Coursework 1 solution

Library Database Management System
ECS740P – Database Systems

Environment: Oracle Live SQL

PART 1: DATABASE DESIGN

I. Conceptual Diagram

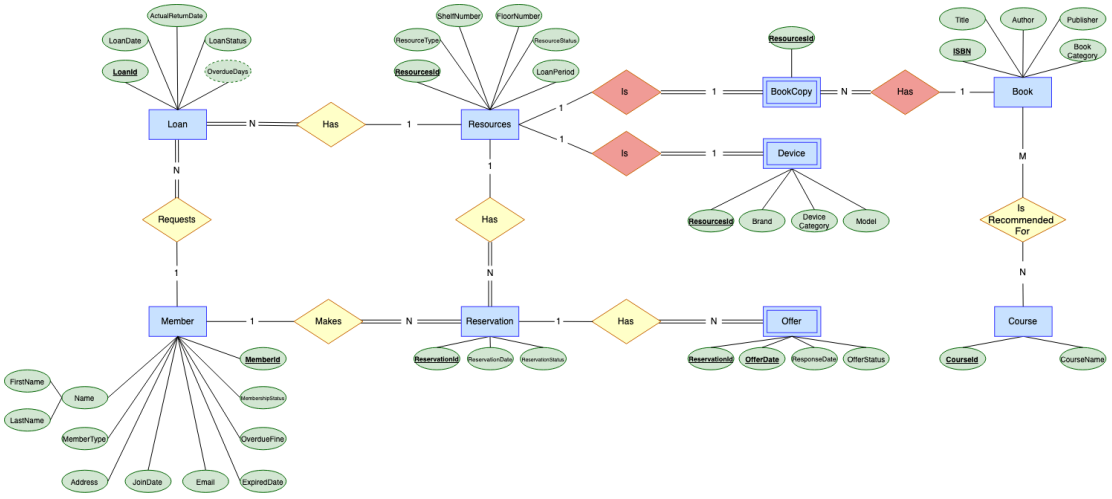
The main purpose of the college library database system is to keep track of:

- Each library member (students and staff), including membership status and overdue fines
- Each library resource (books and devices), including the number of copies (if available), its location, its course numbers if the resource is a book
- Each loan and reservation as well as failures to take up offers on reserved items when they become available

Thus, the main entities required for this system are: **Member, Resources, Device, Book, BookCopy, Course, Loan, Reservation, and Offer**. Based on specific requirements and certain assumptions, the required attributes, primary keys **(bold and underlined)**, and the relationships between entities are identified as follows:

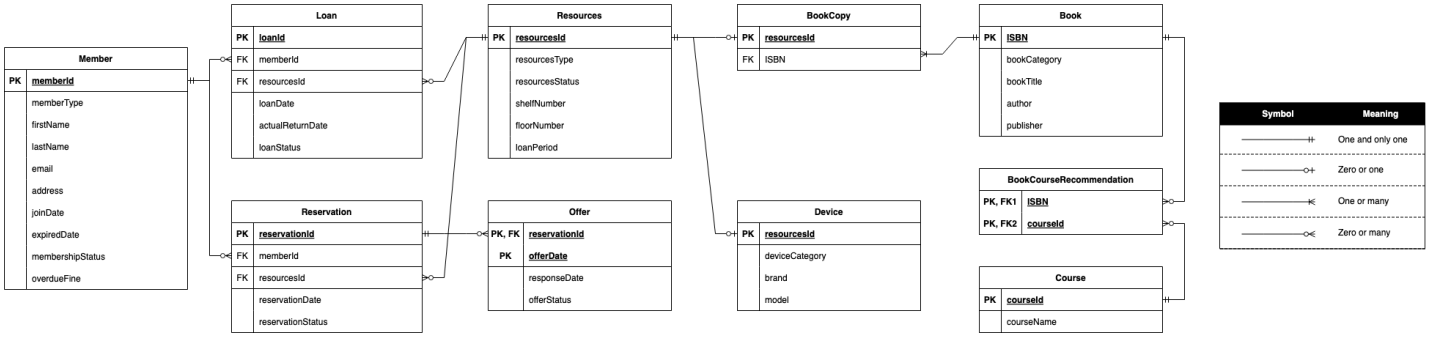
Entity	Attributes	Assumption	Relationship between entities
Member	**MemberId**, MemberType, FirstName, LastName Email, Address, JoinDate, ExpiredDate, MembershipStatus , OverdueFine	1. There are 2 member types: student or staff 2. The full name is split into first and last names to improve search flexibility. 3. Each member has only one email and one address. 4. The address is stored as a single attribute, primarily for basic correspondence, as it is not frequently queried. 5. There are 3 types of membership status: "*Active*", "*Expired*" or "*Suspended*" 6. Overdue fines for each member can be calculated as the sum of overdue days across all of their loans. Although OverdueFine can be derived, it is stored to facilitate frequent tracking and updates.	
Resources	**ResourcesId** ResourcesType, ShelfNumber, FloorNumber, LoanPeriod, ResourcesStatus	7. A resource must be either a device or a book copy. 8. There are 2 types of resource status: *"Available"* and *"Unavailable".* 9. Each resource has a fixed loan period, expressed in days. If a resource can only be used within the library, its loan period is recorded as 0 day.	Resources partially participate in relationship with Book. Similarly, Resources partially participate in relationship with Device.
Book	**ISBN,** BookCategory, BookTitle, Author, Publisher	10. A book can have multiple copies. 11. Every physical book and eBook is associated with an ISBN. 12. Books are categorised (i.e. Literature, Mathematics, Biology, ...) for easier search functionality. 13. Each book only has 1 author and 1 publisher.	Book - BookCopy is a one-to-many relationship and BookCopy totally participates in the relationship.
BookCopy	**ResourcesId**	14. eBooks have a limited number of digital copies, which limits simultaneous access. 15. Each book copy is a library resource.	BookCopy - Resources is one-to-one relationship and BookCopy totally participates in the relationship.
Device	**ResourcesId,** DeviceCategory, Brand, Model	16. Devices are physical resources stored on shelves like physical books. 17. Devices are categorised (i.e. Laptop, Tablet, eBook readers,...) for easier search functionality. 18. Each device is a library resource.	Device - Resources is one-to-one relationship and Device totally participates in the relationship.
Loan	**LoanId,** LoanDate, ActualReturnDate, LoanStatus, OverdueDays	19. If a resource is unavailable, members can't loan the resource. They will have to request a reservation instead. 20. There are 3 types of loan status: "*Loaned"*, *"Returned"*, *"Overdue"* 21. OverdueDays for each resource can be derived using this formula: OverdueDays = ActualReturnDate - (LoanDate + LoanPeriod) 22. A resource can be loaned multiple times. The LoanId is generated each time a member loans a resource.	Resources - Loan is one-to-many relationship and Loan totally participates in the relationship. Member - Loan is one-to-many relationship and Loan totally participates in the relationship.

