**PROJECT-2**

**Part-1 & Part-2:**

**Team Members:**

Priyanka Cheerla(1002082168)

SaiKiran Anumala(1002074049)

**Design Overview**:

1. Entities and Classes:

**Members:** Represents general information about library members.

**LibraryStaff:** Represents staff working in the library.

ReferenceLibrarian , CheckoutStaff, LibraryAssistant, ChiefLibrarian: Subclasses of LibraryStaff , representing specific staff roles.

**Subjects**: Represents subject areas for books.

**Books:** Represents library books with details such as title, author, and description.

**Catalog:** Associates books with specific catalog entries.

**BorrowingActivity**: Tracks borrowing activities, including borrow and return dates.

**SubjectsBooks:** Associates books with subject areas.

**HomeMember:**  Subclass of Members for members with home addresses.

Certainly! Let's define the classes, subclasses, and their corresponding tables along with the entity relationships in a clear way.

**Classes and Subclasses:**

1. Class: Members

- Attributes: MemberID (PK), SSN, CampusAddress, HomeAddress, CompanyAddress, PhoneNumber, MemberCardNumber, ExpirationDate, GracePeriod, Active

2. HomeMember (Subclass of Members)

- Attributes: None additional

3. Subclass: CompanyMember (Subclass of Members)

- Attributes: CompanyName

4. Class: LibraryStaff

- Attributes: StaffID (PK), Name, Role

5. Subclass: ReferenceLibrarian (Subclass of LibraryStaff)

- Attributes: Specialty

6. Subclass: CheckoutStaff (Subclass of LibraryStaff)

- Attributes: CheckoutLimit

7. Subclass: LibraryAssistant (Subclass of LibraryStaff)

- Attributes: Department

8. Subclass: ChiefLibrarian (Subclass of LibraryStaff)

- Attributes: ManagementResponsibility

9. Class: Books

- Attributes: ISBN (PK), Title, Author, SubjectArea, Description, Copies, ReferenceStatus

10. Class: Catalog

- Attributes: CatalogID (PK), ISBN (FK), Description

11. Class: Subjects

- Attributes: SubjectID (PK), Name

12. Class: BorrowingActivity

- Attributes: BorrowID (PK), MemberID (FK), BookISBN (FK), BorrowDate, ReturnDate, Returned

**Entity Relationships**:

1.Members - HomeMember Relationship:

* One-to-One relationship between Members and HomeMember.
* Members.MemberID (PK) relates to HomeMember.MemberID (PK).

2.Members - CompanyMember Relationship:

* One-to-One relationship between Members and CompanyMember.
* Members.MemberID (PK) relates to CompanyMember.MemberID (PK).

3.LibraryStaff - ReferenceLibrarian Relationship:

* One-to-One relationship between LibraryStaff and ReferenceLibrarian.
* LibraryStaff.StaffID (PK) relates to ReferenceLibrarian.StaffID (PK).

4.LibraryStaff - CheckoutStaff Relationship:

* One-to-One relationship between LibraryStaff and CheckoutStaff.
* LibraryStaff.StaffID (PK) relates to CheckoutStaff.StaffID (PK).

5.LibraryStaff - LibraryAssistant Relationship:

* One-to-One relationship between LibraryStaff and LibraryAssistant.
* LibraryStaff.StaffID (PK) relates to LibraryAssistant.StaffID (PK).

6.LibraryStaff - ChiefLibrarian Relationship:

* One-to-One relationship between LibraryStaff and ChiefLibrarian.
* LibraryStaff.StaffID (PK) relates to ChiefLibrarian.StaffID (PK).

7.Books - Catalog Relationship:

* One-to-Many relationship between Books and Catalog.
* Books.ISBN (PK) relates to Catalog.ISBN (FK).

8.Books - Subjects Relationship:

* Many-to-Many relationship between Books and Subjects using an associative table.
* Books.ISBN (PK) relates to SubjectsBooks.ISBN (FK).
* Subjects.SubjectID (PK) relates to SubjectsBooks.SubjectID (FK).

9.Members - BorrowingActivity Relationship:

* One-to-Many relationship between Members and BorrowingActivity.
* Members.MemberID (PK) relates to BorrowingActivity.MemberID (FK).

10.Books - BorrowingActivity Relationship:

* One-to-Many relationship between Books and BorrowingActivity.
* Books.ISBN (PK) relates to BorrowingActivity.BookISBN (FK).

These relationships establish the connections between different classes and subclasses in the EER schema. Adjustments can be made based on specific requirements or clarifications.

**Assumptions:**

1. **Subjects and SubjectsBooks**: Many-to-many relationship between subjects and books. Introduce a separate Subjects. Books table for this association.

2**. HomeMember**: Create a subclass for members with home addresses. Assume home address is different from the campus address.

**3. CheckoutStaff**: Introduce a subclass for staff with checkout responsibilities. Include a checkout limit for each staff member.

**4. ChiefLibrarian:**

Introduce a subclass for the Chief Librarian with management responsibilities.

**5. BorrowingActivity:**

Include borrow and return dates to track borrowing timelines. Assume that returned books are marked with 'Yes' in the Returned field.

**Missing/Incomplete Requirements:**

1. **Detailed Book Information**: Lack details such as publication date, edition, and publisher. Additional attributes might be needed based on the library's requirements.

2. **Member's Company Address**: Include a field for a member's company address, but there's no explicit use or requirement mentioned in the provided data. It might be incomplete if there are specific functionalities or relationships associated with the company address.

3.**Detailed LibraryStaff Roles**: Roles of ReferenceLibrarian, CheckoutStaff, LibraryAssistant, and ChiefLibrarian are defined broadly. Detailed specifications or responsibilities associated with each role may be missing.

4. **Explicit Constraints**: Constraints are implemented, but specific business rules driving these constraints are not explicitly defined.

5. **Additional Relationships**: Assumed relationships between entities, especially subclasses, are sufficient based on common library management scenarios. Any specific relationships unique to the library's operations may be missing.