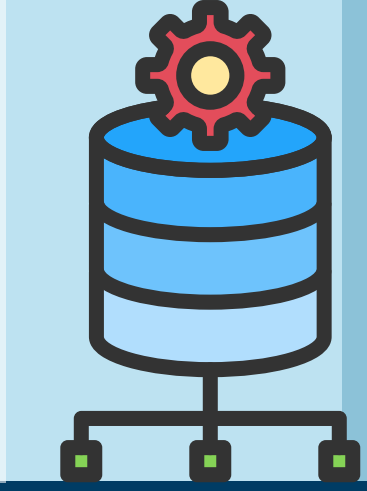
Qassim University
جامعة القصيم
College Of Computer | كلية الحاسب



IT332-Advanced Database project

Submitted by:

Manar almutairi

CONTENTS

1.Requirements Document

1.1 Data Requirement

- **BOOK**

The library has a variety of books, each identified by a unique ISBN (ISBN13 system). The data includes:
- ISBN
- Title
- Author
- Publisher
- Publication Year
- Location (Books are located on shelves 1, 2, and 3)
- Status (Available / Not Available)

- **VIDEO & AUDIO RECORDING**

The library contains video and audio recordings, each identified by a unique ID. The data includes:
- ID
- Type of Material (Video/Audio)
- Title
- Publisher
- Description
- Location (Recordings are located on shelves 4 and 5)
- Status (Available / Not Available)

- **THESIS**

The library also has a collection of theses, each identified by a unique ID. The data includes:
- ID
- Title
- Author
- Description
- Location (Theses are located on shelves 6 and 7)
- Status (Available / Not Available)

1.Requirements Document

- **JOURNAL**

The library possesses many academic journals, each identified by a unique ID. The data includes:
- ID
- Title
- Description
- Location (Journals are located on shelves 8 and 9)
- Status (Available / Not Available)

- **SHORT LOAN**

Some materials in the library cannot be loaned outside, such as literary titles and specialized items. The data includes:
- Item ID
- Title
- Type
- Status (Available / Not Available)
- Maximum Reservation Duration (Can be reserved for up to 2 hours)

- **ROOM**

Various types of rooms (lab, reading, meetings) are available for booking. The data includes:
- Room ID
- Type
- Duration (in hours, cannot be less than 1 hour)
- Status (Available / Not Available)

. MEMBER

To benefit from library services, individuals must become members. The data includes:
- MemberID (PK)
- Name
- DOB
- Address
- Contact
- JoinDate
- ExpiryDate
- Status
- PIN
- MemberTypeID (FK

1.Requirements Document

- ## MEMBER TYPE

This entity defines the types of members and their access levels. The data includes:
- TypeID (PK)
- TypeName
- Description

- ## PRIVILEGE

Privileges are granted based on member type. The data includes:
- PrivilegeID (PK)
- MemberTypeID (FK)
- LoanPeriod (in days, cannot exceed 14 days)
- MaxItems (cannot exceed 5 items)
- MaxRenewals (cannot exceed 3 renewals)

- ## FINE

Fines can be imposed on members who do not adhere to library rules. The data includes:
- FineID (PK)
- MemberID (FK)
- Amount (between 20 and 1000 SR)
- Description
- Status (Paid / Not Paid)

.LOAN

Loan requests are tracked using:
- LoanID (PK)
- MemberID (FK)
- ItemID (FK)
- DateLoaned
- DueDate
- Status (Returned / Not Returned)

1.Requirements Document

- **SHORT LOAN RESERVATION**

Reservations for materials used inside the library are tracked with:
- ReservationID (PK)
- MemberID (FK)
- ItemID (FK)
- StartTime
- EndTime

1.2 Transaction Requirements

. Data Entry:

- Enter details of a new book.
- Enter details of a new video/audio recording.
- Enter details of a new thesis.
- Enter details of a new journal.
- Enter details of a new short loan item.
- Enter details of a new room.
- Enter details of a new member.
- Enter details of a new privilege.
- Enter details of a new fine.
- Enter details of a new loan.
- Enter details of a new reservation.

. Data Update / Deletion:

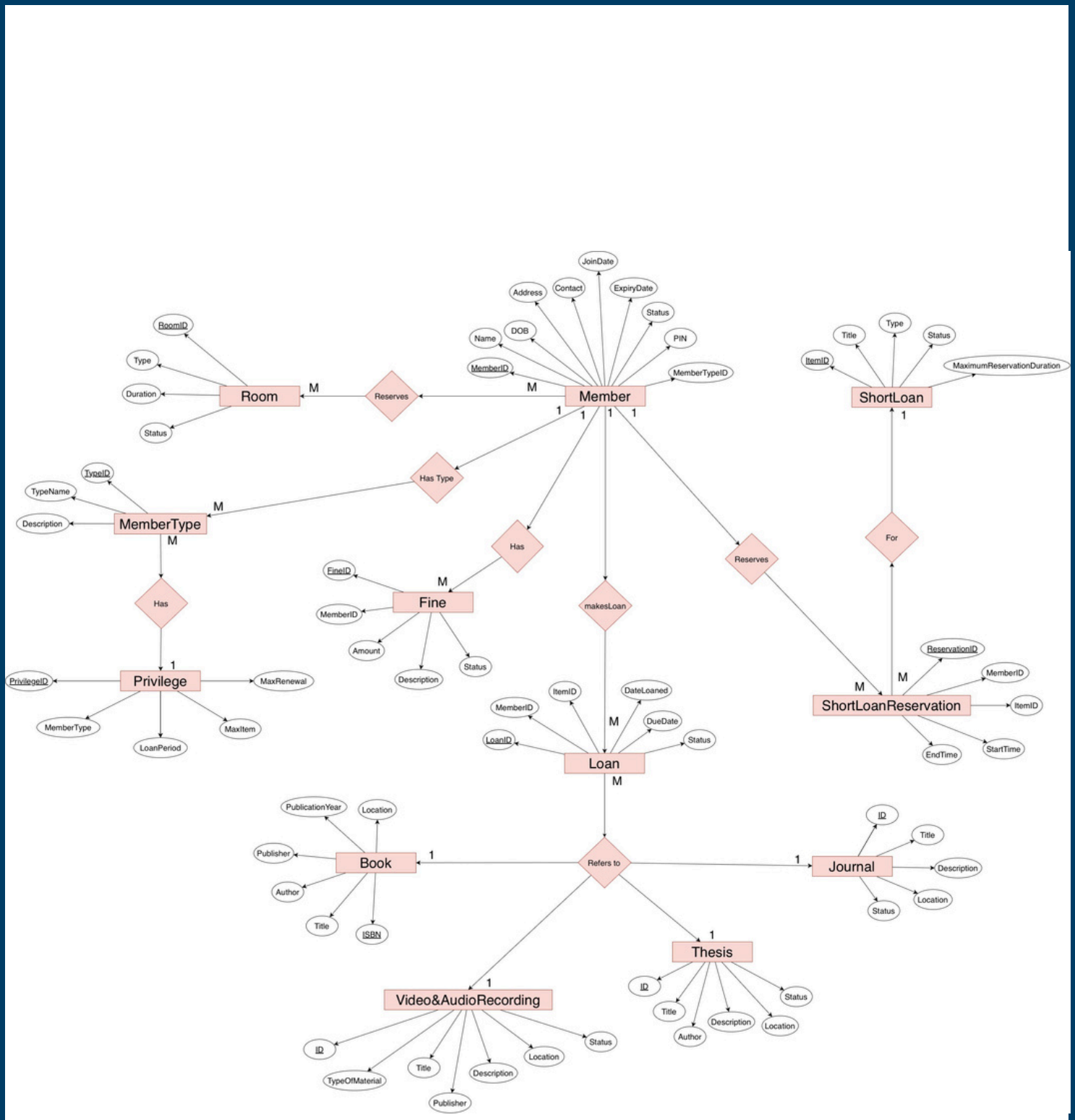
- Update/delete details of a book.
- Update/delete details of a video/audio recording.
- Update/delete details of a thesis.
- Update/delete details of a journal.
- Update/delete details of a short loan item.
- Update/delete details of a room.
- Update/delete details of a member.
- Update/delete details of a privilege.
- Update/delete details of a fine.
- Update/delete details of a loan.
- Update/delete details of a reservation.

