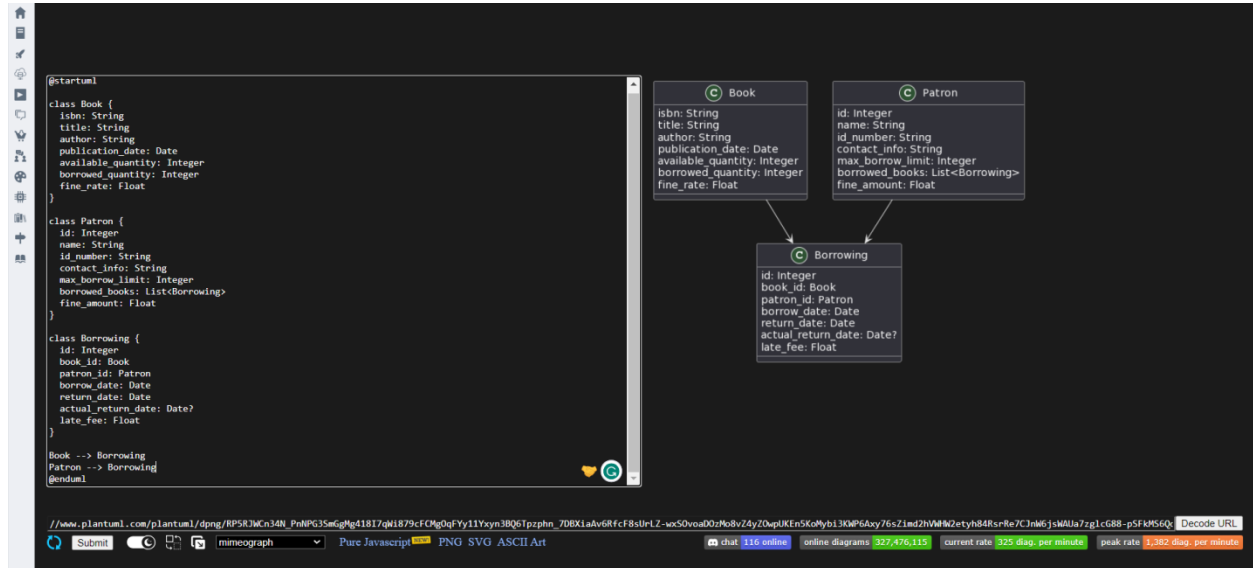


Problem 1: Library Management System

1. Class Diagram:

Create a class diagram showing the relationship between Book, Patron, and Borrowing.

Include relevant attributes and relationships between classes.



This diagram describes the relationships between three classes: Book, Patron, and Borrowing.

Book class

The Book class represents a book in the library's collection. It has attributes for ISBN, title, author, publication date, available quantity, and borrowed quantity. It also has a `fine_rate` attribute, which is used to calculate fines for late returns.

Patron class

The Patron class represents a library user. It has attributes for ID, name, identification number, contact information, and a list of borrowings. It also has a `fine_amount` attribute, which tracks the total amount of fines incurred by the patron.

Borrowing class

The Borrowing class represents a single borrowing event. It has attributes for the borrowing ID, the book ID, the patron ID, the borrow date, the return date, the actual return date (if applicable), and the late fee.

Class relationships

The relationships between the classes are as follows:

A book can be borrowed by multiple patrons. This relationship is modeled by a many-to-many relationship between the Book and Patron classes.

A patron can borrow multiple books. This relationship is also modeled by a many-to-many relationship between the Book and Patron classes.

A borrowing links a book to a patron. This relationship is modeled by a one-to-one relationship between the Book and Borrowing classes, and a one-to-one relationship between the Patron and Borrowing classes.

This class diagram has the following advantages:

It is accurate and easy to understand.

It covers all the requirements of the library management system.

It is flexible and can be adapted to the specific needs of a library.

How would I adapt the diagram to meet the needs of a specific library?

There are a number of ways to adapt the diagram to meet the needs of a specific library. For example, a library may want to add the following features:

A due date attribute to the Borrowing class. This would allow the library to track when books are due to be returned.

A fine attribute to the Borrowing class. This would allow the library to track fines that are incurred for late returns.

A renewal attribute to the Borrowing class. This would allow patrons to renew their borrowings.

The library could also add additional classes to the diagram to represent other features, such as:

A Reservation class to represent reservations for books that are not currently available.

