

Bases de Datos - Examen final

1. Recordemos el script de creación de la tabla Sales:

```
CREATE TABLE sales
(
  stor_id          char(4)          NOT NULL
                REFERENCES stores(stor_id),
  ord_num          varchar(20)      NOT NULL,
  ord_date         datetime         NOT NULL,
  qty             smallint         NOT NULL,
  payterms         varchar(12)     NOT NULL,
  title_id         tid
                REFERENCES titles(title_id),
  CONSTRAINT UPKCL_sales PRIMARY KEY CLUSTERED (stor_id, ord_num, title_id)
)
```

Se solicita crear un trigger PL/pgsql que -cuando alguna sentencia elimine una fila en la tabla Sales- cree una copia de la tabla con el nombre SalesYYYY_M_D, donde YYYY indica el año, M el mes y D el día de la fecha de creación de la tabla.

Por ejemplo: **Sales2024_2_2**.

Sin embargo, se desea que la copia de la tabla posea el estado previo de la tablas Sales a la eliminación en curso.

El nombre de la tabla en realidad debería incluir también la hora o algún código incremental a fin de evitar errores de "archivo existente", pero ignore esta situación a los efectos de este ejercicio.

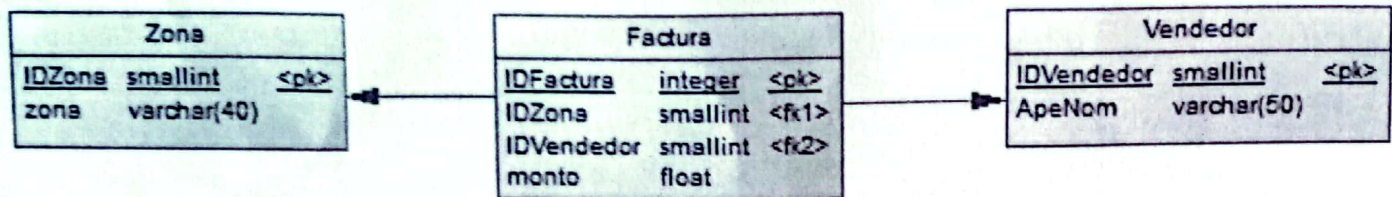
2. ¿Es correcto el siguiente trigger PL/pgsql?. Justifique.

```
CREATE FUNCTION test()
  RETURNS trigger
  LANGUAGE plpgsql
  AS
  $$
  DECLARE
  BEGIN
    NEW.price:=15;
    RETURN NEW;
  END
  $$;

CREATE TRIGGER trTitles4
  BEFORE INSERT ON Titles
  FOR EACH STATEMENT
  EXECUTE PROCEDURE test();
```

3. Describa qué es un non-repeatable read. ¿Que Isolation levels presentan estas situaciones?

4. El siguiente modelo de datos representa las ventas (facturas) de una empresa:



La tabla Factura posee una entrada por cada Factura de venta.

Una factura posee un monto (columna monto) y es atribuible a un vendedor (columna IDVendedor).

Los vendedores realizan ventas en diferentes zonas de la provincia. Para una determinada factura, la columna IDZona identifica la zona sobre la cual se realizó la venta.

Se desea obtener un reporte como el siguiente, que liste las tres mejores facturas (las de mayor monto) de cada zona.

Results		Messages		A	B	C
	IDZona	IdFactura	monto	Vendedor	Observación	
1	1	100	3400	MARANGONI FACUNDO	Supera la media de la zona	
2	1	105	2400	NORIEGA PABLO	Supera la media de la zona	
3	1	114	1100	MARANGONI FACUNDO	No supera la media de la zona	
4	2	106	5000	MARANGONI FACUNDO	Supera la media de la zona	
5	2	101	4000	MARANGONI FACUNDO	Supera la media de la zona	
6	2	108	3700	ULISI DIEGO	Supera la media de la zona	
7	3	115	7000	IBARRA JUAN MANUEL	Supera la media de la zona	
8	3	110	2400	ULISI DIEGO	No supera la media de la zona	
9	3	111	1000	NORIEGA PABLO	No supera la media de la zona	
10	4	102	3800	ULISI DIEGO	Supera la media de la zona	
11	4	104	1900	MENDIETA MARIELA	No supera la media de la zona	
12	4	112	1670	MARANGONI FACUNDO	No supera la media de la zona	

Tal como se muestra en el reporte de ejemplo:

(A) Los montos deben mostrarse de mayor a menor por zona.

(B) Se debe mostrar apellido y nombres del vendedor correspondiente a la factura.

(C) Como observación, se debe indicar si el monto de la factura supera o no el promedio de montos de la zona a la que corresponde.

Resolver utilizando SQL directo, sin utilizar vistas, tablas auxiliares ni extensiones T-SQL o PL/pgSQL.

No utilizar sentencias o comandos no vistos en las Guías de Estudio.