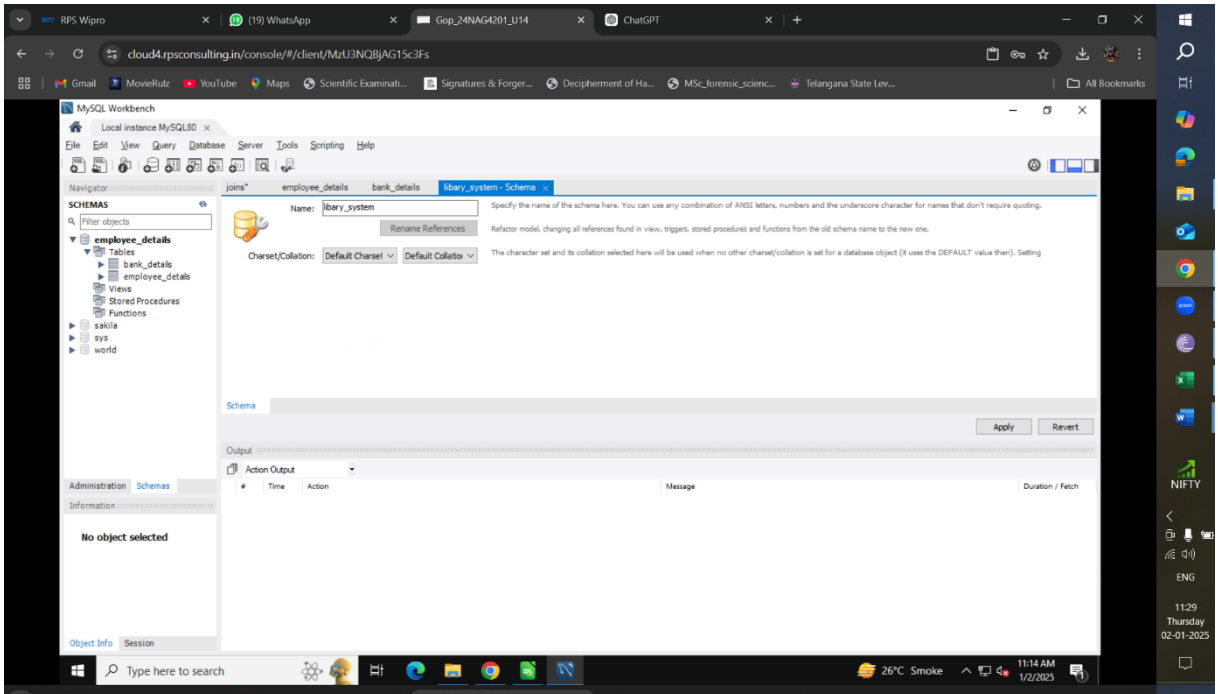
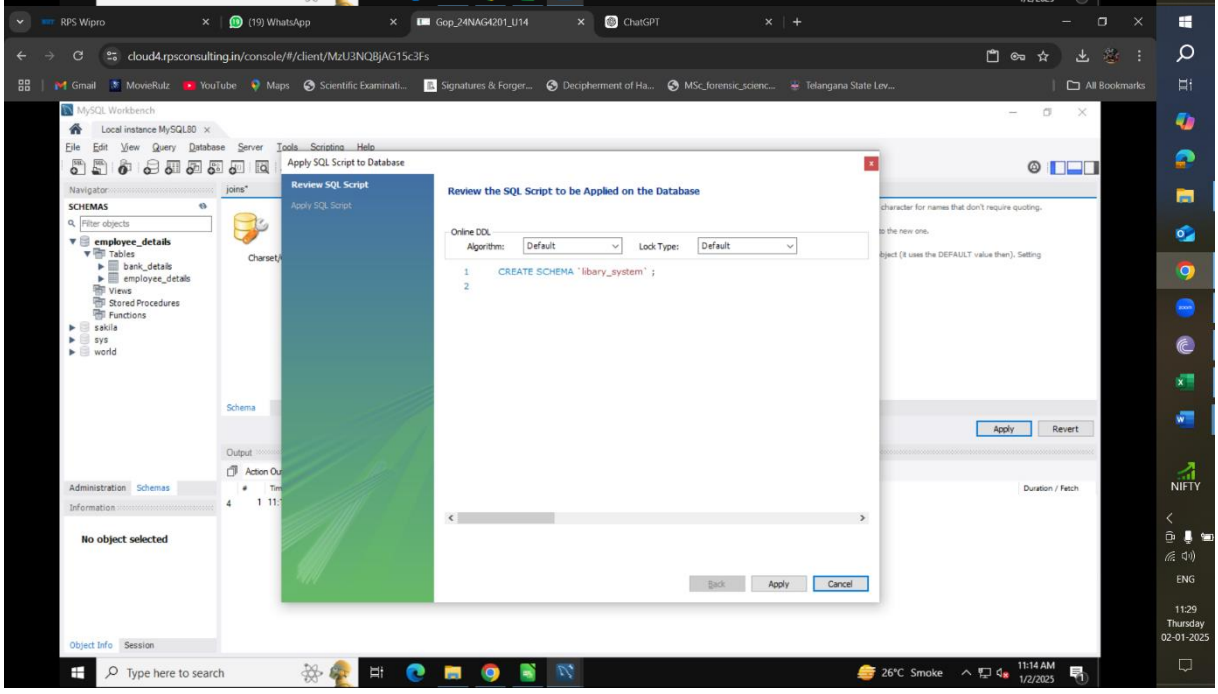
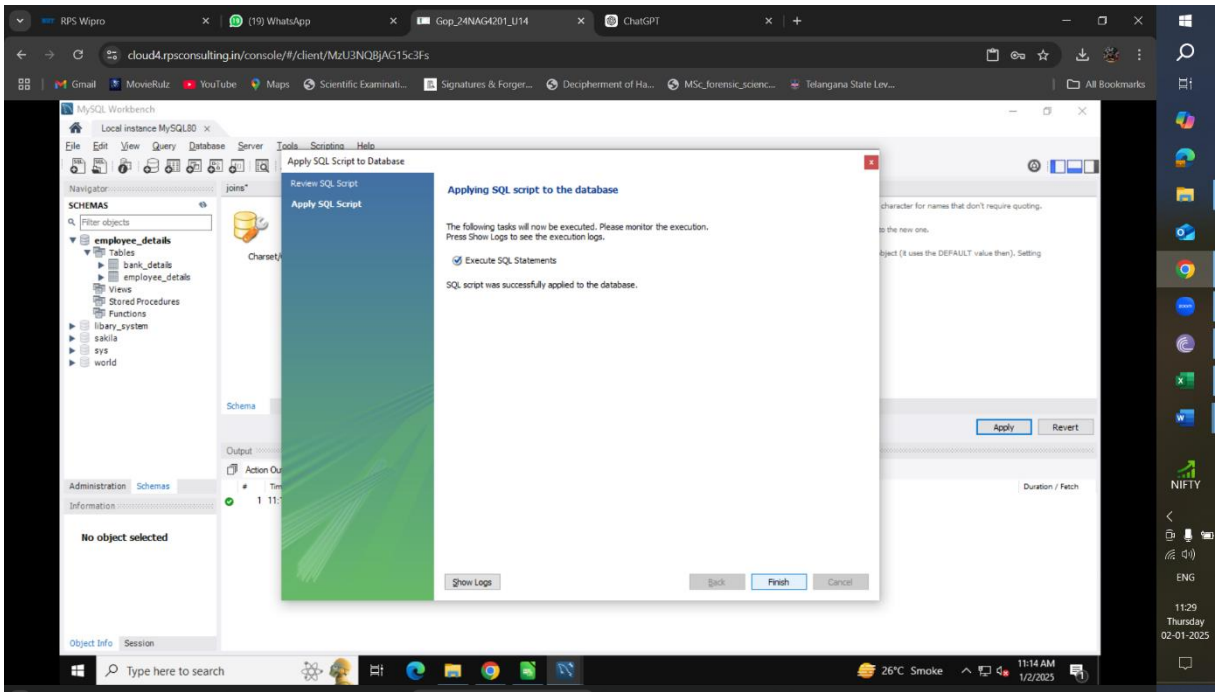
