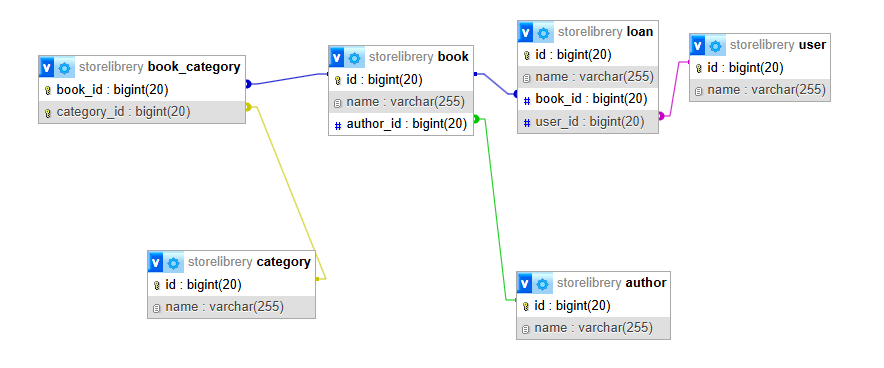
1. **Glosario de términos**
   1. **@JoinTable**: Especifica una tabla auxiliar que gestiona las asociaciones en una relación de muchos a muchos.
   2. **Relación @OneToMany**: Representa una conexión en la que un solo registro puede estar vinculado a múltiples registros de otra entidad.
   3. **Clave primaria (@Id)**: Atributo que permite identificar de manera exclusiva cada fila dentro de una tabla.
   4. **Relación @ManyToOne**: Define una asociación en la que varios registros hacen referencia a un único registro en otra entidad.
   5. **Entidad**: Corresponde a un objeto del modelo de datos que se almacena como una fila en una tabla de la base de datos.
   6. **@JoinColumn**: Especifica qué columna contendrá la referencia a una clave primaria de otra tabla.
   7. **Relación @ManyToMany**: Vínculo en el que múltiples registros de una entidad pueden estar conectados con varios registros de otra entidad.
2. **Código fuente de las entidades**

<https://github.com/JuanCelis2004/JavaBack/tree/main/StoreLibrery>

1. **Diagrama de relaciones entre entidades**



1. **Documento explicativo**

**Entidad Libro (book):**

**Relación con Autor (author):**

Un libro está escrito por un solo autor, lo que se representa con una relación @ManyToOne desde book hacia author. Esto significa que muchos libros pueden estar asociados a un solo autor.

**Relación con Ejemplares Físicos:**

Si consideras que cada libro puede tener varios ejemplares físicos, esto se representaría con una relación @OneToMany desde book hacia una entidad hipotética Ejemplar. Esto indica que un libro puede tener muchos ejemplares.

**Relación con Categorías (category):**

Un libro puede pertenecer a varias categorías, y una categoría puede contener muchos libros. Esto se representa con una relación @ManyToMany entre book y category, utilizando una tabla intermedia book\_category para manejar esta relación.

**Entidad Usuario (user):**

**Relación con Préstamos (loan):**

Un usuario puede tener varios préstamos, lo que se representa con una relación @OneToMany desde user hacia loan. Esto indica que un usuario puede estar asociado a muchos préstamos.

**Entidad Préstamo (loan):**

**Relación con Ejemplar:**

Cada préstamo está asociado a un solo ejemplar físico, lo que se representa con una relación @ManyToOne desde loan hacia la entidad Ejemplar. Esto significa que muchos préstamos pueden estar asociados a un solo ejemplar.

**Relación con Usuario (user):**

Cada préstamo está asociado a un solo usuario, lo que se representa con una relación @ManyToOne desde loan hacia user. Esto indica que muchos préstamos pueden estar asociados a un solo usuario.

En resumen, tiene una estructura en la base de datos:

* Libro tiene una relación @ManyToOne con Autor.
* Libro tiene una relación @OneToMany con Ejemplar.
* Libro tiene una relación @ManyToMany con Categoría.
* Usuario tiene una relación @OneToMany con Préstamo.
* Préstamo tiene una relación @ManyToOne con Ejemplar y Usuario.