

Full Stack Development Program with Generative AI



Designing and Managing Databases with MongoDB



Course-End Project





Designing and Managing Client Databases Using SQL

Objective

To design and implement a relational database system using MySQL by creating structured tables, establishing relationships, designing an Entity-Relationship (ER) diagram, and performing core SQL operations such as insert, update, delete, and querying data using joins.

Description:

This project focuses on creating a database for an architectural firm that needs to manage client details and meetings efficiently. It involves designing an Entity-Relationship (ER) diagram, setting up a database named clientDB in MySQL, and defining tables for storing client and meeting information. These tables will be linked using relational constraints to enforce data integrity. The project helps learners understand database modeling and creation, relational integrity through joins, and core SQL operations for data manipulation and retrieval, equipping them with essential SQL skills for real-world, data-driven applications.



Problem Statement and Motivation



Real-time scenario:

Anna, an architect managing numerous client projects, finds it increasingly difficult to keep track of client information and meeting details manually. Without a centralized system, data becomes fragmented, inconsistent, and difficult to retrieve. This disorganization often leads to missed updates, scheduling conflicts, and reduced productivity.

Solution:

By designing a relational database using MySQL, Anna's firm can digitize and streamline its project management workflow. Starting with the design of an Entity-Relationship (ER) diagram, the system will model the relationships between clients and meetings before implementing them as structured tables. This approach ensures consistent data storage, simplifies recordkeeping, and supports efficient querying and reporting. As a result, the business can manage client information more effectively and scale operations with greater accuracy and ease.

Industry Relevance



The following concepts and tools are vital in today's data-driven industries where structured data management is essential:

1. **MySQL:** MySQL is one of the most popular relational database management systems used across industries. It enables developers to structure, query, and manage data efficiently while ensuring data integrity and security.
2. **Relational Database Design and ER Modeling:** Modeling data through Entity-Relationship (ER) diagrams and structured tables reflects real-world business logic, a crucial skill across industries like finance, healthcare, education, and software development, where relational data must be maintained and queried reliably.
3. **Data Operations (CRUD):** The ability to perform Create, Read, Update, and Delete operations is central to every application, from small-scale CRMs to large enterprise ERP systems.
4. **SQL Joins and Queries:** Writing effective SQL queries to extract meaningful relationships between data sets is a skill in high demand across business analytics, reporting, and application development roles.

Tasks

The following steps outline the process of designing and managing a client database system using MySQL:

1. Create a new database in MySQL to manage client and meeting data
2. Define a table for client details, including fields like ID, name, email, and password
3. Create a table for client meetings, with fields such as meeting topic, number of people, and date
4. Design an Entity-Relationship (ER) Diagram to model the client and meeting entities, their attributes, and the relationships between them
5. Perform SQL operations, including data insertion, updates, deletions, and querying related records across tables
6. Establish a relationship between the two tables using SQL JOINS