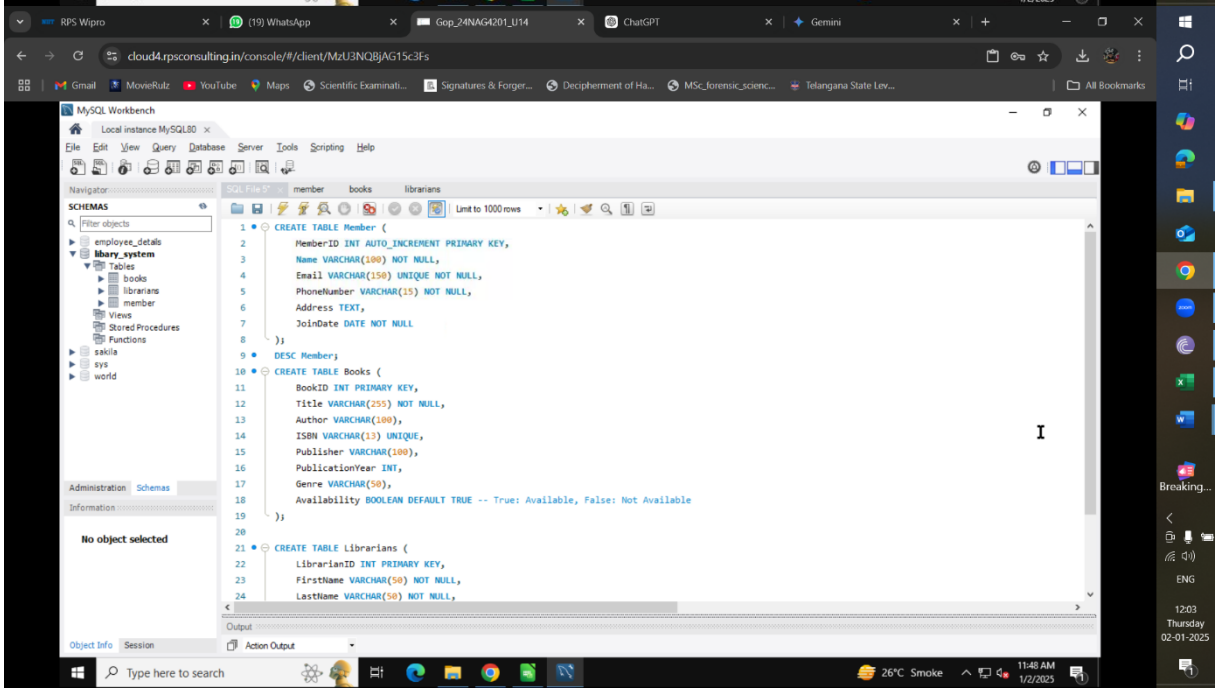
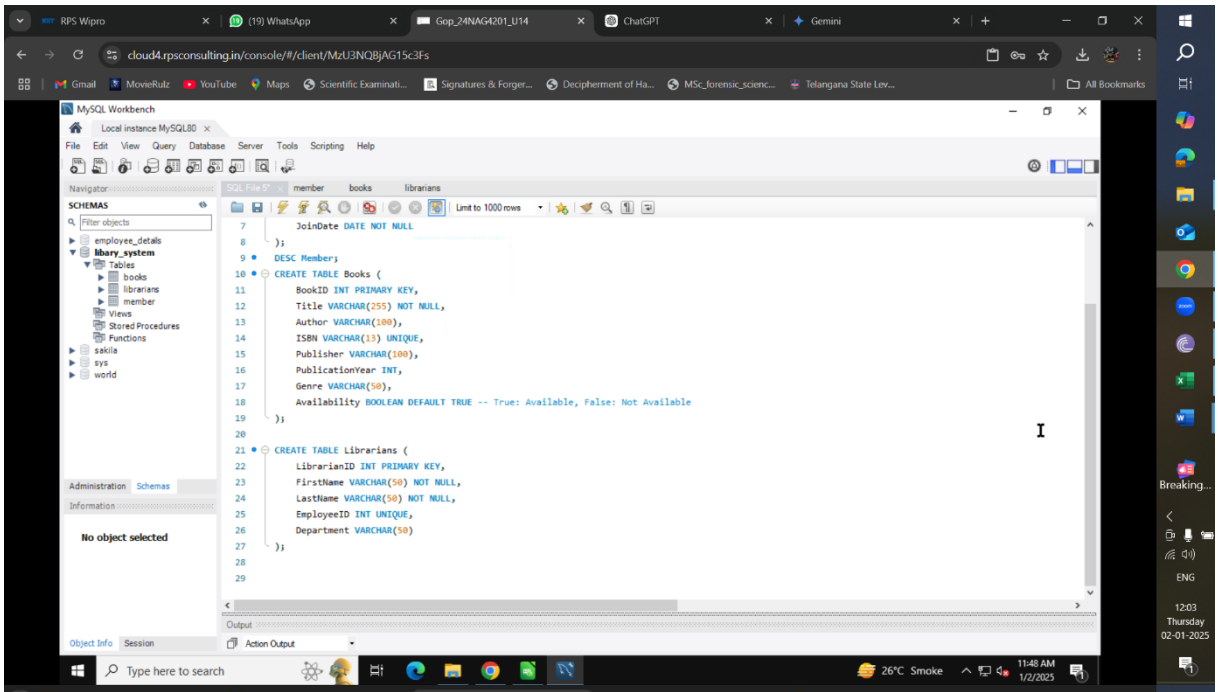
Design a database schema for a library system, including tables, fields, and constraints like NOT NULL, UNIQUE, and CHECK. Include primary and foreign keys to establish relationships between tables.