		23. If a member loans multiple resources on the same ReservationDate, it is considered that reservations were made at the same time. 24. If a member loans multiple resources at the same time, a unique LoanId is generated for each resource loaned. 25. A resource cannot be loaned to different members at the same time and can only be loaned to another member once returned. 26. Each loan must specify a specific resource and a member.	
Reservation	**ReservationId**, ReservationDate, ReservationStatus	27. ReservationId is generated when a member makes a reservation on a resource. If a member reserves multiple resources on the same ReservationDate, it is considered reservations were made at the same time. 28. A reservation must specify a specific resource and a member. 29. A resource can be reserved multiple times. 30. There are 3 types of reservation status: "*Reserved, Completed, Cancelled*"; Reservation is *completed* when a member accepts the offer for the reservation; Reservation is *cancelled* when there are 3 declined or expired offers for that reservation. 31. ReservationId is auto-incremented, which determines the reservation order.	Resources - Reservation is one-to-many relationship and Reservation totally participates in the relationship. Member - Reservation is one-to-many relationship and Reservation totally participates in the relationship.
Offer	**ReservationId**, **OfferDate**, ResponseDate, OfferStatus	32. Offers can not exist without a reservation. 33. A reservation can have up to 3 offers. 34. There are 4 types of OfferStatus: "*Given*, *Accepted, Declined, Expired*". 35. If no response is received within 3 days from OfferDate, the offer is expired. 36. For each reservationId, there is only one offer sent out on the same day.	Offer is a weak entity, that depends on the Reservation entity. Reservation - Offer is one-to-many relationship and Offer totally participates in the relationship
Course	**CourseId**, CourseName	37. A book can be recommended for many courses and a course can recommend many books. Not every book needs to be recommended for a course.	Book and Course has many-to-many relationship.



