimiento

Código

```
-- Activamos la visualización de resultados
SET SERVEROUTPUT ON;

DECLARE
    var_usuario VARCHAR2(30);
    var_fecha    DATE;
BEGIN
    -- Obtenemos el usuario y la fecha/hora del sistema
    SELECT USER, SYSDATE
        INTO var_usuario, var_fecha
        FROM dual;

    -- Mostramos el usuario y la hora actual
    DBMS_OUTPUT.PUT_LINE('Usuario: ' || var_usuario || ' - Fecha y
Hora: ' || TO_CHAR(var_fecha, 'DD/MM/YYYY HH24:MI:SS'));

    -- Ejemplo de uso de la función: para 'C' y para 'Z'
    DBMS_OUTPUT.PUT_LINE('Siguiente letra de C: ' ||
siguiente_letra('C'));
    DBMS_OUTPUT.PUT_LINE('Siguiente letra de Z: ' ||
siguiente_letra('Z'));
END;
/
```

The screenshot shows the Oracle SQL Developer interface. The main window displays a PL/SQL script in the 'Hoja de Trabajo' (Worksheet) tab:

```
-- Activamos la visualización de resultados
SET SERVEROUTPUT ON;

DECLARE
    var_usuario VARCHAR2(30);
    var_fecha DATE;
BEGIN
    -- Obtenemos el usuario y la fecha/hora del sistema
    SELECT USER, SYSDATE
    INTO var_usuario, var_fecha
    FROM dual;

    -- Mostramos el usuario y la hora actual
    DBMS_OUTPUT.PUT_LINE('Usuario: ' || var_usuario || ' - Fecha y Hora: ' || TO_CHAR(var_fecha, 'DD/MM/YYYY HH24:MI:SS'));

    -- Ejemplo de uso de la función: para 'C' y para 'Z'
    DBMS_OUTPUT.PUT_LINE('Siguiente letra de C: ' || siguiente_letra('C'));
    DBMS_OUTPUT.PUT_LINE('Siguiente letra de Z: ' || siguiente_letra('Z'));
END;
/
```

The 'Salida de Script' (Script Output) tab shows the results of the execution:

```
Usuario: C##VENTAS_USER - Fecha y Hora: 09/03/2025 14:12:43
Siguiente letra de C: D
Siguiente letra de Z: A

Procedimiento PL/SQL terminado correctamente.
```

The status bar at the bottom indicates the script was completed in 0,059 seconds at 14:12:43 on 09/03/2025.

Ejercicio 2

Enunciado:

Ejercicio 2: Cursor PL/SQL para obtener información de empleados según oficio y sueldo

Crea un cursor en PL/SQL que muestre los apellidos, oficio y sueldo de los empleados que tengan un oficio concreto y un sueldo mayor que una cantidad dada. Los valores de oficio y sueldo deben ser proporcionados como parámetros de entrada.

Requisitos:

- El cursor debe recibir dos parámetros: Oficio: El oficio de los empleados (por ejemplo, 'PROGRAMADOR').
- Sueldo mínimo: Un valor numérico que representará el sueldo mínimo para filtrar los empleados.
- El cursor debe seleccionar los apellidos, oficio y sueldo de todos los empleados que cumplan con los criterios establecidos (oficio y sueldo mayor que el valor dado).
- El resultado debe ser mostrado para cada empleado que cumpla con los criterios.

Sugerencia: Utiliza un cursor explícito para realizar la selección de los empleados. Despues, recorre el cursor utilizando un LOOP o FETCH.

Ejemplo: Si el parámetro de oficio es 'PROGRAMADOR' y el sueldo mínimo es 3000, el cursor debe devolver todos los empleados con ese oficio y sueldo superior a 3000.