1.Requirements Document

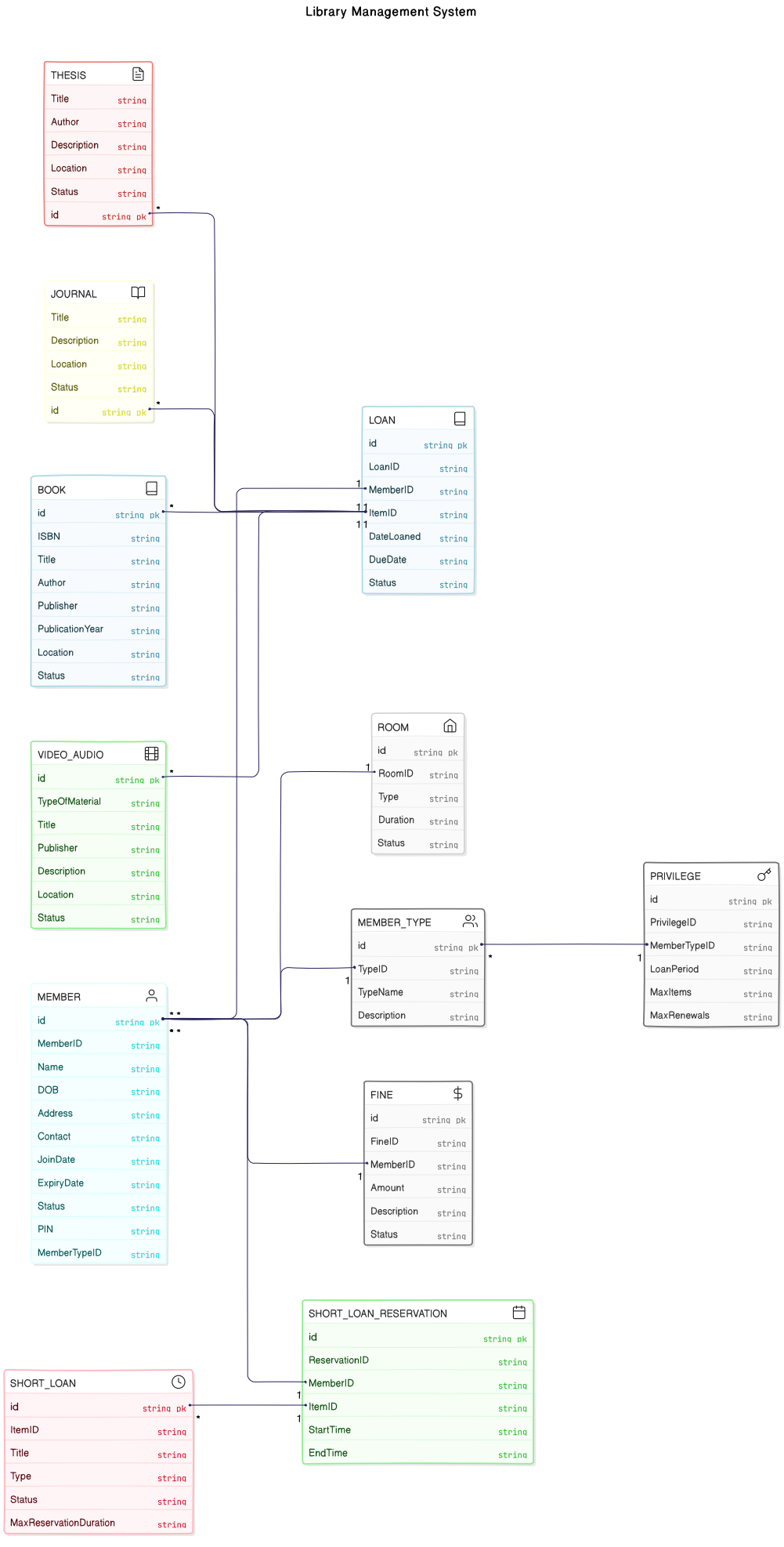
1.3 Data Queries

- Search for items by title, author, or keyword.
- Check availability of specific books or materials.
- View all overdue loans with associated fines.
- Display books written by a specific author.
- Show currently available rooms.
- Display all members with unpaid fines.
- Show details of a room reservation made by a specific member.
- Display all members who joined on a specific date.