II. Relational Schema

- **For strong entity types**: Each strong entity is turned into a relation (table); each simple attribute is turned into a column in that relation (Watt & Eng, 2014). The composite attribute "Name" in Member entity is broken down into "FirstName" and "LastName".
- **For weak entity types**: Weak entities (Offer, BookCopy, Device) inherit primary keys from its parent entities.
  - For BookCopy and Device entities, they inherit the primary key ResourcesId from Resources entity, which can uniquely identify each row in each entity.
  - For entity Offer, the primary key is a composite primary key consisting of the inherited ReservationId from its owner entity and OfferDate, which can uniquely identify each row in the Offer entity.
- **For binary 1-N relationship**:
  - Add MemberId, ResourcesId as foreign keys to Loan and Reservation which are N-side entities in the relationship.
  - Add ReservationId as foreign key to Offer which are N-side entities in Resources - Offer relationship.
  - Add ISBN to BookCopy as foreign key because BookCopy is on the N-side in the relationship.
- **For binary M-N relationship**: Create a new relation BookCourseRecommendation holding foreign keys from both Book and Course. ISBN and CourseId combine as a composite primary key to uniquely identify each row in BookCourseRecommendation.
- **Our final relation schemas** are as below where primary key is bold and underlined, and foreign key is annotated as (FK):
  - Member (**MemberId**, MemberType, FirstName, LastName, Email, Address, JoinDate, ExpiredDate, MembershipStatus, OverdueFine)
  - Resources (**ResourcesId**, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod)
  - Book (**ISBN**, BookCategory, BookTitle, Author, Publisher)
  - BookCopy (**ResourcesId (FK)**, ISBN (FK))
  - Device (**ResourcesId (FK)**, DeviceCategory, Brand, Model)
  - Loan (**LoanId**, MemberId (FK), ResourcesId (FK), LoanDate, ActualReturnDate, LoanStatus)
  - Reservation (**ReservationId**, MemberId (FK), ResourcesId (FK), ReservationDate, ReservationStatus)
  - Offer **(ReservationId (FK), OfferDate,** ResponseDate, OfferStatus)
  - Course (**CourseId**, CourseName)
  - BookCourseRecommendation (**ISBN (FK), CourseId (FK)**)



III. Normalisation

1. Universal Relation

- Based on the specification document, the following are the data attributes required to serve the purpose of the application:
  U (MemberId, MemberType, FirstName, LastName, Email, Address, JoinDate, ExpiredDate, MembershipStatus, OverdueFine, ISBN, BookTitle, Author, Publisher, DeviceCategory, Brand, Model, ResourcesId, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod, LoanId, LoanDate, ActualReturnDate, LoanStatus, ReservationId, ReservationDate, ReservationStatus, OfferDate, ResponseDate, OfferStatus, CourseId, CourseName)
- The primary key of this universal relation is the combination of MemberId, ResourcesId, LoanId, ReservationId, OfferDate, CourseId

2. Functional Dependencies

- Member's ID number can determine the member's type, their contact information, their join date and expired date of library card, membership status and total overdue fine:
  MemberId ⇒ MemberType, FirstName, LastName, Email, Address, JoinDate, ExpiredDate, MembershipStatus, OverdueFine
- ResourcesId can determine resources type, its current status, location and the allowed period for loan, book details if resource is book and device details if resource is device:
  ResourcesId ⇒ *ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod, DeviceCategory, Brand, Model, ISBN, BookCategory, BookTitle, Author, Publisher*
- ISBN can determine resource type and book details: *ISBN ⇒ ResourcesType, BookCategory, BookTitle, Author, Publisher*
- DeviceCategory (i.e tablet, laptop, e-Book reader …) can determine ResourcesType*: DeviceCategory ⇒ ResourcesType*
- Loan's ID number can determine which member requested the loan, which resources were loaned, when it was loaned, the actual date when it was returned and the loan status:
  LoanId ⇒ MemberId, ResourcesId, LoanDate, ActualReturnDate, LoanStatus

- Reservation's ID number can determine which member made the reservation, which resources were reserved when it was reserved, and the reservation status:
  ReservationId ⇒ MemberId, ResourcesId, ReservationDate, ReservationStatus
- ReservationId and OfferDate can determine the offer status, and response date:
  {ReservationId, OfferDate} ⇒ ResponseDate, OfferStatus
- Course's ID can determine the course name:
  CourseId ⇒ CourseName

3. Normalising into 3NF

1. **The universal relation is in 1NF** as it contains simple, single values and no repeating groups (Silberschatz, Korth & Sudarshan, 2019)
2. **Moving into 2NF**:

There are partial dependencies in the universal relation because certain non-key attributes only depend on parts of the primary key and not on the entire primary key of the universal relation (Silberschatz, Korth & Sudarshan, 2019). Therefore, based on functional dependencies, thus those attributes will be separated and included in the new relation only the part of the primary key on which they functionally depend. This leads to the following 2NF relations:

- Member (**MemberId**, MemberType, FirstName, LastName, Email, Address, JoinDate, ExpiredDate, MembershipStatus, OverdueFine) with primary key as MemberId
- Resources (**ResourcesId**, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod, DeviceCategory, Brand, Model, ISBN, BookCategory, BookTitle, Author, Publisher) with ResourcesId as primary key
- Loan (**LoanId**, MemberId, ResourcesId, LoanDate, ActualReturnDate, LoanStatus) with LoanId as primary key.
- Reservation (**ReservationId**, MemberId, ResourcesId, ReservationDate, ReservationStatus) with ReservationId as primary key.
- Offer (**ReservationId, OfferDate**, ResponseDate, OfferStatus) with ReservationId and OfferDate as composite primary key.
- Course (**CourseId**, CourseName) with CourseId as primary key
- BookCourseRecommendation (**CourseId, ISBN**) with a combination of CourseId and ISBN as composite primary key

3. **Moving into 3NF**:

By definition, a relation is in 3NF if for any pair of attributes A & B such that A → B, there is no attribute such that A → X and X → B (Silberschatz, Korth & Sudarshan, 2019). Based on this definition, Resources is not in 3NF. This is because BookCategory, BookTitle, Author, Publisher depend on ResourcesId through ISBN. Hence, attributes that are transitively dependent on the primary key from the original relation will be removed, while keeping the attributes on which the removed attributes are directly dependent to act as a foreign key, linking the old relation to new relations (Silberschatz, Korth & Sudarshan, 2019).