Solución:

Código:

```
-- Creamos el procedimiento que recibe el oficio y sueldo mínimo
CREATE OR REPLACE PROCEDURE mostrar_empleados (
    p_oficio      IN VARCHAR2,
    p_sueldo_min  IN NUMBER
)
IS
    -- Declaración del cursor con los parámetros recibidos
    CURSOR c_empleados IS
        SELECT apellido, oficio, salario
        FROM emple
        WHERE oficio = p_oficio
        AND salario > p_sueldo_min;

    v_apellido  emple.apellido%TYPE;
    v_oficio    emple.oficio%TYPE;
    v_sueldo    emple.salario%TYPE;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Empleados con oficio: ' || p_oficio || '
y sueldo mayor a ' || p_sueldo_min);

    OPEN c_empleados;
    LOOP
        FETCH c_empleados INTO v_apellido, v_oficio, v_sueldo;
        EXIT WHEN c_empleados%NOTFOUND;
        -- Se muestran los datos obtenidos del cursor
        DBMS_OUTPUT.PUT_LINE('Apellido: ' || v_apellido || ' -
Oficio: ' || v_oficio || ' - Sueldo: ' || v_sueldo);
    END LOOP;
    CLOSE c_empleados;
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END;
/
```

The screenshot shows the Oracle SQL Developer interface. The main window displays a PL/SQL script for creating a procedure named 'mostrar_empleados'. The script uses a cursor to select employees whose salary is greater than a minimum value based on their job. The code is as follows:

```
-- Creamos el procedimiento que recibe el oficio y sueldo minimo
CREATE OR REPLACE PROCEDURE mostrar_empleados (
    p_oficio      IN VARCHAR2,
    p_sueldo_min  IN NUMBER
)
IS
    -- Declaración del cursor con los parámetros recibidos
    CURSOR c_empleados IS
        SELECT apellido, oficio, salario FROM emple WHERE oficio = p_oficio AND salario > p_sueldo_min;

    v_apellido  emple.apellido%TYPE;
    v_oficio    emple.oficio%TYPE;
    v_sueldo    emple.salario%TYPE;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Empleados con oficio: ' || p_oficio || ' y sueldo mayor a ' || p_sueldo_min);

    OPEN c_empleados;
    LOOP
        FETCH c_empleados INTO v_apellido, v_oficio, v_sueldo;
        EXIT WHEN c_empleados%NOTFOUND;
        -- Se muestran los datos obtenidos del cursor
        DBMS_OUTPUT.PUT_LINE('Apellido: ' || v_apellido || ' - Oficio: ' || v_oficio || ' - Sueldo: ' || v_sueldo);
    END LOOP;
    CLOSE c_empleados;
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END;
```

The 'Salida de Script' (Script Output) tab at the bottom shows the message 'Procedure MOSTRAR_EMPLEADOS compilado' (Procedure MOSTRAR_EMPLEADOS compiled). The status bar at the bottom right indicates the task was completed in 1,474 seconds at 14:52 on 09/03/2025.

Uso del procedimiento.

Código:

```
-- Activamos la salida de mensajes
SET SERVEROUTPUT ON;

DECLARE
    v_usuario VARCHAR2(30);
    v_fecha    DATE;
BEGIN
    -- Obtenemos el usuario actual y la fecha/hora del sistema
    SELECT USER, SYSDATE
        INTO v_usuario, v_fecha
        FROM dual;

    DBMS_OUTPUT.PUT_LINE('Usuario: ' || v_usuario || ' - Fecha y
Hora: ' || TO_CHAR(v_fecha, 'DD/MM/YYYY HH24:MI:SS'));

    -- Llamamos al procedimiento para el oficio 'EMPLEADO' y sueldo
    -- mínimo de 1000
    mostrar_empleados('EMPLEADO', 1000);
END;
/
```