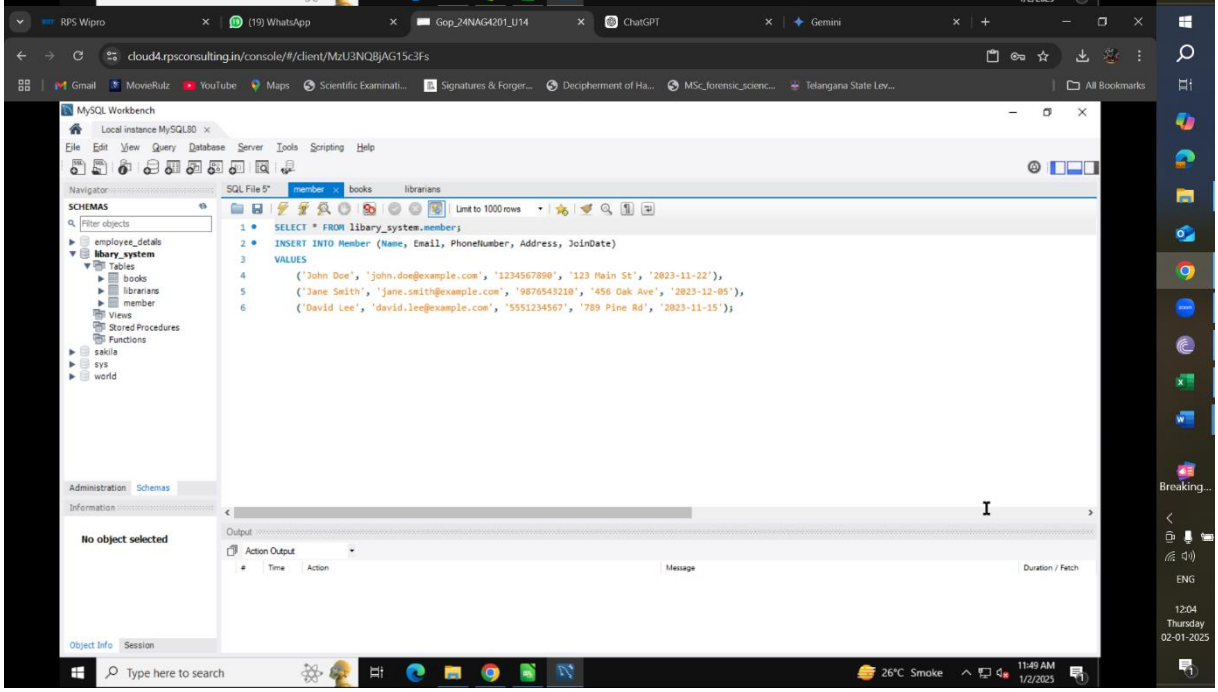


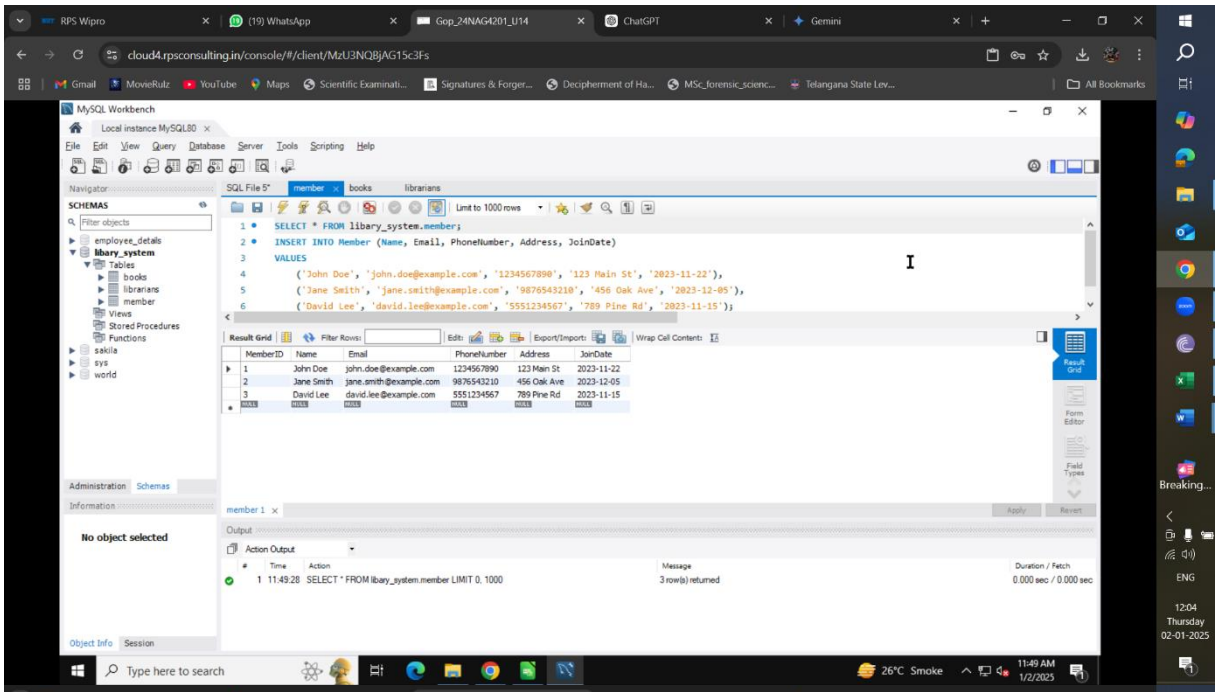


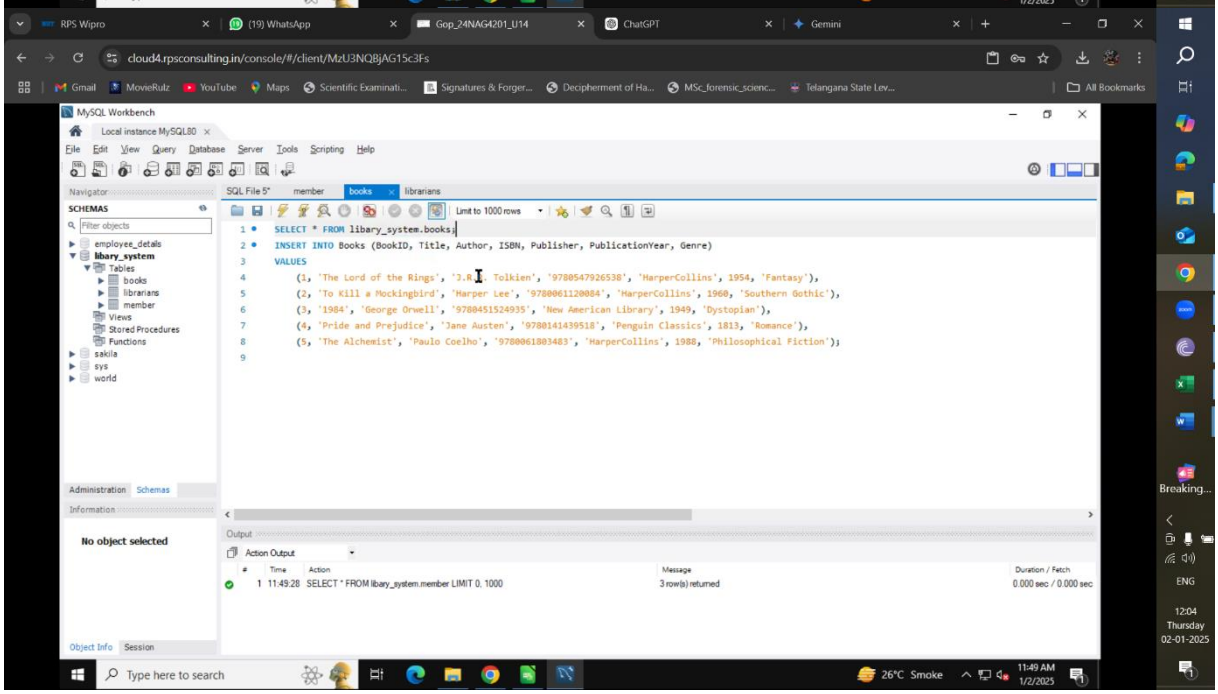

RPS Wipro
(19) WhatsApp
Gop_24NAG4201_U14
ChatGPT
cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Applying SQL script to the database
The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.
Execute SQL Statements
SQL script was successfully applied to the database.
Show Logs
Back
Finish
Cancel
No object selected
Apply
Revert
Type here to search
26°C Smoke
11:14 AM
1/2/2025

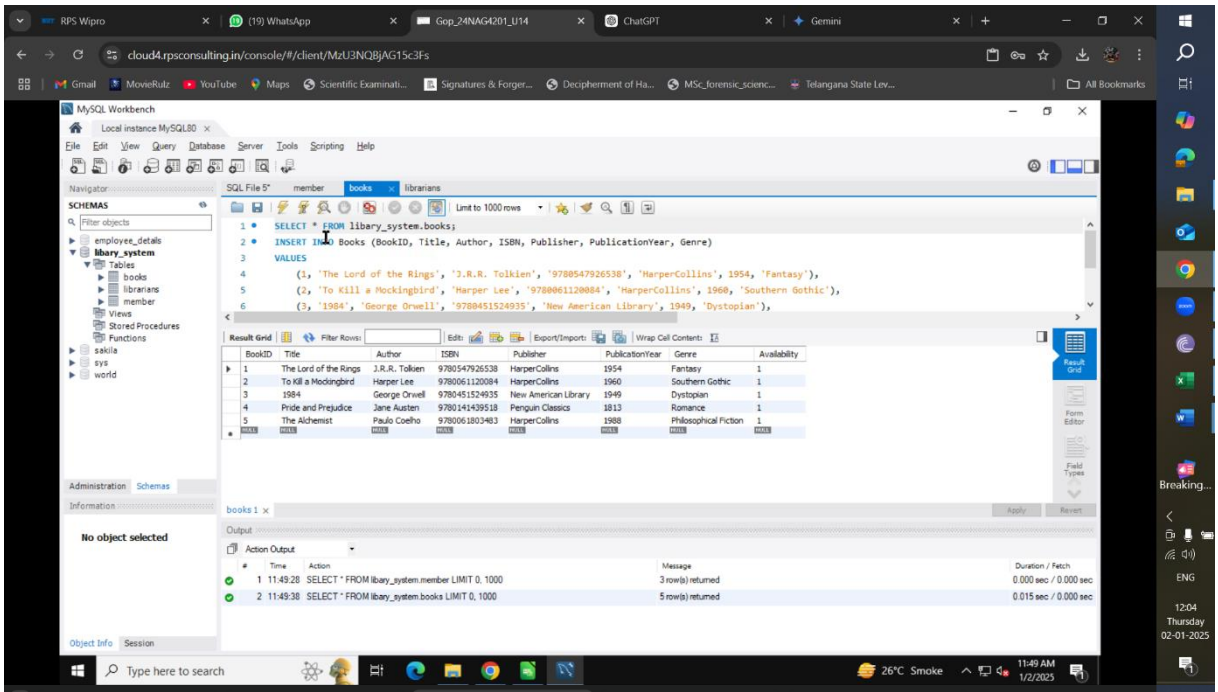

RPS Wipro
(19) WhatsApp
Gop_24NAG4201_U14
ChatGPT
Gemini
cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
SQL File 5*
member
books
librarians
Limit to 1000 rows
1 CREATE TABLE Member (
2 MemberID INT AUTO_INCREMENT PRIMARY KEY,
3 Name VARCHAR(100) NOT NULL,
4 Email VARCHAR(150) UNIQUE NOT NULL,
5 PhoneNumber VARCHAR(15) NOT NULL,
6 Address TEXT,
7 JoinDate DATE NOT NULL
8);
9 DESC Member;
10 CREATE TABLE Books (
11 BookID INT PRIMARY KEY,
12 Title VARCHAR(255) NOT NULL,
13 Author VARCHAR(100),
14 ISBN VARCHAR(13) UNIQUE,
15 Publisher VARCHAR(100),
16 PublicationYear INT,
17 Genre VARCHAR(50),
18 Availability BOOLEAN DEFAULT TRUE -- True: Available, False: Not Available
19);
20
21 CREATE TABLE Librarians (
22 LibrarianID INT PRIMARY KEY,
23 FirstName VARCHAR(50) NOT NULL,
24 LastName VARCHAR(50) NOT NULL,
No object selected
Action Output
Type here to search
26°C Smoke
11:48 AM
1/2/2025