This results in the following relation as follows:

- Resources (**ResourcesId**, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod, DeviceCategory, Brand, Model, ISBN)
- Book (**ISBN**, BookCategory, BookTitle, Author, Publisher)

Resources entity is, however, not in BCNF form. By definition, a relation is in BCNF if and only if every determinant is a candidate key (Silberschatz, Korth & Sudarshan, 2019). There are the following determinants in the Resources relation:

- *ResourcesId ⇒ ResourcesType,* ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod, Brand, Model, ISBN
- *ISBN ⇒ ResourcesType*
- *DeviceCategory ⇒ ResourcesType*

ISBN and DeviceCategory determinants are not candidate keys because they don't uniquely identify all other attributes in the Resources relation. Therefore, the current Resources relation doesn't satisfy the BCNF form. As such, the Resources relation will be decomposed into the following relations such that in new relations, every determinant is also a candidate key:

- Resources (**ResourcesId**, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod)
- Device (**ResourcesId**, DeviceCategory, Brand, Model)
- BookCopy (**ResourcesId,** ISBN)

4. **Our final normalised design** in 3NF & BCNF form, therefore, comprises the following relations:

- Member (**MemberId**, MemberType, FirstName, LastName, Email, Address, JoinDate, ExpiredDate, MembershipStatus, OverdueFine) with primary key as MemberId
- Resources (**ResourcesId**, ResourcesType, ResourcesStatus, ShelfNumber, FloorNumber, LoanPeriod) with ResourcesId as primary key
- Book (**ISBN**, BookCategory, BookTitle, Author, Publisher) with ISBN as primary key
- BookCopy (**ResourcesId**, ISBN) with ResourcesId as primary key
- Device (**ResourcesId**, DeviceCategory, Brand, Model) with ResourcesId as primary key
- Loan (**LoanId**, MemberId, ResourcesId, LoanDate, ActualReturnDate, LoanStatus) with LoanId as primary key.
- Reservation (**ReservationId**, MemberId, ResourcesId, ReservationDate, ReservationStatus) with ReservationId as primary key
- Offer (**ReservationId, OfferDate**, ResponseDate, OfferStatus) with ReservationId and Offerdate as composite primary key
- Course (**CourseId**, CourseName) with CourseId as primary key
- BookCourseRecommendation (**ISBN, CourseId**) with a combination of CourseId and ISBN as composite primary key

PART 2: DATABASE IMPLEMENTATION

I. Listing of all the 'CREATE TABLE' commands with declarative constraints

1. Please refer to the script file (Group1.sql) attached in submission for all CREATE TABLE commands, together with TRIGGER
2. TRIGGER is included in order to validate and update data across multiple tables which are not feasible with normal declarative constraints:

- Requirement: The library charges fines for resources that are loaned for longer than the time allowed for that resource. For each day a resource is overdue the member is fined one pound
  ⇒ Trigger to update OverdueFine (Member table) automatically after ActualReturnDate of a loan (Loan table) is updated
- Requirement: When the amount owed in fines by a member is more than 10 pounds, that member is suspended until all resources have been returned and all fines paid in full
  ⇒ Trigger to update MembershipStatus to 'Suspended' if OverdueFine > 10
- Requirement: The total number of resources student may borrow at a given time must never exceed 5, and 10 for staff
  ⇒ Trigger to ensure loan limit at a time (this involves MemberType from Member table and number of loans count from Loan table)
- Requirement: If a member is unable to take up the offer of a loan 3 times for a given reservation, that reservation is cancelled
  ⇒ Trigger to update ReservationStatus (Reservation table) to 'Cancelled' if there are 3 failed offers in Offer table (offers that are either 'Declined' or 'Expired')
  ⇒ Trigger to ensure no more than 3 offer rows created per reservation

II. The sample test data

The data population (INSERT commands) are included together in the script file above.

