

## ***P4 ~ Database Schema Implementation***

### ***DDL Final Query:***

#### ***Description ~***

This SQL code is designed to manage a comprehensive library database named `DMDD_Group10`. It starts by ensuring the database exists and dropping any existing tables to handle foreign key constraints. Then, it creates a series of tables to store data about libraries, members, authors, publishers, genres, books, staff, copies, transactions, reservations, feedback, fines, and memberships. Each table includes relevant fields and constraints to maintain data integrity, such as primary keys, foreign keys, and check constraints. This setup supports a robust library management system capable of tracking books, member interactions, staff roles, and financial transactions.

#### ***SQL Query ~***

```
USE master;
```

```
GO
```

```
IF EXISTS (SELECT name FROM sys.databases WHERE name = 'DMDD_Group10')
```

```
BEGIN
```

```
    ALTER DATABASE DMDD_Group10 SET SINGLE_USER WITH ROLLBACK  
    IMMEDIATE;
```

```
    DROP DATABASE DMDD_Group10;
```

```
END
```

```
GO
```

```
-- Create database if it doesn't exist
```

```
DROP DATABASE IF EXISTS DMDD_Group10;
```

```
CREATE DATABASE DMDD_Group10;
```

```
GO
```

```
USE DMDD_Group10;
```

```
GO
```

```
-- Drop all tables if they exist (in reverse order of creation to handle foreign key constraints)
```

```
DROP TABLE IF EXISTS FINE;
```

```
DROP TABLE IF EXISTS FEEDBACK;
```

```
DROP TABLE IF EXISTS RESERVATION;
```

```
DROP TABLE IF EXISTS COPY;
```

```
DROP TABLE IF EXISTS [TRANSACTION];
```

```
DROP TABLE IF EXISTS BOOK_AUTHOR;
```

```
DROP TABLE IF EXISTS BOOK;
```

```
DROP TABLE IF EXISTS GENRE;
```

```
DROP TABLE IF EXISTS STAFF;
```

```
DROP TABLE IF EXISTS MEMBER;
```

```
DROP TABLE IF EXISTS MEMBERSHIP;
```

```
DROP TABLE IF EXISTS LIBRARY;
```

```
DROP TABLE IF EXISTS AUTHOR;
```

```
DROP TABLE IF EXISTS PUBLISHER;
```

-- Create LIBRARY table

```
CREATE TABLE LIBRARY (  
    library_id INT PRIMARY KEY IDENTITY(1,1),  
    name VARCHAR(100) NOT NULL,  
    location VARCHAR(255) NOT NULL,  
    phone VARCHAR(20) NOT NULL,  
    email VARCHAR(100) NOT NULL,  
    working_hours VARCHAR(100) NOT NULL  
);
```

-- Create MEMBER table

```
CREATE TABLE MEMBER (  
    member_id INT PRIMARY KEY IDENTITY(1,1),  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    email VARCHAR(100) NOT NULL,  
    phone VARCHAR(20) NOT NULL,  
    join_date DATE NOT NULL  
);
```

-- Create AUTHOR table

```
CREATE TABLE AUTHOR (  
    author_id INT PRIMARY KEY IDENTITY(1,1),  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    date_of_birth DATE,  
    nationality VARCHAR(50) NOT NULL  
);
```

-- Create PUBLISHER table

```
CREATE TABLE PUBLISHER (  
    publisher_id INT PRIMARY KEY IDENTITY(1,1),  
    name VARCHAR(100) NOT NULL,  
    address VARCHAR(255),  
    phone_number VARCHAR(20),  
    email VARCHAR(100) NOT NULL  
);
```

-- Create GENRE table

```
CREATE TABLE GENRE (  
    genre_id INT PRIMARY KEY IDENTITY(1,1),  
    genre_type VARCHAR(50) NOT NULL,  
    description TEXT,
```

```
    popularity_score DECIMAL(3,1),  
  
    created_date DATE NOT NULL,  
  
    CONSTRAINT chk_popularity_score CHECK (popularity_score >= 0 AND  
popularity_score <= 10)  
);
```

-- Create BOOK table

```
CREATE TABLE BOOK (  
  
    book_id INT PRIMARY KEY IDENTITY(1,1),  
  
    library_id INT NOT NULL,  
  
    genre_id INT NOT NULL,  
  
    publisher_id INT NOT NULL,  
  
    title VARCHAR(255) NOT NULL,  
  
    isbn VARCHAR(20) NOT NULL,  
  
    publication_date DATE NOT NULL,  
  
    CONSTRAINT FK_Book_Library FOREIGN KEY (library_id) REFERENCES  
LIBRARY(library_id),  
  
    CONSTRAINT FK_Book_Genre FOREIGN KEY (genre_id) REFERENCES  
GENRE(genre_id),  
  
    CONSTRAINT FK_Book_Publisher FOREIGN KEY (publisher_id) REFERENCES  
PUBLISHER(publisher_id)  
);
```

-- Create BOOK\_AUTHOR junction table

```
CREATE TABLE BOOK_AUTHOR (  

```

```
book_id INT NOT NULL,  
  
author_id INT NOT NULL,  
  
author_role VARCHAR(50) NOT NULL,  
  
contribution_percentage DECIMAL(5,2),  
  
special_acknowledgement TEXT,  
  
PRIMARY KEY (book_id, author_id),  
  
CONSTRAINT FK_BookAuthor_Book FOREIGN KEY (book_id) REFERENCES  
BOOK(book_id),  
  
CONSTRAINT FK_BookAuthor_Author FOREIGN KEY (author_id) REFERENCES  
AUTHOR(author_id),  
  
CONSTRAINT chk_contribution_percentage CHECK (contribution_percentage > 0  
AND contribution_percentage <= 100)  
);
```

-- Create STAFF table

```
CREATE TABLE STAFF (  
  
    staff_id INT PRIMARY KEY IDENTITY(1,1),  
  
    library_id INT NOT NULL,  
  
    first_name VARCHAR(50) NOT NULL,  
  
    last_name VARCHAR(50) NOT NULL,  
  
    role VARCHAR(50) NOT NULL,  
  
    email VARCHAR(100) NOT NULL,  
  
    phone VARCHAR(20) NOT NULL,  
  
    CONSTRAINT FK_Staff_Library FOREIGN KEY (library_id) REFERENCES  
LIBRARY(library_id)
```

);

-- Create COPY table

```
CREATE TABLE COPY (  
    copy_id INT PRIMARY KEY IDENTITY(1,1),  
    book_id INT NOT NULL,  
    condition VARCHAR(20) NOT NULL,  
    purchase_date DATE NOT NULL,  
    availability_status VARCHAR(20) NOT NULL,  
    CONSTRAINT FK_Copy_Book FOREIGN KEY (book_id) REFERENCES  
    BOOK(book_id),  
    CONSTRAINT chk_condition CHECK (condition IN ('New', 'Good', 'Fair', 'Poor',  
    'Damaged')),  
    CONSTRAINT chk_availability_status CHECK (availability_status IN ('Available',  
    'Unavailable', 'Reserved'))  
);
```

-- Create TRANSACTION table

```
CREATE TABLE [TRANSACTION] (  
    transaction_id INT PRIMARY KEY IDENTITY(1,1),  
    member_id INT NOT NULL,  
    copy_id INT NOT NULL,  
    staff_id INT NOT NULL,  
    borrow_date DATE NOT NULL,  
    due_date DATE NOT NULL,
```

```
    return_date DATE,

    transaction_status VARCHAR(20) NOT NULL,

    CONSTRAINT FK_Transaction_Member FOREIGN KEY (member_id) REFERENCES
    MEMBER(member_id),

    CONSTRAINT FK_Transaction_Copy FOREIGN KEY (copy_id) REFERENCES
    COPY(copy_id),

    CONSTRAINT FK_Transaction_Staff FOREIGN KEY (staff_id) REFERENCES
    STAFF(staff_id),

    CONSTRAINT chk_transaction_status CHECK (transaction_status IN ('Active',
    'Completed', 'Overdue', 'Lost'))

);
```

