PRÁCTICA DE LABORATORIO

BASES DE DATOS ORIENTADAS A OBJETOS

PREREQUISITOS

- PostgreSQL instalado (versión 12 o superior)
- pgAdmin 4 o acceso a psql

PARTE 1: MODELADO RELACIONAL TRADICIONAL

Paso 1: Crear la base de datos

✓ Se creó la base de datos lab_bdoo en PostgreSQL.

Paso 2: Crear tablas relacionales tradicionales

Se diseñaron tres tablas:

✓ direcciones: Almacena información de ubicación.

✓ autores: Contiene datos de autores con referencia a direcciones.

✓ libros: Registra libros publicados con referencia a autores.

Paso 3: Insertar datos en el modelo relacional

✓ Se insertó una dirección en Lima.

✓ Se registró el autor *Mario Vargas Llosa*.

✓ Se agregaron dos libros escritos por el autor.

Paso 4: Consultar datos (requiere JOINs)

Data Output | Messages | Notifications

Showing rows: 1 to 2

	titulo character varying (200)	isbn character varying (20)	autor text	ciudad character varying (50)
1	La ciudad y los perros	978-8420471839	Mario Vargas Llosa	Lima
2	Conversación en La Catedral	978-8466326070	Mario Vargas Llosa	Lima

Total rows: 2 | Query complete 00:00:00.071

OBSERVACIÓN: *Note que necesitamos 2 JOINs para obtener información completa.*

PARTE 2: MODELADO OBJETO-RELACIONAL

Paso 5: Crear tipos de datos personalizados

Se definieron dos tipos compuestos:

✓ *tipo_direccion: Contiene calle, número, ciudad y código postal.*

- ✓ *tipo_autor: Incluye nombre, apellido, fecha de nacimiento y dirección (anidada).*

Paso 6: Crear tabla con tipos compuestos

- ✓ Se creó la tabla libros_oo con una columna autor de tipo tipo_autor.

OBSERVACIÓN: *Note que el campo 'autor' es un tipo complejo que contiene toda la información del autor, incluyendo su dirección anidada.*

Paso 7: Insertar datos en el modelo objeto-relacional

- Se insertaron los mismos libros, pero con el autor y dirección como objetos anidados.

Paso 8: Consultar datos sin JOINs (navegación por puntos)

	titulo character varying (200)	isbn character varying (20)	nombre_autor text	ciudad_autor character varying (50)
1	La ciudad y los perros	978-8420471839	Mario Vargas Llosa	Lima
2	Conversación en La Catedral	978-84266326070	Mario Vargas Llosa	Lima

Total rows: 2 Query complete 00:00:00.176

OBSERVACIÓN: *No necesitamos JOINs. Accedemos directamente a los datos anidados usando notación de punto, similar a la programación orientada a objetos.*

PARTE 3: COMPARACIÓN Y ANÁLISIS

Paso 9: Comparar consultas

Ambas consultas devolvieron los mismos resultados, pero con diferencias en la estructura:

- **Relacional:** Usa JOINs explícitos.
- **Objeto-Relacional:** Usa notación de puntos sobre tipos compuestos.

Consulta Relacional:

	titulo character varying (200)	autor text	ciudad character varying (50)
1	La ciudad y los perros	Mario Vargas Llosa	Lima
2	Conversación en La Catedral	Mario Vargas Llosa	Lima

Total rows: 2 Query complete 00:00:00.095

Consulta Objeto-Relacional:

	titulo character varying (200)	nombre_autor text	ciudad_autor character varying (50)
1	La ciudad y los perros	Mario Vargas Llosa	Lima
2	Conversación en La Catedral	Mario Vargas Llosa	Lima

Total rows: 2 Query complete 00:00:00.106

Paso 10: Ventajas y desventajas observadas

Complete la siguiente tabla basándose en su experiencia:

Característica	Modelo Relacional	Modelo Objeto-Relacional
# Tablas necesarias	3 Tablas; cada entidad existe de forma independiente, permitiendo reutilización. Esto da una separación clara de responsabilidades, pero tiende a mayor complejidad en el esquema físico.	1 Tabla principal con tipos compuestos anidados, elimina la necesidad de claves foráneas explícitas y proporciona una estructura más compacta y organizada.
Complejidad de consultas	Alta para consultas con múltiples JOINs, sin embargo esto mismo permite filtrar en cualquier nivel de la jerarquía. Puede degradarse con muchas tablas involucradas.	Baja, debido a la navegación por puntos. Su sintaxis es más intuitiva y puede ser mejor para consultas de lecturas frecuentes, pero es menos flexible para filtros complejos.
Normalización	Alta, es la tercera forma de normalización. • Se eliminan redundancias porque la dirección se almacena una vez por autor. • Las claves foráneas garantizan consistencia, esto asegura la integridad. • Si existieran cambios en dirección, estos afectarían automáticamente a todos los libros del autor	Baja forma de normalización • Si Vargas Llosa cambia de dirección, se debe actualizar cada libro. Por lo tanto, existe un gran riesgo de inconsistencia de datos.
Facilidad de actualización	Alta, la actualización se hace en un solo lugar, pero todos los libros reflejan el cambio inmediatamente, sin embargo, se requiere conocer la estructura completa de las relaciones.	Media, la operación para actualización es más compleja, porque su sintaxis no es tan intuitiva, por ello tiende a fallos; lo que generaría inconsistencia si no se actualizan todos los registros.
Consistencia de datos	• Alta consistencia por restricciones FK. • La integridad se mantiene porque PostgreSQL impide operaciones inválidas.	• Consistencia media, depende del diseño de la aplicación. • No hay verificación automática de parte de PostgreSQL, por lo que no se garantiza integridad.