III. Listing of 4 VIEWS

1. **View all pending overdue loans, including the member's details, the resource information and overdue days**
   This view identifies overdue loans that have not yet been returned. This would help in sending reminders to the members who have overdue loans (Assuming that the system has a scheduled job or transaction that automatically updates the LoanStatus to 'Overdue' based on the loan's due date compared to the current date, which is beyond the scope of the coursework)

```
CREATE VIEW PendingOverdueLoans AS
SELECT l.LoanId, l.LoanDate, l.ActualReturnDate, l.LoanStatus, m.MemberId,
m.FirstName || ' ' || m.LastName AS FullName, m.Email, r.ResourcesId, r.ResourcesType,
trunc(SYSDATE - (l.LoanDate + r.LoanPeriod)) AS OverdueDays
FROM Loan l
JOIN Member m ON l.MemberId = m.MemberId
JOIN Resources r ON l.ResourcesId = r.ResourcesId
WHERE l.LoanStatus = 'Overdue' AND l.ActualReturnDate IS NULL;
```

LOANID	LOANDATE	ACTUALRETURNDATE	LOANSTATUS	MEMBERID	FULLNAME	EMAIL	RESOURCESID	RESOURCESTYPE	OVERDUEDAYS
1	02-NOV-24	-	Overdue	10	Jane Davis	jane.davis@qmul.ac.uk	14	eBook	12
2	11-NOV-24	-	Overdue	20	Rosabella Cunningham	rosabella.cunningham@qmul.ac.uk	19	Device	12
3	12-NOV-24	-	Overdue	5	Diana Johnson	diana.johnson@qmul.ac.uk	2	Book	9

2. **View members who are on the waiting list for resource loan by each resource**

   This view helps to track members who have made reservation for a resource loan and to prioritise members based on the earliest reservation. This view can easily identify:

   - Members waiting for a specific resource loan.
   - The order in which they will be notified (i.e., the member with the earliest reservation gets notified first when the resource becomes available).

```
CREATE VIEW ReservationList AS SELECT m.FirstName, m.LastName, m.email, r.ResourcesId, r.ReservationId
FROM Reservation r
JOIN Member m ON r.MemberId = m.MemberId
WHERE r.ReservationStatus = 'Reserved'
ORDER BY r.ResourcesId, r.ReservationId;
```

FIRSTNAME	LASTNAME	EMAIL	RESOURCESID	RESERVATIONID
Ethan	Williams	ethan.williams@qmul.ac.uk	2	8
Mayu	Kishimoto	mayu.kishimoto@qmul.ac.uk	4	2
Sophia	Taylor	sophia.taylor@qmul.ac.uk	6	10
Bob	Brown	bob.brown@qmul.ac.uk	8	3
Aisha	Gupta	aisha.gupta@qmul.ac.uk	8	4
Diana	Johnson	diana.johnson@qmul.ac.uk	8	5
Sophia	Taylor	sophia.taylor@qmul.ac.uk	11	11
Bob	Brown	bob.brown@qmul.ac.uk	14	1
Diana	Johnson	diana.johnson@qmul.ac.uk	14	6
Ethan	Williams	ethan.williams@qmul.ac.uk	17	7
Ethan	Williams	ethan.williams@qmul.ac.uk	19	9
Sophia	Taylor	sophia.taylor@qmul.ac.uk	19	12

3. **View popular resources by its ranking**
   This view summarises the total number of loans for each resource, showing the number of times each resource has been loaned and ranking them in order of popularity. Resources with the highest number of loans are ranked first. The view can be used by library staff to:
   - Identify the most and least popular resources.
   - Manage resource inventory based on usage.

```
CREATE VIEW PopularResourcesRanking AS SELECT r.ResourcesType,
CASE WHEN r.ResourcesType IN ('Book', 'eBook') THEN b.ISBN || ' ' || b.BookTitle
WHEN r.ResourcesType = 'Device' THEN d.DeviceCategory || ' ' || d.Model
END AS ResourcesName, COUNT(l.loanId) AS LoanCount
FROM Resources r
JOIN Loan l ON r.ResourcesId = l.ResourcesId
LEFT JOIN BookCopy bc ON R.ResourcesId = bc.ResourcesId
LEFT JOIN Book b ON b.ISBN = bc.ISBN
LEFT JOIN Device d ON r.ResourcesId = d.ResourcesId
GROUP BY r.ResourcesType,
CASE WHEN r.ResourcesType IN ('Book', 'eBook') THEN b.ISBN || ' ' || b.BookTitle
WHEN r.ResourcesType = 'Device' THEN d.DeviceCategory || ' ' || d.Model        END
ORDER BY LoanCount DESC;
```

RESOURCESTYPE	RESOURCESNAME	LOANCOUNT
Book	978-0-07-802215-9 Database System Concepts - 7th Edition	5
Book	978-0-74-327356-5 The Great Gatsby	5
eBook	978-1-43-024209-3 Beginning Database Design: From Novice to Professional - 2nd Edition	3
Book	978-0-68-480154-4 Tender is the night	3
eBook	978-1-09-810293-7 Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	2
Device	Laptop Spectre x360	1
Device	Laptop XPS 13	1

4. **View each resource, with its course name (if available), and where these are located in the library**
   This view can be used to:

