**Development Documentation**

**Main Classes and What They Do**

The library app is made up of different parts, called classes. Each class has its own job to help organize everything in the library:

* **Library.java**: This is the main class that runs everything. It keeps track of all the library items (books and magazines), authors, and patrons. Anytime something needs to be added, changed, or removed, the Library class makes it happen.
* **LibraryItem.java**: This is like a general template for any item in the library, like a book or a magazine. It has shared info that all items have, like an ID, title, and publisher.
* **Book.java**: This is a specific type of LibraryItem just for books. It includes extra book details, like the ISBN number, author, and how many copies are available.
* **Periodical.java**: This is also a type of LibraryItem, but it’s for magazines or journals. It’s almost like the Book class but for periodicals, with a few differences for how it’s organized.
* **Author.java**: This class holds details about authors. It keeps their name, birthdate, and a list of items they’ve written. This makes it easier to see all items by the same author.
* **Patron.java**: This is a general class for people who borrow library items. It stores shared info for all patrons, like their name, contact info, and a list of items they’ve borrowed.
* **Student.java and Employee.java**: These are special types of patrons. Each one extends the Patron class and has a few extra details. For example, Student has a student ID, and Employee has an employee ID.
* **Patrons.java**: This class manages the list of patrons. It adds, edits, and removes patrons from the list, making sure everyone is tracked correctly.
* **Demo.java**: This is the main class that runs the app. It shows a menu with options, so we can pick what to do, like adding a new book, viewing all patrons, or borrowing an item.

**A diagram of a library

Description automatically generatedHow We Could Use a Database for Managing Data**

If we wanted to make data management even easier, we could use a **PostgreSQL** database with **pgAdmin** to keep everything organized in tables. Here’s how we’d set it up:

1. **Organizing Data with Tables**  
   PostgreSQL lets us create tables, just like spreadsheets, to store all the info in an organized way. Each table would be for a specific part of the library app:
   * **Books Table**: This table would store details for every book in the library, with columns for things like the title, author, ISBN, publisher, and how many copies are available.
   * **Authors Table**: This table would keep info on all authors, with columns for their name, birthdate, and maybe a list of the books they’ve written. This table could be linked to the Books table to make it easy to see which books each author wrote.
   * **Patrons Table**: This table would list everyone who borrows items, whether they’re students or employees. Each row would hold details like their name, address, phone number, and type (student or employee). This table could also link to the Borrowed Items table to show which items they have checked out.
   * **Borrowed Items Table**: This table would track everything that’s been borrowed. Each row would store details like which patron borrowed which item, the date they borrowed it, and when it’s due to be returned.