Program	:B.tech(CSE)
Specialization	:AIML
Course Title	:AI Assisted Coding
Course Code	:24CS002PC215
Semester	:3rd semester
Academic Session	:2025-2026
Name of Student	:Saini Deepthi
Enrollment No.	:2403A52014
Batch No.	:02
Date	:04/11/2025

#LAB ASSIGNMENT-16.2

#TASK DESCRIPTION-1:

Task: Ask AI to design a schema for a Library Management System
(Tables: Books, Members, Loans).
SQL Code

```sql
CREATE TABLE Members (
    member_id INT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100) UNIQUE,
    join_date DATE
);


CREATE TABLE Books (
    book_id INT PRIMARY KEY,
    title VARCHAR(200),
    author VARCHAR(100),
    available BOOLEAN
);


CREATE TABLE Loans (
    loan_id INT PRIMARY KEY,
    member_id INT,
    book_id INT,
    loan_date DATE,
    return_date DATE,
    FOREIGN KEY (member_id) REFERENCES Members(member_id),
    FOREIGN KEY (book_id) REFERENCES Books(book_id)
);
```

#PROMPT:

Design an **SQL schema for a Library Management System** that includes three tables —
Members, Books, and Loans.
Each table should have appropriate **primary keys**, **foreign keys**, and relevant columns such as

member details, book details, and loan records.

Write complete and executable **SQL CREATE TABLE statements**.

Also include **sample INSERT statements** to add example data into each table.

#CODE:

```sql
-- Create Members table
CREATE TABLE Members (
    member_id INT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15),
    join_date DATE NOT NULL,
    status VARCHAR(20) DEFAULT 'ACTIVE'
);

-- Create Books table
CREATE TABLE Books (
    book_id INT PRIMARY KEY,
    title VARCHAR(200) NOT NULL,
    author VARCHAR(100) NOT NULL,
    isbn VARCHAR(13) UNIQUE NOT NULL,
    publication_year INT,
    genre VARCHAR(50),
    total_copies INT NOT NULL,
    available_copies INT NOT NULL
);
```

```sql
-- Create Loans table
CREATE TABLE Loans (
    loan_id INT PRIMARY KEY,
    member_id INT,
    book_id INT,
    loan_date DATE NOT NULL,
    due_date DATE NOT NULL,
    return_date DATE,
    FOREIGN KEY (member_id) REFERENCES Members(member_id),
    FOREIGN KEY (book_id) REFERENCES Books(book_id)
);

-- Insert sample data into Members table
INSERT INTO Members (member_id, first_name, last_name, email, phone, join_date)
VALUES
(1, 'John', 'Doe', 'john.doe@email.com', '555-0101', '2023-01-15'),
(2, 'Jane', 'Smith', 'jane.smith@email.com', '555-0102', '2023-02-20'),
(3, 'Alice', 'Johnson', 'alice.j@email.com', '555-0103', '2023-03-25');

-- Insert sample data into Books table
INSERT INTO Books (book_id, title, author, isbn, publication_year, genre, total_copies, available_copies)
VALUES
```

```sql
-- Insert sample data into Books table
INSERT INTO Books (book_id, title, author, isbn, publication_year, genre, total_copies, available_copies)
VALUES
(1, 'The Great Gatsby', 'F. Scott Fitzgerald', '9780743273565', 1925, 'Fiction', 3, 2),
(2, '1984', 'George Orwell', '9780451524935', 1949, 'Science Fiction', 5, 4),
(3, 'Pride and Prejudice', 'Jane Austen', '9780141439518', 1813, 'Romance', 4, 3);

-- Insert sample data into Loans table
INSERT INTO Loans (loan_id, member_id, book_id, loan_date, due_date, return_date)
VALUES
(1, 1, 1, '2023-06-01', '2023-06-15', '2023-06-14'),
(2, 2, 2, '2023-06-10', '2023-06-24', NULL),
(3, 3, 3, '2023-06-12', '2023-06-26', NULL);
```

OUTPUT:

```
[1:49:15 PM] [SQL Tools Service] Initializing SQL tools service for the mssql extension.
[1:49:15 PM] [SQL Tools Service] Note: mssql commands will be available after installing the service.
[1:49:15 PM] [SQL Tools Service]
[1:49:15 PM] [SQL Tools Service] Platform: win32, x86_64 (Windows)
[1:49:15 PM] [SQL Tools Service]
[1:49:15 PM] [SQL Tools Service] Installing SQL tools service to c:\Users\deeks\.vscode\extensions\ms-mssql.mssql-1.36.
0\sqltoolsservice\5.0.20250828.1\Windows.
[1:49:15 PM] [SQL Tools Service] Downloading https://github.com/Microsoft/sqltoolsservice/releases/download/5.0.20250828.1/microsoft.
sqltools.servicelayer-win-x64-net8.0.zip
[1:49:17 PM] [SQL Tools Service] (80557 KB) ................... Done!
[1:49:32 PM] [SQL Tools Service] Installing ...
[1:49:32 PM] [SQL Tools Service] Extracted: Azure.AI.OpenAI.dll
[1:49:32 PM] [SQL Tools Service] Extracted: Azure.Core.dll
[1:49:32 PM] [SQL Tools Service] Extracted: Azure.Identity.dll
[1:49:32 PM] [SQL Tools Service] Extracted: Azure.ResourceManager.dll
[1:49:32 PM] [SQL Tools Service] Extracted: Azure.ResourceManager.Sql.dll
[1:49:33 PM] [SQL Tools Service] Extracted: Azure.Storage.Blobs.dll
[1:49:33 PM] [SQL Tools Service] Extracted: Azure.Storage.Common.dll
[1:49:33 PM] [SQL Tools Service] Extracted: clretwrc.dll
[1:49:33 PM] [SQL Tools Service] Extracted: clrgc.dll
[1:49:33 PM] [SQL Tools Service] Extracted: clrjit.dll
[1:49:33 PM] [SQL Tools Service] Extracted: coreclr.dll
[1:49:33 PM] [SQL Tools Service] Extracted: createdump.exe
[1:49:33 PM] [SOL Tools Service] Extracted: cs-CZ/Microsoft.SqlServer.TransactSql.ScriptDom.Nrt.resources.dll
```

