**Trabajo Practico N°1 - Persistencia**

**Desarrollo de creación de JPA**

**Creación del Proyecto Spring Boot:**

1. Fui al sitio web de Spring Initializer ([https://start.spring.io/](https://start.spring.io/" \t "https://chat.openai.com/c/_new)).
2. Configuré mi proyecto seleccionando "Maven Project", "Java", y estableciendo el nombre del proyecto.
3. Agregué las dependencias que necesitaba, como Spring Boot Devtools, Lombok, Spring Web, H2 Database y Spring Data JPA.

**Explicación de las Dependencias:**

Spring Boot Devtools: Me ayuda durante el desarrollo para que los cambios en el código se reflejen automáticamente en la aplicación sin necesidad de reiniciarla manualmente.

Lombok: Simplifica la escritura de código Java, reduciendo la necesidad de crear métodos repetitivos como getters y setters.

Spring Web: Facilita la creación de aplicaciones web, manejando las solicitudes HTTP y la gestión de controladores.

H2 Database: Es una base de datos en memoria útil para desarrollar y probar aplicaciones localmente.

Spring Data JPA: Hace más fácil interactuar con bases de datos relacionales utilizando JPA.

**Creación del codigo:**

Entidades:

Creé varias clases Java que representan las entidades de mi aplicación, como Cliente, Domicilio, Factura, Pedido, Producto, Rubro, y DetallePedido.

Usé anotaciones de JPA para mapear estas clases a tablas en la base de datos.

La clase BaseEntidad es una clase base para otras entidades y contiene un campo id que se genera automáticamente.

Repositorios:

Creé repositorios para cada entidad utilizando Spring Data JPA, como ClienteRepository, DomicilioRepository, etc. Estos repositorios me permiten realizar operaciones de base de datos de manera sencilla.

Main:

En la clase principal TpjpaApplication, configuré la inyección de dependencias para los repositorios y definí un método init como un bean CommandLineRunner que se ejecuta al iniciar la aplicación.

En el método init, creé instancias de las entidades (clientes, domicilios, facturas, etc.), las relacioné entre sí y las guardé en la base de datos.

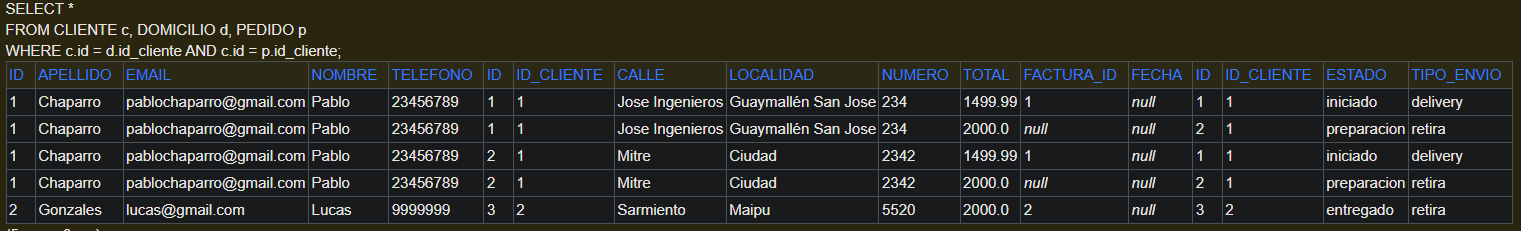
También mostré algunos ejemplos de cómo recuperar y mostrar datos de la base de datos, como recuperar un cliente por su ID.

**Consulta para verificar:**

1. Un cliente puede tener muchos domicilios:



1. Un cliente puede hacer varios pedidos



1. Pedido tiene una factura



1. Pedido tiene muchos detalle y cada detalle tiene un solo producto

