Class Definitions:

1. Class Name: Book

- Properties:

- `title` (String)

- `author` (String)

- `ISBN` (String)

- Methods:

- `toString()` - To provide a string representation of the book.

- Special Considerations:

- Abstract class for other book types.

- Implements polymorphism for various book types.

2. Class Name: FictionBook

- Properties: Inherits `title`, `author`, and `ISBN` from the `Book` class.

- Special Considerations: Represents a type of book with specific details.

3. Class Name: NonFictionBook

- Properties: Inherits `title`, `author`, and `ISBN` from the `Book` class.

- Special Considerations: Represents a type of book with specific details.

4. Class Name: Library

- Properties:

- `books` (List<Book>) - Composition relationship.

- Methods:

- `addBook(Book book)`

- `getBook(String ISBN)`

- `updateBook(String ISBN, Book newBook)`

- `deleteBook(String ISBN)`

- Special Considerations: Demonstrates composition, implements CRUD operations.

5. Class Name: BookOperationsImpl

- Properties:

- `database` (SQLite) - Data storage.

- Methods:

- `createBook(Book book)`

- `readBook(String ISBN)`

- `updateBook(String ISBN, Book newBook)`

- `deleteBook(String ISBN)`

- Special Considerations: Implements the `BookOperations` interface, provides database operations, handles data storage.

6. Interface: BookOperations

* Methods:
  + createBook(Book book)
  + readBook(String ISBN)
  + updateBook(String ISBN, Book **newBook)**
  + deleteBook(String ISBN)
* Special Considerations: This class defines the interface with methods for CRUD operations.

Data Storage:

- Table Name: BookCollection

- Fields:

- `ISBN` (TEXT, PRIMARY KEY) - Unique identifier for the book.

- `title` (TEXT) - Title of the book.

- `author` (TEXT) - Author of the book.

- `type` (TEXT) - Indicates if the book is fiction or non-fiction.

- Relationships:

- None

Class Hierarchy and Relationships:

- The `FictionBook` and `NonFictionBook` classes inherit from the `Book` abstract class, implementing specific details for their respective types.

- The `Library` class has a composition relationship with the `Book` class, managing a collection of books.

- The `BookOperationsImpl` class implements the `BookOperations` interface, which defines the methods for CRUD operations.

Access Specifiers:

- Access specifiers will be used to encapsulate class members appropriately, allowing controlled access to properties and methods.