/\* 1-Realiza una función llamada incretusinciales que reciba como parámetro de entrada un código de un comercial e incremente su comisión un 7% y devuelva dicha comisión incrementada. Tened en cuenta que hay comerciales que tienen comisión igual a NULL y que para hacer operaciones matemáticas con dicho valor tenéis que usar la función NVL(a,0) que lo que hace es que si el valor de a es NULL devuelve un 0. También tenéis que controlar la excepción de no existencia del código de comercial. Si ejecutáis la función con un código de comercial cuya comisión IS NULL, el resultado ha de ser igual a 0.

Si ejecutamos la función con un código comercial cuya comisión IS NOT NULL, el resultado ha de ser la comisión incrementada.

Si ejecutamos la función con un código de comercial que no exista debe salir el siguiente mensaje de error controlando la EXCEPCIÓN adecuada y también devolver el valor -1.

\*/

CREATE OR REPLACE FUNCTION INCREDGL (CODCOMERCIAL NUMBER)

RETURN NUMBER AS

COMISION COMERCIAL.COMISION%TYPE;

BEGIN

SELECT NVL(COMISION\*1.07,0) INTO COMISION

FROM COMERCIAL

WHERE COMERCIAL.ID=CODCOMERCIAL;

RETURN COMISION;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('El comercial no existe');

RETURN -1;

END;

/

/\*2-Realiza un procedimiento llamado nuevacomistusiniciales que reciba como parámetro de entrada un código de comercial y si ese comercial tiene asignados 3 o más pedidos se le incremente la comisión un 7% haciendo uso de la función anterior.

Debe mostrar el nombre del comercial, los apellidos, la cantidad de pedidos , la comisión antigua y la nueva. También debe actualizar dicho valor en la tabla comercial.

Tened en cuenta que la excepción de no existencia del código comercial ya está controlada en la función.

Por ejemplo

● Si ejecutamos el procedimiento para el comercial de código 1. Este debe ser el resultado:

● Y si consultamos la tabla comercial, debe aparecer el nuevo valor de la comisión:

● Si ejecutamos el procedimiento con un código de comercial que no tenga 3 o más pedidos deberá devolver el siguiente mensaje.

● Si ejecutamos el procedimiento con un código de comercial que no tengamos registrado debe devolver el siguiente mensaje:

CREATE OR REPLACE PROCEDURE NUEVACOMISDGL(CODCOMERCIAL COMERCIAL.ID%TYPE) AS

NUEVACOMIS NUMBER;

NUMPEDIDOS NUMBER;

BEGIN

NUEVACOMIS:=INCREFINCH(CODCOMERCIAL);

IF NUEVACOMIS <> -1 THEN

SELECT COUNT(\*) INTO NUMPEDIDOS

FROM PEDIDOS

WHERE ID.COMERCIAL=CODCOMERCIAL;

IF NUMPEDIDOS>=3 THEN

SELECT NOMBRE,APELLIDO1,APELLIDO2,NVL(COMISION,0) INTO NOMBRE,APE1,AP2,COMI

FROM COMERCIAL

WHERE COMERCIAL.ID=CODCOMERCIAL;

DBMS\_OUTPUT.PUT\_LINE('Comercial: '||nombre||' '||ape1||' '||ape2||''||'Cantidad de pedidos: '||numPedidos||', comision antigua:');

END;

/

/\*3-

\*/

CREATE OR REPLACE FUNCTION PASSDGL (NOMBRECOMER COMERCIAL.NOMBRE%TYPE)

RETURN VARCHAR2 AS

CONTRASENA VARCHAR2(20);

BEGIN

SELECT RPAD(NOMBREECOMER,10,CONCAT (SUBSTR(NOMBRE,3,5),SUBSTR(NOMBRE,1,2))

FROM COMERCIAL

WHERE COMERCIAL.NOMBRE=NOMBRE;

RETURN CONTRASENA;

END;

/