-- Create RESERVATION table

```
CREATE TABLE RESERVATION (

    reservation_id INT PRIMARY KEY IDENTITY(1,1),

    member_id INT NOT NULL,

    book_id INT NOT NULL,

    staff_id INT NOT NULL,

    reservation_date DATE NOT NULL,

    reservation_status VARCHAR(20) NOT NULL,

    CONSTRAINT FK_Reservation_Member FOREIGN KEY (member_id) REFERENCES
    MEMBER(member_id),

    CONSTRAINT FK_Reservation_Book FOREIGN KEY (book_id) REFERENCES
    BOOK(book_id),

    CONSTRAINT FK_Reservation_Staff FOREIGN KEY (staff_id) REFERENCES
    STAFF(staff_id),
```



```
        CONSTRAINT chk_reservation_status CHECK (reservation_status IN ('Pending',  
'Fulfilled', 'Cancelled', 'Expired'))  
);
```

-- Create FEEDBACK table

```
CREATE TABLE FEEDBACK (  
    feedback_id INT PRIMARY KEY IDENTITY(1,1),  
    book_id INT NOT NULL,  
    member_id INT NOT NULL,  
    feedback_date DATE NOT NULL,  
    rating INT NOT NULL,  
    comment TEXT,  
    CONSTRAINT FK_Feedback_Book FOREIGN KEY (book_id) REFERENCES  
BOOK(book_id),  
    CONSTRAINT FK_Feedback_Member FOREIGN KEY (member_id) REFERENCES  
MEMBER(member_id),  
    CONSTRAINT chk_rating CHECK (rating >= 1 AND rating <= 5)  
);
```

-- Create FINE table

```
CREATE TABLE FINE (  
    fine_id INT PRIMARY KEY IDENTITY(1,1),  
    transaction_id INT NOT NULL,  
    amount DECIMAL(10,2) NOT NULL,  
    issue_date DATE NOT NULL,
```

```

    payment_status VARCHAR(20) NOT NULL,

    CONSTRAINT FK_Fine_Transaction FOREIGN KEY (transaction_id) REFERENCES
[TRANSACTION](transaction_id),

    CONSTRAINT chk_amount CHECK (amount > 0),

    CONSTRAINT chk_payment_status CHECK (payment_status IN ('Paid', 'Unpaid',
'Waived'))

);

```

```

CREATE TABLE MEMBERSHIP (

    member_library_id INT PRIMARY KEY IDENTITY(1,1), -- Use IDENTITY for
auto-increment in SQL Server

    library_id INT NOT NULL,

    member_id INT NOT NULL,

    registration_date DATE NOT NULL,

    membership_status VARCHAR(20) NOT NULL,

    CONSTRAINT chk_membership_status CHECK (membership_status IN ('Active',
'Expired', 'Suspended', 'Pending')),

    CONSTRAINT FK_Membership_Member FOREIGN KEY (member_id)
REFERENCES MEMBER(member_id),

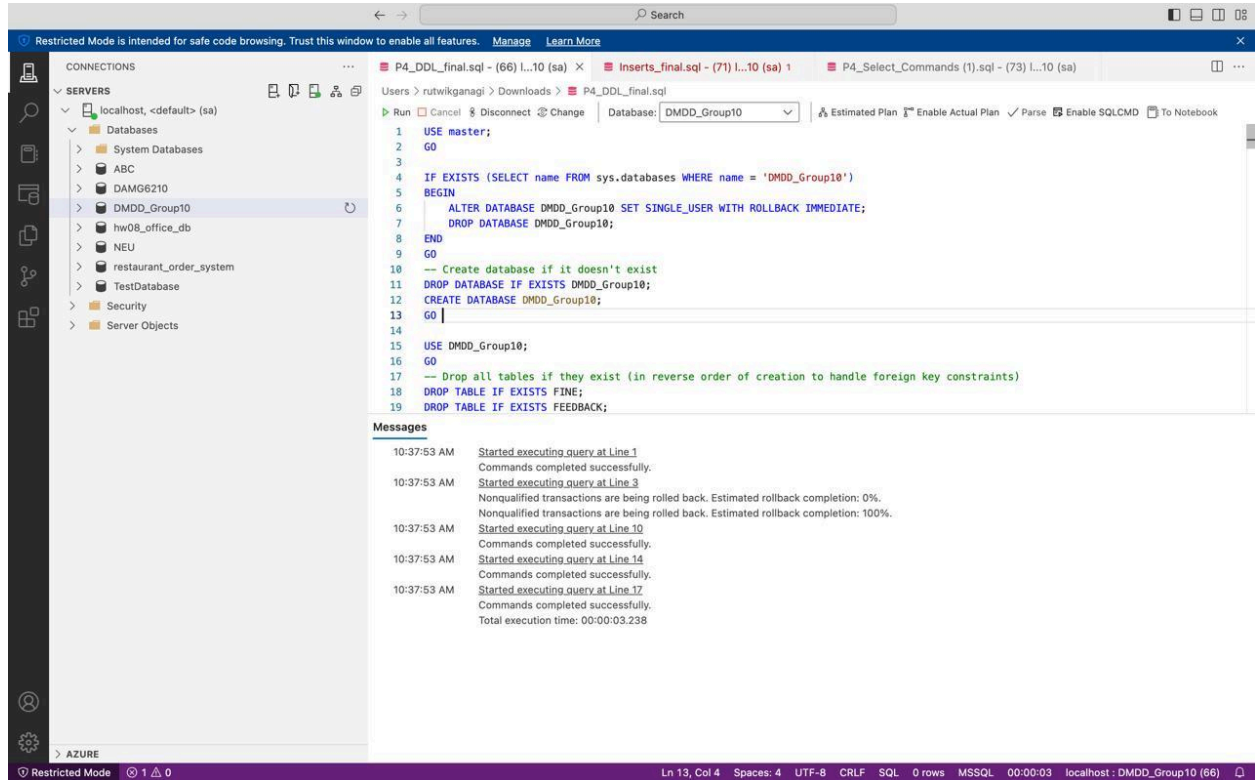
    CONSTRAINT FK_Membership_Library FOREIGN KEY (library_id) REFERENCES
LIBRARY(library_id)

);

```

\*\*\*\*\*

## Output (Screenshot) ~



\*\*\*\*\*

## ***Insertion Commands:***

### ***Description ~***

The provided SQL code is designed to populate various tables in a database named `DMDD_Group10`. It includes a series of `INSERT INTO` statements that add data to tables such as `LIBRARY`, `MEMBER`, `AUTHOR`, `PUBLISHER`, `GENRE`, `BOOK`, `MEMBERSHIP`, `BOOK_AUTHOR`, `STAFF`, `COPY`, `TRANSACTION`, `RESERVATION`, `FEEDBACK`, and `FINE`. Each statement inserts specific data into the respective tables, setting up a comprehensive dataset for a library management system. This data includes library locations, member details, book information, staff roles, and transaction records, among others.

