

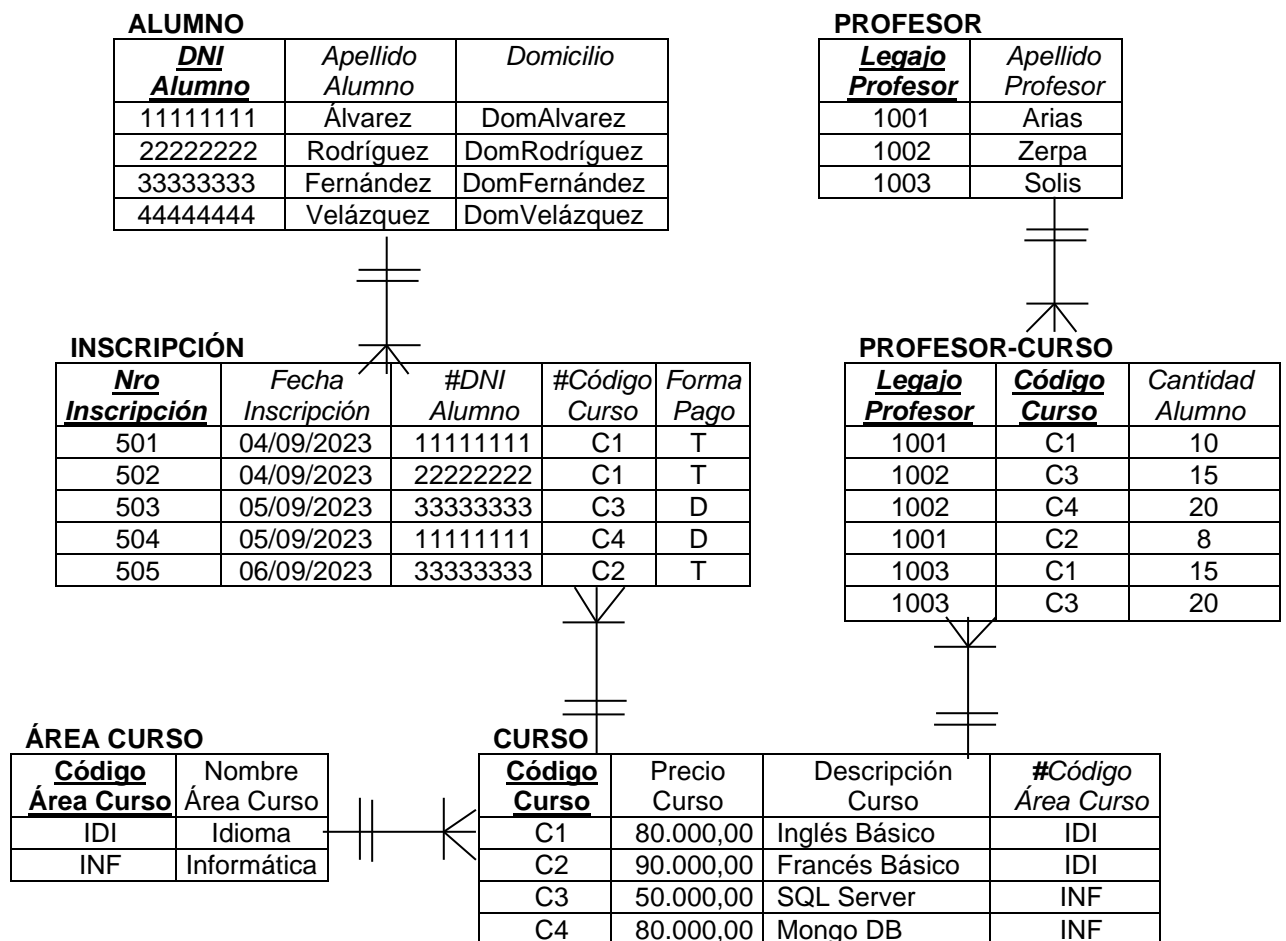
TRABAJO PRÁCTICO Nº 5

Fecha Inicio: 02/10/2023

Tema: PostgreSQL - PL SQL

Actividades

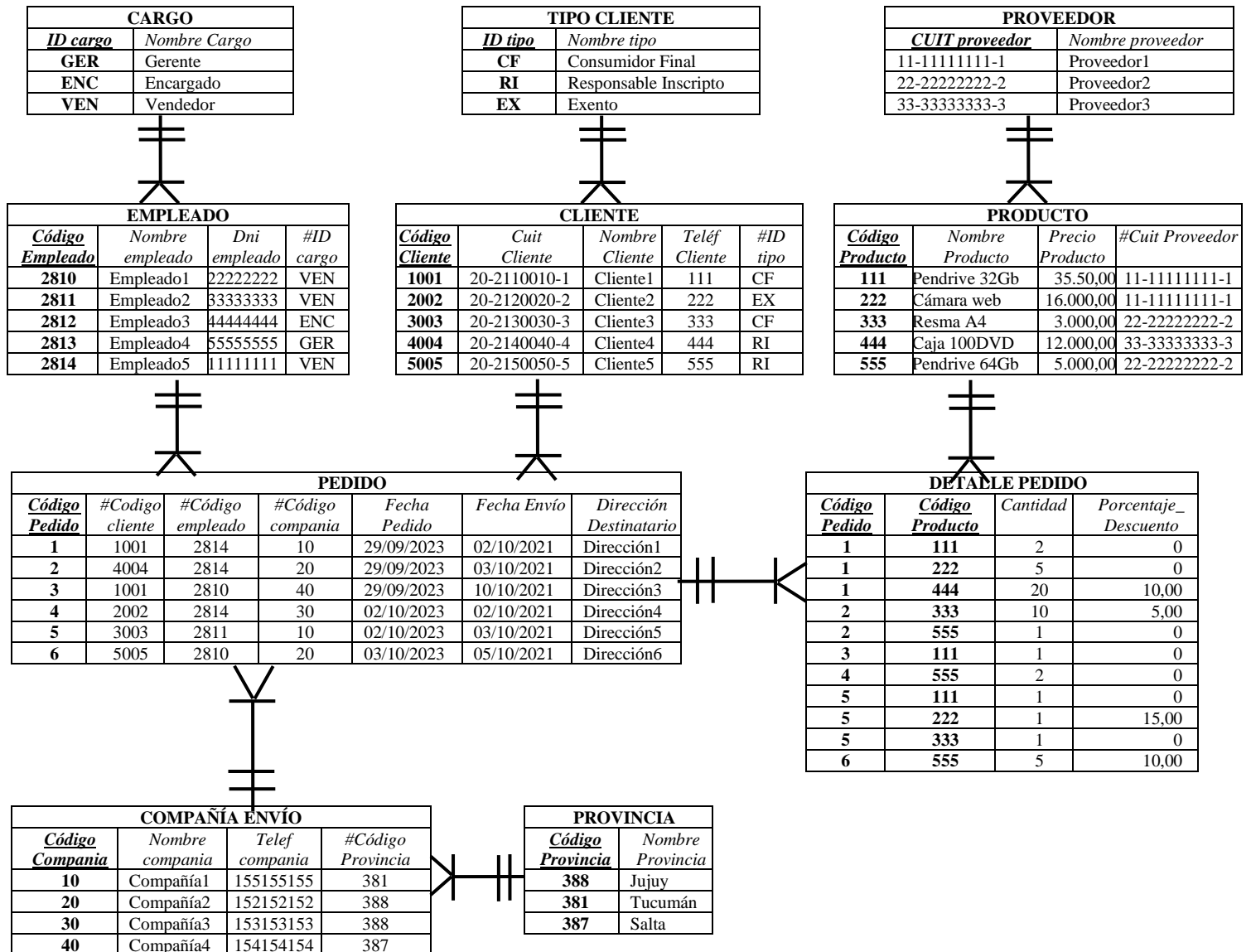
Utilizando PostgreSQL resuelva los siguientes ejercicios (por cada punto DEBE capturar las pantallas, tanto de la sentencia SQL como del resultado obtenido). Emplee el diagrama que se presenta en el Trabajo Práctico Nº 4.