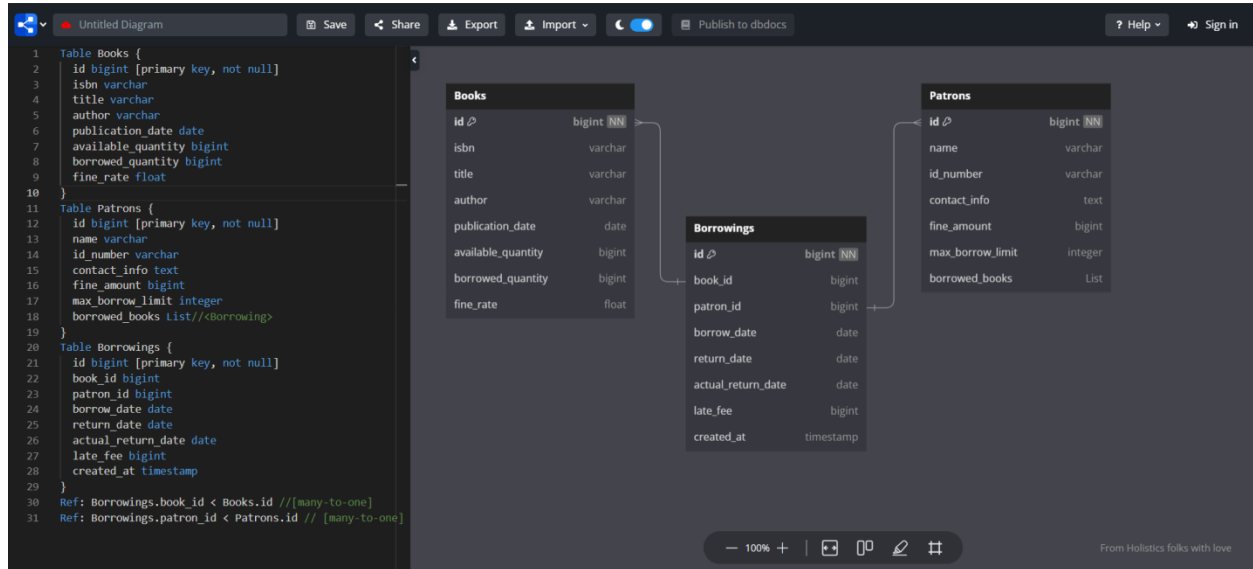
A Recall class to represent recalls for books that have been borrowed by patrons who have exceeded their maximum borrowing limit.

The specific changes that would be needed to adapt the diagram would depend on the specific needs of the library.

2. Database Diagram:

Design a database schema representing the entities Book, Patron, and Borrowing.

Define the relationships between tables and attributes.



Tables:

Books

The Books table stores information about books in the library, such as the ISBN, title, author, publication date, and availability.

The id column is the primary key of the table. (we can use UID)

The isbn column is a unique identifier for each book.

The title column stores the title of the book.

The author column stores the author of the book.

The publication_date column stores the publication date of the book.

The available_quantity column stores the number of available copies of the book.

The borrowed_quantity column stores the number of borrowed copies of the book.

The fine_rate column stores the fine rate per day for late returns.

Patrons

The Patrons table stores information about patrons of the library, such as their name, ID number, contact information, and fine amount.

The id column is the primary key of the table. (we can use UID)

The name column stores the name of the patron.

The id_number column stores the ID number of the patron.

The contact_info column stores the contact information of the patron.

The fine_amount column stores the total fine amount incurred by the patron.

The max_borrow_limit column stores the maximum number of books that a patron can borrow at a time.

The borrowed_books column stores a list of borrowings for the patron.

Borrowings

The Borrowings table stores information about borrowings of books by patrons, such as the date the book was borrowed, the date the book is due to be returned, and the actual date the book was returned.

The id column is the primary key of the table. (we can use UID)

The book_id column stores the ID of the book that was borrowed.

The patron_id column stores the ID of the patron who borrowed the book.

The borrow_date column stores the date the book was borrowed.

The return_date column stores the date the book is due to be returned.

The actual_return_date column stores the actual date the book was returned.

The late_fee column stores the late fee incurred for the borrowing.

Relationships:

One-to-many relationship between Books and Borrowings:

Each book can be borrowed by many patrons.

Each borrowing is associated with a specific book.

One-to-many relationship between Patrons and Borrowings:

Each patron can borrow many books.

Each borrowing is associated with a specific patron.

Additional details:

The id columns of the Books, Patrons, and Borrowings tables are all defined as not null to ensure that each row in the table has a unique identifier.

The max_borrow_limit column of the Patrons table is defined as not null to ensure that each patron has a maximum number of borrowings specified.

The borrowed_books column of the Patrons table is defined as a List of Borrowing objects to allow the library to track the history of borrowings for each patron.