SportSphere:

An Online Sports Equipment Store

Tung Duc Vu (2023558), Gaurav Shankar (2023219), Aditya Chaudhary (2023042), Turbold Amarbat (2023559)

Deadline 1: Project Scope and Requirements

SportSphere is an online sports equipment store with dedicated portals for customers, admins, and delivery agents. It offers secure login, real-time tracking, inventory management, and performance analytics. Built using MySQL and Python tkinter for the UI, the system ensures scalability, data integrity, and a seamless shopping experience with efficient delivery and feedback systems

01

Project Scope

SportSphere is an online sports equipment store designed to streamline shopping for customers and simplify operations for admins and delivery agents. It supports secure logins, product browsing, order management, real-time delivery tracking, and customer feedback. Admins manage inventory and analyze performance, while agents handle timely deliveries. Built on a structured ER-based DBMS, the system emphasizes scalability, data integrity, and accessibility to ensure a reliable and efficient platform for all stakeholders.

02

Technical requirements

SportsSphere is built using a modern tech stack to ensure high performance, scalability, and security:

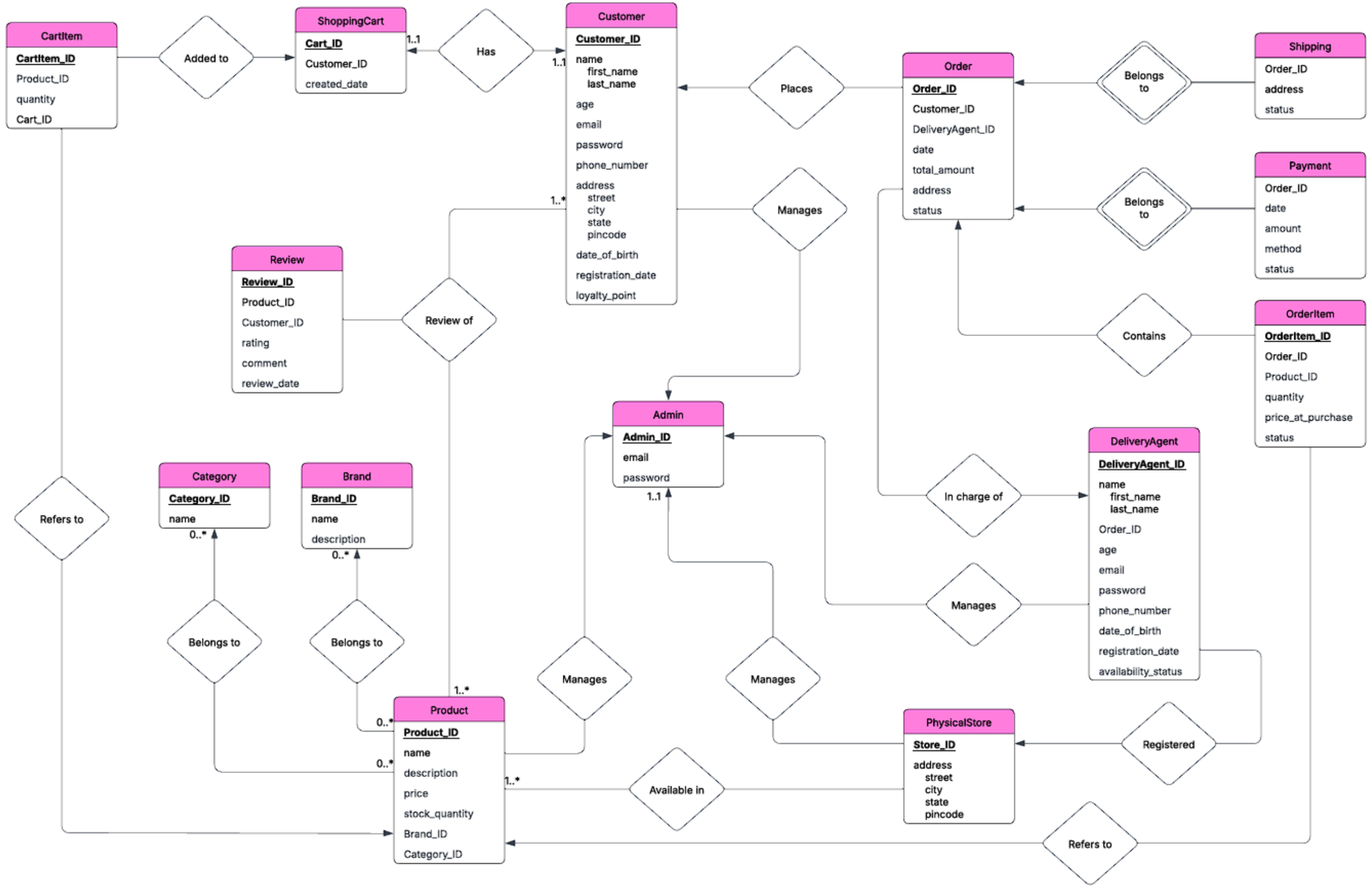
- Database Management: MySQL
- Backend Development: Python
- Frontend Development: HTML, CSS, JavaScript, ReactJS

03

System Functionality

SportSphere supports secure signup and login for customers, admins, and delivery agents. Customers can browse products, use a shopping cart, track orders in real-time, earn loyalty points, and submit reviews. Admins manage inventory, store operations, and delivery agents, while analyzing orders, sales, and customer feedback. Delivery partners receive order assignments based on availability, provide live status updates, and track tasks via a dashboard. The system ensures smooth coordination across all roles for efficient store management and a seamless user experience.

Deadline 1: ER Diagram



The ER model outlines key entities in SportSphere: Customers, Admins, Delivery Agents, and Physical Stores. Customers can browse products (linked to categories and brands), manage carts, place orders, and leave reviews. Orders contain items, are linked to payments and shipping, and are managed by delivery agents. Admins oversee inventory, store operations, and agent performance. Delivery agents update real-time order status. The design ensures efficient data flow, scalability, and seamless coordination among system components to support all core functionalities of the platform.

