To get the book’s ISBN\_13 from Google API:

1. Enter the corresponding book *title* in <https://www.googleapis.com/books/v1/volumes?q=intitle:1984+inauthor:Orwell>
2. From the response, grab results.data.docs[0].cover\_edition\_key

To get the book’s ISBN\_13 from OpenLibrary API:

1. Enter the corresponding book *title* and make a get request in axios to <https://openlibrary.org/search.json?title=the+lord+of+the+rings>
2. From the response, get the isbn number using result.data.docs[0].ia.find(item => item.startsWith(“ISBN”)).slice(5)

Then use to replace the value in <https://covers.openlibrary.org/b/isbn/0385472579-L.jpg> and use the link in the src for the img tag for the book title.

There is an option to do this from the front end or the back end.

However, since we only need to use this to get the image for the book cover, it may be better to do this in the back end.

It may even be a good idea to get the ISBN number in the back end and link it in the SQL database for easier management.

In the front end, it may be a good idea to provide an option to enter a book title in the search bar and only allow the user to enter books for which there is an isbn number available. For this, a hover-box with the search results may need to be provided for good user-experience.

Google api sometimes gives 403 forbidden error when the request is sent through axios. This is because google apis sometimes require an api key to make the requests.

OpenLibraryAPI allows us to get the isbn information without the use of api keys.

However, the results returned from the openlibrary api often have multiple isbn numbers corresponding to different editions and versions (such as soft cover, hard cover, etc.) for the same book. It also does not have a consistent format structure of where this isbn number is stored in the result. Hence we have to use the array.search function to find the matches.

However, ChatGPT is saying the ISBNs listed under “ia” section may not be actual ISBN numbers.

~~Instead, we need to grab the "edition\_key" from results.data.docs[0].cover\_edition\_key” first. Then enter this number into~~ [~~https://openlibrary.org/books/OL26331930M.json~~](https://openlibrary.org/books/OL26331930M.json)~~.~~

But again, the results from the above did not have any isbn field in it.

So, will be relying on searching the values through the ia array that starts with isbn.

Need to use result.data.docs[0].ia.find(item => item.startsWith(“ISBN”)).slice(5)

Created a new database named permalist and created a table named bookreviewlist in that database using the Schema:

CREATE TABLE bookreviewlist (

id SERIAL UNIQUE,

Title varchar(100) NOT NULL,

Author varchar(100) NOT NULL,

Rating int,

Summary TEXT,

Isbn VARCHAR(13),

date timestamptz,

PRIMARY KEY (title, author)

);