### ***SQL Query ~***

```
USE DMDD_Group10;
```

```
-- Insert data into LIBRARY table
```

```
INSERT INTO LIBRARY (name, location, phone, email, working_hours)
```

```
VALUES
```

```
('Snell Library', '360 Huntington Ave, Boston, MA 02115', '617-373-8778',  
'ill@northeastern.edu', 'Mon-Thu 8am-10pm, Fri 8am-9pm, Sat 11am-9pm, Sun 11am-10pm'),
```

```
('F. W. Olin Library', '5000 MacArthur Blvd, Oakland, CA 94613', '510-430-2196',  
'fwolin@northeastern.edu', 'Mon-Thu 8am-10pm, Fri 8am-9pm, Sat 11am-9pm, Sun  
11am-10pm'),
```

```
('Northeastern University Health Sciences Library', '101 Medical Plaza, Boston, MA 02115',  
'617-555-1010', 'healthlibrary@northeastern.edu', 'Mon-Fri: 8am-8pm; Sat: 10am-4pm; Sun:  
Closed'),
```

('Northeastern University Silicon Valley Library', '123 Tech Way, San Jose, CA 95112', '408-555-9876', 'siliconvalleylibrary@northeastern.edu', 'Mon-Fri: 9am-6pm; Sat: 10am-4pm; Sun: Closed'),

('Northeastern University London Library', '50 City Rd, London, EC1Y 1AB, UK', '020-7946-1234', 'londonlibrary@northeastern.ac.uk', 'Mon-Fri: 9am-5pm; Sat-Sun: Closed'),

('Northeastern University Charlotte Library', '400 Trade St, Charlotte, NC 28202', '704-555-1234', 'charlottelibrary@northeastern.edu', 'Mon-Fri: 8am-6pm; Sat-Sun: Closed'),

('Northeastern University Arlington Library', '1501 N Royal St, Arlington, VA 22209', '703-555-5678', 'arlingtonlibrary@northeastern.edu', 'Mon-Fri: 8am-8pm; Sat: 10am-4pm; Sun: Closed'),

('Northeastern University Burlington Library', '1200 Maple St, Burlington, MA 01803', '781-555-7890', 'burlingtonlibrary@northeastern.edu', 'Mon-Fri: 9am-5pm; Sat-Sun: Closed'),

('Northeastern University Seattle Library', '500 Pine St, Seattle, WA 98101', '206-555-2468', 'seattlelibrary@northeastern.edu', 'Mon-Fri: 9am-6pm; Sat: 10am-4pm; Sun: Closed'),

('Northeastern University Toronto Library', '100 King St W, Toronto, ON M5X 1A9, Canada', '416-555-1357', 'torontolibrary@northeastern.edu', 'Mon-Fri: 9am-5pm; Sat-Sun: Closed');

GO

-- Insert data into MEMBER table

INSERT INTO MEMBER (first\_name, last\_name, email, phone, join\_date)

VALUES

('John', 'Smith', 'john.smith@email.com', '555-111-2222', '2020-03-15'),

('Sarah', 'Johnson', 'sarah.j@email.com', '555-222-3333', '2019-07-22'),

('Michael', 'Williams', 'mwilliams@email.com', '555-333-4444', '2021-01-10'),

('Emily', 'Brown', 'emily.brown@email.com', '555-444-5555', '2018-11-05'),

('David', 'Jones', 'david.jones@email.com', '555-555-6666', '2022-02-28'),

('Jessica', 'Garcia', 'jgarcia@email.com', '555-666-7777', '2020-09-17'),

```
('Robert', 'Miller', 'rmiller@email.com', '555-777-8888', '2021-05-03'),  
( 'Jennifer', 'Davis', 'jdavis@email.com', '555-888-9999', '2019-12-20'),  
( 'William', 'Rodriguez', 'wrodriguez@email.com', '555-999-0000', '2022-04-11'),  
( 'Amanda', 'Martinez', 'amartinez@email.com', '555-000-1111', '2020-08-07'),  
( 'Thomas', 'Wilson', 'twilson@email.com', '555-112-2233', '2021-11-15'),  
( 'Elizabeth', 'Anderson', 'eanderson@email.com', '555-223-3344', '2019-03-29');  
GO
```

```
-- Insert data into AUTHOR table
```

```
INSERT INTO AUTHOR (first_name, last_name, date_of_birth, nationality)
```

```
VALUES
```

```
('Jane', 'Austen', '1775-12-16', 'British'),  
( 'Ernest', 'Hemingway', '1899-07-21', 'American'),  
( 'Gabriel', 'Garc a M rquez', '1927-03-06', 'Colombian'),  
( 'Haruki', 'Murakami', '1949-01-12', 'Japanese'),  
( 'Chimamanda', 'Adichie', '1977-09-15', 'Nigerian'),  
( 'Stephen', 'King', '1947-09-21', 'American'),  
( 'J.K.', 'Rowling', '1965-07-31', 'British'),  
( 'Toni', 'Morrison', '1931-02-18', 'American'),  
( 'Margaret', 'Atwood', '1939-11-18', 'Canadian'),  
( 'Paulo', 'Coelho', '1947-08-24', 'Brazilian');  
GO
```

-- Insert data into PUBLISHER table

INSERT INTO PUBLISHER (name, address, phone\_number, email)

VALUES

('Penguin Random House', '1745 Broadway, New York, NY 10019', '212-782-9000',  
'info@penguinrandomhouse.com'),

('HarperCollins', '195 Broadway, New York, NY 10007', '212-207-7000',  
'contact@harpercollins.com'),

('Simon & Schuster', '1230 Avenue of the Americas, New York, NY 10020', '212-698-7000',  
'info@simonandschuster.com'),

('Macmillan Publishers', '120 Broadway, New York, NY 10271', '646-307-5151',  
'info@macmillan.com'),

('Hachette Book Group', '1290 Avenue of the Americas, New York, NY 10104', '212-364-1100',  
'contact@hachettebookgroup.com'),

('Oxford University Press', 'Great Clarendon Street, Oxford OX2 6DP, UK', '+44-1865-353535',  
'info@oup.com'),

('Scholastic', '557 Broadway, New York, NY 10012', '212-343-6100',  
'scholastic@scholastic.com'),

('Wiley', '111 River Street, Hoboken, NJ 07030', '201-748-6000', 'info@wiley.com'),

('Pearson Education', '221 River Street, Hoboken, NJ 07030', '201-236-7000',  
'info@pearson.com'),

('McGraw-Hill Education', '1325 Avenue of the Americas, New York, NY 10019',  
'646-766-2000', 'customer.service@mheducation.com');

GO

-- Insert data into GENRE table

INSERT INTO GENRE (genre\_type, description, popularity\_score, created\_date)

VALUES

('Fiction', 'Narrative works created from imagination', 8.5, '2018-01-15'),  
('Non-Fiction', 'Works based on facts and real events', 7.2, '2018-01-15'),  
('Mystery', 'Stories focused on solving a crime or puzzle', 8.0, '2018-02-20'),  
('Science Fiction', 'Speculative fiction dealing with advanced science and technology', 7.8, '2018-02-21'),  
('Fantasy', 'Works involving magical or supernatural elements', 8.3, '2018-03-05'),  
('Biography', 'Account of a person's life written by someone else', 6.5, '2018-03-10'),  
('History', 'Works focused on past events', 6.8, '2018-04-12'),  
('Romance', 'Stories centered on romantic relationships', 7.5, '2018-05-15'),  
('Self-Help', 'Books offering advice for personal improvement', 6.2, '2018-06-20'),  
('Poetry', 'Literary works with rhythmic qualities of language', 5.5, '2018-07-25');  
GO

-- Insert data into BOOK table

```
INSERT INTO BOOK (library_id, genre_id, publisher_id, title, isbn, publication_date)
VALUES
```

```
(1, 1, 1, 'Pride and Prejudice', '9780141439518', '1813-01-28'),
(2, 3, 6, 'The Old Man and the Sea', '9780684801223', '1952-09-01'),
(1, 1, 3, 'One Hundred Years of Solitude', '9780060883287', '1967-05-30'),
(3, 4, 2, 'Norwegian Wood', '9780375704024', '1987-09-04'),
(4, 1, 5, 'Americanah', '9780307455925', '2013-05-14'),
(5, 5, 7, 'The Shining', '9780307743657', '1977-01-28'),
(6, 5, 7, 'Harry Potter and the Philosopher's Stone', '9780747532743', '1997-06-26'),
(7, 1, 3, 'Beloved', '9781400033416', '1987-09-02'),
```



```
(8, 4, 1, 'The Handmaid"s Tale', '9780385490818', '1985-06-01'),  
(9, 1, 2, 'The Alchemist', '9780062315007', '1988-04-25'),  
(10, 2, 9, 'Sapiens: A Brief History of Humankind', '9780062316097', '2011-02-10'),  
(1, 8, 5, 'The Notebook', '9780553816716', '1996-10-01');
```

GO

-- Insert data into MEMBERSHIP table

```
INSERT INTO MEMBERSHIP (library_id, member_id, registration_date, membership_status)
```

