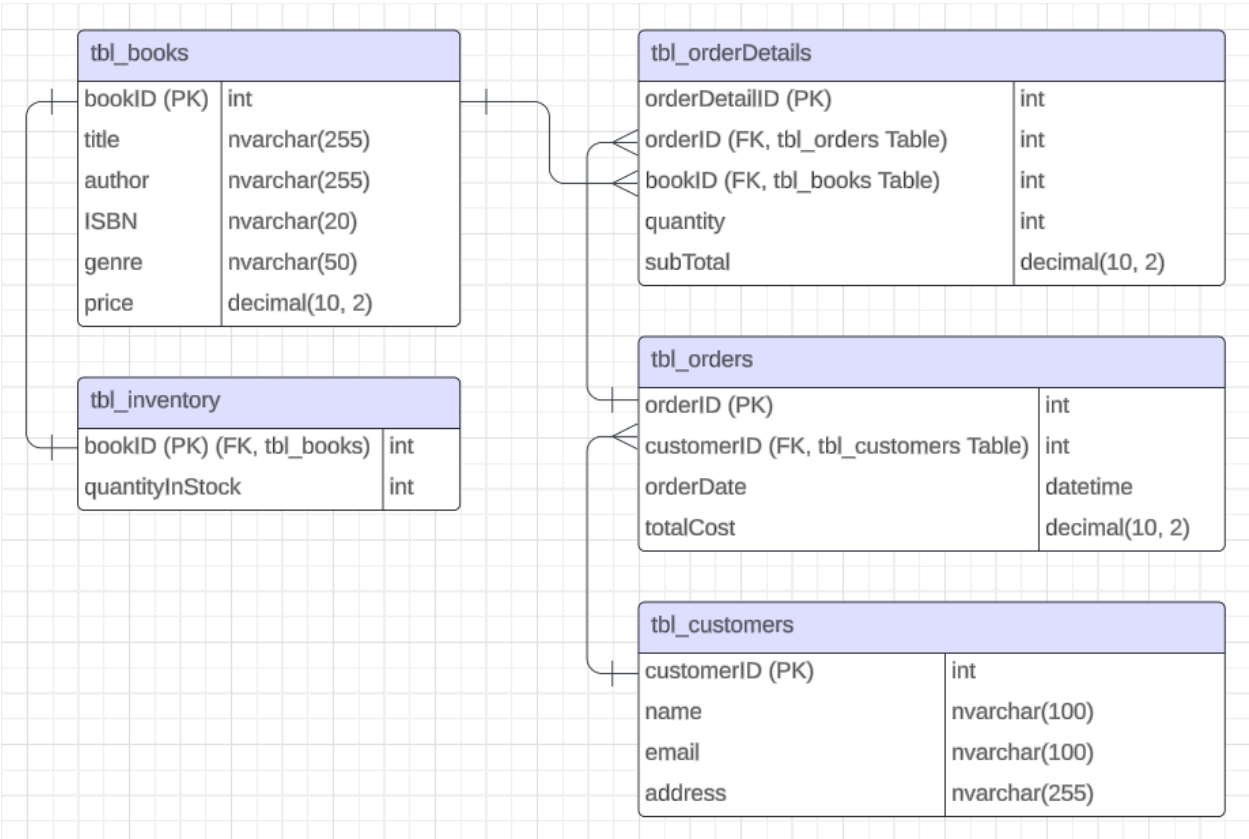
Phase 1: Database Design
Group 3 Members: Johnson Benedict Corpus, Reynold Ocampo, Glenn Perez, Rod Stephen Espiritu

Case Study: Online Bookstore
The Online Bookstore sells books, manages customers, track orders, and maintains inventory.

- The database should store information about books, including title, author, ISBN, genre, and price.
- Customers should be able to register accounts with the bookstore, providing their name, email, and address information.
- Customers should be able to place orders for one or more books.
- Each order should contain information such as the customer who placed the order, the date and time of the order, and the total cost.
- The database should maintain inventory information, including the quantity of each book in stock.

ER DIAGRAM



RELATIONSHIPS BETWEEN TABLES

1. tbl_books – tbl_orders (One-to-Many)
    - Each order in the tbl_orders table can consist of multiple books from the tbl_books table. One order can contain multiple books, but each book can be associated with only one order.
    - The cardinality is indicated as “1:N”.

2. tbl_customers – tbl_orders (One-to-Many)
    - Each customer in the tbl_customers table can place multiple orders in the tbl_orders table. One customer can place multiple orders, but each order is associated with only one customer.
    - The cardinality is indicated as “1:N”.
3. tbl_books – tbl_orderDetails (One-to-Many)
    - Each book in the tbl_books table can appear in multiple order details in the tbl_orderDetails table. One book can be included in multiple order details, but each order detail is associated with only one book.
    - The cardinality is indicated as “1:N”.

4. tbl_orders – tbl_orderDetails (One-to-Many)
    - Each order in the tbl_orders table can consist of multiple order details in the tbl_orderDetails table. One order can have multiple order details, but each order detail is associated with only one order.
    - The cardinality is indicated as “1:N”.

5. tbl_books – tbl_inventory (One-to-One)
    - Each book in the tbl_books table has a corresponding entry in the tbl_inventory table, representing the quantity of that book in stock. Each book has exactly one inventory entry, and each inventory entry corresponds to exactly one book.
    - The cardinality is indicated as “1:1”.

SPECIAL CONTRAINTS THAT ENTITIES HAVE

1. tbl_books table
    - bookID (Primary Key): Unique identifier for each book.
    - ISBN (Unique Constraint): Ensures that each book has a unique ISBN.
    - price (Check Constraint): Ensures that the price of a book is always greater than zero.

2. tbl_customers table
    - customerID (Primary Key): Unique identifier for each customer.

3. tbl_orders table
    - orderID (Primary Key): Unique identifier for each order.
    - customerID (Foreign Key): Establishes a relationship with the tbl_customers table, ensuring that orders are associated with existing customers.
    - orderDate (Default Constraint): Automatically sets the current date and time when a new order is inserted.

4. tbl_orderDetails table
    - orderDetailID (Primary Key): Unique identifier for each order detail.
    - orderID (Foreign Key): Establishes a relationship with the tbl_orders table, ensuring that order details are associated with existing orders.
    - bookID (Foreign Key): Establishes a relationship with the tbl_books table, ensuring that order details are associated with existing books.
5. tbl_inventory table
    - bookID (Primary Key, Foreign Key): Unique identifier for each book and establishes a relationship with the tbl_books table.
    - quantityInStock (Not Null Constraint): Ensures that the quantity in stock is always provided and cannot be null.