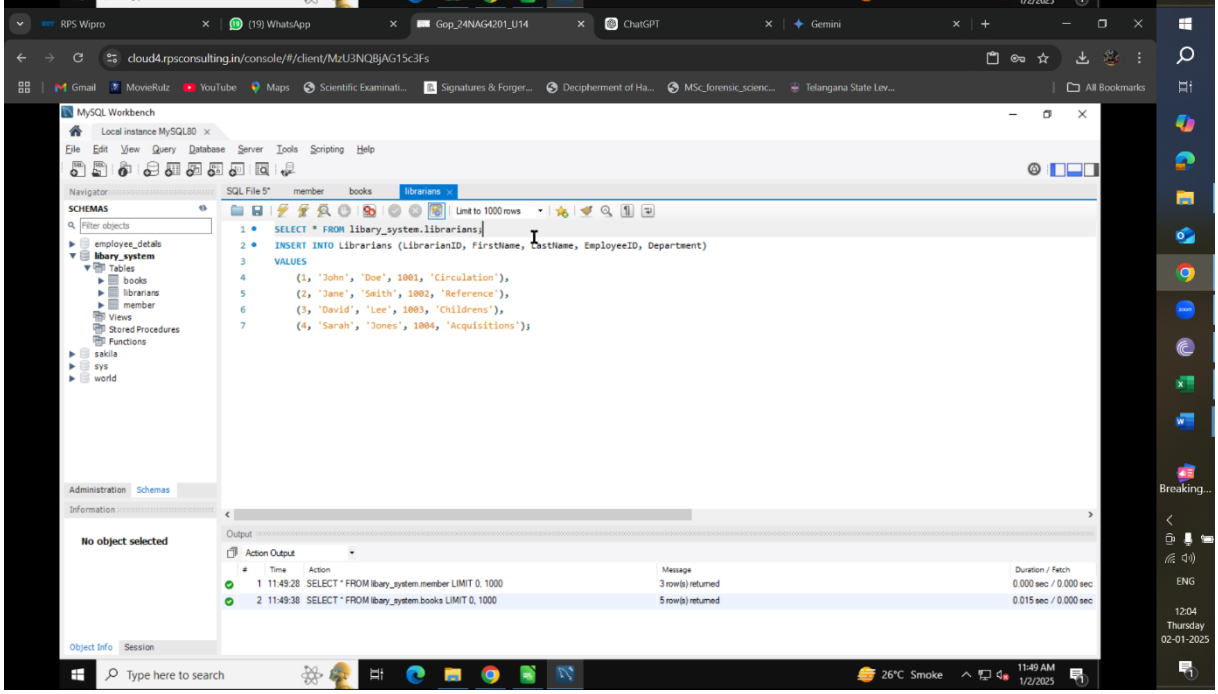

RPS Wipro
(19) WhatsApp
Gop_24NAG4201_U14
ChatGPT
Gemini
cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
SQL File 5*
member
books
librarians
Limit to 1000 rows
SCHEMAS
employee_details
libary_system
Tables
books
librarians
member
Views
Stored Procedures
Functions
sakila
sys
world
Administration Schemas
Information
No object selected
Object Info Session
Output
Action Output
JoinDate DATE NOT NULL
);
DESC Member;
CREATE TABLE Books (
BookID INT PRIMARY KEY,
Title VARCHAR(255) NOT NULL,
Author VARCHAR(100),
ISBN VARCHAR(13) UNIQUE,
Publisher VARCHAR(100),
PublicationYear INT,
Genre VARCHAR(50),
Availability BOOLEAN DEFAULT TRUE -- True: Available, False: Not Available
);
CREATE TABLE Librarians (
LibrarianID INT PRIMARY KEY,
FirstName VARCHAR(50) NOT NULL,
LastName VARCHAR(50) NOT NULL,
EmployeeID INT UNIQUE,
Department VARCHAR(50)
);
Type here to search
26°C Smoke
11:48 AM
1/2/2025
ENG
12:03
Thursday
02-01-2025