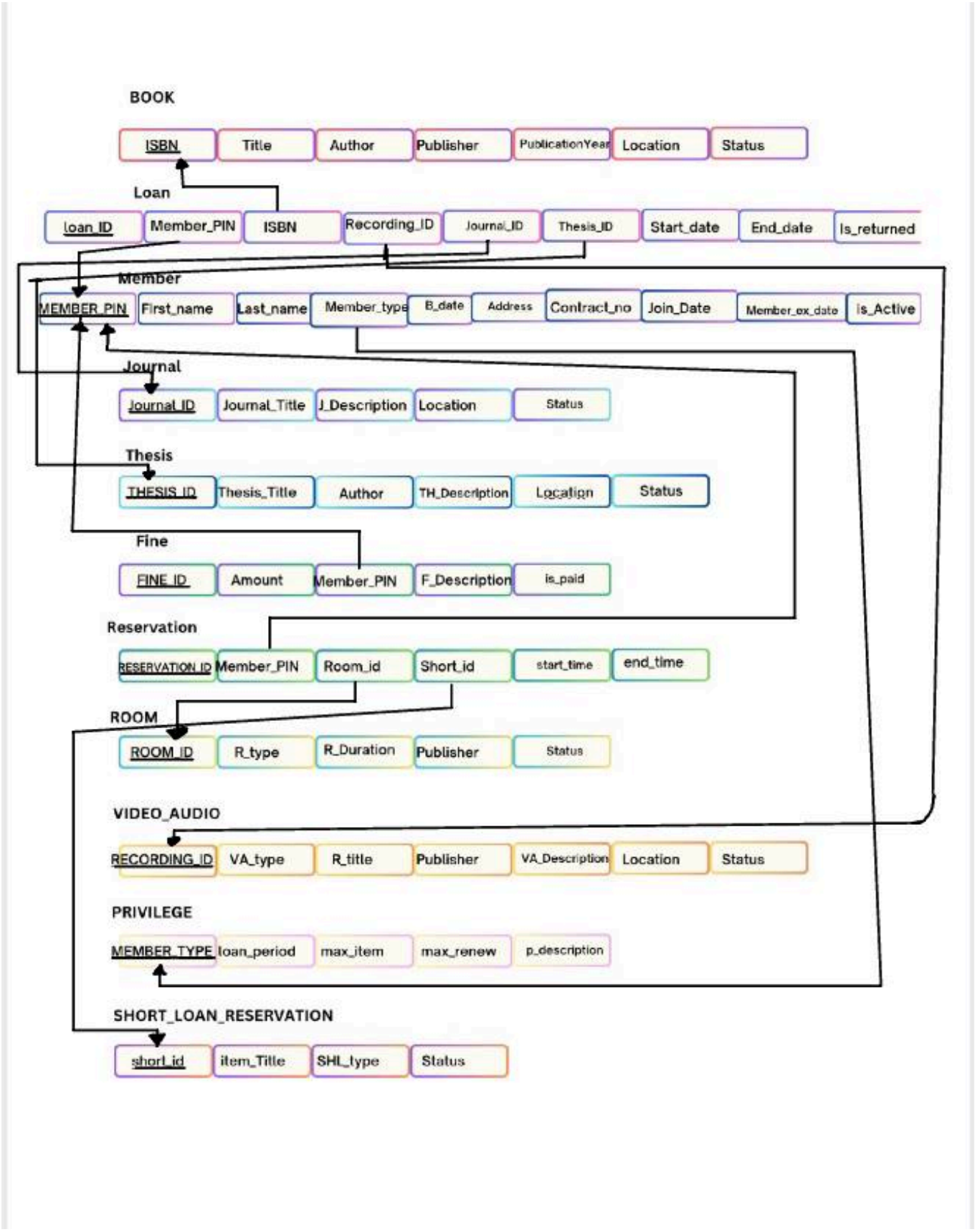
2.ERR Diagram



2.UML Diagram



3. Relational Mapping



4. Normalization

Book:

1NF:

BOOK TABLE:

ISBN	Title	Notes	Status	Cname

AUTHOR_BOOK TABLE:

ISBN	Author	Publisher	Publication_year

2NF:

BOOK TABLE:

ISBN	Title	Notes	Status	Cname

AUTHOR_BOOK TABLE:

ISBN	Author

PUBLISHER_BOOK TABLE:

ISBN	Publisher	Publisher_year

4. Normalization

VID_AUD_REC:

1NF:

VID_AUD_REC TABLE:

Recording_ID	VA_Type	Publisher	VA_Description	Status	Shelf	Row

2NF:

VID_AUD_REC TABLE:

Recording_ID	VA_Type	Publisher	VA_Description	Status

REC_LOCATION TABLE:

Recording_ID	Shelf	Row

THESIS:

1NF:

THESIS TABLE:

Thesis_ID	Thesis_Title	TH_Description	Author	Status	Shelf	Row

4. Normalization

2NF:

THESIS TABLE:

Thesis_ID	Thesis_Title	TH_Description	Author	Status

THESIS_AUTHOR TABLE:

Thesis_ID	Author

THESIS_Location TABLE:

Thesis_ID	Shelf	Row

JOURNAL

1NF:

JOURNAL TABLE:

Journal_ID	Journal_Title	J_Description	Status	Shelf	Row

4. Normalization

2NF:

JOURNAL TABLE:

Journal_ID	Journal_Title	J_Description	Status

JOURNAL_LOCATION TABLE:

Journal_ID	Shelf	Row

JOURNAL_LOCATION TABLE:

1NF = 2NF

SHORT_LOAN_ITEMS TABLE:

Short_ID	Item_Title	SHL_Type	Status

ROOM:

1NF = 2NF

Room_ID	R_Type	R_Duration	Status

4. Normalization

PRIVILEGE

1NF = 2NF

Member_Type	Loan_period	Max_Items	Max_Renew	P_Description

MEMBER:

1NF:

MEMBER TABLE:

Member_PIN	Member_Fname	Member_Lname	Member_Type	B_Date	Address	Contact_No	Join_Date	Member_EX_Date	Is_Active

2NF:

MEMBER TABLE:

Member_PIN	Member_Fname	Member_Lname	Member_Type	B_Date	Address	Contact_No

MEMBERSHIP_DETAILS TABLE:

Member_PIN	Join_Date	Member_EX_Date	Is_Active

4. Normalization

FINE

1NF = 2NF

FINE TABLE:

Fine_ID	Amount	Member_PIN	F_Description	Is_Paid

LOAN + Sub-loans:

1NF:

LOAN TABLE:

Loan_ID	Member_PIN	Item_ID	Start_Date	End_Date	Is_Returned

2NF:

LOAN TABLE:

Loan_ID	Member_PIN	BOOK_LOAN	REC_LOAN	THESIS_LOAN

JOURNAL_LOAN TABLES:

Loan_ID	Item_ID	Start_Date	End_Date	Is_Returned

4. Normalization

RESERVATION + Sub-reservations

1NF:

RESERVATION TABLE:

Reservation_ID	Member_PIN	Item_ID	Start_Time	End_Time

2NF:

RESERVATION TABLE:

Reservation_ID	Member_PIN	ROOM_RESERVATION

SHORT_RESERVATION TABLES:

Reservation_ID	Room	Short_ID	Start_Time	End_Time

5.Table Creation

```
CREATE TABLE Privilege (
    Member_Type VARCHAR(20) PRIMARY KEY,
    Loan_period INT,
    Max_Items INT,
    Max_Renew INT,
    P_Description VARCHAR(200)
);
```

```
CREATE TABLE Member (
    Member_PIN VARCHAR(20) NOT NULL,
    Member_Fname VARCHAR(50) NOT NULL,
    Member_Lname VARCHAR(50) NOT NULL,
    Member_Type VARCHAR(20),
    B_Date DATE,
    Address VARCHAR(100),
    Contact_No VARCHAR(15),
    PRIMARY KEY (Member_PIN),
    FOREIGN KEY (Member_Type)
        REFERENCES Privilege(Member_Type)
        ON DELETE SET NULL
);
```

```
CREATE TABLE Membership_Details (
    Member_PIN VARCHAR(10) PRIMARY KEY,
    Join_Date DATE,
    Member_EX_Date DATE,
    Is_Active CHAR(1),
    FOREIGN KEY (Member_PIN) REFERENCES Member(Member_PIN) ON DELETE CASCADE
);
```

```
CREATE TABLE Book (
    ISBN VARCHAR(20) PRIMARY KEY,
    Book_Title VARCHAR(100),
    Publisher VARCHAR(50),
    Publication_Year INT,
    Status VARCHAR(15)
);
CREATE TABLE Book_Author (
    Author_ID VARCHAR(10) PRIMARY KEY,
    Author_Fname VARCHAR(30),
    Author_Lname VARCHAR(30),
    ISBN VARCHAR(20),
    FOREIGN KEY (ISBN) REFERENCES Book(ISBN) ON DELETE SET NULL
);
CREATE TABLE Book_location (
    ISBN VARCHAR(20) PRIMARY KEY,
    Shelf VARCHAR(10),
    Roww VARCHAR(10),
    FOREIGN KEY (ISBN) REFERENCES Book(ISBN) ON DELETE CASCADE
);
CREATE TABLE Vid_Aud_Rec (
    Recording_ID VARCHAR(10) PRIMARY KEY,
    VA_Type VARCHAR(20),
    Publisher VARCHAR(50),
    VA_Description VARCHAR(200),
    Status VARCHAR(15)
);
```

