**Creating Relations: Assignment**

This checkpoint should take you between 60 and 90 minutes. If you spend longer than that on this checkpoint, reach out for help!

**Overview**

At *Thinkful Books*, your business is booming. However, your *Customer Support* team has been struggling to handle an increasing number of questions from customers about the books available at your store. Currently, your inventory is recorded in a physical notebook, so your team has to consult and look up records manually to find answers to your customers' questions.

Customers would like to know, for example, if *Thinkful Books* currently has any historical books written during the Second World War between 1939 - 1945 in stock. Other customers might like to view all the books in your inventory written by a certain author. Your employees might also need a list of books that are out of stock so they can plan next month's inventory accordingly.

To help answer these questions more efficiently, you have decided to create and link a few additional tables in your database so that your team can easily access the information desired by your customers and your employees.

**Instructions**

For this checkpoint to considered complete, you must get all the tests to pass. Follow the instructions below to get the tests to pass.

NOTE: Make sure you have the correct punctuation and spelling in your queries, which must exactly match the question prompts. The tests are looking for *exact* string matches in the queries.

**Set up a database (optional)**

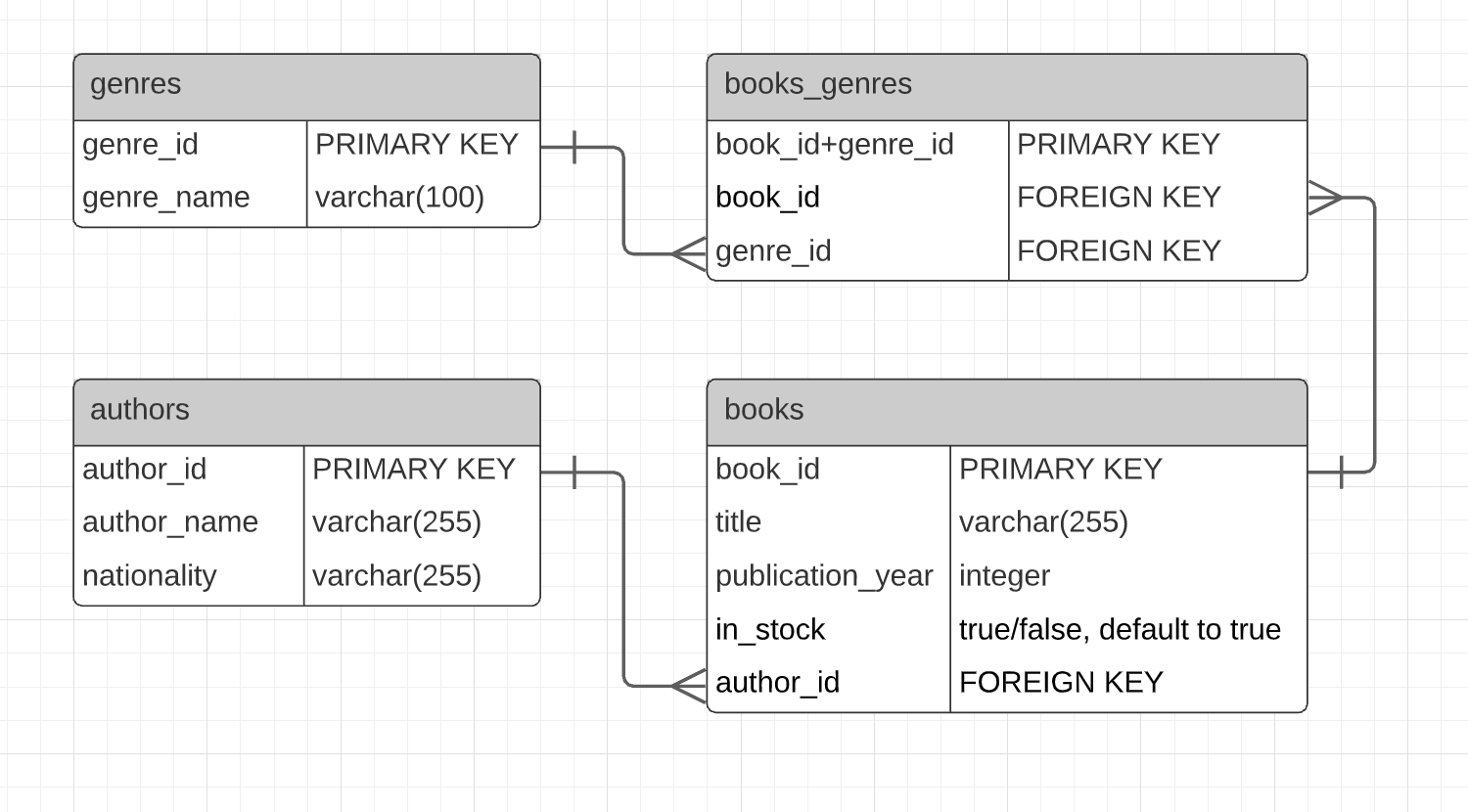
This setup is optional, but if you would like a way to debug your queries, you can setup a database and connect it to DBeaver on your local machine. You can then execute any queries you write for this assignment and verify their outputs in DBeaver on your local machine.

1. Set up a new instance called thinkful\_books on ElephantSQL, if you haven't already set up one while you were working through the module. The instructions for creating a new database instance can be found in the "Creating and deleting databases" checkpoint of the previous module.
2. Connect DBeaver to your database instance and rename the database connection to thinkful\_books for easy reference. The instructions for connecting DBeaver to your instance can be found in the "Installing DBeaver" checkpoint of the previous module.
3. After creating all the tables described below, you can run the setup/seed.sql script in DBeaver to seed your database. Then you can use the SELECT \* FROM <table\_name> command to retrieve the records from the tables and check that the tables were properly populated. As you're creating your queries for this assignment, you can execute your queries in DBeaver to see if they're retrieving the datasets properly.

**Tasks**

After brainstorming with your employees, you created the following ERD to represent the addtional tables you would need:

1:1



And now, you are ready to translate the ERD above into SQL tables.

For each of the following files, create the queries as described.

1. setup/authors.sql, setup/books.sql, setup/genres.sql, setup/books\_genres.sql: Write queries to create the new tables in your database based on the ERD provided above.
2. src/get-all-books-by-author.sql: Write a query to return a list of all book and author-related fields for all books written by "Amy Tan".
3. src/get-all-books-not-in-stock.sql: Write a query to return a list of all book and author-related fields for all books that are currently out of stock.
4. src/get-all-books-by-country.sql: Write a query to return a list of all book and author-related fields for all books written by an author from "China" or "Turkey".
5. src/count-books.sql: Write a query to return the number of books "Leo Tolstoy" wrote.
6. src/count-books-before-year.sql: Write a query to return the number of books written before 2000.
7. src/get-all-books-before-year-by-nationality.sql: Write a query to return a list of all book and author-related fields for all books written before 2005 by non-US authors.
8. src/get-all-authors-by-title-length.sql: Write a query to return a list of all authors who have written books whose titles are longer than 25 characters.
9. src/get-all-books-by-author-by-genre.sql: Write a query to return a list of all book and author-related fields for all books written by "Leo Tolstoy" that belong to either the "autobiography" or "history" genres.