The screenshot shows the Oracle SQL Developer interface. The top menu bar includes 'Origen', 'Equipo', 'Herramientas', 'Ventana', and 'Ayuda'. The title bar shows 'Página de bienvenida' and 'TAREA06'. The main area is titled 'Hoja de Trabajo' and contains a 'Generador de Consultas' tab. The code area contains the following PL/SQL script:

```
-- Activamos la salida de mensajes
SET SERVEROUTPUT ON;

DECLARE
    v_usuario VARCHAR2(30);
    v_fecha DATE;
BEGIN
    -- Obtenemos el usuario actual y la fecha/hora del sistema
    SELECT USER, SYSDATE
    INTO v_usuario, v_fecha
    FROM dual;

    DBMS_OUTPUT.PUT_LINE('Usuario: ' || v_usuario || ' - Fecha y Hora: ' || TO_CHAR(v_fecha, 'DD/MM/YYYY HH24:MI:SS'));

    -- Llamamos al procedimiento para el oficio 'PROGRAMADOR' y sueldo mínimo de 3000
    mostrar_empleados('EMPLEADO', 1000);
END;
/
```

The output window below shows the results of the script execution:

```
Salida de Script x
Usuario: C##VENTAS_USER - Fecha y Hora: 09/03/2025 15:02:05
Empleados con oficio: EMPLEADO y sueldo mayor a 1000
Apellido: SANCHEZ - Oficio: EMPLEADO - Sueldo: 1040
Apellido: ALONSO - Oficio: EMPLEADO - Sueldo: 1430
Apellido: JIMENO - Oficio: EMPLEADO - Sueldo: 1335
Apellido: MUÑOZ - Oficio: EMPLEADO - Sueldo: 1690

Procedimiento PL/SQL terminado correctamente.
```

The status bar at the bottom right indicates the date and time: 09/03/2025 15:05.

Ejercicio 3

Enunciado:

Ejercicio 3: Auditoría de cambios en la tabla de empleados

Crea un trigger en PL/SQL que registre todos los cambios de oficio en la tabla EMPLE. Cada vez que se actualice el oficio de un empleado, se debe insertar un registro en una nueva tabla llamada AUDITA_EMPLE, con una descripción del cambio y la fecha en la que se realizó el cambio.

Requisitos:

- Crea una tabla llamada AUDITA_EMPLE con las siguientes columnas: ID_AUDIT (clave primaria autogenerada).
- DESCRIPCION (debería describir el cambio, por ejemplo, 'Cambio de oficio de ADMINISTRATIVO a PROGRAMADOR').
- FECHA_CAMBIO (fecha en la que se realizó el cambio, usa SYSDATE).

Crea un trigger que se dispare después de una actualización en la tabla EMPLE y registre en AUDITA_EMPLE un nuevo registro cada vez que un empleado cambie su oficio.

- La descripción del cambio debe indicar el oficio anterior y el nuevo oficio, y debe incluir la fecha del cambio.

Sugerencia: Recuerda usar los registros OLD y NEW en el trigger para acceder a los valores antiguos y nuevos de la columna OFICIO.