5.Table Creation

```
CREATE TABLE Rec_location (
    Recording_ID VARCHAR(10) PRIMARY KEY,
    Shelf VARCHAR(10),
    Roww VARCHAR(10),
    FOREIGN KEY (Recording_ID) REFERENCES Vid_Aud_Rec(Recording_ID) ON DELETE CASCADE
);
CREATE TABLE Thesis (
    Thesis_ID VARCHAR(10) PRIMARY KEY,
    Thesis_Title VARCHAR(100),
    TH_Description VARCHAR(200),
    Status VARCHAR(15)
);
```

```
CREATE TABLE Thesis_Author (
    Author_ID NUMBER NOT NULL,
    Author_Fname VARCHAR(50) NOT NULL,
    Author_Lname VARCHAR(50) NOT NULL,
    Thesis_ID VARCHAR(10),
    PRIMARY KEY (Author_ID),
    FOREIGN KEY (Thesis_ID) REFERENCES Thesis(Thesis_ID) ON DELETE SET NULL
);
CREATE TABLE Thesis_location (
    Thesis_ID VARCHAR(10) PRIMARY KEY,
    Shelf VARCHAR(10),
    Roww VARCHAR(10),
    FOREIGN KEY (Thesis_ID) REFERENCES Thesis(Thesis_ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Journal (
    Journal_ID VARCHAR(10) PRIMARY KEY,
    Journal_Title VARCHAR(100),
    J_Description VARCHAR(200),
    Status VARCHAR(15)
);
```

```
CREATE TABLE Journal_Location (
    Journal_ID VARCHAR(10) PRIMARY KEY,
    Shelf VARCHAR(10),
    Roww VARCHAR(10),
    FOREIGN KEY (Journal_ID) REFERENCES Journal(Journal_ID) ON DELETE CASCADE
);
CREATE TABLE Short_Loan_Items (
    Shrt_ID VARCHAR(10) PRIMARY KEY,
    Item_Title VARCHAR(100),
    SHL_Type VARCHAR(20),
    Status VARCHAR(15)
);
```

```
CREATE TABLE Room (
    Room_ID VARCHAR(10) PRIMARY KEY,
    R_Type VARCHAR(20),
    R_Duration INT,
    Status VARCHAR(15)
);
CREATE TABLE Fine (
    Fine_ID VARCHAR(10) PRIMARY KEY,
    Amount DECIMAL(10,2),
    Member_PIN VARCHAR(10),
    F_Description VARCHAR(200),
    Is_Paid CHAR(1),
    FOREIGN KEY (Member_PIN) REFERENCES Member(Member_PIN) ON DELETE CASCADE );
```

```
CREATE TABLE Loan (
    Loan_ID VARCHAR(10) PRIMARY KEY,
    Member_PIN VARCHAR(10),
    FOREIGN KEY (Member_PIN) REFERENCES Member(Member_PIN) ON DELETE CASCADE
);
CREATE TABLE Book_Loan (
    Book_Loan_ID VARCHAR(10) PRIMARY KEY,
    Loan_ID VARCHAR(10),
    ISBN VARCHAR(20),
    Start_date DATE,
    End_date DATE,
    Is_Returned CHAR(1),
    FOREIGN KEY (Loan_ID) REFERENCES Loan(Loan_ID),
```

FOREIGN KEY (ISBN) REFERENCES Book(ISBN) ON DELETE CASCADE);

5.Table Creation

```
CREATE TABLE Rec_Loan (
    Rec_Loan_ID VARCHAR(10) PRIMARY KEY,
    Loan_ID VARCHAR(10),
    Recording_ID VARCHAR(10),
    Start_date DATE,
    End_date DATE,
    Is_Returned CHAR(1),
    FOREIGN KEY (Loan_ID) REFERENCES Loan(Loan_ID),
    FOREIGN KEY (Recording_ID) REFERENCES Vid_Aud_Rec(Recording_ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Thesis_Loan (
    Thesis_Loan_ID VARCHAR(10) PRIMARY KEY,
    Loan_ID VARCHAR(10),
    Thesis_ID VARCHAR(10),
    Start_date DATE,
    End_date DATE,
    Is_Returned CHAR(1),
    FOREIGN KEY (Loan_ID) REFERENCES Loan(Loan_ID),
    FOREIGN KEY (Thesis_ID) REFERENCES Thesis(Thesis_ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Journal_Loan (
    Journal_Loan_ID VARCHAR(10) PRIMARY KEY,
    Loan_ID VARCHAR(10),
    Journal_ID VARCHAR(10),
    Start_date DATE,
    End_date DATE,
    Is_Returned CHAR(1),
    FOREIGN KEY (Loan_ID) REFERENCES Loan(Loan_ID),
    FOREIGN KEY (Journal_ID) REFERENCES Journal(Journal_ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Reservation (
    Reservation_ID VARCHAR(10) PRIMARY KEY,
    Member_PIN VARCHAR(10),
    FOREIGN KEY (Member_PIN) REFERENCES Member(Member_PIN) ON DELETE CASCADE
);
```

```
);
CREATE TABLE Room_Reservation (
    Room_Reservation_ID VARCHAR(10) PRIMARY KEY,
    Reservation_ID VARCHAR(10),
    Room_ID VARCHAR(10),
    Start_Time TIMESTAMP,
    End_Time TIMESTAMP,
    FOREIGN KEY (Reservation_ID) REFERENCES Reservation(Reservation_ID),
    FOREIGN KEY (Room_ID) REFERENCES Room(Room_ID) ON DELETE CASCADE
);
```

```
);
CREATE TABLE Shrt_Reservation (
    Shrt_Reservation_ID VARCHAR(10) PRIMARY KEY,
    Reservation_ID VARCHAR(10),
    Shrt_ID VARCHAR(10),
    Start_Time TIMESTAMP,
    End_Time TIMESTAMP,
    FOREIGN KEY (Reservation_ID) REFERENCES Reservation(Reservation_ID),
    FOREIGN KEY (Shrt_ID) REFERENCES Short_Loan_Items(Shrt_ID) ON DELETE CASCADE
);
```