Deadline 2: Relational Model

01

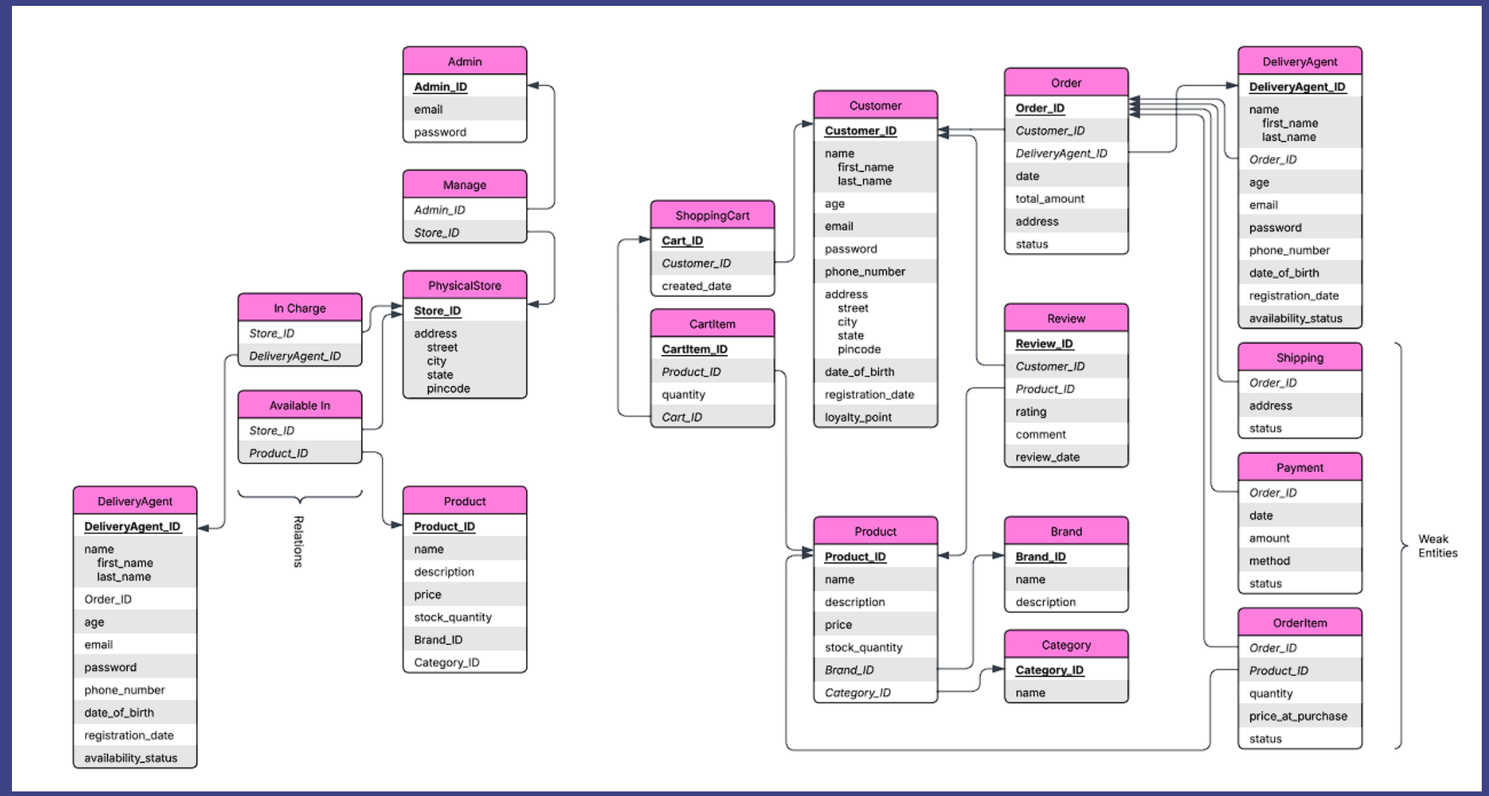
Customer-Centric Experience:

Customers can securely register, shop using a cart system, place and track orders, and give product reviews. All relationships are well-connected, ensuring a seamless shopping experience and reliable order lifecycle management from cart to delivery.

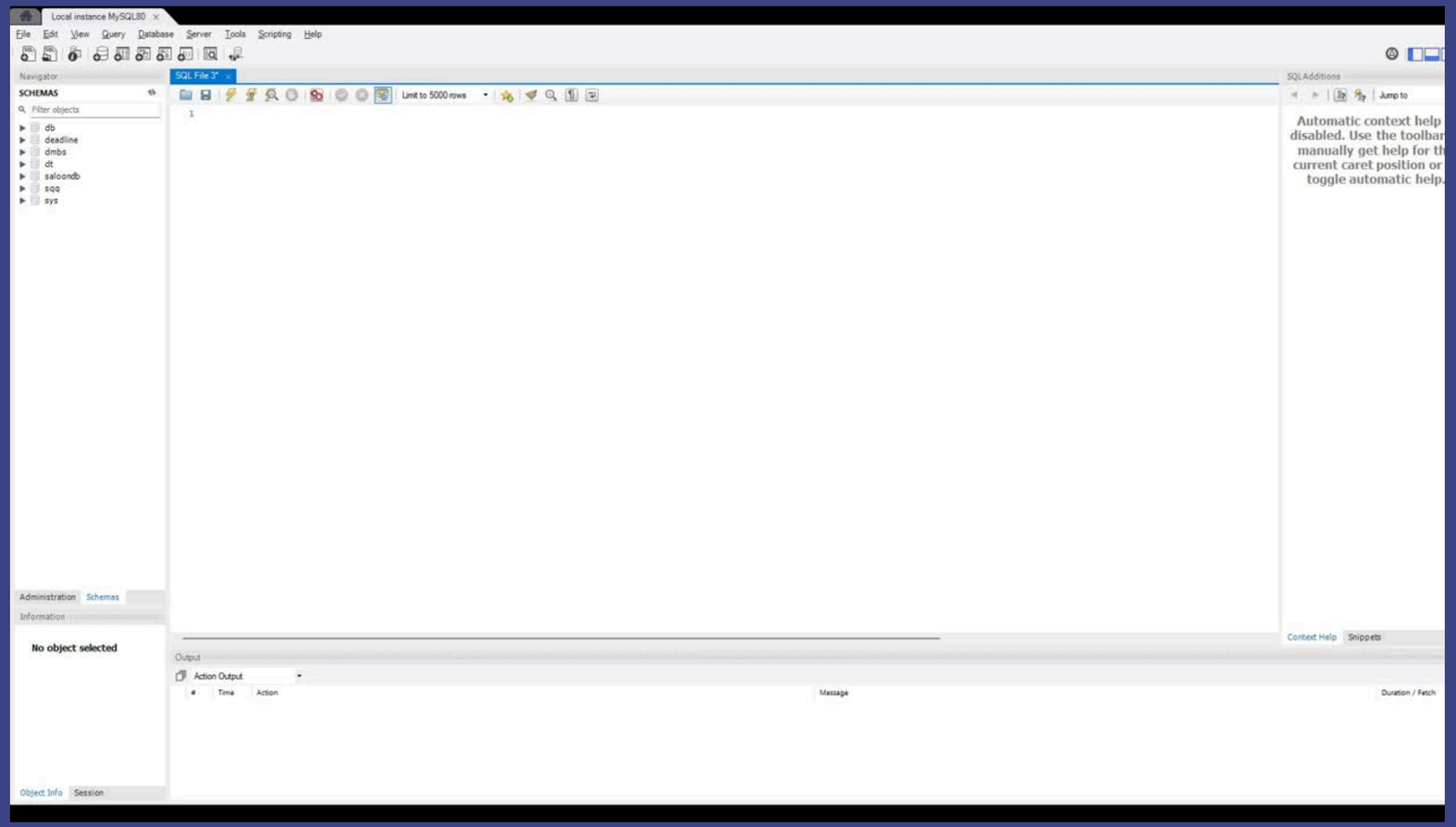
02

Enhanced Operational Flow:

Admins manage physical stores, while Delivery Agents are assigned to stores via the In Charge relationship. The Available In relation between stores and products allows real-time inventory tracking across locations, supporting smooth order fulfillment.

