📓 Project Title: Quotation and Invoice Generator

🧑‍💻 Developed By: Kavya Pratap Singh Chauhan

📆 Submission Date: 19-06-2025

🎗️ For: DevifyX SQL Internship Assignment

1. Objective

The objective of this project is to design and implement a SQL-only backend system to generate and manage client quotations and invoices using **MySQL**. This includes client management, product catalog, quotation creation, invoice generation, tax and discount handling, status tracking, and audit logging — all without any frontend/backend code.

🔷 2. Tools & Technology

- MySQL 8.0
- MySQL Workbench

🔷 3. Database Tables

➤ clients

Stores client details like name, contact info, and address.

Column	Type	Description
id	INT (PK)	Unique client ID
name	VARCHAR	Full name
email	VARCHAR	Email address
phone	VARCHAR	Phone number
address	TEXT	Address info

➤ products

Stores product/service information.

Column	Type	Description
id	INT (PK)	Unique product ID
name	VARCHAR	Product name
description	TEXT	Description
price	DECIMAL	Unit price

➤ **quotations**

Stores quotations given to clients.

Column	Type	Description
id	INT (PK)	Unique quotation ID
client_id	INT (FK)	Linked client
date	DATE	Quotation date
status	VARCHAR	Draft, Sent, Approved
discount_type	VARCHAR	PERCENT or FIXED
discount_value	DECIMAL	Discount value

➤ **quotation_items**

Stores line items in each quotation.

Column	Type	Description
id	INT (PK)	Unique item ID
quotation_id	INT (FK)	Linked quotation
product_id	INT (FK)	Product or service
quantity	INT	Quantity quoted
tax_rate	DECIMAL	GST/VAT rate
price	DECIMAL	Price per unit

➤ **invoices**

Stores invoices generated from quotations.

Column	Type	Description
id	INT (PK)	Unique invoice ID
quotation_id	INT (FK)	Source quotation
invoice_number	VARCHAR	Unique invoice code
date	DATE	Invoice date
status	VARCHAR	Unpaid, Paid, Overdue

➤ **audit_logs**

Tracks all important actions (e.g., create, approve, pay).

Column	Type	Description
id	INT (PK)	Log entry ID
action_type	VARCHAR	Type of action
entity	VARCHAR	Affected table (quotation, invoice)
entity_id	INT	Related record ID
timestamp	DATETIME	Time of action
notes	TEXT	Extra message

◈ 4. Triggers Used

✅ **after_invoice_insert**

Logs when a new invoice is created.

✅ **after_quotation_insert**

Logs when a new quotation is created.

✅ **after_quotation_approval**

Logs when a quotation is approved.

✅ **after_invoice_paid**

Logs when invoice status changes to Paid.

◈ 5. Sample Data

- 3 Clients (e.g., Aditi, Rohan, Sneha)
- 3 Products (e.g., Hosting, SEO, Website)
- 1 Quotation with 2 items
- 1 Invoice linked to that quotation

🔷 6. Sample Output Screenshots (Optional)

Insert screenshots of:

- Table data from SELECT * FROM ...
- ER Diagram (drawn by you or from dbdiagram.io)

🔷 7. Instructions to Test the System

1. Open MySQL Workbench
2. Run quotation_invoice_system.sql
3. Use the following to check your data:

sql

CopyEdit

USE quotation_db;

SELECT * FROM clients;

SELECT * FROM quotations;

SELECT * FROM quotation_items;

SELECT * FROM invoices;

SELECT * FROM audit_logs;

4. Update status of quotations or invoices to test triggers

🔷 8. Conclusion

This project demonstrates a complete invoice management system built entirely with MySQL. It showcases skills in database design, data relationships, SQL queries, and automation through triggers. The system is designed to be scalable, traceable, and easy to use for real-world applications.

🔹 9. *Final Note*

This project was completed as part of the DevifyX SQL Internship Assignment. I learned how to build a fully functional SQL-only backend system and appreciated the opportunity to apply best practices in database design and automation.

End of Report