5.Table Creation

```
Table PRIVILEGE created.

Table MEMBER created.

Table MEMBERSHIP_DETAILS created.

Table BOOK created.
```

```
Table THESIS_LOCATION created.

Table THESIS_AUTHOR created.
```

```
Table THESIS_LOCATION created.

Table THESIS_AUTHOR created.
```

```
Table JOURNAL created.

Table JOURNAL_LOCATION created.

Table SHORT_LOAN_ITEMS created.

Table ROOM created.
```

```
Table BOOK_LOCATION created.

Table VID_AUD_REC created.

Table REC_LOCATION created.

Table THESIS created.
```

```
Table BOOK_LOCATION created.

Table VID_AUD_REC created.

Table REC_LOCATION created.

Table THESIS created.
```

```
Table MEMBER created.

Table MEMBERSHIP_DETAILS created.

Table BOOK created.
```

```
Table THESIS_LOCATION created.

Table THESIS_AUTHOR created.
```

6.insert into tables

```
INSERT INTO Privilege VALUES ('Student', 14, 5, 2, 'Standard student privilege');
INSERT INTO Privilege VALUES ('Teacher', 30, 10, 4, 'Extended loan period');
INSERT INTO Privilege VALUES ('Researcher', 21, 8, 3, 'Special research privileges');


INSERT INTO Member VALUES ('M001', 'Sara', 'Ali', 'Student', TO_DATE('2001-05-10',
'YYYY-MM-DD'), 'Riyadh', '0551234567');
INSERT INTO Member VALUES ('M002', 'Fahad', 'Omar', 'Teacher', TO_DATE('1980-02-15',
'YYYY-MM-DD'), 'Jeddah', '0567891234');
INSERT INTO Member VALUES ('M003', 'Noura', 'Khalid', 'Researcher', TO_DATE('1990-11-
20', 'YYYY-MM-DD'), 'Dammam', '0543219876');


INSERT INTO Membership_Details VALUES ('M001', TO_DATE('2021-01-01', 'YYYY-MM-
DD'), TO_DATE('2025-01-01', 'YYYY-MM-DD'), 'Y');
INSERT INTO Membership_Details VALUES ('M002', TO_DATE('2020-06-15', 'YYYY-MM-
DD'), TO_DATE('2024-06-15', 'YYYY-MM-DD'), 'Y');
INSERT INTO Membership_Details VALUES ('M003', TO_DATE('2022-03-10', 'YYYY-MM-
DD'), TO_DATE('2026-03-10', 'YYYY-MM-DD'), 'Y');


INSERT INTO Book VALUES ('ISBN001', 'Database Systems', 'O'Reilly', 2020, 'Available');
INSERT INTO Book VALUES ('ISBN002', 'Operating Systems', 'Pearson', 2019,
'Borrowed');
INSERT INTO Book VALUES ('ISBN003', 'Artificial Intelligence', 'MIT Press', 2021,
'Available');

INSERT INTO Book_Author VALUES ('A001', 'Elmasri', 'Navathe', 'ISBN001');
INSERT INTO Book_Author VALUES ('A002', 'Silberschatz', 'Galvin', 'ISBN002');
INSERT INTO Book_Author VALUES ('A003', 'Stuart', 'Russell', 'ISBN003');


INSERT INTO Book_location VALUES ('ISBN001', 'A1', '3');
INSERT INTO Book_location VALUES ('ISBN002', 'B2', '5');
INSERT INTO Book_location VALUES ('ISBN003', 'C3', '2');


INSERT INTO Vid_Aud_Rec VALUES ('VA001', 'Video', 'Oxford', 'Intro to Databases',
'Available');
INSERT INTO Vid_Aud_Rec VALUES ('VA002', 'Audio', 'Cambridge', 'OS Concepts
Summary', 'Available');
INSERT INTO Vid_Aud_Rec VALUES ('VA003', 'Video', 'Stanford', 'AI Lectures',
'Borrowed');
```

6.insert into tables

```
INSERT INTO Rec_location VALUES ('VA001', 'C1', '1');
INSERT INTO Rec_location VALUES ('VA002', 'C2', '4');
INSERT INTO Rec_location VALUES ('VA003', 'C3', '2');


INSERT INTO Thesis VALUES ('T001', 'AI in Education', 'Study of AI techniques in modern learning',
'Available');
INSERT INTO Thesis VALUES ('T002', 'Blockchain Security', 'Research on blockchain systems',
'Borrowed');
INSERT INTO Thesis VALUES ('T003', 'Quantum Computing', 'Advances in quantum algorithms',
'Available');


INSERT INTO Thesis_Author VALUES (101, 'Layla', 'Hassan', 'T001');
INSERT INTO Thesis_Author VALUES (102, 'Mohammed', 'Salem', 'T002');
INSERT INTO Thesis_Author VALUES (103, 'Ahmed', 'Farooq', 'T003');


INSERT INTO Thesis_location VALUES (1, 'T001', 'D1', '2');
INSERT INTO Thesis_location VALUES (2, 'T002', 'D2', '3');
INSERT INTO Thesis_location VALUES (3, 'T003', 'D3', '1');


INSERT INTO Journal VALUES ('J001', 'Tech Today', 'Latest trends in IT', 'Available');
INSERT INTO Journal VALUES ('J002', 'Cyber Weekly', 'Cybersecurity updates', 'Borrowed');
INSERT INTO Journal VALUES ('J003', 'Data Science Review', 'DS research papers', 'Available');


INSERT INTO Journal_Location VALUES ('J001', 'E1', '1');
INSERT INTO Journal_Location VALUES ('J002', 'E2', '2');
INSERT INTO Journal_Location VALUES ('J003', 'E3', '3');


INSERT INTO Short_Loan_Items VALUES ('SH001', 'Data Brief', 'Report', 'Available');
INSERT INTO Short_Loan_Items VALUES ('SH002', 'Fast Learn OS', 'Notes', 'Available');
INSERT INTO Short_Loan_Items VALUES ('SH003', 'Quick AI Guide', 'Handbook', 'Borrowed');


INSERT INTO Room VALUES ('R001', 'Study', 2, 'Available');
INSERT INTO Room VALUES ('R002', 'Conference', 3, 'Occupied');
INSERT INTO Room VALUES ('R003', 'Meeting', 1, 'Available');
INSERT INTO Fine VALUES ('F001', 15.00, 'M001', 'Late return', 'N');
INSERT INTO Fine VALUES ('F002', 10.00, 'M002', 'Lost item', 'Y');
INSERT INTO Fine VALUES ('F003', 5.00, 'M003', 'Damaged item', 'N');


INSERT INTO Loan VALUES ('L001', 'M001');
INSERT INTO Loan VALUES ('L002', 'M002');
INSERT INTO Loan VALUES ('L003', 'M003');
```