#TASK DESCRIPTION-2:

Error Insert Data
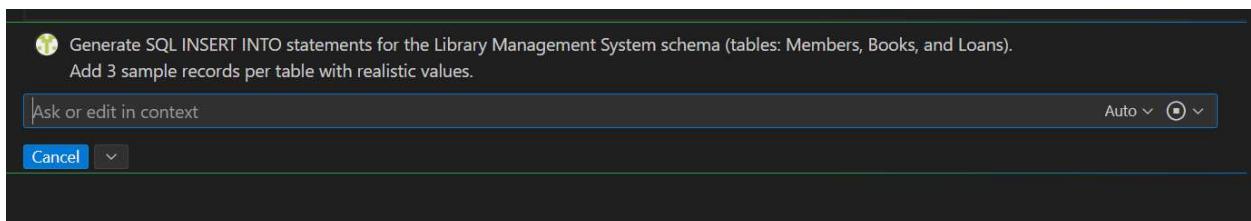
Task: Ask AI to generate INSERT INTO queries for the schema above

(3 sample records per table).   Give me simple and perfect prompt for this

#PROMPT:

Generate SQL INSERT INTO statements for the Library Management System
schema (tables: Members, Books, and Loans).
Add **3 sample records per table** with realistic values.

#CODE:

#TASK DESCRIPTION-3:

Basic Queries
Task: Use AI to generate a query to list all books borrowed by a specific member

#PROMPT:

Generate an SQL query to list all books borrowed by a specific member from the Library Management System database (tables: Members, Books, and Loans).
The query should display the member's name, book title, author, and loan date.

```
SELECT
   🌐 Generate an SQL query to list all books borrowed by a specific member from the Library Management System database (tables: Members, Books,
      and Loans).
      The query should display the member's name, book title, author, and loan date.

   Ask or edit in context                                                                                    Auto ∨  ▷ ∨

   Cancel   ∨
```

#CODE:

```sql
    ♀ Connect to MSSQL
 1   SELECT
 2       Members.name AS member_name,
 3       Books.title AS book_title,
 4       Books.author AS book_author,
 5       Loans.loan_date AS loan_date
 6   FROM
 7       Members
 8   JOIN
 9       Loans ON Members.id = Loans.member_id
10   JOIN
11       Books ON Loans.book_id = Books.id
12   WHERE
13       Members.id = ?;  -- Replace ? with the specific member's ID
```

#OUTPUT:

#TASK DESCRIPTION-4:

Update and Delete Queries
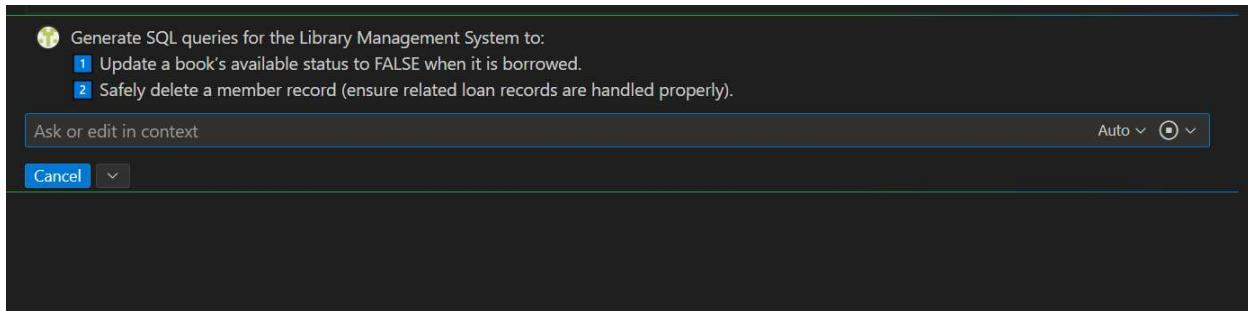
Task: Generate queries with AI for:

• Updating a book's availability to FALSE when borrowed.

• Deleting a member record safely.   Give me perfect and simple prompt

#PROMPT:

Generate SQL queries for the Library Management System to:
1. Update a book's available status to FALSE when it is borrowed.
2. Safely delete a member record (ensure related loan records are handled properly).

#CODE:

```
⏻ Connect to MSSQL
-- 1  Update a book's available status to FALSE when it is borrowed
UPDATE books
SET available = FALSE
WHERE book_id = ?;  -- Replace ? with the actual book ID

-- 2  Safely delete a member record (ensure related loan records are handled properly)
DELETE FROM loans
WHERE member_id = ?;  -- Replace ? with the actual member ID

DELETE FROM members
WHERE member_id = ?;  -- Replace ? with the actual member ID
```

#OUTPUT:

```
⏻ Connect to MSSQL
1   -- 1  Update a book's available status to FALSE when it is borrowed
2   UPDATE books
3   SET available = FALSE
4   WHERE book_id = ?;  -- Replace ? with the actual book ID
5
6   -- 2  Safely delete a member record (ensure related loan records are handled properly)
7   DELETE FROM loans
8   WHERE member_id = ?;  -- Replace ? with the actual member ID
9
10  DELETE FROM members
11  WHERE member_id = ?;  -- Replace ? with the actual member ID
```

------- Thank You-------