Project References

Task 1: Lesson 2

Task 2: Lesson 2

Task 3: Lesson 2

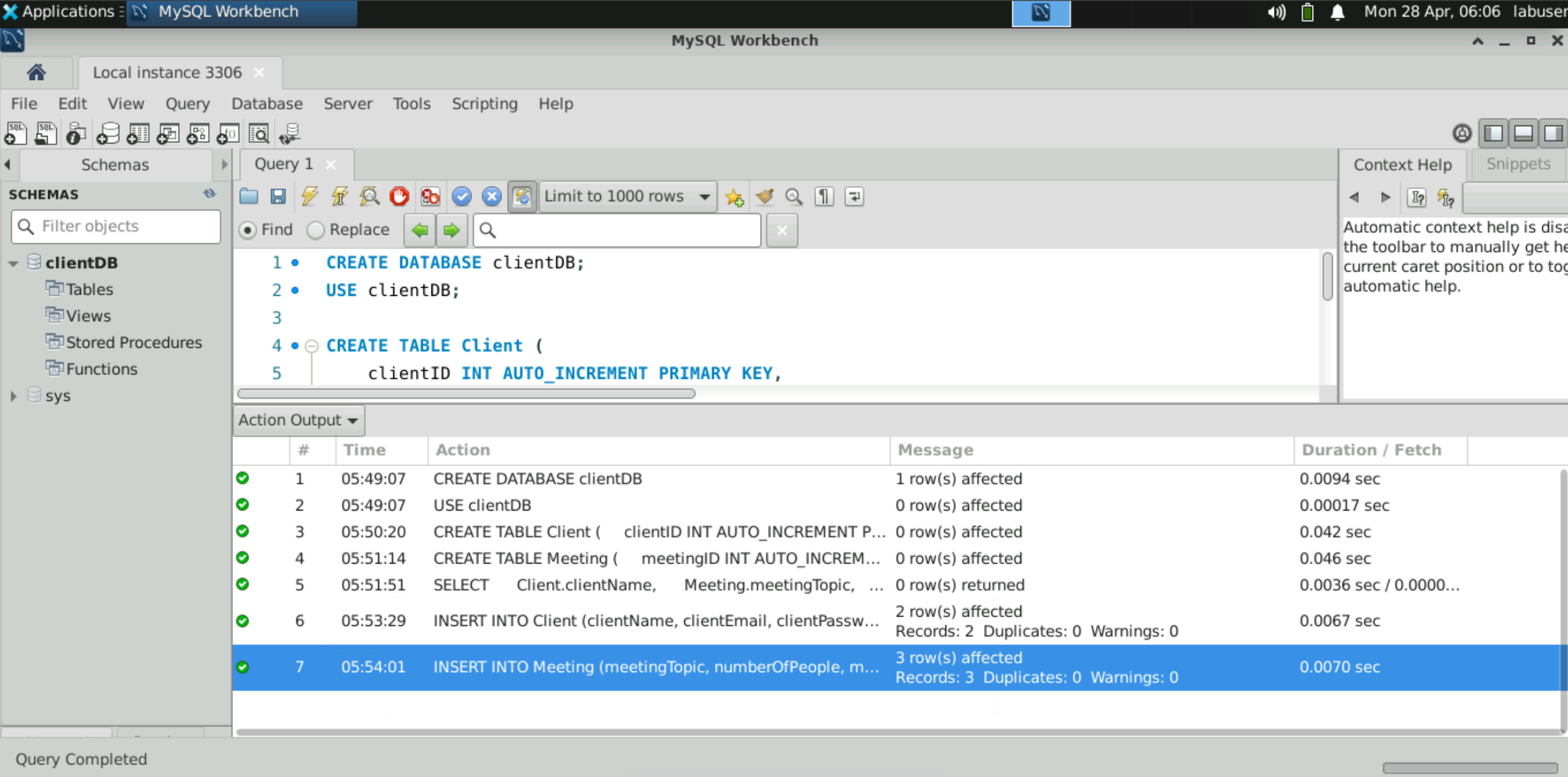
Task 4: Lesson 2

Task 5 and Task 6: Lesson 3



Output Screenshots

Database and tables created with records inserted in MySQL Workbench:



The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' panel with 'clientDB' expanded, showing 'Tables', 'Views', 'Stored Procedures', and 'Functions'. The main query editor shows the following SQL script:

```
1 • CREATE DATABASE clientDB;
2 • USE clientDB;
3
4 • CREATE TABLE Client (
5   clientID INT AUTO_INCREMENT PRIMARY KEY,
```

The 'Action Output' panel at the bottom shows the results of the executed queries:

#	Time	Action	Message	Duration / Fetch
1	05:49:07	CREATE DATABASE clientDB	1 row(s) affected	0.0094 sec
2	05:49:07	USE clientDB	0 row(s) affected	0.00017 sec
3	05:50:20	CREATE TABLE Client (clientID INT AUTO_INCREMENT P...	0 row(s) affected	0.042 sec
4	05:51:14	CREATE TABLE Meeting (meetingID INT AUTO_INCREM...	0 row(s) affected	0.046 sec
5	05:51:51	SELECT Client.clientName, Meeting.meetingTopic, ...	0 row(s) returned	0.0036 sec / 0.0000...
6	05:53:29	INSERT INTO Client (clientName, clientEmail, clientPassw...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.0067 sec
7	05:54:01	INSERT INTO Meeting (meetingTopic, numberOfPeople, m...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0070 sec

The status bar at the bottom indicates 'Query Completed'.

Output Screenshots

Joined client and meeting data retrieved using SQL JOIN in MySQL Workbench:

The screenshot displays the MySQL Workbench interface. The 'Schemas' panel on the left shows the 'clientDB' database. The 'Query 1' tab is active, showing a SQL query that has been executed. The 'Result Grid' displays the output of the query, which is a JOIN of client and meeting data. The 'Action Output' panel at the bottom shows the execution log, including the creation of the database, tables, and the execution of the SELECT query.

Result Grid

#	clientName	meetingTopic	meetingDate
1	Anna Smith	Project Kickoff	2025-05-05
2	Anna Smith	Design Review	2025-05-10
3	John Doe	Final Presentation	2025-05-15

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	05:49:07	CREATE DATABASE clientDB	1 row(s) affected	0.0094 sec
✓ 2	05:49:07	USE clientDB	0 row(s) affected	0.00017 sec
✓ 3	05:50:20	CREATE TABLE Client (clientID INT AUTO_INCREMENT P...	0 row(s) affected	0.042 sec
✓ 4	05:51:14	CREATE TABLE Meeting (meetingID INT AUTO_INCREM...	0 row(s) affected	0.046 sec
✓ 5	05:51:51	SELECT Client.clientName, Meeting.meetingTopic, ...	0 row(s) returned	0.0036 sec / 0.0000...
✓ 6	05:53:29	INSERT INTO Client (clientName, clientEmail, clientPassw...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.0067 sec
✓ 7	05:54:01	INSERT INTO Meeting (meetingTopic, numberOfPeople, m...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0070 sec
✓ 8	06:09:11	SELECT Client.clientName, Meeting.meetingTopic, ...	3 row(s) returned	0.00039 sec / 0.000...

Query Completed



Thank you