VALUES

```
(1, 1, '2020-03-15', 'Active'),  
(1, 2, '2019-07-22', 'Active'),  
(2, 3, '2021-01-10', 'Active'),  
(3, 4, '2018-11-05', 'Expired'),  
(2, 5, '2022-02-28', 'Active'),  
(4, 6, '2020-09-17', 'Suspended'),  
(5, 7, '2021-05-03', 'Active'),  
(6, 8, '2019-12-20', 'Active'),  
(7, 9, '2022-04-11', 'Pending'),  
(8, 10, '2020-08-07', 'Active'),  
(9, 11, '2021-11-15', 'Active'),  
(10, 12, '2019-03-29', 'Expired');
```

GO

-- Insert data into BOOK\_AUTHOR table

```
INSERT INTO BOOK_AUTHOR (book_id, author_id, author_role, contribution_percentage,  
special_acknowledgement)
```

VALUES

(1, 1, 'Primary Author', 100.00, NULL),

(2, 2, 'Primary Author', 100.00, NULL),

(3, 3, 'Primary Author', 100.00, 'Winner of Nobel Prize in Literature'),

(4, 4, 'Primary Author', 100.00, NULL),

(5, 5, 'Primary Author', 100.00, NULL),

(6, 6, 'Primary Author', 100.00, NULL),

(7, 7, 'Primary Author', 100.00, NULL),

(8, 8, 'Primary Author', 100.00, 'Winner of Pulitzer Prize'),

(9, 9, 'Primary Author', 100.00, NULL),

(10, 10, 'Primary Author', 100.00, NULL);

GO

-- Insert data into STAFF table

```
INSERT INTO STAFF (library_id, first_name, last_name, role, email, phone)
```

VALUES

(1, 'Patricia', 'Wilson', 'Head Librarian', 'pwilson@library.org', '555-121-2323'),

(1, 'Richard', 'Moore', 'Reference Librarian', 'rmoore@library.org', '555-131-2424'),

(2, 'Susan', 'Taylor', 'Library Assistant', 'staylor@library.org', '555-141-2525'),

(3, 'James', 'Anderson', 'Library Technician', 'janderson@library.org', '555-151-2626'),

(4, 'Lisa', 'Thomas', 'Head Librarian', 'lthomas@library.org', '555-161-2727'),

(5, 'Mark', 'Jackson', 'Reference Librarian', 'mjackson@library.org', '555-171-2828'),  
(6, 'Karen', 'White', 'Library Assistant', 'kwhite@library.org', '555-181-2929'),  
(7, 'Daniel', 'Harris', 'Library Technician', 'dharris@library.org', '555-191-3030'),  
(8, 'Nancy', 'Martin', 'Head Librarian', 'nmartin@library.org', '555-212-3131'),  
(9, 'Paul', 'Thompson', 'Reference Librarian', 'pthompson@library.org', '555-232-3232'),  
(10, 'Laura', 'Garcia', 'Library Assistant', 'lgarcia@library.org', '555-242-3333');

GO

-- Insert data into COPY table

INSERT INTO COPY (book\_id, [condition], purchase\_date, availability\_status)

VALUES

(1, 'Good', '2019-06-15', 'Available'),  
(1, 'Fair', '2017-03-22', 'Available'),  
(2, 'Good', '2020-01-10', 'Available'),  
(3, 'New', '2022-02-05', 'Reserved'),  
(4, 'Good', '2021-07-18', 'Available'),  
(5, 'Poor', '2018-09-12', 'Available'),  
(6, 'Good', '2019-11-30', 'Unavailable'),  
(7, 'New', '2022-03-15', 'Available'),  
(8, 'Fair', '2020-05-22', 'Available'),  
(9, 'Good', '2021-04-14', 'Reserved'),  
(10, 'New', '2022-01-08', 'Available'),  
(11, 'Good', '2021-08-19', 'Available'),

```
(12, 'Fair', '2020-10-25', 'Available');
```

```
GO
```

```
-- Insert data into TRANSACTION table
```

```
INSERT INTO [TRANSACTION] (member_id, copy_id, staff_id, borrow_date, due_date,  
return_date, transaction_status)
```

```
VALUES
```

```
(1, 1, 1, '2023-01-15', '2023-02-15', '2023-02-10', 'Completed'),
```

```
(2, 3, 2, '2023-02-05', '2023-03-05', '2023-03-02', 'Completed'),
```

```
(3, 5, 3, '2023-03-10', '2023-04-10', NULL, 'Active'),
```

```
(4, 7, 4, '2023-01-20', '2023-02-20', '2023-03-01', 'Overdue'),
```

```
(5, 9, 5, '2023-04-05', '2023-05-05', NULL, 'Active'),
```

```
(6, 11, 6, '2023-02-25', '2023-03-25', '2023-03-20', 'Completed'),
```

```
(7, 2, 7, '2023-03-15', '2023-04-15', NULL, 'Active'),
```

```
(8, 4, 8, '2023-02-10', '2023-03-10', '2023-04-05', 'Overdue'),
```

```
(9, 6, 9, '2023-04-01', '2023-05-01', NULL, 'Active'),
```

```
(10, 8, 10, '2023-01-05', '2023-02-05', NULL, 'Lost');
```

```
GO
```

```
-- Insert data into RESERVATION table
```

```
INSERT INTO RESERVATION (member_id, book_id, staff_id, reservation_date,  
reservation_status)
```

```
VALUES
```

```
(1, 3, 1, '2023-04-01', 'Pending'),
```

(2, 9, 2, '2023-03-25', 'Pending'),  
(3, 6, 3, '2023-03-15', 'Cancelled'),  
(4, 1, 4, '2023-02-10', 'Fulfilled'),  
(5, 7, 5, '2023-04-05', 'Pending'),  
(6, 2, 6, '2023-03-01', 'Expired'),  
(7, 10, 7, '2023-04-10', 'Pending'),  
(8, 5, 8, '2023-03-20', 'Fulfilled'),  
(9, 8, 9, '2023-02-15', 'Cancelled'),  
(10, 4, 10, '2023-04-02', 'Pending');  
GO

-- Insert data into FEEDBACK table

INSERT INTO FEEDBACK (book\_id, member\_id, feedback\_date, rating, comment)

VALUES

(1, 1, '2023-02-12', 5, 'A timeless classic that never gets old!'),  
(2, 2, '2023-03-04', 4, 'A beautiful story about perseverance.'),  
(3, 4, '2023-01-10', 5, 'One of the greatest novels ever written.'),  
(4, 6, '2023-03-22', 3, 'I found it a bit slow, but the writing is beautiful.'),  
(5, 8, '2023-04-06', 5, 'A powerful exploration of identity and belonging.'),  
(6, 3, '2023-02-20', 4, 'Genuinely scary and psychologically complex.'),  
(7, 5, '2023-03-18', 5, 'Magical and captivating from start to finish.'),  
(8, 7, '2023-01-25', 5, 'A profound and moving masterpiece.'),  
(9, 9, '2023-02-28', 4, 'Disturbing but thought-provoking.'),

```
(10, 10, '2023-04-03', 5, 'Life-changing wisdom in a simple story.');
```

```
GO
```

```
-- Insert data into FINE table
```

```
INSERT INTO FINE (transaction_id, amount, issue_date, payment_status)
```

```
VALUES
```

```
(4, 5.00, '2023-03-02', 'Paid'),
```

```
(8, 10.00, '2023-04-06', 'Unpaid'),
```

```
(10, 25.00, '2023-02-06', 'Unpaid'),
```

```
(1, 2.50, '2023-02-16', 'Waived'),
```

```
(2, 3.00, '2023-03-06', 'Paid'),
```

```
(3, 7.50, '2023-04-11', 'Unpaid'),
```

```
(5, 4.00, '2023-05-06', 'Unpaid'),
```

