SQL Internship Task Solutions

By: Bhavesh Rathod

✅ Task 1: Database Schema Design

Objective:
Design a normalized relational database schema for a Library Management System.

Deliverables:

- library_schema.sql
- er_diagram.png

Summary:

- 7 tables: Books, Authors, Categories, Publishers, Members, Staff, Loans
- Used InnoDB, PRIMARY KEY, FOREIGN KEY, CHECK, DEFAULT, and AUTO_INCREMENT
- Generated ER diagram using MySQL Workbench

🧠 Task 1: Q&A

1. **What is a Database Schema?**
   A structure that defines tables, relationships, and constraints in a database.
2. **Why use Foreign Keys?**
   To maintain referential integrity between related tables.
3. **What is the use of AUTO_INCREMENT?**
   Automatically increases numeric values for primary key fields.
4. **What is the purpose of ER Diagrams?**
   To visually represent the structure and relationships of a database.
5. **What engine supports foreign keys in MySQL?**
   InnoDB storage engine supports foreign key constraints.

Status: ✅ Completed

✅ Task 2: Data Insertion & Manipulation

Objective:
Practice INSERT, UPDATE, DELETE, NULL handling, and maintaining FK relationships

Deliverables:

- task2_data_manipulation.sql

Summary:

- Inserted multiple rows into parent tables (Authors, Categories, Publishers)
- Ensured correct FK values before inserting into Books
- Demonstrated safe UPDATE and DELETE using PRIMARY KEY
- Handled errors: Error 1175 (Safe Mode), Error 1452 (FK violation)

🧠 **Task 2: Q&A**

1. **Why insert into parent tables first?**
   To avoid foreign key constraint errors when inserting into child tables.
2. **What is Error 1175?**
   Safe update mode blocks updates/deletes without a key column in WHERE clause.
3. **What is Error 1452?**
   Foreign key constraint failure due to missing reference value in parent table.
4. **Difference between DELETE and TRUNCATE?**
   DELETE removes specific rows with WHERE; TRUNCATE removes all rows quickly.
5. **Why use PRIMARY KEY in UPDATE/DELETE?**
   To safely affect only one specific row and avoid unintended changes.

Status: ✅ Completed

✅ **Task 3: SELECT Queries & Filtering**

Objective:
Retrieve data using various SELECT queries with conditions, sorting, and limits

Deliverables:

- task3_select_queries.sql

Summary:

- Used SELECT *, specific columns, WHERE, AND, OR, LIKE, BETWEEN
- Applied ORDER BY, LIMIT, DISTINCT, aliasing, and simple JOINs
- Fixed mismatched comments (e.g., corrected FullName/Email logic)

🧠 **Task 3: Q&A**

1. **What does SELECT * do?**
   Retrieves all columns from the specified table.
2. **How does WHERE clause work?**
   Filters records that match a specific condition.
3. **What does LIKE '%abc%' do?**
   Finds rows where the column contains the substring 'abc'.
4. **Why use ORDER BY?**
   To sort results in ascending (ASC) or descending (DESC) order.
5. **What is the use of LIMIT?**
   Restricts the number of rows returned by a query.

Status: ✅ Completed

✅ Task 4: Aggregate Functions + Interview Q&A

Objective:
Use aggregate functions like SUM, COUNT, AVG, and GROUP BY

Deliverables:

- task4_aggregates.sql
- task4_interview_QA.txt

Summary:

- Wrote queries using: COUNT(), SUM(), AVG(), GROUP BY, HAVING, ROUND()
- Grouped data by Author, Category, Publisher
- Calculated totals, averages, and distinct counts

🧠 Task 4: Interview Q&A

1. **What is SQL?**
   Structured Query Language used to manage and interact with databases.
2. **What is a Primary Key?**
   A unique identifier for table rows. Cannot be NULL or duplicate.
3. **What is a Foreign Key?**
   A column that refers to a Primary Key in another table to maintain relationships.

4. **What are Constraints?**
   Rules like NOT NULL, UNIQUE, CHECK, DEFAULT, and key constraints to ensure data integrity.
5. **Difference between WHERE and HAVING?**
   WHERE filters rows before grouping. HAVING filters groups after GROUP BY.
6. **Difference between DELETE, TRUNCATE, DROP?**
   - DELETE: Removes specific rows
   - TRUNCATE: Deletes all rows (faster, resets identity)
   - DROP: Deletes entire table
7. **What are aggregate functions?**
   Functions like SUM(), AVG(), COUNT() used to perform calculations on a set of rows.
8. **Difference between COUNT(*) and COUNT(column)?**
   - COUNT(*): Counts all rows
   - COUNT(column): Ignores NULLs in that column
9. **Use of GROUP BY?**
   Groups rows that share a property, enabling aggregation per group
10. **Use of ORDER BY?**
    Sorts the query result in ascending or descending order

Status: ✅ Completed

GitHub Repo Link: *[Add your link here]*

Final Submission Status: ✅ All 4 tasks completed and documented