   - Help library members identify resources relevant to their academic needs (i.e. when applying further filters based on a specific course) and quickly locate where the resource is

```
CREATE VIEW ResourcesDetails AS
SELECT r.*,
CASE WHEN r.ResourcesType IN ('Book', 'eBook') THEN b.BookTitle
WHEN r.ResourcesType = 'Device' THEN d.Brand || ' ' || d.Model
END AS ResourcesName, c.CourseName
FROM Resources r
LEFT JOIN Device d ON r.resourcesId = d.resourcesId
LEFT JOIN BookCopy bc ON r.resourcesId = bc.resourcesId
LEFT JOIN Book b ON bc.ISBN = b.ISBN
LEFT JOIN BookCourseRecommendation bcr ON b.ISBN = bcr.ISBN
LEFT JOIN Course c ON bcr.CourseId = c.CourseId;
```

RESOURCESID	RESOURCESTYPE	RESOURCESSTATUS	FLOORNUMBER	SHELFNUMBER	LOANPERIOD	RESOURCESNAME	COURSENAME
6	Book	Available	3	10	21	Database System Concepts – 7th Edition	Database Systems
7	Book	Available	3	11	21	Database System Concepts – 7th Edition	Database Systems
6	Book	Available	3	10	21	Database System Concepts – 7th Edition	Computer Programming
7	Book	Available	3	11	21	Database System Concepts – 7th Edition	Computer Programming
4	Book	Available	2	13	14	Tender is the night	American Literature
5	Book	Available	2	14	14	Tender is the night	American Literature
1	Book	Available	2	10	7	The Great Gatsby	American Literature
2	Book	Unavailable	2	11	7	The Great Gatsby	American Literature
3	Book	Unavailable	2	12	7	The Great Gatsby	American Literature
1	Book	Available	2	10	7	The Great Gatsby	The Jazz Age in Film and Literature
2	Book	Unavailable	2	11	7	The Great Gatsby	The Jazz Age in Film and Literature
3	Book	Unavailable	2	12	7	The Great Gatsby	The Jazz Age in Film and Literature
11	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Database Systems
12	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Database Systems
13	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Database Systems
14	eBook	Unavailable	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Database Systems
11	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Applied Mathematics
12	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Applied Mathematics
13	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Applied Mathematics
14	eBook	Unavailable	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Applied Mathematics
11	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Big Data Analytics
12	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Big Data Analytics
13	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Big Data Analytics
14	eBook	Unavailable	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Big Data Analytics
11	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Introduction to AI/ML
12	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Introduction to AI/ML
13	eBook	Available	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Introduction to AI/ML
14	eBook	Unavailable	–	–	14	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Introduction to AI/ML
8	eBook	Available	–	–	7	Beginning Database Design: From Novice to Professional – 2nd Edition	Database Systems
9	eBook	Available	–	–	7	Beginning Database Design: From Novice to Professional – 2nd Edition	Database Systems
10	eBook	Unavailable	–	–	7	Beginning Database Design: From Novice to Professional – 2nd Edition	Database Systems

15	Device	Available	1	1	0	Apple iPad Pro	-
17	Device	Available	1	3	5	Dell XPS 13	-
18	Device	Available	1	4	5	Apple MacBook Air M1	-
20	Device	Available	1	6	3	Amazon Kindle Paperwhite	-
19	Device	Unavailable	1	5	5	HP Spectre x360	-
16	Device	Available	1	2	0	Samsung Galaxy Tab S8	-

IV. Simple queries (SELECT and WHERE)

1. **Query 1: List all books that are related to 'database'.**
   The query retrieves all books in the library system that contain 'database' in their title.

```
SELECT ISBN, BookTitle, Author, Publisher FROM Book WHERE LOWER(BookTitle) LIKE '%database%';
```

ISBN	BOOKTITLE	AUTHOR	PUBLISHER
978-0-07-802215-9	Database System Concepts - 7th Edition	Abraham Silberschatz	McGraw-Hill Education
978-1-43-024209-3	Beginning Database Design: From Novice to Professional - 2nd Edition	Clare Churcher	Apress

2. **Query 2: Find resources that can only be used within the library.**

```
SELECT ResourcesId, ResourcesType, ShelfNumber, FloorNumber
FROM Resources WHERE loanPeriod = 0;
-- assumption that resources that can only be used within library has 0 day loan period
```

RESOURCESID	RESOURCESTYPE	SHELFNUMBER	FLOORNUMBER
15	Device	1	1
16	Device	2	1

3. **Query 3: Find all members whose membership status is "Suspended"**
   This query helps to retrieve information on all suspended members

```
SELECT MemberId, MemberType, FirstName, LastName, OverdueFine, MembershipStatus
FROM Member WHERE MembershipStatus = 'Suspended';
```

MEMBERID	MEMBERTYPE	FIRSTNAME	LASTNAME	OVERDUEFINE	MEMBERSHIPSTATUS
5	Student	Diana	Johnson	11	Suspended
10	Student	Jane	Davis	14	Suspended
20	Staff	Rosabella	Cunningham	14	Suspended

