

Step 2:

This database schema represents a system for managing a library's resources, events, and personnel. It comprises several entities, each with their attributes and relationships, providing efficient organisation and tracking of various aspects within the library.

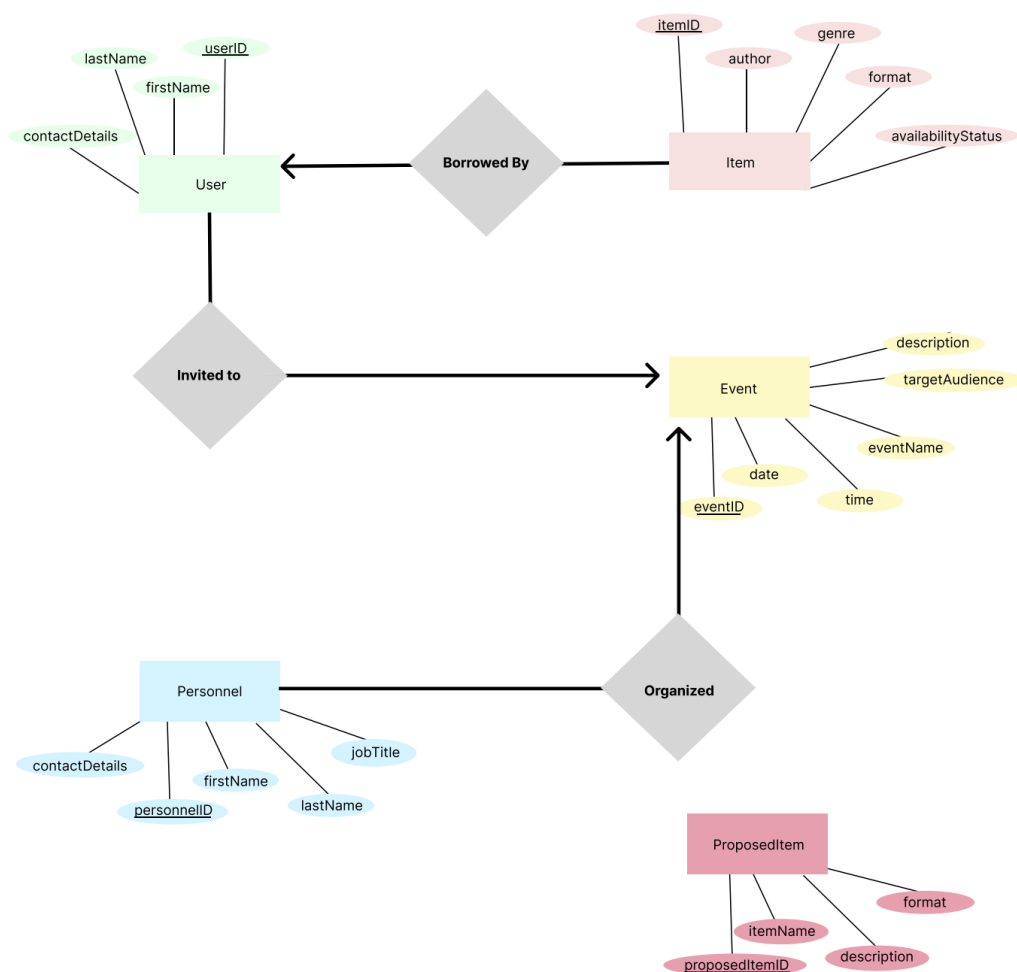
Here's an overview of our database design:

1. **Item:** This entity captures essential details about the library's collection, including itemID(primary key), title, author, genre, format, and availabilityStatus. The attributes allow users to easily search and locate specific items, whether they are print books, online books, CDs, records, or other materials.
2. **User:** The user entity is designed to store information about library users. Each user is assigned a unique userID as the primary key. We record their first name, last name, and contact details, such as email and phone number.
3. **Borrowing:** The borrowing entity is crucial in tracking the borrowing transactions made by users. With borrowingID as the primary key, we can uniquely identify each transaction. The foreign keys, userID, and itemID, establish relationships with the User and Item entities, respectively. This allows us to associate each borrowing record with the corresponding user and item. Additionally, we record the borrowing date, due date, return date, and any applicable fines, enabling efficient management of borrowed items and tracking fines.
4. **Event:** Each event is assigned a unique eventID as the primary key. Key attributes like eventName, description, date, time, and targetAudience provide information about library events. The targetAudience attribute ensures that events are recommended to specific audiences, helping users find events tailored to their interests.
5. **Personnel:** Our Personnel entity stores information about library staff members. Each staff member is assigned a unique personnelID as the primary key. We record their first name, last name, job title, and contact details to facilitate internal communication and management.
6. **Organised:** This relationship table connects personnel to the events they organise. It establishes a many-to-many relationship between Personnel and Events entities using foreign keys personnelID and eventID, respectively. This enables multiple staff members to be associated with various events they organise.

7. Invited: The Invited relationship table establishes a many-to-many relationship between Users and Events entities(using foreign keys userID and eventID). It enables users to attend multiple events and events to have multiple attendees.

8. ProposedItem: For future library expansion, we have introduced the ProposedItem entity. Each proposed item is assigned a unique proposedItemID as the primary key. We record the item's name, description, and format, providing insights into potential additions to the library.

Step 3: E-R Diagram



Step 4:

Our design is currently in BCNF, and it does not have any non-trivial functional dependencies, which means it does not allow anomalies. We have carefully analyzed all the entities and their attributes to ensure that each table is free of bad functional dependencies. Each table has a primary key that uniquely identifies each record, and all other attributes in each table are functionally dependent on the

primary key. Moreover, we have used proper foreign key constraints to establish relationships between tables without creating any unintended dependencies. This approach ensures data integrity and avoids redundancy or update anomalies in the database. Therefore, our final design meets the requirements and is free from anomalies.

Step 5/6: See MiniProj.ipynb

Step 7: See step7.py