6.insert into tables

```
INSERT INTO Book_Loan VALUES ('BL001', 'L001', 'ISBN001', TO_DATE('2025-04-01', 'YYYY-MM-DD'),
TO_DATE('2025-04-10', 'YYYY-MM-DD'), 'N');
INSERT INTO Book_Loan VALUES ('BL002', 'L002', 'ISBN002', TO_DATE('2025-03-15', 'YYYY-MM-DD'),
TO_DATE('2025-03-25', 'YYYY-MM-DD'), 'Y');
INSERT INTO Book_Loan VALUES ('BL003', 'L003', 'ISBN003', TO_DATE('2025-04-05', 'YYYY-MM-DD'),
TO_DATE('2025-04-20', 'YYYY-MM-DD'), 'N');

INSERT INTO Rec_Loan VALUES ('RL001', 'L001', 'VA001', TO_DATE('2025-04-01', 'YYYY-MM-DD'),
TO_DATE('2025-04-05', 'YYYY-MM-DD'), 'N');
INSERT INTO Rec_Loan VALUES ('RL002', 'L002', 'VA002', TO_DATE('2025-03-20', 'YYYY-MM-DD'),
TO_DATE('2025-03-30', 'YYYY-MM-DD'), 'Y');
INSERT INTO Rec_Loan VALUES ('RL003', 'L003', 'VA003', TO_DATE('2025-04-10', 'YYYY-MM-DD'),
TO_DATE('2025-04-15', 'YYYY-MM-DD'), 'N');


INSERT INTO Thesis_Loan VALUES ('TL001', 'L001', 'T001', TO_DATE('2025-04-01', 'YYYY-MM-DD'),
TO_DATE('2025-04-15', 'YYYY-MM-DD'), 'N');
INSERT INTO Thesis_Loan VALUES ('TL002', 'L002', 'T002', TO_DATE('2025-03-01', 'YYYY-MM-DD'),
TO_DATE('2025-03-10', 'YYYY-MM-DD'), 'Y');
INSERT INTO Thesis_Loan VALUES ('TL003', 'L003', 'T003', TO_DATE('2025-04-03', 'YYYY-MM-DD'),
TO_DATE('2025-04-18', 'YYYY-MM-DD'), 'N');


INSERT INTO Journal_Loan VALUES ('JL001', 'L001', 'J001', TO_DATE('2025-04-01', 'YYYY-MM-DD'),
TO_DATE('2025-04-05', 'YYYY-MM-DD'), 'N');
INSERT INTO Journal_Loan VALUES ('JL002', 'L002', 'J002', TO_DATE('2025-03-10', 'YYYY-MM-DD'),
TO_DATE('2025-03-15', 'YYYY-MM-DD'), 'Y');
INSERT INTO Journal_Loan VALUES ('JL003', 'L003', 'J003', TO_DATE('2025-04-08', 'YYYY-MM-DD'),
TO_DATE('2025-04-12', 'YYYY-MM-DD'), 'N');


INSERT INTO Reservation VALUES ('RS001', 'M001');
INSERT INTO Reservation VALUES ('RS002', 'M002');
INSERT INTO Reservation VALUES ('RS003', 'M003');


INSERT INTO Room_Reservation VALUES ('RR001', 'RS001', 'R001', TO_TIMESTAMP('2025-04-10
10:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-10 12:00:00', 'YYYY-MM-DD
HH24:MI:SS'));
INSERT INTO Room_Reservation VALUES ('RR002', 'RS002', 'R002', TO_TIMESTAMP('2025-04-11
14:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-11 17:00:00', 'YYYY-MM-DD
HH24:MI:SS'));
INSERT INTO Room_Reservation VALUES ('RR003', 'RS003', 'R003', TO_TIMESTAMP('2025-04-12
09:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-12 10:00:00', 'YYYY-MM-DD
HH24:MI:SS'));


INSERT INTO Shrt_Reservation VALUES ('SR001', 'RS001', 'SH001', TO_TIMESTAMP('2025-04-12 09:00:00',
'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-12 11:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Shrt_Reservation VALUES ('SR002', 'RS002', 'SH002', TO_TIMESTAMP('2025-04-13 13:00:00',
'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-13 15:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Shrt_Reservation VALUES ('SR003', 'RS003', 'SH003', TO_TIMESTAMP('2025-04-14 10:00:00',
'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2025-04-14 12:00:00', 'YYYY-MM-DD HH24:MI:SS'));
```

6.insert into tables

SELECT * TABLE Privilege;

MEMBER_TYPE	LOAN_PERIOD	MAX_ITEMS	MAX_RENEW	P_DESCRIPTION
Student	14	5	2	Standard student privilege
Teacher	30	10	4	Extended loan period
Researcher	21	8	3	Special research privileges

SELECT * FROM Member;

MEMBER_PIN	MEMBER_FNAME	MEMBER_LNAME	MEMBER_TYPE	B_DATE	ADDRESS	CONTACT_NO
M001	Sara	Ali	Student	10-MAY-01	Riyadh	0551234567
M002	Fahad	Omar	Teacher	15-FEB-80	Jeddah	0567891234
M003	Noura	Khalid	Researcher	20-NOV-90	Dammam	0543219876

SELECT * FROM Membership_Details;

MEMBER_PIN	JOIN_DATE	MEMBER_EX_DATE	IS_ACTIVE
M001	01-JAN-21	01-JAN-25	Y
M002	15-JUN-20	15-JUN-24	Y
M003	10-MAR-22	10-MAR-26	Y

SELECT * FROM Book;

ISBN	BOOK_TITLE	PUBLISHER	PUBLICATION_YEAR	STATUS
ISBN001	Database Systems	O'Reilly	2020	Available
ISBN002	Operating Systems	Pearson	2019	Borrowed
ISBN003	Artificial Intelligence	MIT Press	2021	Available

SELECT * FROM Book_Author;

AUTHOR_ID	AUTHOR_FNAME	AUTHOR_LNAME	ISBN
A001	Elmasri	Navathe	ISBN001
A002	Silberschatz	Galvin	ISBN002
A003	Stuart	Russell	ISBN003

SELECT * FROM Book_location;

ISBN	SHELF	ROWW
ISBN001	A1	3
ISBN002	B2	5
ISBN003	C3	2

6.insert into tables

SELECT * FROM Vid_Aud_Rec;

RECORDING_ID	VA_TYPE	PUBLISHER	VA_DESCRIPTION	STATUS
VA001	Video	Oxford	Intro to Databases	Available
VA002	Audio	Cambridge	OS Concepts Summary	Available
VA003	Video	Stanford	AI Lectures	Borrowed

SELECT * FROM Rec_location;

RECORDING_ID	SHELF	ROWW
VA001	C1	1
VA002	C2	4
VA003	C3	2

SELECT * FROM Thesis;

THESIS_ID	THESIS_TITLE	TH_DESCRIPTION	STATUS
T001	AI in Education	Study of AI techniques in modern learning	Available
T002	Blockchain Security	Research on blockchain systems	Borrowed
T003	Quantum Computing	Advances in quantum algorithms	Available

SELECT * FROM Thesis_Author;

AUTHOR_ID	AUTHOR_FNAME	AUTHOR_LNAME	THESIS_ID
101	Layla	Hassan	T001
102	Mohammed	Salem	T002
103	Ahmed	Farooq	T003

SELECT * FROM Thesis_location;

LOCATION_ID	THESIS_ID	SHELF	ROWW
1	T001	D1	2
2	T002	D2	3
3	T003	D3	1

SELECT * FROM Journal;

JOURNAL_ID	JOURNAL_TITLE	J_DESCRIPTION	STATUS
J001	Tech Today	Latest trends in IT	Available
J002	Cyber Weekly	Cybersecurity updates	Borrowed
J003	Data Science Review	DS research papers	Available