4. **Query 4: Find all available resources of device type**

```
SELECT ResourcesId, ResourcesType, ResourcesStatus, LoanPeriod FROM Resources
WHERE ResourcesType = 'Device' AND ResourcesStatus = 'Available';
```

RESOURCESID	RESOURCESTYPE	RESOURCESSTATUS	LOANPERIOD
15	Device	Available	0
16	Device	Available	0
17	Device	Available	5
18	Device	Available	5
20	Device	Available	3

V. Intermediate queries (JOIN)

1. **Query 5: List all active overdue loans, along with a due date for return**
   This query is intended to query all loans whose due date has passed but not yet returned.

```
SELECT l.LoanId, l.MemberId, r.ResourcesType, l.LoanDate, l.ActualReturnDate,
l.LoanDate + r.LoanPeriod AS DueDate, l.LoanStatus
FROM Loan l
JOIN Resources r ON l.ResourcesId = r.ResourcesId WHERE l.LoanStatus = 'Overdue';
```

LOANID	MEMBERID	RESOURCESTYPE	LOANDATE	DUEDATE	LOANSTATUS
1	10	eBook	02-NOV-24	16-NOV-24	Overdue
2	20	Device	11-NOV-24	16-NOV-24	Overdue
3	5	Book	12-NOV-24	19-NOV-24	Overdue

2. **Query 6: Find all physical book copies of 'Literature' category**
   This query retrieves all the copies of books that fall under the 'Literature' category

```
SELECT r.ResourcesId, r.ResourcesType, b.BookTitle, b.BookCategory, r.ResourcesStatus, r.ShelfNumber, r.FloorNumber
FROM Resources r
JOIN BookCopy bc ON r.ResourcesId = bc.ResourcesId
JOIN Book b ON bc.ISBN = b.ISBN
WHERE r.ResourcesType = 'Book' AND b.BookCategory = 'Literature';
```

RESOURCESID	RESOURCESTYPE	BOOKTITLE	BOOKCATEGORY	RESOURCESSTATUS	SHELFNUMBER	FLOORNUMBER
1	Book	The Great Gatsby	Literature	Available	10	2
2	Book	The Great Gatsby	Literature	Unavailable	11	2
3	Book	The Great Gatsby	Literature	Unavailable	12	2
4	Book	Tender is the night	Literature	Available	13	2
5	Book	Tender is the night	Literature	Available	14	2

3. **Query 7: Show all active reservations made by a specific member**
   This query retrieves all active reservations and details of the resources that were reserved for memberId 2.

```
SELECT res.ReservationId, res.ReservationDate, res.MemberId,
m.FirstName || ' ' || m.LastName AS MemberName, r.ResourcesId, r.ResourcesType,
CASE WHEN r.ResourcesType IN ('Book', 'eBook') THEN b.BookTitle
WHEN r.ResourcesType = 'Device' THEN d.deviceCategory || ' ' || d.Model
END AS ResourcesName, res.ReservationStatus
FROM Reservation res
JOIN Resources r ON res.ResourcesId = r.ResourcesId
LEFT JOIN BookCopy bc ON r.ResourcesId = bc.ResourcesId
LEFT JOIN Book b ON bc.ISBN = b.ISBN
LEFT JOIN Device d ON r.ResourcesId = d.ResourcesId
JOIN Member m ON res.MemberId = m.MemberId
WHERE res.MemberId = 2 AND res.ReservationStatus = 'Reserved';
```

RESERVATIONID	RESERVATIONDATE	MEMBERID	MEMBERNAME	RESOURCESID	RESOURCESTYPE	RESOURCESNAME	RESERVATIONSTATUS
1	28-NOV-24	2	Bob Brown	14	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Reserved
3	28-NOV-24	2	Bob Brown	8	eBook	Beginning Database Design: From Novice to Professional – 2nd Edition	Reserved

4. **Query 8: Find available recommended book copies for a specific course**
   List all available book copies recommended for data courses with book title, author, its location and loan period

```
SELECT c.CourseName, r.ResourcesId, r.ResourcesType, b.BookTitle, b.Author, r.floorNumber, r.shelfNumber, r.loanPeriod
FROM BookCourseRecommendation bcr
JOIN Course c ON bcr.CourseId = c.CourseId
JOIN Book b ON bcr.ISBN = b.ISBN
JOIN BookCopy bc ON b.ISBN = bc.ISBN
JOIN Resources r ON bc.ResourcesId = r.ResourcesId
WHERE LOWER(c.CourseName) LIKE '%data%' AND r.ResourcesStatus = 'Available';
```