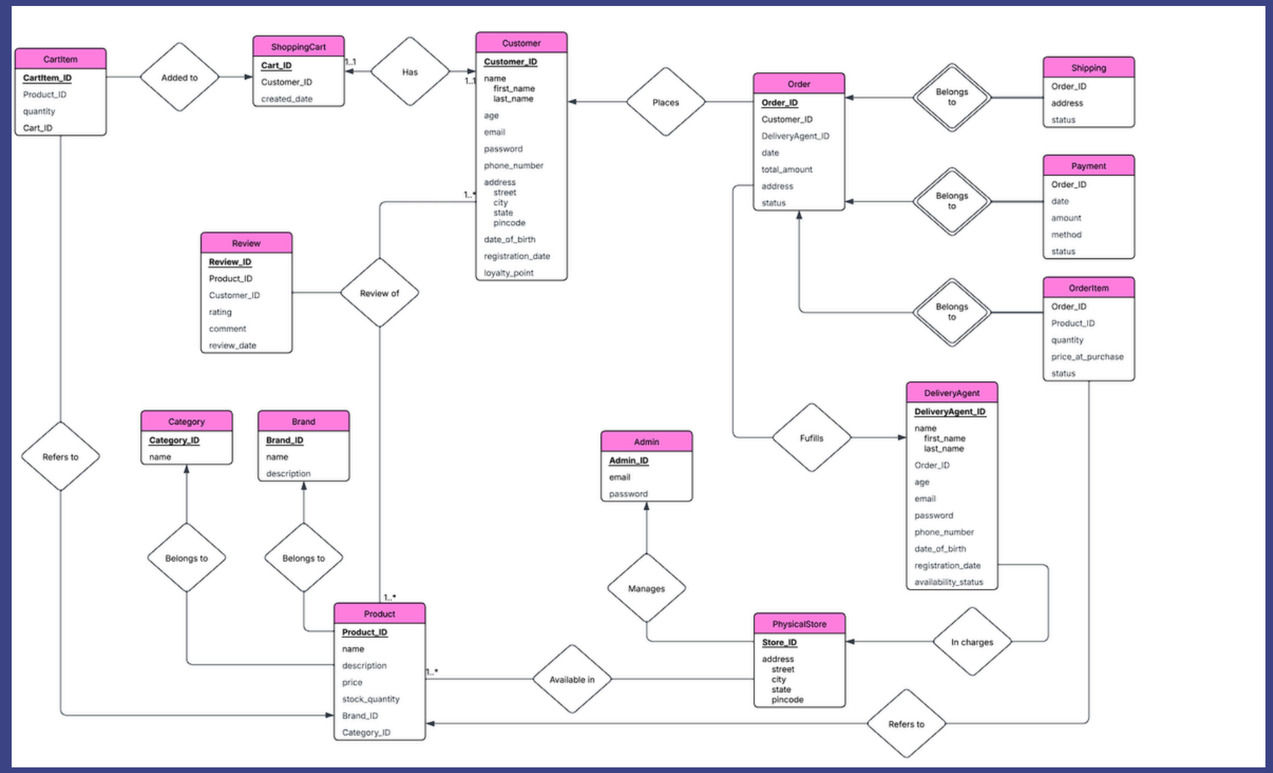


Deadline2: SQL



We designed and created the sports_store_db database to manage an online and offline sports equipment store. It includes core entities like Customer, Admin, Product, Brand, Category, DeliveryAgent, and PhysicalStore, with relationships modeled using foreign keys and associative tables such as OrderItem, Available_in, Manages, and InCharges. Constraints like CHECK, UNIQUE, and NOT NULL ensure data integrity. We populated the tables by executing SQL scripts in MySQL Workbench, using INSERT INTO statements with sample data for customers, products, and store info to test all functionalities.

Refinement: ER Model



01

Customer Flow: Customers manage carts, place orders, track deliveries, write reviews, and earn loyalty points. Each order links to payment, shipping, and delivery status.

02

Admin Oversight: Admins manage inventory, stores, and monitor product availability, ensuring operational efficiency.

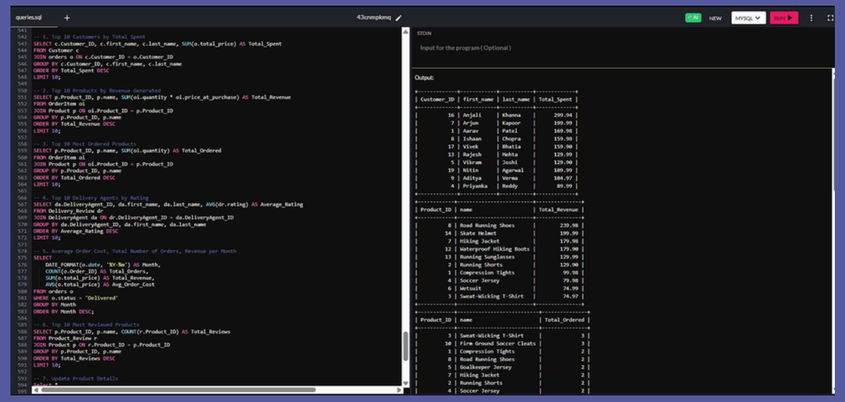
03

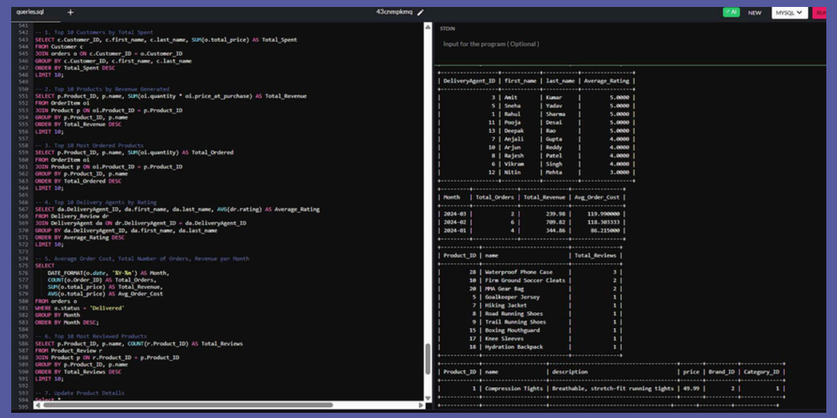
Delivery Fulfillment: Delivery agents are assigned to orders and update fulfillment status. Products belong to categories and brands, and are stocked in physical stores.

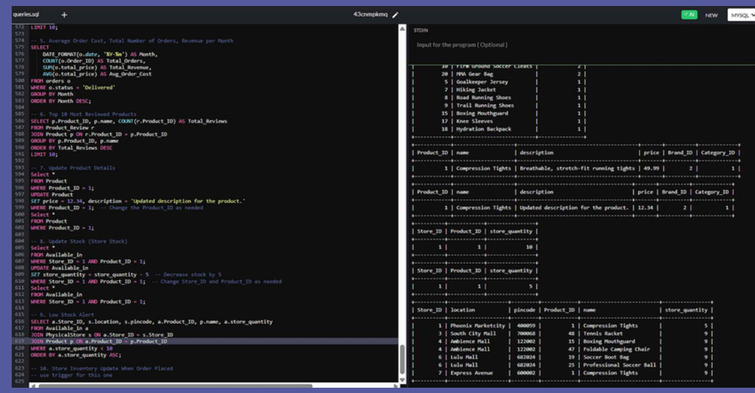
Main Changes from the First ER Model:

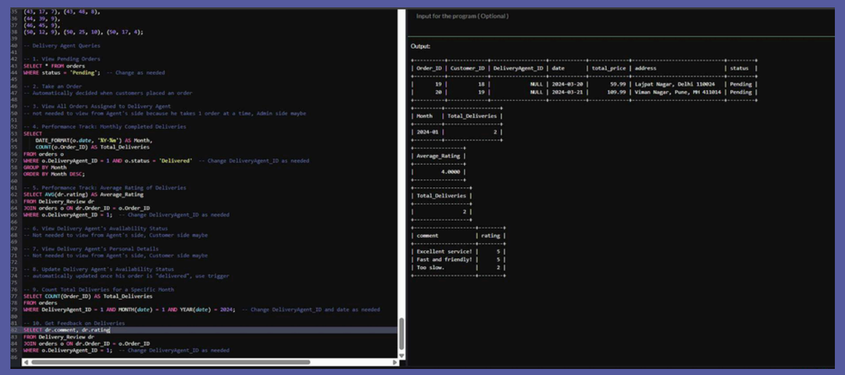
The updated ER diagram introduces a more streamlined structure with clearer relationships and fewer redundancies. The “Manages” relationships are now simplified — Admins manage stores directly, and Delivery Agents fulfill orders instead of being in charge of multiple entities. The “Order” entity is now more centralized, linking directly to OrderItem, Payment, and Shipping, ensuring modular tracking of each order component. Additionally, “PhysicalStore” is better integrated to represent stock availability, and the “Product” entity is now more focused with direct relations to Brand and Category. These refinements improve clarity, data integrity, and system efficiency.

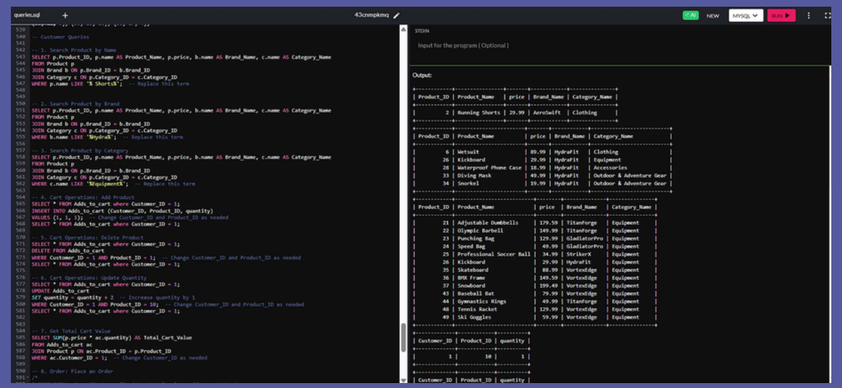
Deadline 3: Functional Queries

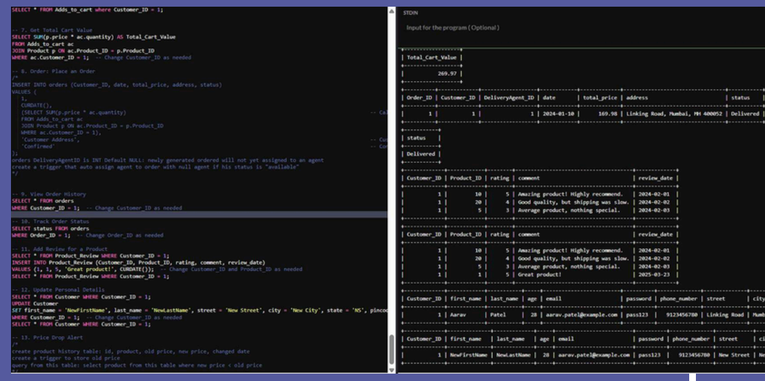






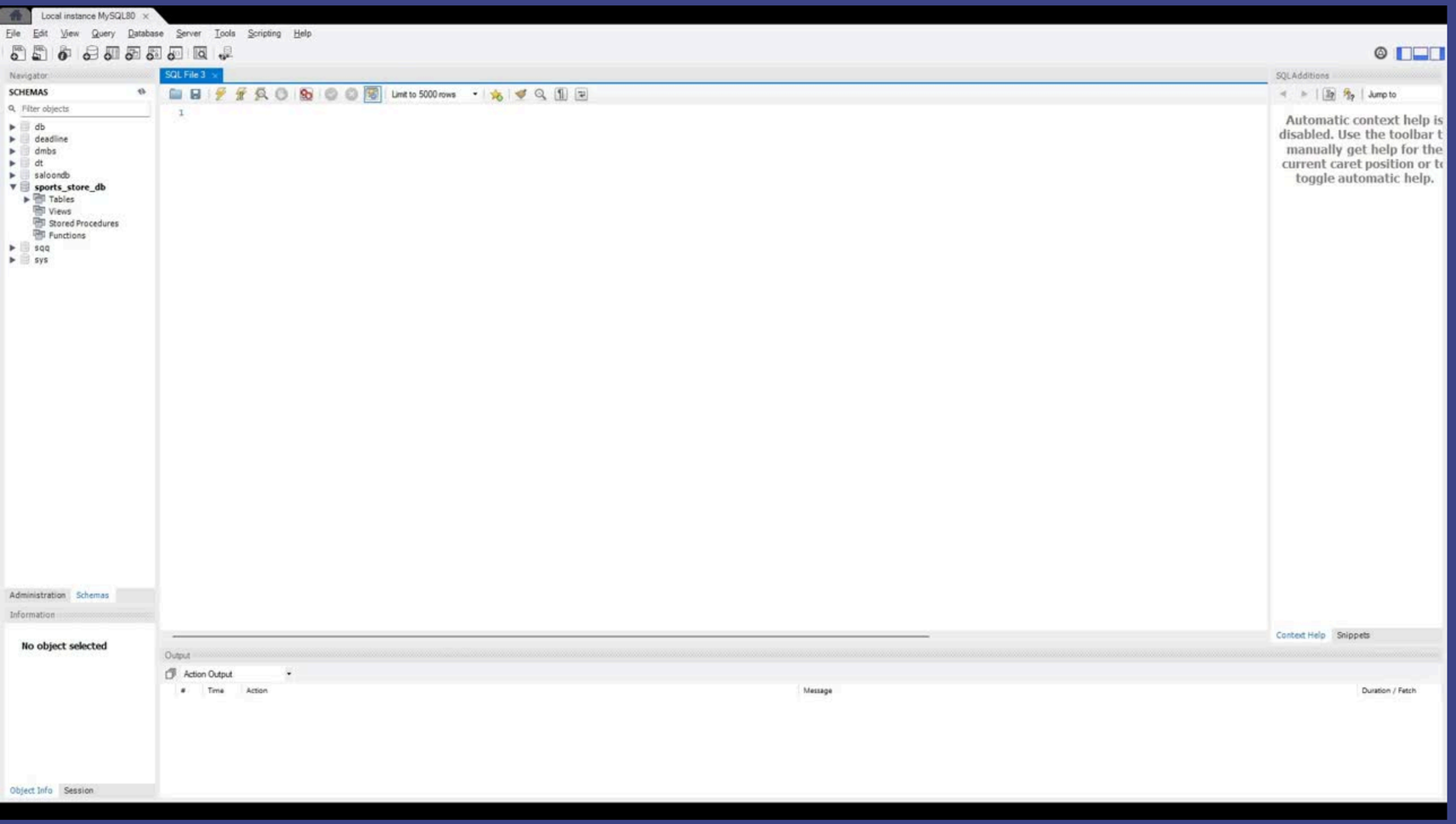






This SQL script supports various customer operations in the online sports store database. First, it retrieves all items in a customer's cart using SELECT * FROM Adds_to_cart WHERE Customer_ID = 1;. Next, it calculates the total cart value by joining the cart and product tables, multiplying price and quantity. When placing an order, it inserts a new record into the orders table with the current date, total amount, and address, setting the status as ‘Confirmed’. Order history is fetched using a simple SELECT on the orders table filtered by Customer_ID. Order delivery status can be tracked using SELECT status FROM orders WHERE Order_ID = 1;. Customers can add product reviews with ratings and comments using an INSERT into the Product_Review table. Personal details like name, address, and city can be updated using the UPDATE command. Finally, a concept is suggested for price drop alerts by maintaining a product history table and comparing new prices with old ones to notify customers when prices decrease.