6.insert into tables

SELECT * FROM Journal_Location;

JOURNAL_ID	SHELF	ROWW
J001	E1	1
J002	E2	2
J003	E3	3

SELECT * FROM Short_Loan_Items;

SHRT_ID	ITEM_TITLE	SHL_TYPE	STATUS
SH001	Data Brief	Report	Available
SH002	Fast Learn OS	Notes	Available
SH003	Quick AI Guide	Handbook	Borrowed

SELECT * FROM Room;

ROOM_ID	R_TYPE	R_DURATION	STATUS
R001	Study	2	Available
R002	Conference	3	Occupied
R003	Meeting	1	Available

SELECT * FROM Fine;

FINE_ID	AMOUNT	MEMBER_PIN	F_DESCRIPTION	IS_PAID
F001	15	M001	Late return	N
F002	10	M002	Lost item	Y
F003	5	M003	Damaged item	N

SELECT * FROM Loan;

LOAN_ID	MEMBER_PIN
L001	M001
L002	M002
L003	M003

SELECT * FROM Book_Loan;

BOOK_LOAN_ID	LOAN_ID	ISBN	START_DATE	END_DATE	IS_RETURNED
BL001	L001	ISBN001	01-APR-25	10-APR-25	N
BL002	L002	ISBN002	15-MAR-25	25-MAR-25	Y
BL003	L003	ISBN003	05-APR-25	20-APR-25	N

6.insert into tables

SELECT * FROM Rec_Loan;

REC_LOAN_ID	LOAN_ID	RECORDING_ID	START_DATE	END_DATE	IS_RETURNED
RL001	L001	VA001	01-APR-25	05-APR-25	N
RL002	L002	VA002	20-MAR-25	30-MAR-25	Y
RL003	L003	VA003	10-APR-25	15-APR-25	N

SELECT * FROM Thesis_Loan;

THESIS_LOAN_ID	LOAN_ID	THESIS_ID	START_DATE	END_DATE	IS_RETURNED
TL001	L001	T001	01-APR-25	15-APR-25	N
TL002	L002	T002	01-MAR-25	10-MAR-25	Y
TL003	L003	T003	03-APR-25	18-APR-25	N

SELECT * FROM Journal_Loan;

JOURNAL_LOAN_ID	LOAN_ID	JOURNAL_ID	START_DATE	END_DATE	IS_RETURNED
JL001	L001	J001	01-APR-25	05-APR-25	N
JL002	L002	J002	10-MAR-25	15-MAR-25	Y
JL003	L003	J003	08-APR-25	12-APR-25	N

SELECT * FROM Reservation;

RESERVATION_ID	MEMBER_PIN
RS001	M001
RS002	M002
RS003	M003

SELECT * FROM Room_Reservation;

ROOM_RESERVATION_ID	RESERVATION_ID	ROOM_ID	START_TIME	END_TIME
RR001	RS001	R001	10-APR-25 10.00.00.000000 AM	10-APR-25 12.00.00.000000 PM
RR002	RS002	R002	11-APR-25 02.00.00.000000 PM	11-APR-25 05.00.00.000000 PM
RR003	RS003	R003	12-APR-25 09.00.00.000000 AM	12-APR-25 10.00.00.000000 AM

SELECT * FROM Shrt_Reservation;

SHRT_RESERVATION_ID	RESERVATION_ID	SHRT_ID	START_TIME	END_TIME
SR001	RS001	SH001	12-APR-25 09.00.00.000000 AM	12-APR-25 11.00.00.000000 AM
SR002	RS002	SH002	13-APR-25 01.00.00.000000 PM	13-APR-25 03.00.00.000000 PM
SR003	RS003	SH003	14-APR-25 10.00.00.000000 AM	14-APR-25 12.00.00.000000 PM

7. view Creation and Testing

- *View to show all available books with their authors and status*

```
CREATE VIEW vBookStatus AS
SELECT
    b.ISBN,
    b.Book_Title,
    b.Publisher,
    b.Publication_Year,
    b.Status AS Book_Status,
    ba.Author_Fname || ' ' || ba.Author_Lname AS Author_Name,
    bl.Shelf,
    bl.Roww
FROM
    Book b
JOIN
    Book_Author ba ON b.ISBN = ba.ISBN
JOIN
    Book_location bl ON b.ISBN = bl.ISBN
WHERE
    b.Status = 'Available';
```

7. view Creation and Testing

- *Shows active members with their details and privileges.*

```
CREATE VIEW Active_Members_View AS
SELECT m.Member_PIN, m.Member_Fname, m.Member_Lname, m.Member_Type,
    p.P_Description, md.Join_Date, md.Member_EX_Date
FROM Member m
JOIN Membership_Details md ON m.Member_PIN = md.Member_PIN
JOIN Privilege p ON m.Member_Type = p.Member_Type
WHERE md.Is_Active = 'Y';
```

- *Lists available books with author and location details.*

```
CREATE VIEW Available_Books_View AS
SELECT b.ISBN, b.Book_Title, b.Publisher, b.Publication_Year,
    ba.Author_Fname || ' ' || ba.Author_Lname AS Author_Name,
    bl.Shelf, bl.Roww
FROM Book b
JOIN Book_Author ba ON b.ISBN = ba.ISBN
JOIN Book_location bl ON b.ISBN = bl.ISBN
WHERE b.Status = 'Available';
```

- *Displays currently borrowed items (books, theses, journals, recordings).*

```
CREATE VIEW Current_Loans_View AS
SELECT l.Loan_ID, m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname AS
Member_Name,
    COALESCE(b.Book_Title, r.VA_Description, t.Thesis_Title, j.Journal_Title) AS Item_Title,
    COALESCE(bl.End_date, rl.End_date, tl.End_date, jl.End_date) AS Due_Date,
    CASE
      WHEN bl.Loan_ID IS NOT NULL THEN 'Book'
      WHEN rl.Loan_ID IS NOT NULL THEN 'Recording'
      WHEN tl.Loan_ID IS NOT NULL THEN 'Thesis'
      WHEN jl.Loan_ID IS NOT NULL THEN 'Journal'
    END AS Item_Type
FROM Loan l
JOIN Member m ON l.Member_PIN = m.Member_PIN
LEFT JOIN Book_Loan bl ON l.Loan_ID = bl.Loan_ID AND bl.Is_Returned = 'N'
LEFT JOIN Book b ON bl.ISBN = b.ISBN
LEFT JOIN Rec_Loan rl ON l.Loan_ID = rl.Loan_ID AND rl.Is_Returned = 'N'
LEFT JOIN Vid_Aud_Rec r ON rl.Recording_ID = r.Recording_ID
LEFT JOIN Thesis_Loan tl ON l.Loan_ID = tl.Loan_ID AND tl.Is_Returned = 'N'
LEFT JOIN Thesis t ON tl.Thesis_ID = t.Thesis_ID
LEFT JOIN Journal_Loan jl ON l.Loan_ID = jl.Loan_ID AND jl.Is_Returned = 'N'
LEFT JOIN Journal j ON jl.Journal_ID = j.Journal_ID
WHERE (bl.Loan_ID IS NOT NULL OR rl.Loan_ID IS NOT NULL OR
    tl.Loan_ID IS NOT NULL OR jl.Loan_ID IS NOT NULL);
```