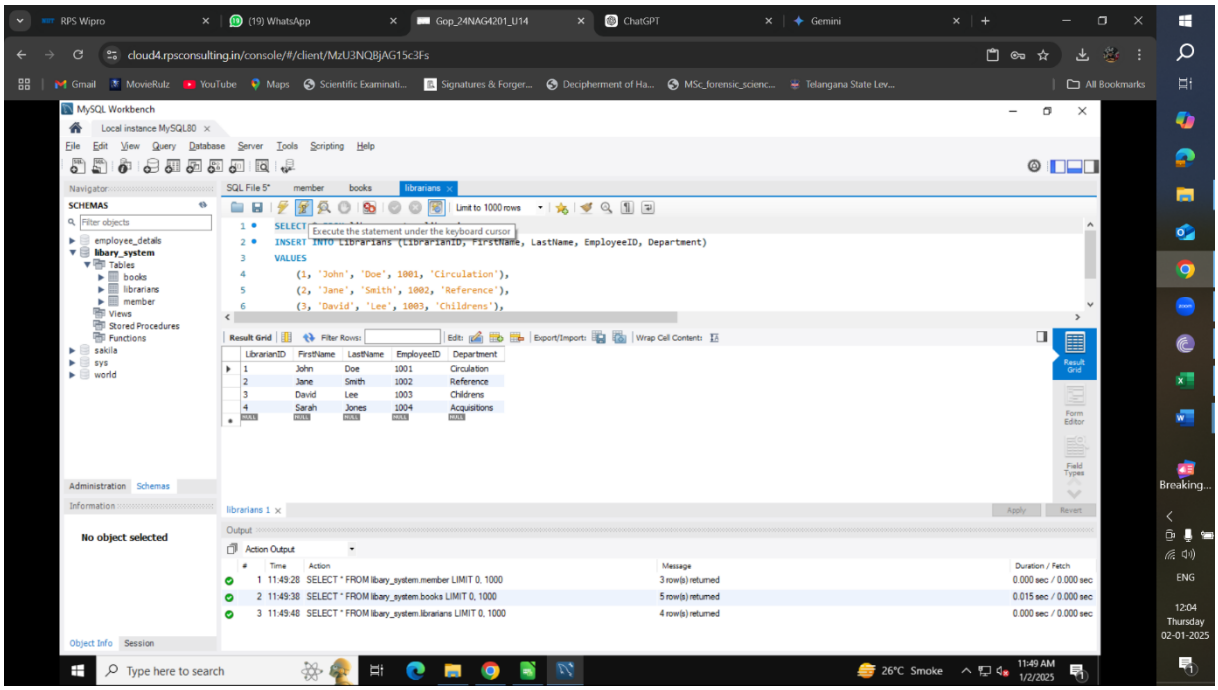

RPS Wipro
(19) WhatsApp
Gop_24NAG4201_U14
ChatGPT
Gemini
cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
SQL File 5*
member
books
librarians
Limit to 1000 rows
SCHEMAS
employee_details
libary_system
Tables
books
librarians
member
Views
Stored Procedures
Functions
sakila
sys
world
Administration Schemas
Information
No object selected
Object Info Session
Output
Action Output
Time Action Message Duration / Fetch
SELECT * FROM libary_system.member;
INSERT INTO Member (Name, Email, PhoneNumber, Address, JoinDate)
VALUES
('John Doe', 'john.doe@example.com', '1234567890', '123 Main St', '2023-11-22'),
('Jane Smith', 'jane.smith@example.com', '9876543210', '456 Oak Ave', '2023-12-05'),
('David Lee', 'david.lee@example.com', '5551234567', '789 Pine Rd', '2023-11-15');
Type here to search
26°C Smoke
11:49 AM
1/2/2025
ENG
12:04
Thursday
02-01-2025


cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
SQL File 5*
member
books
librarians
SELECT * FROM libary_system.member;
INSERT INTO Member (Name, Email, PhoneNumber, Address, JoinDate)
VALUES
('John Doe', 'john.doe@example.com', '1234567890', '123 Main St', '2023-11-22'),
('Jane Smith', 'jane.smith@example.com', '9876543210', '456 Oak Ave', '2023-12-05'),
('David Lee', 'david.lee@example.com', '5551234567', '789 Pine Rd', '2023-11-15');
MemberID Name Email PhoneNumber Address JoinDate
1 John Doe john.doe@example.com 1234567890 123 Main St 2023-11-22
2 Jane Smith jane.smith@example.com 9876543210 456 Oak Ave 2023-12-05
3 David Lee david.lee@example.com 5551234567 789 Pine Rd 2023-11-15
No object selected
1 11:49:28 SELECT * FROM libary_system.member LIMIT 0, 1000
3 row(s) returned
0.000 sec / 0.000 sec


cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
SQL File 5*
member
books
librarians
SELECT * FROM libary_system.books;
INSERT INTO Books (BookID, Title, Author, ISBN, Publisher, PublicationYear, Genre)
VALUES
(1, 'The Lord of the Rings', 'J.R.R. Tolkien', '9780547926538', 'HarperCollins', 1954, 'Fantasy'),
(2, 'To Kill a Mockingbird', 'Harper Lee', '9780061120084', 'HarperCollins', 1960, 'Southern Gothic'),
(3, '1984', 'George Orwell', '9780451524935', 'New American Library', 1949, 'Dystopian'),
(4, 'Pride and Prejudice', 'Jane Austen', '9780141439518', 'Penguin Classics', 1813, 'Romance'),
(5, 'The Alchemist', 'Paulo Coelho', '9780061803483', 'HarperCollins', 1988, 'Philosophical Fiction');
No object selected
1 11:49:28 SELECT * FROM libary_system.member LIMIT 0, 1000
3 row(s) returned
0.000 sec / 0.000 sec


cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
SELECT * FROM libary_system.books;
INSERT INTO Books (BookID, Title, Author, ISBN, Publisher, PublicationYear, Genre)
VALUES
(1, 'The Lord of the Rings', 'J.R.R. Tolkien', '9780547926538', 'HarperCollins', 1954, 'Fantasy'),
(2, 'To Kill a Mockingbird', 'Harper Lee', '9780061120084', 'HarperCollins', 1960, 'Southern Gothic'),
(3, '1984', 'George Orwell', '9780451524935', 'New American Library', 1949, 'Dystopian'),
No object selected
1 11:49:28 SELECT * FROM libary_system.member LIMIT 0, 1000 3 row(s) returned 0.000 sec / 0.000 sec
2 11:49:38 SELECT * FROM libary_system.books LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec
Type here to search
26°C Smoke
11:49 AM 1/2/2025


cloud4.rpsconsulting.in/console/#/client/MzU3NQBjAG15c3Fs
MySQL Workbench
Local instance MySQL80
SELECT * FROM libary_system.librarians;
INSERT INTO Librarians (LibrarianID, FirstName, LastName, EmployeeID, Department)
VALUES
(1, 'John', 'Doe', 1001, 'Circulation'),
(2, 'Jane', 'Smith', 1002, 'Reference'),
(3, 'David', 'Lee', 1003, 'Childrens'),
(4, 'Sarah', 'Jones', 1004, 'Acquisitions');
No object selected
1 11:49:28 SELECT * FROM libary_system.member LIMIT 0, 1000 3 row(s) returned 0.000 sec / 0.000 sec
2 11:49:38 SELECT * FROM libary_system.books LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec
Type here to search
26°C Smoke
11:49 AM 1/2/2025