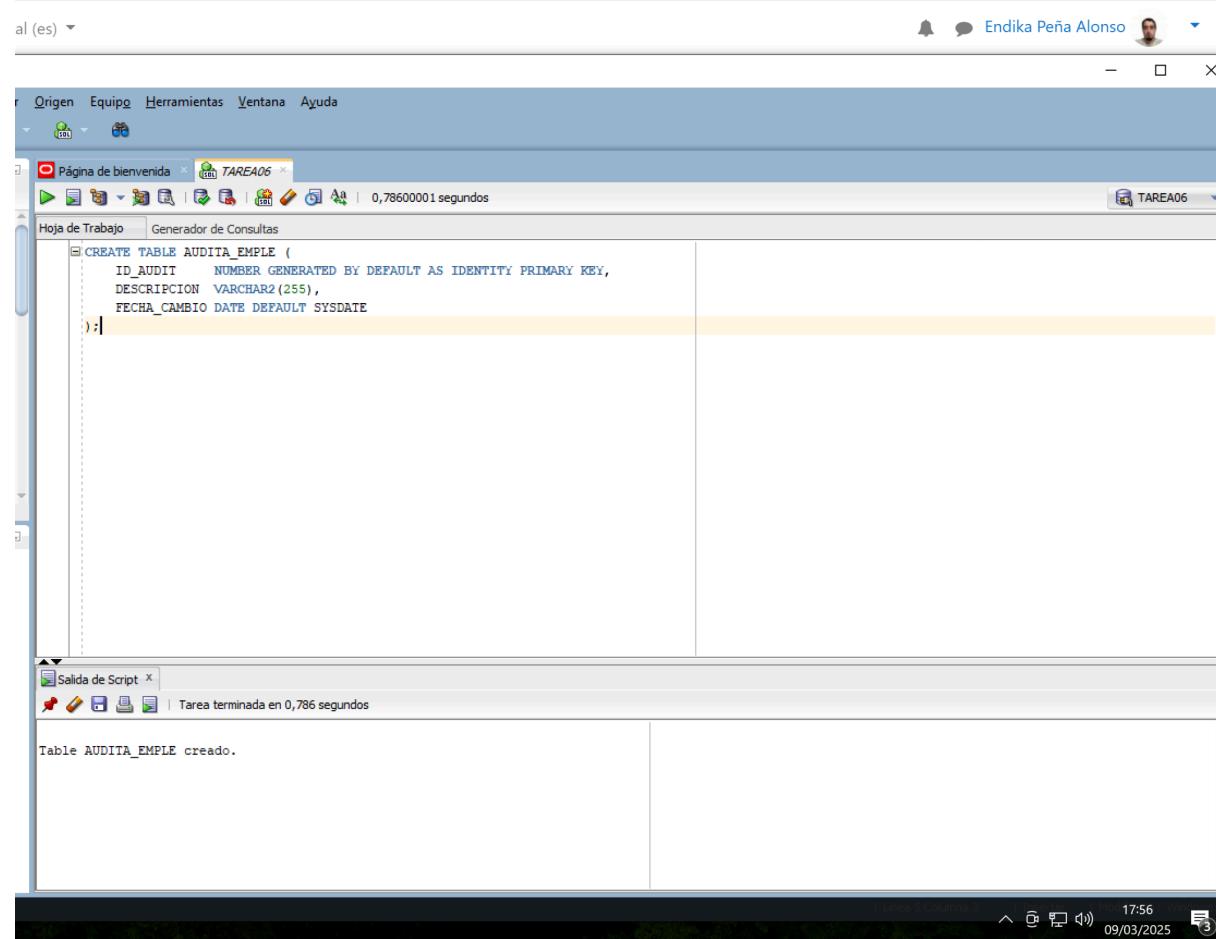
Ejemplo: Si un empleado tiene el oficio 'ADMINISTRATIVO' y se cambia a 'PROGRAMADOR', la tabla AUDITA_EMPLE debe registrar un cambio como 'Cambio de oficio de ADMINISTRATIVO a PROGRAMADOR' y la fecha actual.

Solución:

Creación de la tabla:

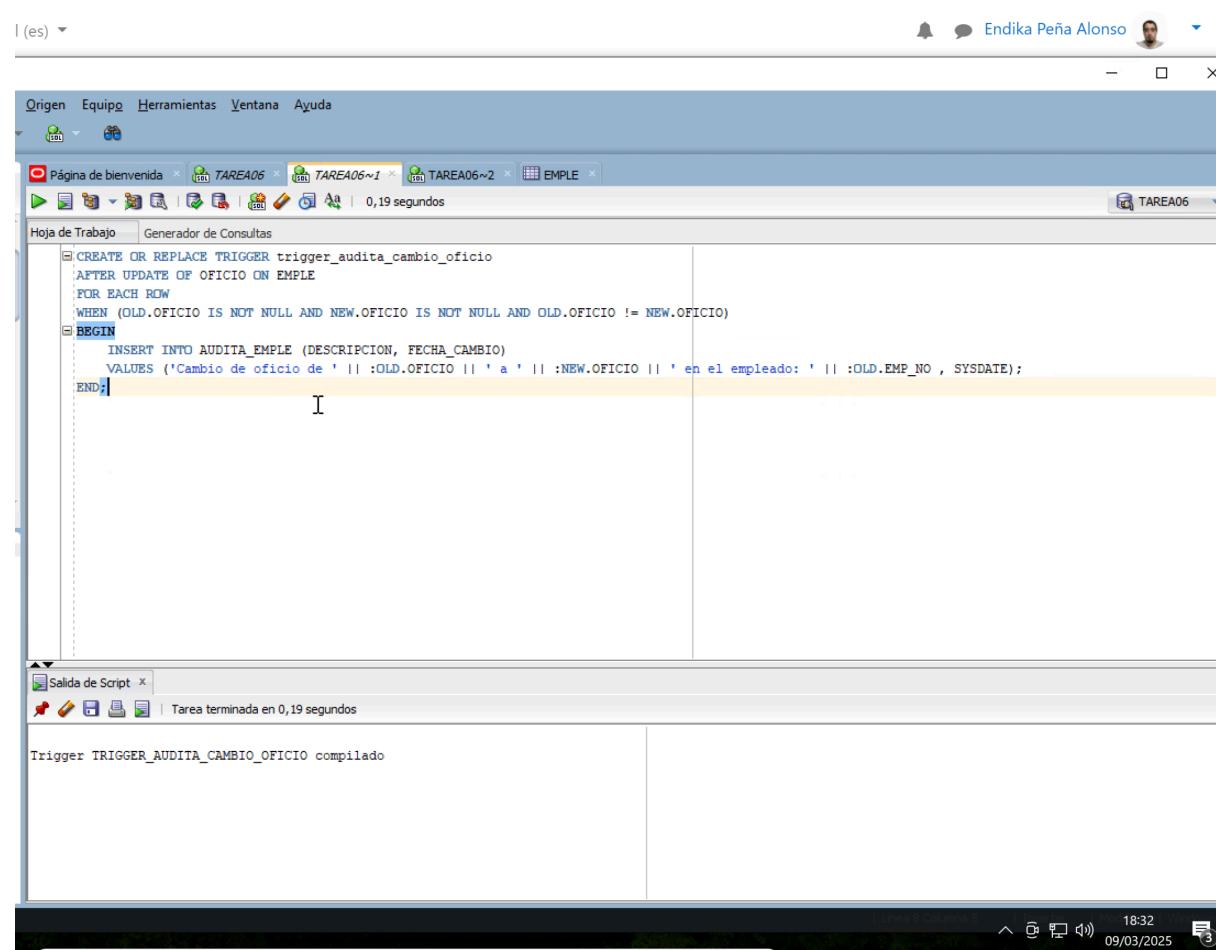
Código para crear la tabla

```
CREATE TABLE AUDITA_EMPLE (
    ID_AUDIT      NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY
KEY,
    DESCRIPCION   VARCHAR2(255),
    FECHA_CAMBIO DATE DEFAULT SYSDATE
);
```



Código del trigger

```
CREATE OR REPLACE TRIGGER trigger_audita_cambio_oficio
AFTER UPDATE OF OFICIO ON EMPLE
FOR EACH ROW
WHEN (OLD.OFICIO IS NOT NULL AND NEW.OFICIO IS NOT NULL AND
OLD.OFICIO != NEW.OFICIO)
BEGIN
    INSERT INTO AUDITA_EMPLE (DESCRIPCION, FECHA_CAMBIO)
    VALUES ('Cambio de oficio de ' || :OLD.OFICIO || ' a ' ||
:NEW.OFICIO || ' en el empleado: ' || :OLD.EMP_NO , SYSDATE);
END;
```



