Introduction:

Project Overview: The system will manage a diverse collection of loanable items, track various types of memberships, enforce borrowing rules, and generate meaningful reports.

Scope Statement: From the project overview: This project aims to provide hands-on experience designing, implementing, and managing a relational database system for a small library. The system will manage a diverse collection of loanable items, track various types of memberships, enforce borrowing rules, and generate meaningful reports.

Get Ready for ER Modeling.

Here's a draft of the Data Entities section for our Library Management System project, outlining key entities, attributes, data types, constraints, and relationships:

Entities

1. Item

Represents physical and digital items available for loan.

- ItemID ((Primary Key) – INT, Auto-increment: Unique identifier for each book.
- Title – VARCHAR(255): Title of the item.
- PublicationDate – DATE: The date the item was published.
- Genre – VARCHAR(100): The genre of the item.
- AvailabilityStatus – BOOLEAN: Indicates whether the item is available (TRUE) or loaned out (FALSE).
- Digital/PhysicalType -VARCHAR(10): Type of the item ('Digital', 'Physical')

  Sub-Entities of Item
  - Book - Represents books in the library.
    - ISBN – VARCHAR(13): International Standard Book Number, must be unique.
  - Magazine - Represents magazines in the library.
    - IssueNumber - INT: Issue number or volume of the magazine.
  - DigitalMedia - Represents digital media in the library
    - MediaType - VARCHAR(10): Type of digital media ('AudioBook', 'Movie', 'Podcast')

2. Author

Represents individuals or groups who create items available in the library

- AuthorID (Primary Key) – INT, Auto-increment: Unique identifier for each author.
- Name – (Complex Attribute)
    - FirstName - VARCHAR(255): The author's first name.
    - LastName - VARCHAR(255): The author's last name.

3. Staff

Represents staff members employed by the library.

- StaffID (Primary Key) – INT, Auto-increment: Unique identifier for each staff member.
- Role - VARCHAR(10): Position of the staff member ('Manager', 'Employee', 'Owner')
- PhoneNumber – CHAR(10), exactly 10 numbers
- Name - (Complex Attribute)
    - FirstName - VARCHAR(100): The author's first name.
    - LastName - VARCHAR(100): The author's last name.

4. Client

Represents library users who can borrow items.

- ClientID (Primary Key) – INT, Auto-increment: Unique identifier for each client.
- Name – (Complex Attribute)
    - FirstName - VARCHAR(255): The client's first name.
    - LastName - VARCHAR(255): The client's last name.
- PhoneNumber – CHAR(10), exactly 10 numbers
- MembershipType – VARCHAR(50): Membership type (e.g., 'Regular', 'Student', 'Senior').
- AccountStatus – VARCHAR(20): Status of the client account (e.g., 'Active', 'Suspended', or 'Pending Deletion')

Relations

1. Loans

Represents the borrowing of Items

- LoanID (Primary Key) – INT, Auto-increment: Unique identifier for each loan transaction.
- ItemID – INT, Auto-increment: Unique identifier for each book.

- ClientID – INT, Auto-increment: Unique identifier for each client.
- LoanDate – DATE: The date the item was borrowed.
- DueDate – DATE: The date the item is due to be returned.
- ReturnDate – DATE, Nullable: The actual return date of the item (if returned).
- LateFee – DECIMAL, Nullable: The fee incurred if the item is returned late.

2. Reserves
Represents reservations made by Clients

- ReservationID (Primary Key) – INT, Auto-increment: Unique identifier for each reservation.
- ItemID – INT, Auto-increment: Unique identifier for each book.
- ClientID – INT, Auto-increment: Unique identifier for each client.
- ReservationDate – DATE: The date when the reservation was made.
- ReservationDueDate – DATE: The date the item must be picked up or the reservation will expire

Staff manages Client

- One Staff member can manage multiple Clients.
- Cardinality: One-to-Many

Client reserves Item

- Each client can reserve multiple items, and multiple clients can reserve each item.
- Cardinality: Many-to-Many

Author creates Item

- Each author can create multiple items, and multiple authors can create each item.
- Cardinality: Many-to-Many

Staff loans Item

- Each staff member can loan multiple items, and multiple staff members can loan each item.
- Cardinality: Many-to-Many