1. Realice una función que reciba como parámetro la descripción curso y devuelva la cantidad de alumnos inscriptos en ese curso.
2. Realice una función que reciba como parámetro el apellido del profesor y devuelva el monto total cobrado según el precio de cursos que dicta.
3. Realice una función que reciba como parámetros el "Nombre Área Curso", cargue el cursor con los datos de los profesores que están en esa área de curso, lea el cursor y grabe dichos datos en la tabla "profesor_por_area".

4. Realice una función que reciba como parámetros una “Fecha inscripción desde” y una “Fecha inscripcion hasta”, cargue el cursor con los datos de la inscripción, lea el cursor y grabe dichos datos en la tabla “resumen_inscripcion”
5. Realice una función que reciba como parámetro el Código Curso, cargue un cursor con los datos del curso “profesor_curso”, luego debe abrir el cursor, y calcular el precio por cada producto ($\text{Precio Curso} * \text{Cantidad Alumno}$), sume el total y cargue la variable que será devuelta por la función.
6. Realice un trigger que antes de guardar un nuevo registro en profesor_curso, cuando el nombre de curso es “inglés básico”, la cantidad de alumnos permitida en ese curso es hasta 15 alumnos inclusive.
7. Realice un trigger que guarde un registro en la tabla histórico_precio_curso (codigo de curso, descripción curso, precio de curso, fecha de modificación, cada vez que se actualiza el precio de curso. Debe crear la tabla histórico_precio_curso previamente sin clave primaria.
8. Realice un trigger que permita sumar el precio curso cada vez que se ingresa un registro a la tabla Curso cuyo campo “Descripcion Curso” sea “Mongo DB”. Debe crear la tabla “curso_mongo” con los campos “nombre” y “precio_total”. El campo “nombre” tiene como valor “Mongo DB” y el campo “precio_total” se actualiza cada vez que se ingresa un nuevo registro a la tabla Curso.

Utilizando PostgreSQL resuelva los siguientes ejercicios (por cada punto DEBE capturar las pantallas, tanto de la sentencia SQL como del resultado obtenido). Emplee el diagrama que se presenta a continuación:



- Realice una función que muestre por pantalla la cantidad total de registros de todas las tablas.
- Realice una función que reciba como parámetro el código de provincia y devuelva el nombre de las compañías asociadas con el mismo.
- Realice una función que reciba como parámetro una fecha de pedido y devuelva el monto total de pedidos para esa fecha.

12. Realice una función que reciba como parámetros una “Fecha Pedido desde” y una “Fecha Pedido hasta”, cargue el cursor con los datos de pedidos, lea el cursor y grabe dichos datos en la tabla “pedido_especial”
13. Realice una función que reciba como parámetro el “Codigo Compañía”, cargue un cursor con todos los pedidos realizados por esa compañía, lea el cursor y devuelva los nombres de dichas compañías.
14. Realice una función que reciba como parámetro el Cuit de proveedor, cargue un cursor con los productos y el detalle pedido, luego debe abrir el cursor, y calcular el precio por cada producto ($\text{Precio Producto} * \text{Cantidad} - \text{Porcentaje Descuento}$), sume el total y cargue la variable que será devuelta por la función.
15. Realice un trigger que antes de guardar un nuevo detalle de pedido con un producto cuyo precio sea superior a \$10.000,00; le aplique un porcentaje descuento a 10% (valor a guardar en la columna respectiva).
16. Realice la creación de la tabla “Pedidos Pendientes” con los siguientes campos: código pedido, nombre cliente, nombre empleado, nombre compania, fecha pedido. Crear un trigger que luego de insertar un pedido sin fecha de envío, le asigne la fecha actual, guarde sus datos en la tabla “Pedidos Pendientes”.
17. Realice un trigger que guarde un registro en una tabla AuditoriaProducto (Codigo de Producto, Nombre de producto, Precio de Producto, Tipo de Operacion cuyos valores serán 'Nuevo', 'Modificado' o 'Eliminado') cada vez que se crea, actualiza o elimina un Producto. Debe crear la tabla AuditoriaProducto previamente sin clave primaria.
18. Realice un trigger que guarde un registro en una tabla AuditoriaProducto (creada en el punto anterior) cada vez que se agregue un producto cuyo precio sea mayor a \$50.000,00. En este caso en Tipo de operación debe guardarse “NuevoProductoDeLujo”.