Deadline3: SQL

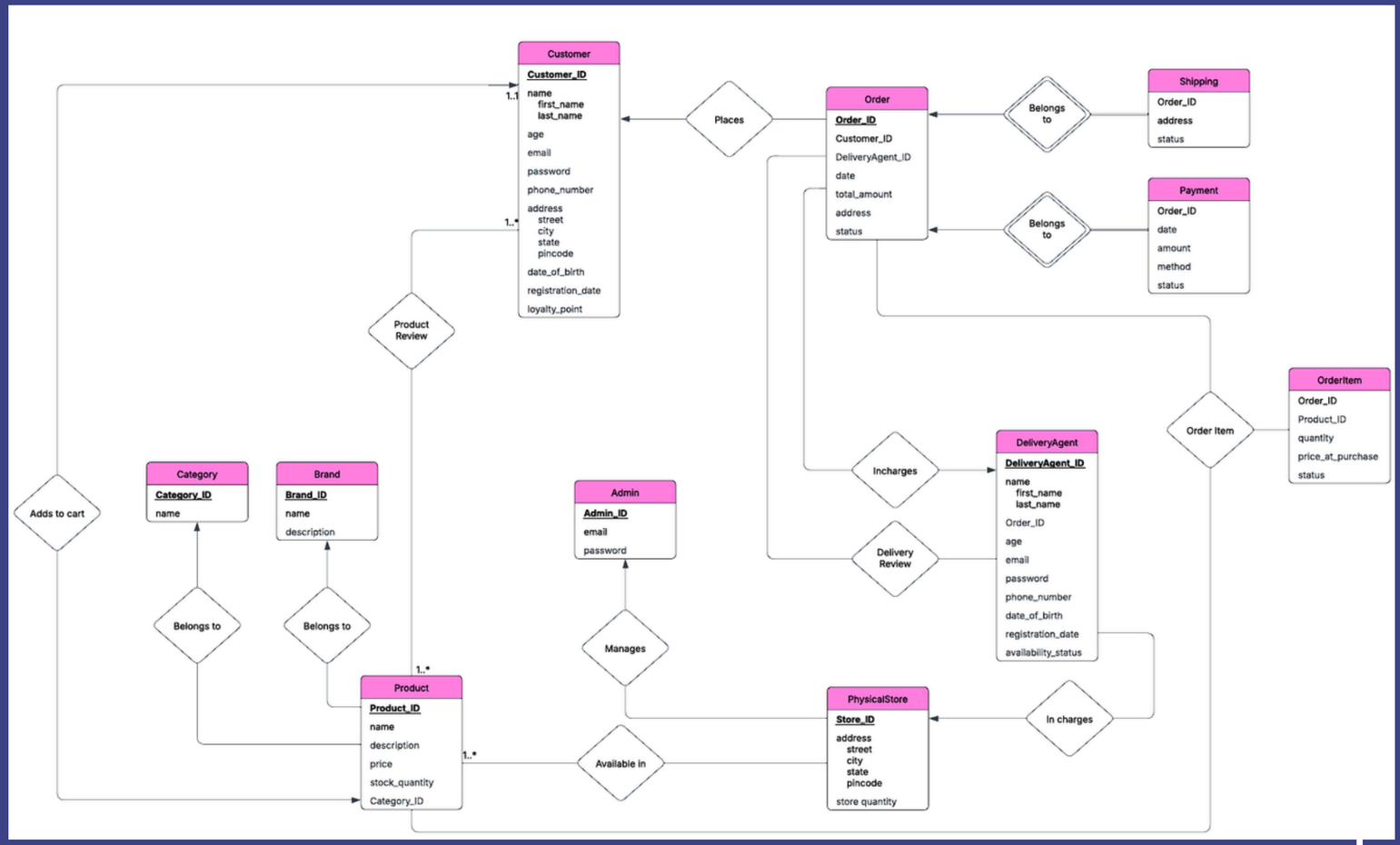


- **Admin Operations:**
  - Top Insights: Identify top 10 customers, products (by revenue and order count), and delivery agents (by rating).
  - **Analytics**: Track average order cost and monthly revenue; monitor stock with low inventory alerts.
  - **Inventory & Product Management:** Update product details and stock levels and maintain product price history.
- **Customer Operations:**
  - **Search & Browse**: Filter products by name, brand, or category.
  - **Cart Management**: Add, remove, update quantity, and calculate total cart value.
  - **Ordering**: Place orders directly from cart; view history and track order status.
  - **Profile & Feedback**: Update personal details, leave product reviews
- **Delivery Agent Operations:**
  - **Order Operations:** Take order, view pending order, delivering history
  - **Performance Tracking**: Monthly delivery count, average ratings, and customer feedback.
  - **Profile:** View and update personal informations