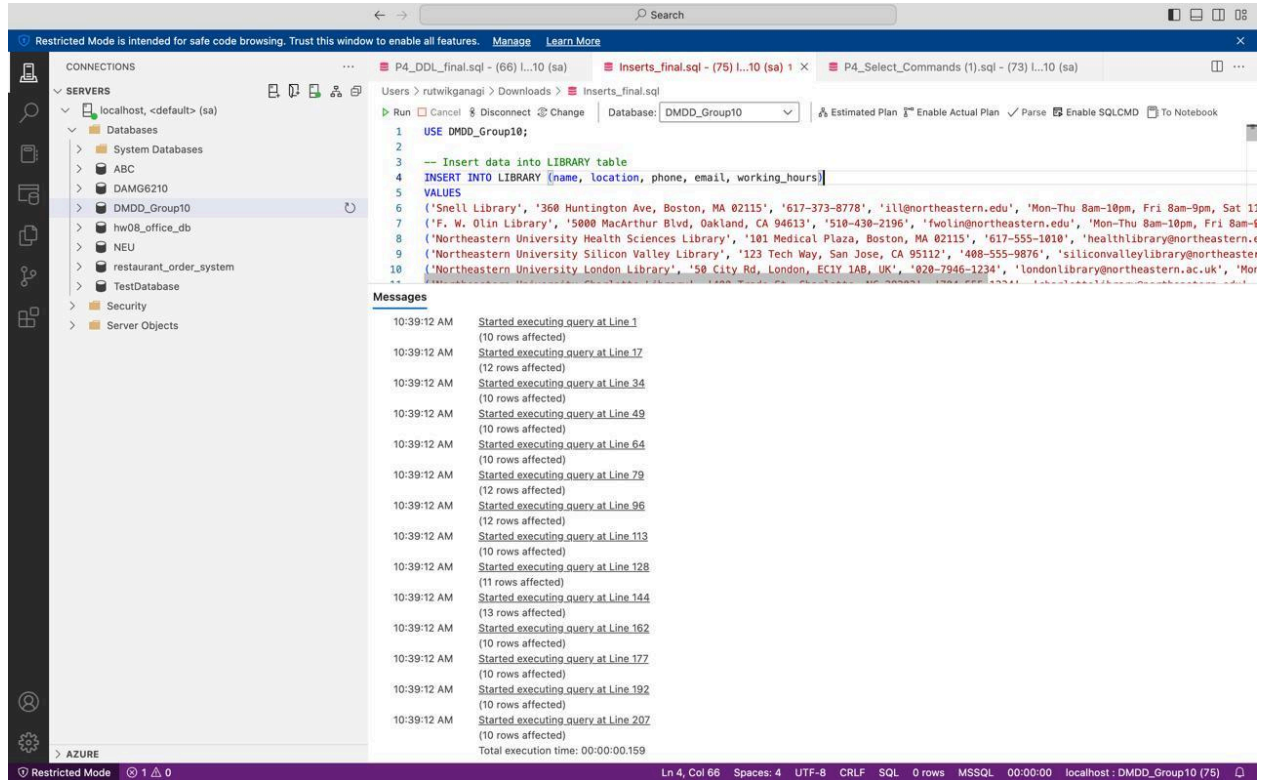
```
(6, 2.00, '2023-03-26', 'Paid'),
```

```
(7, 6.00, '2023-04-16', 'Unpaid'),
```

```
(9, 8.00, '2023-05-02', 'Unpaid');
```

```
*****
```

## Output Screenshot ~



\*\*\*\*\*

## ***Selection Command:***

### ***Description ~***

The provided SQL code is designed to retrieve all records from various tables within a database named `DMDD_Group10`. It includes queries for tables such as `LIBRARY`, `MEMBER`, `AUTHOR`, `PUBLISHER`, `GENRE`, `BOOK`, `MEMBERSHIP`, `BOOK_AUTHOR`, `STAFF`, `COPY`, `TRANSACTION`, `RESERVATION`, `FEEDBACK`, and `FINE`. Each query uses the `SELECT *` statement to fetch all columns and rows from each respective table, allowing for a comprehensive overview of the data stored in the database. This can be useful for data exploration, debugging, or initial setup of a database management system.

### ***SQL Query ~***

```
USE DMDD_Group10;
```

```
-- Select all records from LIBRARY table
```

```
SELECT * FROM LIBRARY;
```

```
-- Select all records from MEMBER table
```

```
SELECT * FROM MEMBER;
```

```
-- Select all records from AUTHOR table
```

```
SELECT * FROM AUTHOR;
```

```
-- Select all records from PUBLISHER table
```

```
SELECT * FROM PUBLISHER;
```



-- Select all records from GENRE table

```
SELECT * FROM GENRE;
```

-- Select all records from BOOK table

```
SELECT * FROM BOOK;
```

-- Select all records from MEMBERSHIP table

```
SELECT * FROM MEMBERSHIP;
```

-- Select all records from BOOK\_AUTHOR table

```
SELECT * FROM BOOK_AUTHOR;
```

-- Select all records from STAFF table

```
SELECT * FROM STAFF;
```

-- Select all records from COPY table

```
SELECT * FROM COPY;
```

-- Select all records from TRANSACTION table

```
SELECT * FROM [TRANSACTION];
```

-- Select all records from RESERVATION table

```
SELECT * FROM RESERVATION;
```

-- Select all records from FEEDBACK table

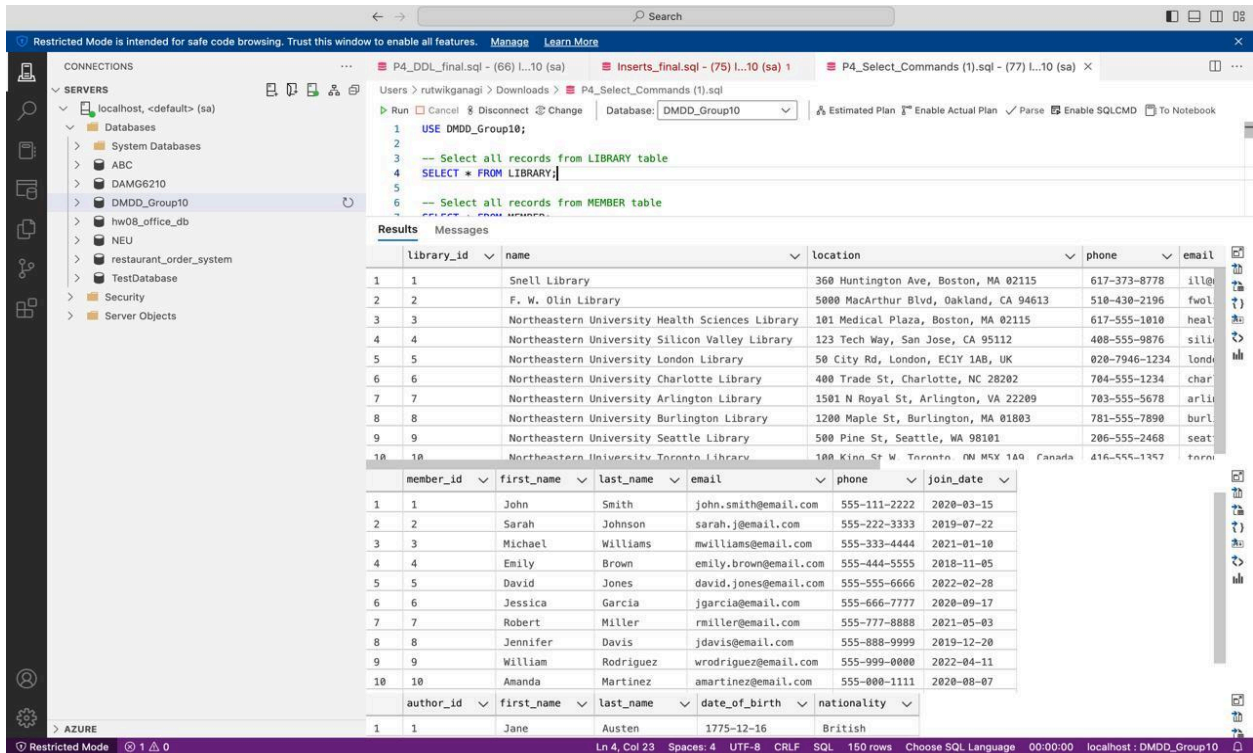
SELECT \* FROM FEEDBACK;

