**Documentación de la Base de Datos**

**1. Base de Datos Relacional**

**Modelo de Entidad-Relación**

1. **Tabla: Category**
   * **Descripción**: Almacena las categorías de productos.
   * **Columnas**:
     + id (Long): Clave primaria, valor autogenerado.
     + name (String): Nombre de la categoría.
2. **Tabla: Product**
   * **Descripción**: Almacena información sobre los productos.
   * **Columnas**:
     + id (Long): Clave primaria, valor autogenerado.
     + name (String): Nombre del producto.
     + description (String): Descripción del producto.
     + price (Double): Precio del producto.
     + category\_id (Long): Clave foránea que referencia a la tabla Category.
     + stock (Integer): Cantidad en stock.
3. **Tabla: Order**
   * **Descripción**: Representa las órdenes realizadas por los usuarios.
   * **Columnas**:
     + id (Long): Clave primaria, valor autogenerado.
     + userId (Long): ID del usuario que realiza la orden.
     + orderDate (LocalDateTime): Fecha y hora de la orden.
     + status (String): Estado de la orden.
     + total (Double): Total de la orden.
4. **Tabla: OrderItem**
   * **Descripción**: Almacena los elementos de cada orden.
   * **Columnas**:
     + id (Long): Clave primaria, valor autogenerado.
     + productId (Long): Clave foránea que referencia a la tabla Product.
     + quantity (Integer): Cantidad del producto en la orden.
     + price (Double): Precio del producto en la orden.
     + order\_id (Long): Clave foránea que referencia a la tabla Order.

**Relaciones**

* **Category a Product**: Relación de muchos a uno (ManyToOne). Un producto pertenece a una categoría.
* **Order a OrderItem**: Relación de uno a muchos (OneToMany). Una orden puede tener múltiples elementos.
* **OrderItem a Product**: Relación de muchos a uno (ManyToOne). Un ítem de una orden está asociado a un producto.

**Detalles Técnicos**

* **@Entity**: Marca las clases como entidades de base de datos.
* **@Id y @GeneratedValue**: Definen el campo de clave primaria y su generación automática.
* **@ManyToOne y @OneToMany**: Definen las relaciones entre entidades.
* **@JoinColumn**: Define la columna que se usa como clave foránea en la tabla actual.
* **@Cascade**: Controla la propagación de operaciones en las entidades relacionadas.
* **@FetchType**: Estrategia de carga para relaciones (EAGER o LAZY).

**2. Base de Datos No Relacional (Ejemplo: MongoDB)**

**Modelo de Documentos**

1. **Colección: categories**
   * **Descripción**: Almacena los documentos que representan las categorías de productos.

:

{

"id": "Long",

"name": "String"

}

1. **Colección: products**
   * **Descripción**: Almacena los documentos que representan los productos.
   * **Documentos**:

{

"id": "Long",

"name": "String",

"description": "String",

"price": "Double",

"category": {

"id": "Long",

"name": "String"

},

"stock": "Integer"

}

1. **Colección: orders**
   * **Descripción**: Almacena los documentos que representan las órdenes realizadas.
   * **Documentos**:

{

"id": "Long",

"userId": "Long",

"orderDate": "ISODate",

"status": "String",

"total": "Double",

"items": [

{

"productId": "Long",

"quantity": "Integer",

"price": "Double"

}

]

}

**Relaciones**

* **Referencias**: Los documentos de products pueden incluir una referencia anidada a categories, y los documentos de orders pueden incluir una lista de elementos (items) con detalles de productos.