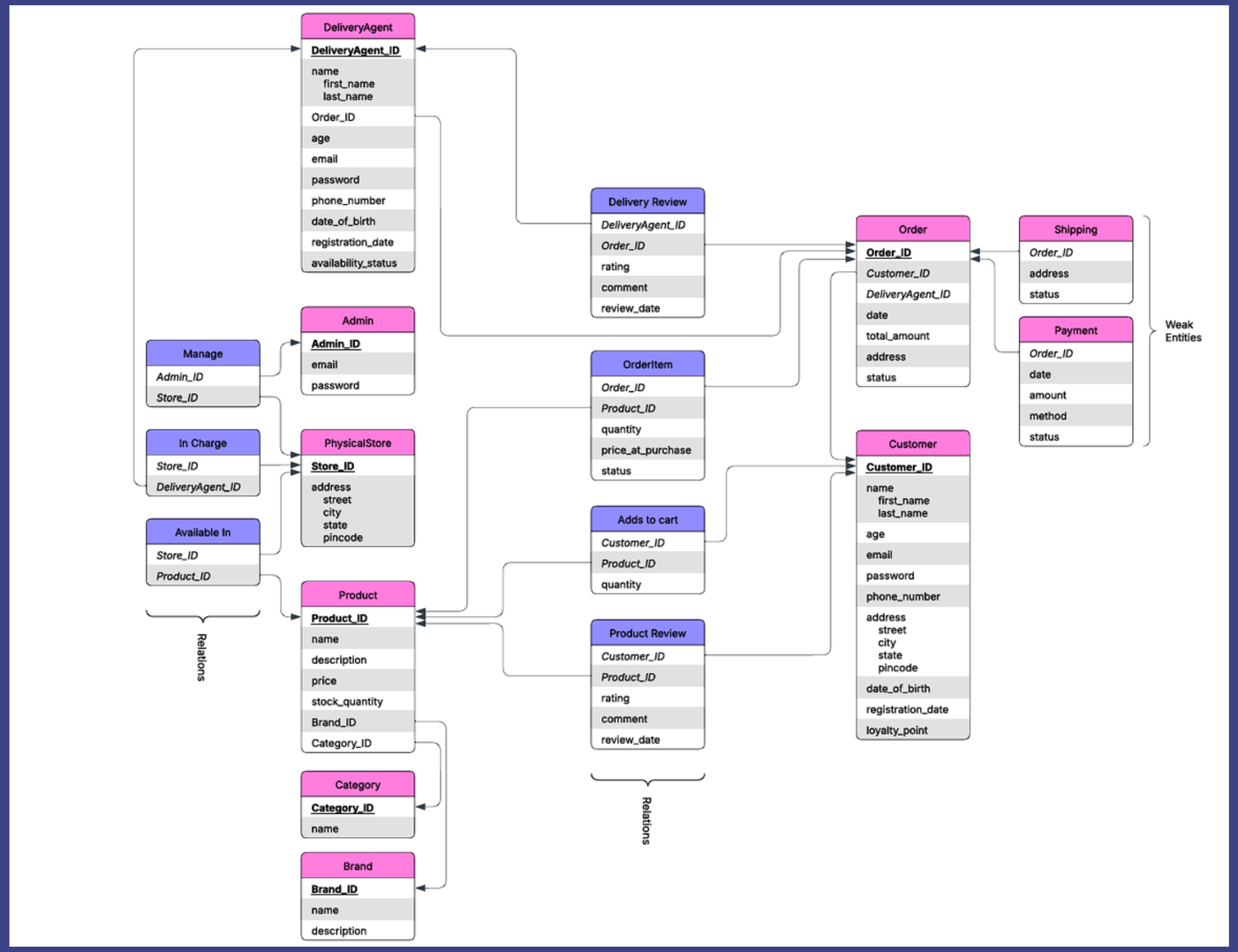
Refinement: ER Diagram

Improved ER Diagram: Smarter & More Scalable Design

This updated ER model is a significant improvement over our earlier version, designed for better normalization, flexibility, and role-based operations.
Customers can now easily browse products (by brand or category), manage carts, place orders, and leave reviews.
The Order entity is enhanced with connections to Payment, Shipping, and OrderItem, making it easier to track the complete order lifecycle.
We've introduced DeliveryAgent integration, where agents are auto-assigned and reviewed post-delivery—supporting automation through triggers.
A major enhancement is the store-wise inventory system via the Available_in relation, enabling Admins to manage stock per store instead of globally.
The model clearly separates Brand and Category for products, improving search functionality and analytics.



Refinement: Relational Model

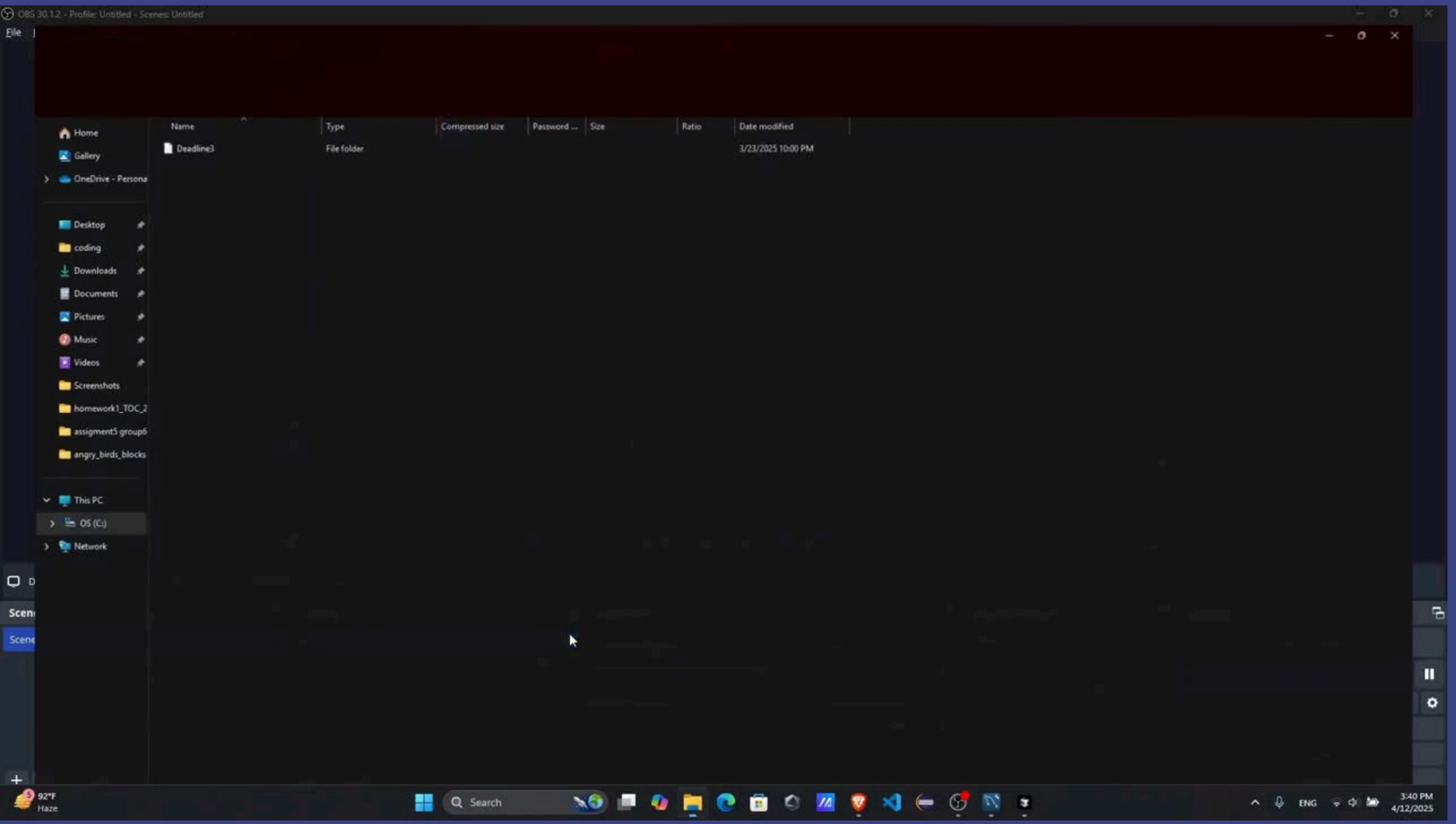


Refined Relational Schema: Clean Structure & Improved Relationships

This updated relational schema builds upon our earlier ER diagram, translating it into a well-structured, normalized relational model. Every entity is now mapped to a relation with clear foreign key connections, ensuring data integrity and referential consistency.

- Customer, Order, and Product remain at the core, now linked via junction tables like OrderItem, AddsToCart, and ProductReview, allowing many-to-many relationships to be handled efficiently.
- Weak entities like Shipping and Payment depend on Order_ID as foreign keys, capturing one-to-one optional relationships in a compact format.
- Roles are separated more clearly: Admins manage stores, and DeliveryAgents are linked to orders and reviewed post-delivery.
- The new design introduces better role-based access and a normalized AvailableIn table to track product availability per physical store, improving inventory handling.