-- Select all records from FINE table

SELECT \* FROM FINE;

\*\*\*\*\*

## Output (Screenshots) ~



Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS: Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Database: DMDD\_Group10

Run [X] Cancel [X] Disconnect [X] Change [X] Estimated Plan [X] Enable Actual Plan [X] Parse [X] Enable SQLCMD [X] To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

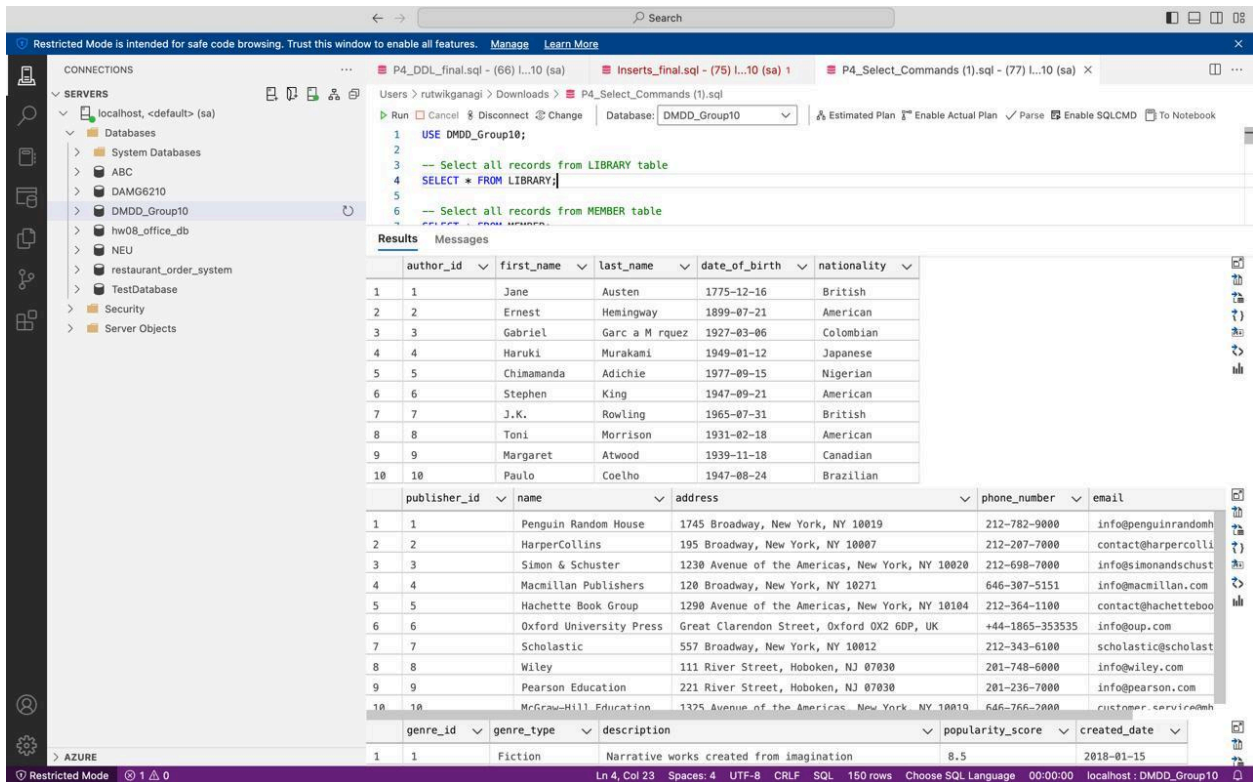
library_id	name	location	phone	email
1	Snell Library	368 Huntington Ave, Boston, MA 02115	617-373-8778	ill@
2	F. W. Olin Library	5000 MacArthur Blvd, Oakland, CA 94613	510-430-2196	fwol
3	Northeastern University Health Sciences Library	101 Medical Plaza, Boston, MA 02115	617-555-1010	heal
4	Northeastern University Silicon Valley Library	123 Tech Way, San Jose, CA 95112	408-555-9876	sili
5	Northeastern University London Library	50 City Rd, London, EC1Y 1AB, UK	020-7946-1234	lond
6	Northeastern University Charlotte Library	400 Trade St, Charlotte, NC 28202	704-555-1234	char
7	Northeastern University Arlington Library	1501 N Royal St, Arlington, VA 22209	703-555-5678	arlin
8	Northeastern University Burlington Library	1200 Maple St, Burlington, MA 01803	781-555-7890	burl
9	Northeastern University Seattle Library	500 Pine St, Seattle, WA 98101	206-555-2468	seatt
10	Northeastern University Toronto Library	100 King St W, Toronto, ON M5X 1A9, Canada	416-555-1357	toron

member_id	first_name	last_name	email	phone	join_date
1	John	Smith	john.smith@email.com	555-111-2222	2020-03-15
2	Sarah	Johnson	sarah.j@email.com	555-222-3333	2019-07-22
3	Michael	Williams	mwilliams@email.com	555-333-4444	2021-01-10
4	Emily	Brown	emily.brown@email.com	555-444-5555	2018-11-05
5	David	Jones	david.jones@email.com	555-555-6666	2022-02-28
6	Jessica	Garcia	jgarcia@email.com	555-666-7777	2020-09-17
7	Robert	Miller	rmiller@email.com	555-777-8888	2021-05-03
8	Jennifer	Davis	jdavis@email.com	555-888-9999	2019-12-20
9	William	Rodriguez	wrodriguez@email.com	555-999-0000	2022-04-11
10	Amanda	Martinez	amartinez@email.com	555-000-1111	2020-08-07

author\_id | first\_name | last\_name | date\_of\_birth | nationality

author_id	first_name	last_name	date_of_birth	nationality
1	Jane	Austen	1775-12-16	British

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10



Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS: Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Database: DMDD\_Group10

Run [X] Cancel [X] Disconnect [X] Change [X] Estimated Plan [X] Enable Actual Plan [X] Parse [X] Enable SQLCMD [X] To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

author_id	first_name	last_name	date_of_birth	nationality
1	Jane	Austen	1775-12-16	British
2	Ernest	Hemingway	1899-07-21	American
3	Gabriel	Garc a M rquez	1927-03-06	Colombian
4	Haruki	Murakami	1949-01-12	Japanese
5	Chimamanda	Adichie	1977-09-15	Nigerian
6	Stephen	King	1947-09-21	American
7	J.K.	Rowling	1965-07-31	British
8	Toni	Morrison	1931-02-18	American
9	Margaret	Atwood	1939-11-18	Canadian
10	Paulo	Coelho	1947-08-24	Brazilian

publisher_id	name	address	phone_number	email
1	Penguin Random House	1745 Broadway, New York, NY 10019	212-782-9000	info@penguinrandomh
2	HarperCollins	195 Broadway, New York, NY 10007	212-207-7000	contact@harpercoll
3	Simon & Schuster	1230 Avenue of the Americas, New York, NY 10020	212-698-7000	info@simonandschust
4	Macmillan Publishers	120 Broadway, New York, NY 10271	646-307-5151	info@macmillan.com
5	Hachette Book Group	1290 Avenue of the Americas, New York, NY 10104	212-364-1100	contact@hachetteboo
6	Oxford University Press	Great Clarendon Street, Oxford OX2 6DP, UK	+44-1865-353535	info@oup.com
7	Scholastic	557 Broadway, New York, NY 10012	212-343-6100	scholastic@scholast
8	Wiley	111 River Street, Hoboken, NJ 07030	201-748-6000	info@wiley.com
9	Pearson Education	221 River Street, Hoboken, NJ 07030	201-236-7000	info@pearson.com
10	McGraw-Hill Education	1375 Avenue of the Americas, New York, NY 10019	646-766-7000	customer.servicedeh

genre_id	genre_type	description	popularity_score	created_date
1	Fiction	Narrative works created from imagination	8.5	2018-01-15