COURSENAME	RESOURCESID	RESOURCESTYPE	BOOKTITLE	AUTHOR	FLOORNUMBER	SHELFNUMBER	LOANPERIOD
Database Systems	6	Book	Database System Concepts – 7th Edition	Abraham Silberschatz	3	10	21
Database Systems	7	Book	Database System Concepts – 7th Edition	Abraham Silberschatz	3	11	21
Database Systems	8	eBook	Beginning Database Design: From Novice to Professional – 2nd Edition	Clare Churcher	–	–	7
Database Systems	9	eBook	Beginning Database Design: From Novice to Professional – 2nd Edition	Clare Churcher	–	–	7
Database Systems	11	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14
Big Data Analytics	11	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14
Database Systems	12	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14
Big Data Analytics	12	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14
Database Systems	13	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14
Big Data Analytics	13	eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	Thomas Nield	–	–	14

VI. Advanced Queries (JOIN & GROUP BY)

1. **Query 9: Count the number of resources currently borrowed by each member**
   Lists the number of resources currently loaned by each member, including their membership type

```
SELECT m.MemberType, m.MemberId, m.FirstName || ' ' || m.LastName AS FullName,
COUNT(l.LoanId) AS ActiveLoanCount
FROM Loan l
JOIN Member m ON l.MemberId = m.MemberId
WHERE l.LoanStatus in ('Loaned', 'Overdue')
GROUP BY m.MemberType, m.MemberId, m.FirstName || ' ' || m.LastName
ORDER BY m.MemberType, ActiveLoanCount DESC;
```

MEMBERTYPE	MEMBERID	FULLNAME	ACTIVELOANCOUNT
Staff	20	Rosabella Cunningham	1
Student	5	Diana Johnson	5
Student	2	Bob Brown	1
Student	10	Jane Davis	1

2. **Query 10: Find the popular loaned books for the current month**
   List details of the top 5 books that was loaned the most for the current month

```
SELECT r.ResourcesType, b.BookTitle, TO_CHAR(l.LoanDate, 'YYYY-MM') AS LoanMonth, COUNT(l.LoanId) AS LoanCount
FROM Resources r JOIN Loan l ON r.ResourcesId = l.ResourcesId
JOIN BookCopy bc ON r.ResourcesId = bc.ResourcesId
JOIN Book b ON bc.ISBN = b.ISBN
WHERE TO_CHAR(l.LoanDate, 'YYYY-MM') = TO_CHAR(SYSDATE, 'YYYY-MM')
GROUP BY r.ResourcesType, b.BookTitle, TO_CHAR(l.LoanDate, 'YYYY-MM')
ORDER BY LoanCount DESC FETCH FIRST 5 ROWS ONLY;
```

RESOURCESTYPE	BOOKTITLE	LOANMONTH	LOANCOUNT
eBook	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	2024-11	2
eBook	Beginning Database Design: From Novice to Professional – 2nd Edition	2024-11	1
Book	Database System Concepts – 7th Edition	2024-11	1
Book	The Great Gatsby	2024-11	1
Book	Tender is the night	2024-11	1

3. **Query 11: Find the top 3 members who have borrowed the most resources**
   Identify the top 3 members who have borrowed the highest number of resources

```
SELECT m.MemberId, m.FirstName || ' ' || m.LastName AS FullName, m.MemberType,
COUNT(l.LoanId) AS TotalLoans
FROM Loan l
JOIN Member m ON l.MemberId = m.MemberId
GROUP BY m.MemberId, m.FirstName, m.LastName, m.MemberType
ORDER BY TotalLoans DESC
FETCH FIRST 3 ROWS ONLY;
```

MEMBERID	FULLNAME	MEMBERTYPE	TOTALLOANS
5	Diana Johnson	Student	5
2	Bob Brown	Student	4
4	Aisha Gupta	Student	3

4. **Query 12: Find the number of book copies of the recommended books for each course**
   Identify which book is recommended for which course and how many copies are available

```
SELECT c.CourseName, b.BookTitle, COUNT(bc.ResourcesId) AS CopiesCount
FROM BookCourseRecommendation bcr
JOIN Course c ON bcr.CourseId = c.CourseId
JOIN Book b ON bcr.ISBN = b.ISBN
JOIN BookCopy bc ON b.ISBN = bc.ISBN
GROUP BY c.CourseName, b.BookTitle
ORDER BY c.CourseName;
```

COURSENAME	BOOKTITLE	COPIESCOUNT
American Literature	Tender is the night	2
American Literature	The Great Gatsby	3
Applied Mathematics	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	4
Big Data Analytics	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	4
Computer Programming	Database System Concepts – 7th Edition	2
Database Systems	Beginning Database Design: From Novice to Professional – 2nd Edition	3
Database Systems	Database System Concepts – 7th Edition	2
Database Systems	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	4
Introduction to AI/ML	Essential Math for Data Science: Take Control of Your Data with Fundamental Linear Algebra, Probability, and Statistics	4
The Jazz Age in Film and Literature	The Great Gatsby	3

Bibliography

Silberschatz, A., Korth, H.F. & Sudarshan, S. (2019) *Database System Concepts*. 7th ed. New York: McGraw-Hill Education.

Watt, A. & Eng, N. (2014) *Database Design*. 2nd ed. Victoria, B.C.: BCcampus.