Pruebas de funcionamiento:

Hemos editado desde SQL Developer un registro de VENDEDOR A DEVELOPER

The screenshot shows the SQL Developer interface with the EMPLE table open. Row 5, which originally had 'OFICIO' as 'VENDEDOR', has been changed to 'DEVELOPERS'. This row is highlighted with a blue selection bar and has an asterisk (*) preceding its row number. An icon with a red circle and a minus sign is circled in red, indicating it's used for undoing changes. The 'Mensajes - Log' pane at the bottom shows the SQL statements used to make the changes:

```
DELETE FROM "C##VENTAS_USER"."AUDITA_EMPL" WHERE ROWID = 'AAAS42AAHAAAAK/AAD' AND ORA_ROWSCN = '13050965' and ( "ID_AUDIT" is null or "ID_AUDIT" is not null)
DELETE FROM "C##VENTAS_USER"."AUDITA_EMPL" WHERE ROWID = 'AAAS42AAHAAAAK/AAE' AND ORA_ROWSCN = '13050965' and ( "ID_AUDIT" is null or "ID_AUDIT" is not null)
Confirmación Correcta

UPDATE "C##VENTAS_USER"."EMPLE" SET OFICIO = 'DEVELOPER' WHERE ROWID = 'AAAS41AAHAAAAXJXAC' AND ORA_ROWSCN = '13050966'
UPDATE "C##VENTAS_USER"."EMPLE" SET OFICIO = 'DEVELOPER' WHERE ROWID = 'AAAS41AAHAAAAXJXAE' AND ORA_ROWSCN = '13050966'

Confirmación Correcta
```

Para confirmarlo vamos a pinchar en el ícono marcado con el círculo rojo, además si nos fijamos la fila seleccionada aparece marcada con un asterisco que indica que hay cambios que aún no se han confirmado.

Verificamos el cambio se realizado a generado un registro nuevo en la tabla de AUDITA_EMPL

The screenshot shows the Oracle SQL Developer interface. The top navigation bar includes 'Equipo', 'Herramientas', 'Ventana', and 'Ayuda'. The title bar indicates the connection is to 'USER.AUDITA_EMPL@TAREA06'. The main window displays the 'AUDITA_EMPL' table with two rows of data:

ID_AUDIT	DESCRIPCION	FECHA_C...
1	6 Cambio de oficio de DIRECTOR a DEVELOPER en el empleado: 7521	09/03/25
2	7 Cambio de oficio de VENDEDOR a DEVELOPER en el empleado: 7654	09/03/25

Below the table, the 'Mensajes - Log' pane shows the audit log entries:

```
DELETE FROM "C##VENTAS_USER"."AUDITA_EMPL" WHERE ROWID = 'AAAS42AAHAAAAK/AAD' AND ORA_ROWSCN = '13050985' and ( "ID_AUDIT" is null or "ID_AUDIT" is not nu
DELETE FROM "C##VENTAS_USER"."AUDITA_EMPL" WHERE ROWID = 'AAAS42AAHAAAAK/AAE' AND ORA_ROWSCN = '13050985' and ( "ID_AUDIT" is null or "ID_AUDIT" is not nu
Confirmación Correcta

UPDATE "C##VENTAS_USER"."EMPLE" SET OFICIO = 'DEVELOPER' WHERE ROWID = 'AAAS41AAHAAAAXKAAC' AND ORA_ROWSCN = '13050966'
UPDATE "C##VENTAS_USER"."EMPLE" SET OFICIO = 'DEVELOPER' WHERE ROWID = 'AAAS41AAHAAAAXKAAE' AND ORA_ROWSCN = '13050966'

Confirmación Correcta

Tarea Cancelada:
```

The bottom status bar shows the time as 18:39 and the date as 09/03/2025.