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS

SERVERS

localhost, <default> (sa)

Databases

System Databases

ABC

DAMG6210

DMDD\_Group10

hw08\_office\_db

NEU

restaurant\_order\_system

TestDatabase

Security

Server Objects

Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Run Cancel Disconnect Change Database: DMDD\_Group10 Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

genre_id	genre_type	description	popularity_score	created_date
1	Fiction	Narrative works created from imagination	8.5	2018-01-15
2	Non-Fiction	Works based on facts and real events	7.2	2018-01-15
3	Mystery	Stories focused on solving a crime or puzzle	8.0	2018-02-20
4	Science Fiction	Speculative fiction dealing with advanced science an...	7.8	2018-02-21
5	Fantasy	Works involving magical or supernatural elements	8.3	2018-03-05
6	Biography	Account of a person's life written by someone else	6.5	2018-03-10
7	History	Works focused on past events	6.8	2018-04-12
8	Romance	Stories centered on romantic relationships	7.5	2018-05-15
9	Self-Help	Books offering advice for personal improvement	6.2	2018-06-20
10	Poetry	Literary works with rhythmic qualities of language	5.5	2018-07-25

book_id	library_id	genre_id	publisher_id	title	isbn	publication_d...
1	1	1	1	Pride and Prejudice	9780141439518	1813-01-28
2	2	2	3	The Old Man and the Sea	9780608488123	1952-09-01
3	3	1	1	One Hundred Years of Solitude	9780060883287	1967-05-30
4	4	3	4	Norwegian Wood	9780375784024	1987-09-04
5	5	4	1	Americanah	9780307455925	2013-05-14
6	6	5	5	The Shining	9780307743657	1977-01-28
7	7	6	5	Harry Potter and the Philosopher's Stone	9780747532743	1997-06-26
8	8	7	1	Beloved	9781400033416	1987-09-02
9	9	8	4	The Handmaid's Tale	9780385490818	1985-06-01
10	10	9	1	The Alchemist	9780062711507	1988-04-25

member_library_id	library_id	member_id	registration_date	membership_status
3	3	2	2021-01-10	Active
4	4	3	2018-11-05	Expired
5	5	2	2022-02-28	Active
6	6	4	2020-09-17	Suspended
7	7	5	2021-05-03	Active
8	8	6	2019-12-20	Active
9	9	7	2022-04-11	Pending
10	10	8	2020-08-07	Active
11	11	9	2021-11-15	Active
12	12	10	2019-03-29	Expired

book_id	author_id	author_role	contribution_percentage	special_acknowledgement
1	1	Primary Author	100.00	NULL
2	2	Primary Author	100.00	NULL
3	3	Primary Author	100.00	Winner of Nobel Prize in Literature
4	4	Primary Author	100.00	NULL
5	5	Primary Author	100.00	NULL
6	6	Primary Author	100.00	NULL
7	7	Primary Author	100.00	NULL
8	8	Primary Author	100.00	Winner of Pulitzer Prize
9	9	Primary Author	100.00	NULL
10	10	Primary Author	100.00	NULL

staff_id	library_id	first_name	last_name	role	email	phone
1	1	Patricia	Wilson	Head Librarian	pwilson@library.org	555-121-2323

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS

SERVERS

localhost, <default> (sa)

Databases

System Databases

ABC

DAMG6210

DMDD\_Group10

hw08\_office\_db

NEU

restaurant\_order\_system

TestDatabase

Security

Server Objects

Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Run Cancel Disconnect Change Database: DMDD\_Group10 Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

member_library_id	library_id	member_id	registration_date	membership_status
3	3	2	2021-01-10	Active
4	4	3	2018-11-05	Expired
5	5	2	2022-02-28	Active
6	6	4	2020-09-17	Suspended
7	7	5	2021-05-03	Active
8	8	6	2019-12-20	Active
9	9	7	2022-04-11	Pending
10	10	8	2020-08-07	Active
11	11	9	2021-11-15	Active
12	12	10	2019-03-29	Expired

book_id	author_id	author_role	contribution_percentage	special_acknowledgement
1	1	Primary Author	100.00	NULL
2	2	Primary Author	100.00	NULL
3	3	Primary Author	100.00	Winner of Nobel Prize in Literature
4	4	Primary Author	100.00	NULL
5	5	Primary Author	100.00	NULL
6	6	Primary Author	100.00	NULL
7	7	Primary Author	100.00	NULL
8	8	Primary Author	100.00	Winner of Pulitzer Prize
9	9	Primary Author	100.00	NULL
10	10	Primary Author	100.00	NULL

staff_id	library_id	first_name	last_name	role	email	phone
1	1	Patricia	Wilson	Head Librarian	pwilson@library.org	555-121-2323

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS

SERVERS

localhost <default> (sa)

Databases

System Databases

ABC

DAMG6210

DMDD\_Group10

hw08\_office\_db

NEU

restaurant\_order\_system

TestDatabase

Security

Server Objects

Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Database: DMDD\_Group10

Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

	staff_id	library_id	first_name	last_name	role	email	phone
2	2	1	Richard	Moore	Reference Librarian	rmoore@library.org	555-131-2424
3	3	2	Susan	Taylor	Library Assistant	staylor@library.org	555-141-2525
4	4	3	James	Anderson	Library Technician	janderson@library.org	555-151-2626
5	5	4	Lisa	Thomas	Head Librarian	lthomas@library.org	555-161-2727
6	6	5	Mark	Jackson	Reference Librarian	mjackson@library.org	555-171-2828
7	7	6	Karen	White	Library Assistant	kwhite@library.org	555-181-2929
8	8	7	Daniel	Harris	Library Technician	dharris@library.org	555-191-3030
9	9	8	Nancy	Martin	Head Librarian	nmartin@library.org	555-212-3131
10	10	9	Paul	Thompson	Reference Librarian	pthompson@library.org	555-232-3232
11	11	10	Laura	Garcia	Library Assistant	lgarcia@library.org	555-242-3333

	copy_id	book_id	condition	purchase_date	availability_status
4	4	3	New	2022-02-05	Reserved
5	5	4	Good	2021-07-18	Available
6	6	5	Poor	2018-09-12	Available
7	7	6	Good	2019-11-30	Unavailable
8	8	7	New	2022-03-15	Available
9	9	8	Fair	2020-05-22	Available
10	10	9	Good	2021-04-14	Reserved
11	11	10	New	2022-01-08	Available
12	12	11	Good	2021-08-19	Available
13	13	12	Fair	2020-10-25	Available

	transaction_id	member_id	copy_id	staff_id	borrow_date	due_date	return_date	transaction_status
1	1	1	1	1	2023-01-15	2023-02-15	2023-02-10	Completed

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS

SERVERS

localhost <default> (sa)

Databases

System Databases

ABC

DAMG6210

DMDD\_Group10

hw08\_office\_db

NEU

restaurant\_order\_system

TestDatabase

Security

Server Objects

Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Database: DMDD\_Group10

Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

	transaction_id	member_id	copy_id	staff_id	borrow_date	due_date	return_date	transaction_status
1	1	1	1	1	2023-01-15	2023-02-15	2023-02-10	Completed
2	2	2	3	2	2023-02-05	2023-03-05	2023-03-02	Completed
3	3	3	5	3	2023-03-10	2023-04-10	NULL	Active
4	4	4	7	4	2023-01-20	2023-02-20	2023-03-01	Overdue
5	5	5	9	5	2023-04-05	2023-05-05	NULL	Active
6	6	6	11	6	2023-02-25	2023-03-25	2023-03-20	Completed
7	7	7	2	7	2023-03-15	2023-04-15	NULL	Active
8	8	8	4	8	2023-02-10	2023-03-10	2023-04-05	Overdue
9	9	9	6	9	2023-04-01	2023-05-01	NULL	Active
10	10	10	8	10	2023-01-05	2023-02-05	NULL	Lost

	reservation_id	member_id	book_id	staff_id	reservation_date	reservation_status
1	1	1	3	1	2023-04-01	Pending
2	2	2	9	2	2023-03-25	Pending
3	3	3	6	3	2023-03-15	Cancelled
4	4	4	1	4	2023-02-10	Fulfilled
5	5	5	7	5	2023-04-05	Pending
6	6	6	2	6	2023-03-01	Expired
7	7	7	10	7	2023-04-10	Pending
8	8	8	5	8	2023-03-20	Fulfilled
9	9	9	8	9	2023-02-15	Cancelled
10	10	10	4	10	2023-04-02	Pending

	feedback_id	book_id	member_id	feedback_date	rating	comment
1	1	1	1	2023-02-12	5	A timeless classic that never gets old!