CREATE TABLE Member: This statement creates a new table named "Member".
MemberID INT AUTO_INCREMENT PRIMARY KEY:
MemberID: This column will store a unique integer value for each member.
INT: Specifies that the MemberID column will store integer values.
AUTO_INCREMENT: This keyword automatically generates a unique integer value for each new row inserted into the table. The first row will have MemberID of 1, the second will have 2, and so on.
PRIMARY KEY: This constraint ensures that the MemberID column uniquely identifies each row in the table and cannot contain null values.
Name VARCHAR(100) NOT NULL:
Name: This column will store the member's name.
VARCHAR(100): Specifies that the Name column can store a variable-length string of up to 100 characters.
NOT NULL: This constraint ensures that the Name column cannot contain null values (i.e., it must always have a value).
Email VARCHAR(150) UNIQUE NOT NULL:
Email: This column will store the member's email address.
VARCHAR(150): Specifies that the Email column can store a variable-length string of up to 150 characters.
UNIQUE: This constraint ensures that each email address in the table is unique.
NOT NULL: This constraint ensures that the Email column cannot contain null values.
PhoneNumber VARCHAR(15) NOT NULL:
PhoneNumber: This column will store the member's phone number.
VARCHAR(15): Specifies that the PhoneNumber column can store a variable-length string of up to 15 characters.
NOT NULL: This constraint ensures that the PhoneNumber column cannot contain null values.
Address TEXT:
Address: This column will store the member's address.
TEXT: Specifies that the Address column can store a large amount of text.
JoinDate DATE NOT NULL:
JoinDate: This column will store the date when the member joined the library.
DATE: Specifies that the JoinDate column will store a date value (e.g., '2024-12-31').
NOT NULL: This constraint ensures that the JoinDate column cannot contain null values.

DESC Member: This statement displays the column definitions of the "Member" table. It provides information about each column, including its name, data type, length, whether it allows null values, and any constraints (such as PRIMARY KEY, UNIQUE, etc.).

CREATE TABLE Books: This statement creates a new table named "Books".
BookID: This column will store a unique integer value for each book.
INT: Specifies that the BookID column will store integer values.
PRIMARY KEY: This constraint ensures that the BookID column uniquely identifies each row in the table and cannot contain null values.
Title VARCHAR(255) NOT NULL:
Title: This column will store the title of the book.
VARCHAR(255): Specifies that the Title column can store a variable-length string of up to 255 characters.
NOT NULL: This constraint ensures that the Title column cannot contain null values.
Author VARCHAR(100):
Author: This column will store the author of the book.
VARCHAR(100): Specifies that the Author column can store a variable-length string of up to 100 characters.
ISBN VARCHAR(13) UNIQUE:
ISBN: This column will store the International Standard Book Number (ISBN) of the book.
VARCHAR(13): Specifies that the ISBN column can store a variable-length string of up to 13 characters.
UNIQUE: This constraint ensures that each ISBN in the table is unique.
Publisher VARCHAR(100):
Publisher: This column will store the publisher of the book.
VARCHAR(100): Specifies that the Publisher column can store a variable-length string of up to 100 characters.
PublicationYear INT:
PublicationYear: This column will store the year the book was published.
INT: Specifies that the PublicationYear column will store integer values.
Genre VARCHAR(50):
Genre: This column will store the genre of the book (e.g., "Fantasy", "Mystery", "Science Fiction").
VARCHAR(50): Specifies that the Genre column can store a variable-length string of up to 50 characters.
Availability BOOLEAN DEFAULT TRUE:
Availability: This column will indicate whether the book is currently available for borrowing.
BOOLEAN: Specifies that the Availability column can store a boolean value (TRUE or FALSE).
DEFAULT TRUE: Sets the default value for the Availability column to TRUE, meaning books are initially considered available.

CREATE TABLE Librarians: This statement creates a new table named "Librarians".
LibrarianID INT PRIMARY KEY:
LibrarianID: This column will store a unique integer value for each librarian.
INT: Specifies that the LibrarianID column will store integer values.
PRIMARY KEY: This constraint ensures that the LibrarianID column uniquely identifies each row in the table and cannot contain null values.
FirstName VARCHAR(50) NOT NULL:
FirstName: This column will store the first name of the librarian.
VARCHAR(50): Specifies that the FirstName column can store a variable-length string of up to 50 characters.
NOT NULL: This constraint ensures that the FirstName column cannot contain null values.
LastName VARCHAR(50) NOT NULL:
LastName: This column will store the last name of the librarian.
VARCHAR(50): Specifies that the LastName column can store a variable-length string of up to 50 characters.
NOT NULL: This constraint ensures that the LastName column cannot contain null values.
EmployeeID INT UNIQUE:
EmployeeID: This column will store the employee ID of the librarian.
INT: Specifies that the EmployeeID column will store integer values.

UNIQUE: This constraint ensures that each employee ID in the table is unique.
Department VARCHAR(50):
Department: This column will store the department in which the librarian works (e.g., "Circulation", "Reference").
VARCHAR(50): Specifies that the Department column can store a variable-length string of up to 50 characters.
These SQL statements define the structure of three tables: "Member", "Books", and "Librarians". Each table has specific columns with appropriate data types and constraints to ensure data integrity and consistency.