7. view Creation and Testing

- *Shows all active loans for a specific member*

```
CREATE VIEW Member_Loans_View AS
SELECT m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname AS
Member_Name,
    b.Book_Title AS Item_Title, 'Book' AS Item_Type,
    bl.Start_date, bl.End_date AS Due_Date
FROM Member m
JOIN Loan l ON m.Member_PIN = l.Member_PIN
JOIN Book_Loan bl ON l.Loan_ID = bl.Loan_ID
JOIN Book b ON bl.ISBN = b.ISBN
WHERE bl.Is_Returned = 'N'
UNION ALL
SELECT m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname,
    r.VA_Description, 'Recording',
    rl.Start_date, rl.End_date
FROM Member m
JOIN Loan l ON m.Member_PIN = l.Member_PIN
JOIN Rec_Loan rl ON l.Loan_ID = rl.Loan_ID
JOIN Vid_Aud_Rec r ON rl.Recording_ID = r.Recording_ID
WHERE rl.Is_Returned = 'N'
UNION ALL
SELECT m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname,
    t.Thesis_Title, 'Thesis',
    tl.Start_date, tl.End_date
FROM Member m
JOIN Loan l ON m.Member_PIN = l.Member_PIN
JOIN Thesis_Loan tl ON l.Loan_ID = tl.Loan_ID
JOIN Thesis t ON tl.Thesis_ID = t.Thesis_ID
WHERE tl.Is_Returned = 'N'
UNION ALL
SELECT m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname,
    j.Journal_Title, 'Journal',
    jl.Start_date, jl.End_date
FROM Member m
JOIN Loan l ON m.Member_PIN = l.Member_PIN
JOIN Journal_Loan jl ON l.Loan_ID = jl.Loan_ID
JOIN Journal j ON jl.Journal_ID = j.Journal_ID
WHERE jl.Is_Returned = 'N';
```

7. view Creation and Testing

- ***Lists upcoming room reservations with member details.***

```
CREATE VIEW Unpaid_Fines_View AS
SELECT f.Fine_ID, f.Amount,
    m.Member_PIN, m.Member_Fname || ' ' || m.Member_Lname AS Member_Name,
    f.F_Description, f.Is_Paid
FROM Fine f
JOIN Member m ON f.Member_PIN = m.Member_PIN
WHERE f.Is_Paid = 'N';
```

7. view Creation and Testing

- ***SELECT * FROM vBookStatus;***

ISBN	BOOK_TITLE	PUBLISHER	PUBLICATION_YEAR	BOOK_STATUS	AUTHOR_NAME	SHELF	ROWW
ISBN001	Database Systems	O'Reilly	2020	Available	Elmasri Navathe	A1	3
ISBN003	Artificial Intelligence	MIT Press	2021	Available	Stuart Russell	C3	2

7. view Creation and Testing

- ***SELECT * FROM Active_Members_View;***

MEMBER_PIN	MEMBER_FNAME	MEMBER_LNAME	MEMBER_TYPE	P_DESCRIPTION	JOIN_DATE	MEMBER_EX_DATE
M001	Sara	Ali	Student	Standard student privilege	01-JAN-21	01-JAN-25
M002	Fahad	Omar	Teacher	Extended loan period	15-JUN-20	15-JUN-24
M003	Noura	Khalid	Researcher	Special research privileges	10-MAR-22	10-MAR-26

- ***SELECT * FROM Available_Books_View;***

ISBN	BOOK_TITLE	PUBLISHER	PUBLICATION_YEAR	AUTHOR_NAME	SHELF	ROWW
ISBN001	Database Systems	O'Reilly	2020	Elmasri Navathe	A1	3
ISBN003	Artificial Intelligence	MIT Press	2021	Stuart Russell	C3	2

- ***SELECT * FROM Current_Loans_View;***

LOAN_ID	MEMBER_PIN	MEMBER_NAME	ITEM_TITLE	DUE_DATE	ITEM_TYPE
L001	M001	Sara Ali	Database Systems	10-APR-25	Book
L003	M003	Noura Khalid	Artificial Intelligence	20-APR-25	Book

- ***SELECT * FROM Member_Loans_View WHERE Member_PIN = 'M001';***

MEMBER_PIN	MEMBER_NAME	ITEM_TITLE	ITEM_TYPE	START_DATE	DUE_DATE
M001	Sara Ali	Database Systems	Book	01-APR-25	10-APR-25
M001	Sara Ali	Intro to Databases	Recording	01-APR-25	05-APR-25
M001	Sara Ali	AI in Education	Thesis	01-APR-25	15-APR-25
M001	Sara Ali	Tech Today	Journal	01-APR-25	05-APR-25

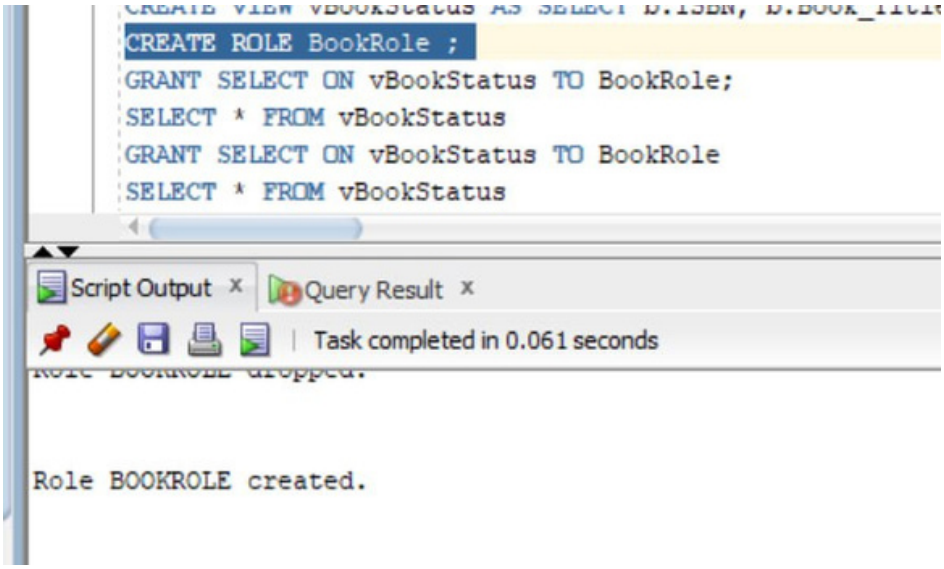
- ***SELECT * FROM Unpaid_Fines_View;***

FINE_ID	AMOUNT	MEMBER_PIN	MEMBER_NAME	F_DESCRIPTION	IS_PAID
F001	15	M001	Sara Ali	Late return	N
F003	5	M003	Noura Khalid	Damaged item	N

8.Role Creation and Permissions Setup

- *Create the BookRole*

CREATE ROLE BookRole;



- *Grant SELECT permission on vBookStatus to BookRole*

GRANT SELECT ON vBookStatus TO BookRole;