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

CONNECTIONS

SERVERS

- localhost, <default> (sa)
- Databases
  - System Databases
  - ABC
  - DAMG6210
  - DMDD\_Group10
  - hw08\_office\_db
  - NEU
  - restaurant\_order\_system
  - TestDatabase
  - Security
  - Server Objects

Users > rutwikganagi > Downloads > P4\_Select\_Commands (1).sql

Database: DMDD\_Group10

Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

```
1 USE DMDD_Group10;
2
3 -- Select all records from LIBRARY table
4 SELECT * FROM LIBRARY;
5
6 -- Select all records from MEMBER table
7 SELECT * FROM MEMBER;
```

Results Messages

9	9	9	8	9	2023-02-15	Cancelled
10	10	10	4	10	2023-04-02	Pending

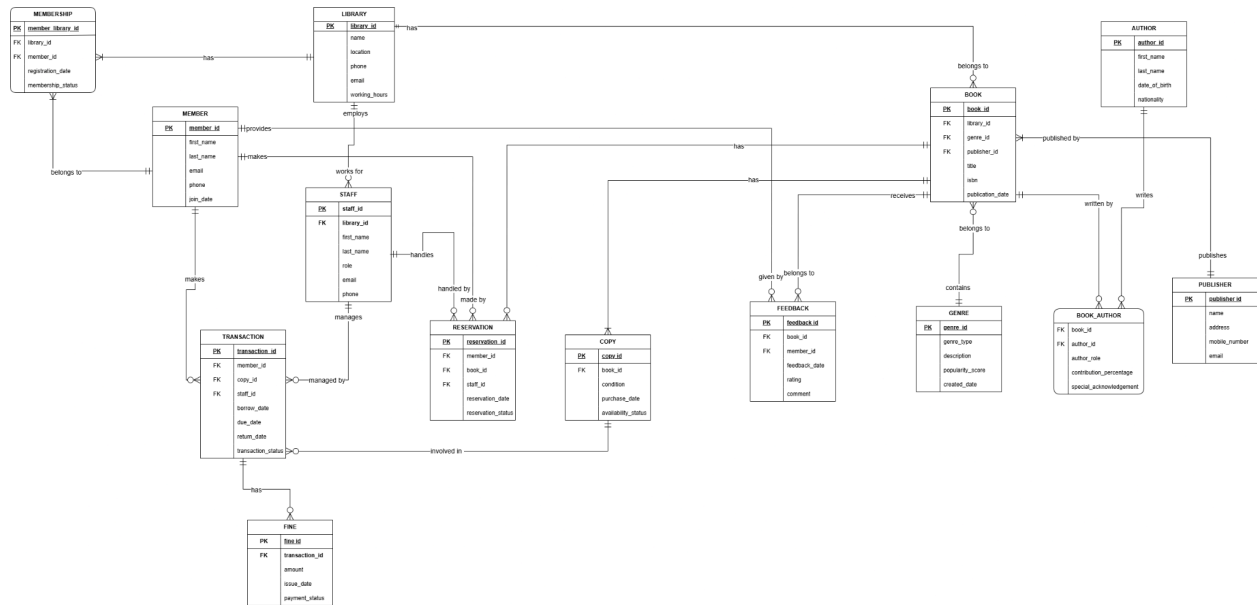
	feedback_id	book_id	member_id	feedback_date	rating	comment
1	1	1	1	2023-02-12	5	A timeless classic that never gets old!
2	2	2	2	2023-03-04	4	A beautiful story about perseverance.
3	3	3	4	2023-01-10	5	One of the greatest novels ever written.
4	4	4	6	2023-03-22	3	I found it a bit slow, but the writing is beautiful.
5	5	5	8	2023-04-06	5	A powerful exploration of identity and belonging.
6	6	6	3	2023-02-20	4	Genuinely scary and psychologically complex.
7	7	7	5	2023-03-18	5	Magical and captivating from start to finish.
8	8	8	7	2023-01-25	5	A profound and moving masterpiece.
9	9	9	9	2023-02-28	4	Disturbing but thought-provoking.
10	10	10	10	2023-04-03	5	Life-changing wisdom in a simple story.

	fine_id	transaction_id	amount	issue_date	payment_status
1	1	4	5.00	2023-03-02	Paid
2	2	8	10.00	2023-04-06	Unpaid
3	3	10	25.00	2023-02-06	Unpaid
4	4	1	2.50	2023-02-16	Waived
5	5	2	3.00	2023-03-06	Paid
6	6	3	7.50	2023-04-11	Unpaid
7	7	5	4.00	2023-05-06	Unpaid
8	8	6	2.00	2023-03-26	Paid
9	9	7	6.00	2023-04-16	Unpaid
10	10	9	8.00	2023-05-02	Unpaid

Ln 4, Col 23 Spaces: 4 UTF-8 CRLF SQL 150 rows Choose SQL Language 00:00:00 localhost: DMDD\_Group10

\*\*\*\*\*

## Updated ERD:



Link ~ [ERD - P4](#)



## *Changes from P3 to P4 ERD ~*

The differences between P3 and P4 entity-relationship diagrams (ERDs) are:

### 1. AUTHOR Table Changes:

P3: The AUTHOR table contains `author_id`, `first_name`, and `last_name`.

P4: The AUTHOR table has additional attributes:

`date_of_birth`

`nationality`

### 2. BOOK\_AUTHOR Table Changes:

P3: The BOOK\_AUTHOR table contains `book_id` and `author_id`.

P4: The BOOK\_AUTHOR table has additional attributes:

`author_role`

`contribution_percentage`

`special_acknowledgement`

### 3. GENRE Table Changes:

P3: The GENRE table contains `genre_id`, `genre_type`, and `description`.

P4: The GENRE table has additional attributes:

`popularity_score`

`created_date`

### 4. COPY Table Changes:

P3: The COPY table contains `copy_id`, `book_id`, `condition`, and `purchase_date`.

P4: The COPY table has an additional attribute:

`availability_status`



## 5. BOOK Table Changes:

P3: The BOOK table contains book\_id, library\_id, category\_id, publisher\_id, title, isbn, publication\_date, and availability\_status.

P4: The BOOK table replaces category\_id with genre\_id and availability\_status is removed.

## 6. Minor Layout Adjustments:

The overall structure remains similar, but P4 contains additional attributes for better granularity in authorship, genre tracking, and book details.

## **Summary ~**

P4 is an improved version of P3, incorporating more details in the AUTHOR, BOOK\_AUTHOR, GENRE, COPY, and BOOK tables. These modifications enhance tracking of author contributions, book categorization, and availability.

***Group 10 (Team Members ):***

1) Name: Rutwik Ganagi

Email: [ganagi.r@northeastern.edu](mailto:ganagi.r@northeastern.edu)

2) Name: Sachin Vishaul Baskar

Email: [baskar.sa@northeastern.edu](mailto:baskar.sa@northeastern.edu)

3) Name: Ashwin Badamikar

Email: [badamikar.a@northeastern.edu](mailto:badamikar.a@northeastern.edu)

4) Name: Dennis sharon

Email: [cheruvathoorshaj.d@northeastern.edu](mailto:cheruvathoorshaj.d@northeastern.edu)

5) Name: Chetan Warad

Email: [warad.c@northeastern.edu](mailto:warad.c@northeastern.edu)

Github URL - <https://github.com/rutwikganagi2000/DMDD-Group-10>