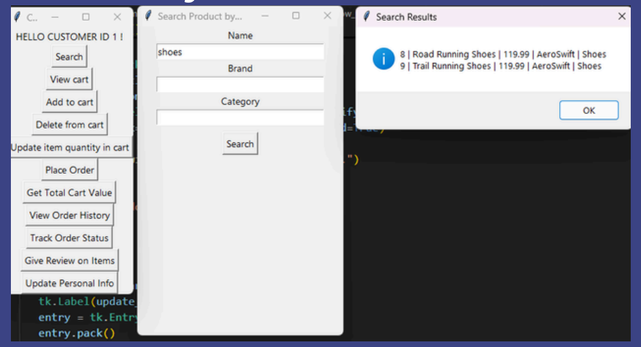
Refinement: Database SQL



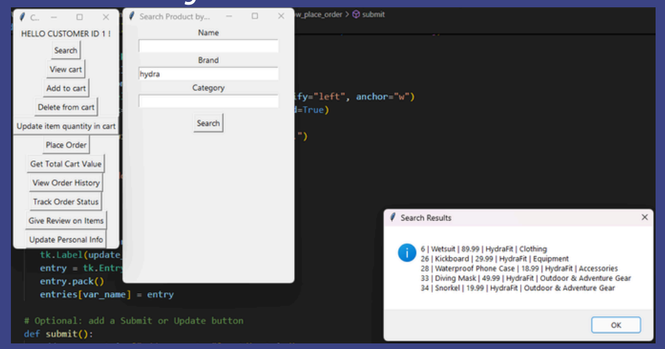
Due to the changes required for Deadline 3, we have updated our ER and relational models, along with the corresponding SQL code. These modifications were necessary to enhance data integrity, normalization, and overall system efficiency. To ensure clarity and transparency in our progress, we have decided to present the revised models and updated implementation in this demo video and through additional slides. This will provide a clearer understanding of the improvements made and demonstrate how the current design better aligns with the project's evolving requirements and real-world functionality.

