

APELLIDOS: _____

NOMBRE: _____

FECHA: 05/05/23

Curso: 2022/23

Convocatoria: 2 Evaluación

Módulo: Base de Datos

Ciclo Formativo: Seleccionar el que proceda

- o Desarrollo de aplicaciones multiplataforma
- o Desarrollo de aplicaciones web

Instrucciones.

- La prueba consta de 10 puntos.
- Cada ejercicio o sub-ejercicio tiene su puntuación.
- Debe realizar con bolígrafo azul o negro, el incumplimiento de esta norma implicará la calificación de 0 puntos en el examen.
- El DNI tiene que estar sobre la mesa.
- Se dispone de 1 hora y 30 minutos para la realización.
- Se debe responder en el lugar habilitado para ello.
- En caso de necesitar más folios, se debe indicar el número de hojas que se entrega adicionales al enunciado del examen. Estas deberán numerarse y escribir claramente el nombre del alumno y el ejercicio que se está resolviendo.

Entrego ____ hojas adicionales al examen.

Ejercicio 1. (2,5 puntos)

Crear una función, llamada ejercicio1, que reciba como un solo parámetro: el código del producto y calcule el número de unidades que se han vendido de ese producto. Se valora la eficiencia de la solución.

En caso de que el producto no se halla vendido o no exista devolverá el cero. En otras palabras, la función no generará ninguna excepción.

```
CREATE OR REPLACE FUNCTION ejercicio1 (codproducto CHAR)
RETURN NUMBER IS
existe INTEGER;
sumaUnidades INTEGER;
BEGIN
    SELECT COUNT(*) INTO existe
    FROM producto
    WHERE codigo_producto = codproducto;
    SELECT SUM(cantidad) INTO sumaUnidades
    FROM detalle_pedido
    WHERE codigo_producto = codproducto;
    RETURN sumaUnidades;
END;
/
```

Ejercicio 2. (2,5 puntos)

Crear un procedimiento, llamado ejercicio2, que dado un código de pedido, calcule su importe y genere el pago correspondiente. En el caso de que no exista ese pedido se genera la siguiente excepción:

- **Código:** -20011
- **Mensaje.** El pedido no existe.

```
CREATE OR REPLACE PROCEDURE ejercicio2 (codPedido CHAR)
AS
    esta INTEGER;
    importe NUMBER;
    codCliente NUMBER;
BEGIN
    SELECT COUNT(*) INTO esta
    FROM pedido
    WHERE codigo_pedido = codPedido;
    IF esta = 0 THEN
        RAISE_APPLICATION_ERROR (-20011, 'El pedido no
        existe');
    END IF;
    SELECT SUM(cantidad*precio_unidad) INTO importe
    FROM detalle_pedido
    WHERE codigo_pedido = codPedido;
    SELECT codigo_cliente INTO codCliente
    FROM pedido
    WHERE codigo_pedido = codPedido;
    INSERT INTO pago (codigo_cliente, total) VALUES
    (codCliente, importe);
END;
```

Ejercicio 3. (2,5 puntos)

Crear un disparador, llamado ejercicio3, que cuando se inserta una fila se comprueba que existe tanto el pedido al cual está asociado como el producto existe.

Excepciones:

- **Código:** -20011
- **Mensaje:** El pedido no existe.
- **Código:** -20012
- **Mensaje:** El producto no existe.

```
CREATE OR REPLACE TRIGGER ejercicio03
  BEFORE INSERT
  ON DETALLE_PEDIDO
  FOR EACH ROW
DECLARE
  v_pedido INTEGER;
  v_producto INTEGER;
BEGIN
  SELECT COUNT(*) INTO v_pedido
  FROM pedido
  WHERE codigo_pedido = :new.codigo_pedido;
  IF v_pedido = 0 THEN
    RAISE_APPLICATION_ERROR(-20011, 'El pedido no
    existe');
  END IF;
  SELECT COUNT(*) INTO v_producto
  FROM producto
  WHERE codigo_producto = :new.codigo_producto;
  IF v_producto = 0 THEN
    RAISE_APPLICATION_ERROR(-20012, 'El producto no
    existe');
  END IF;
END;
/
```

Ejercicio 4. (2,5 puntos)

- a) Escriba la instrucción para borrar la oficina cuyo código es: 'BCN-ES';

```
DELETE FROM oficina WHERE codigo_oficina='BCN-ES';
```

- b) Escriba la instrucción para subir un 10% el precio de los productos cuya gama es 'Herramientas'

```
UPDATE producto  
  SET precio_venta = 1.1*precio_venta  
  WHERE gama='Herramientas';
```

- c) Escriba las instrucciones para insertar el siguiente registro en la tabla Gama (no puede aparecer el valor null).

Gama	'Ferretería'
Descripcion_text	'Herramientas para el campo de mano'

```
INSERT INTO gama (gama, descripcion_text)  
  VALUES  
    ('Ferretería', 'Herramientas para el campo de mano');
```