|  |  |  |
| --- | --- | --- |
| **Nombre**: Jesús Andrés Rosales Medina | | **Matrícula**: 2991597 |
| **Nombre del curso:**  Bases de datos | **Nombre del profesor**:  Francisco Gómez Rubio | |
| **Módulo**:  Módulo 2 | **Actividad**:  Actividad 7 | |

1. Analiza la siguiente función:  
     
   FUNCTION convertir\_moneda  
   (cantidad IN NUMBER(12,3),  
   moneda\_original IN VARCHAR2,  
   moneda\_destino IN VARCHAR2) RETURNS NUMBER(12,3)  
   IS /\* ahora comienza la declaración \*/  
   conversion IN NUMBER(12,3);  
   datos\_mal EXCEPTION;  
   BEGIN  
   /\*Inicio de la sección ejecutable de un bloque\*/  
   IF cantidad > 0 THEN  
   . . .;  
   ELSE  
   **. . .;**  
   END IF;  
   EXCEPTION  
   /\*Inicio de la sección de manejo de excepciones \*/  
   WHEN datos\_mal THEN  
   . . .;  
   END;
2. Agrega las instrucciones necesarias PL/SQL para que la función pueda hacer conversiones de dólares a euros.

Código:

CREATE OR [REPLACE] FUNCTION convertir\_moneda

(cantidad IN NUMBER(12,3),

moneda\_original IN VARCHAR2,

moneda\_destino IN VARCHAR2) RETURN NUMBER(12,3)

IS

/\* ahora comienza la declaración \*/

conversion NUMBER(12,3);

datos\_mal EXCEPTION;

BEGIN

/\*Inicio de la sección ejecutable de un bloque\*/

IF cantidad > 0 THEN

IF moneda\_original = 'USD' AND moneda\_destino = 'EUR' THEN

conversion := cantidad \* 0.82; -- 1 USD = 0.82 EUR (tipo de cambio actualizado al 14 de mayo de 2023)

ELSIF moneda\_original = 'EUR' AND moneda\_destino = 'USD' THEN

conversion := cantidad \* 1.22; -- 1 EUR = 1.22 USD (tipo de cambio actualizado al 14 de mayo de 2023)

ELSE

RAISE datos\_mal;

END IF;

ELSE

RAISE datos\_mal;

END IF;

EXCEPTION

/\*Inicio de la sección de manejo de excepciones \*/

WHEN datos\_mal THEN

RETURN NULL;

END;

1. Analiza el siguiente procedimiento almacenado:

CREATE OR REPLACE PROCEDURE cambios\_estudiantes  
(p\_legajo IN NUMBER,  
p\_apellido IN VARCHAR2,  
p\_nombre IN VARCHAR2) AS  
BEGIN  
UPDATE estudiantes  
SET nombre = p\_nombre, apellido = p\_apellido  
WHERE legajo = p\_legajo;  
EXCEPTION  
WHEN no\_data\_found THEN  
DBMS\_OUTPUT.PUT\_LINE(‘No existe el empleado con el legajo’|| p\_legajo)  
END;

1. Modifica el procedimiento almacenado del punto 3 para que actualice, también, dirección, teléfono y correo electrónico.

CREATE OR REPLACE PROCEDURE cambios\_estudiantes

(p\_legajo IN NUMBER,

p\_apellido IN VARCHAR2,

p\_nombre IN VARCHAR2,

p\_direccion IN VARCHAR2,

p\_telefono IN VARCHAR2,

p\_correo IN VARCHAR2) AS

BEGIN

UPDATE estudiantes

SET nombre = p\_nombre,

apellido = p\_apellido,

direccion = p\_direccion,

telefono = p\_telefono,

correo = p\_correo

WHERE legajo = p\_legajo;

EXCEPTION

WHEN no\_data\_found THEN

DBMS\_OUTPUT.PUT\_LINE('No existe el estudiante con el legajo'|| p\_legajo)

END;

1. Desarrolla una función en PL/SQL llamada Cantidad\_Letra que reciba, como parámetro, un valor de tipo numérico que reciba valores como 1250 y retorne su equivalente en cadena como “Un mil doscientos cincuenta pesos 00/M. N.”

IF v\_entero >= 1000000000 THEN

v\_unidades := Cantidad\_Letra(TRUNC(v\_entero / 1000000000)) || 'mil millones ';

v\_entero := MOD(v\_entero, 1000000000);

END IF;

IF v\_entero >= 1000000 THEN

v\_unidades := v\_unidades || Cantidad\_Letra(TRUNC(v\_entero / 1000000)) || 'millones ';

v\_entero := MOD(v\_entero, 1000000);

END IF;

IF v\_entero >= 1000 THEN

v\_unidades := v\_unidades || Cantidad\_Letra(TRUNC(v\_entero / 1000)) || 'mil ';

v\_entero := MOD(v\_entero, 1000);

END IF;

IF v\_entero >= 100 THEN

v\_unidades := v\_unidades || Cantidad\_Letra(TRUNC(v\_entero / 100)) || 'cien ';

v\_entero := MOD(v\_entero, 100);

END IF;

IF v\_entero >= 30 THEN

CASE TRUNC

FUNCTION convertir\_moneda(cantidad IN NUMBER(12,3), moneda\_original IN VARCHAR2, moneda\_destino IN VARCHAR2) RETURNS NUMBER(12,3) IS

conversion IN NUMBER(12,3);

datos\_mal EXCEPTION;

BEGIN

IF cantidad > 0 THEN

IF moneda\_original = 'USD' AND moneda\_destino = 'EUR' THEN

conversion := cantidad / 1.22; -- convertir de dólares a euros

ELSE

conversion := cantidad; -- no es una conversión de dólares a euros, devolver la cantidad original

END IF;

RETURN conversion;

ELSE

RAISE datos\_mal;

END IF;

EXCEPTION

WHEN datos\_mal THEN

DBMS\_OUTPUT.PUT\_LINE('Error: la cantidad debe ser mayor que cero');

RETURN NULL;

END;