Deadline 4: Customer UI

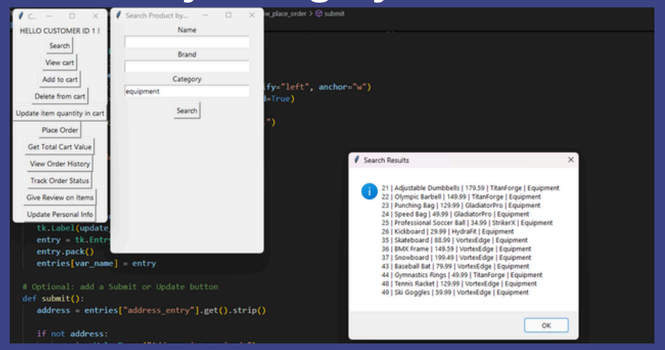
Search by Name



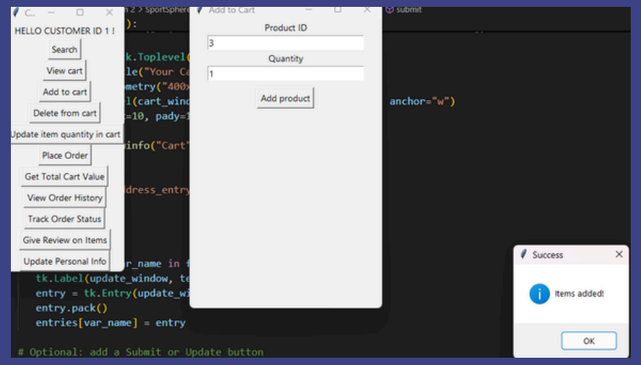
Search by Brand



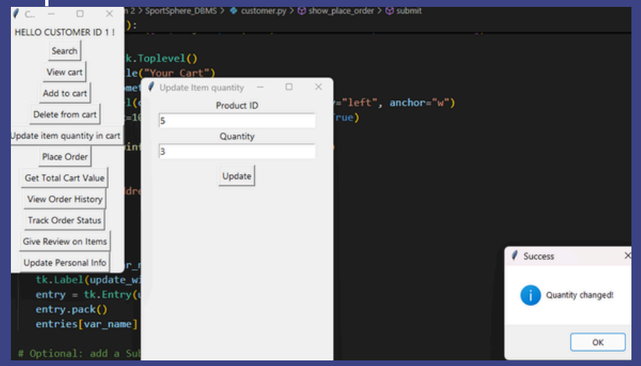
Search by Category



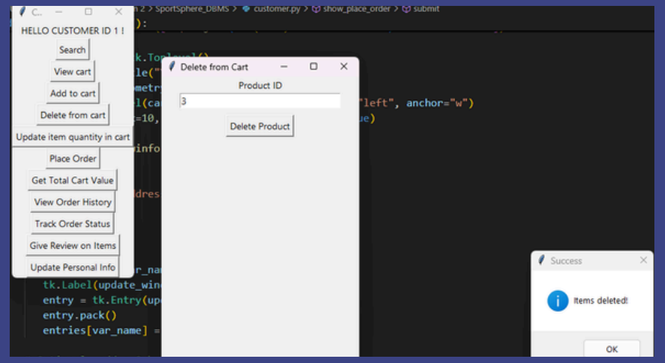
Add Item to Cart



Update Item in Cart

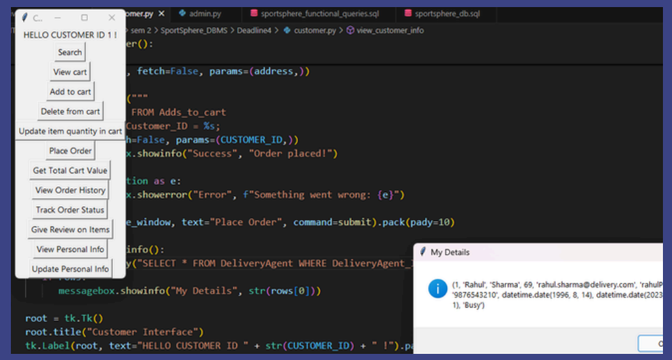


Delete Product from Cart



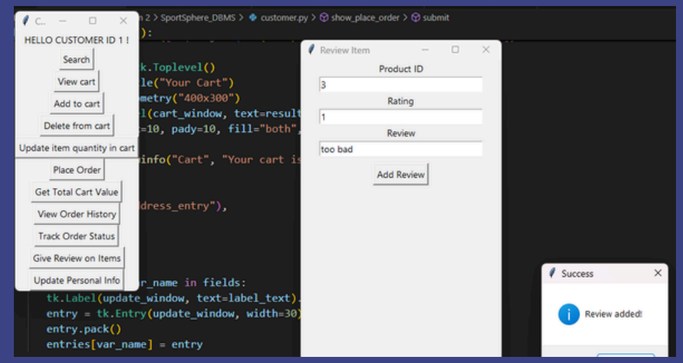
Deadline 4: Customer UI

View Personal Info

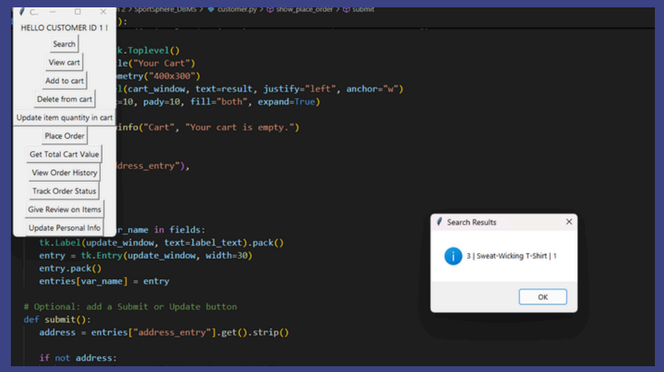


Get Cart Value

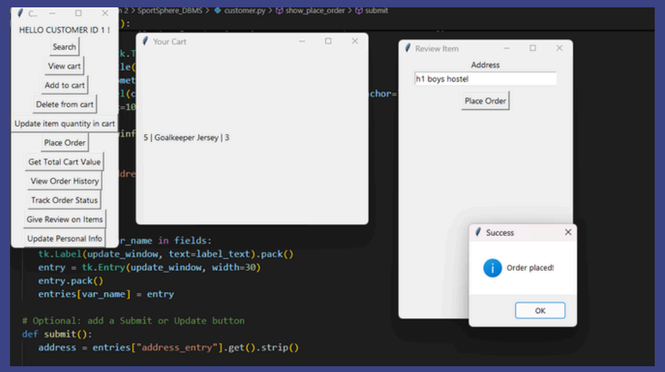
Review Product



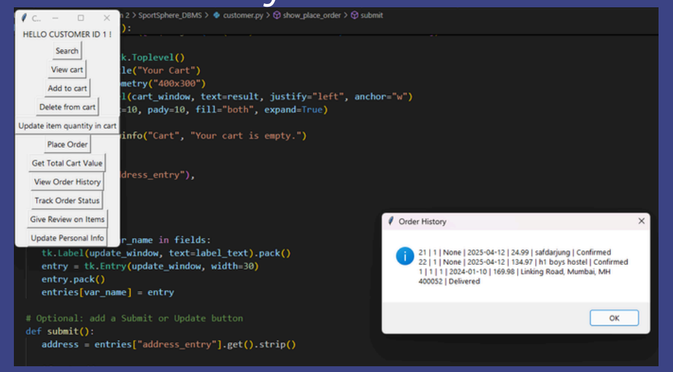
View Cart



Place Order

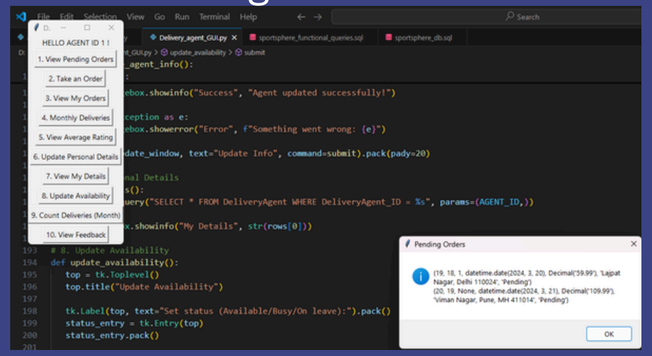


Order History

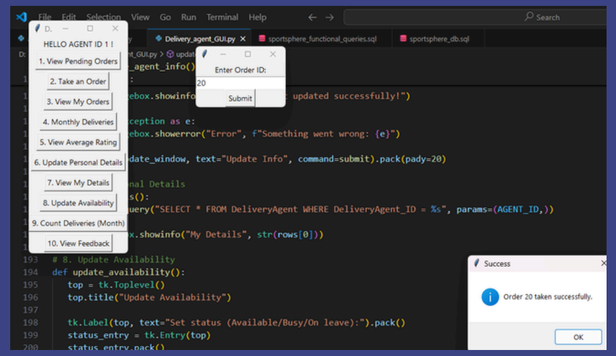


Deadline 4: Agent UI

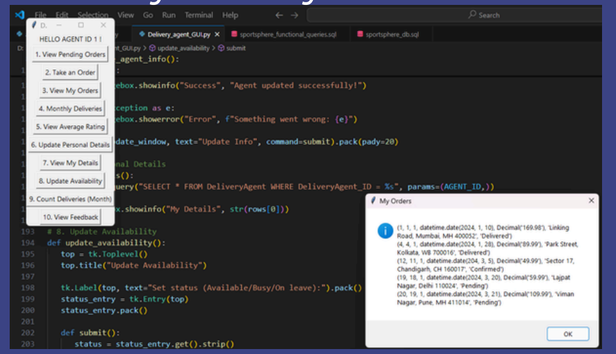
View Pending Order



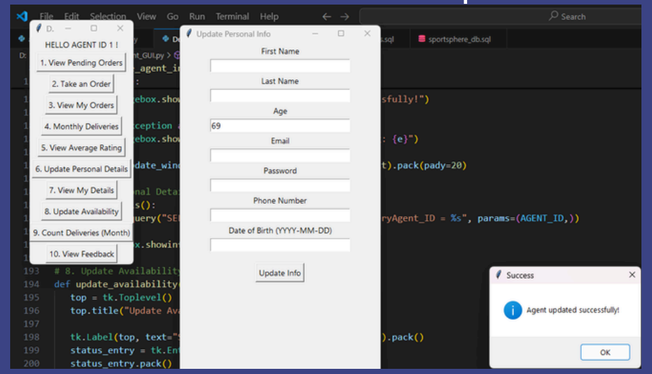
Take Order



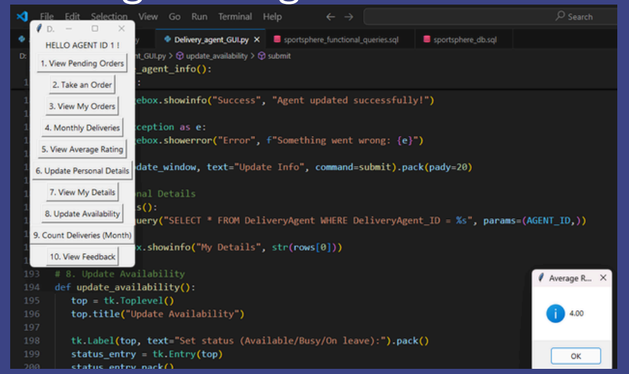
Delivery History



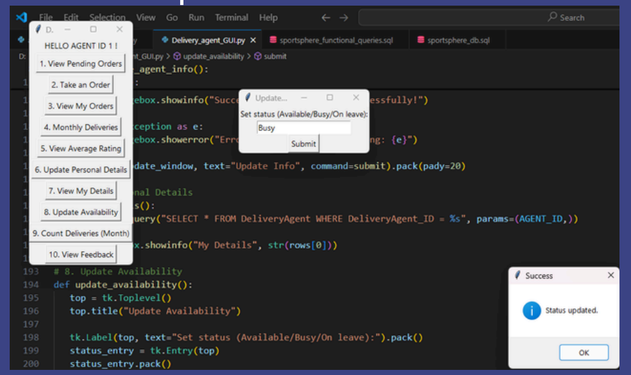
Personal Information Update



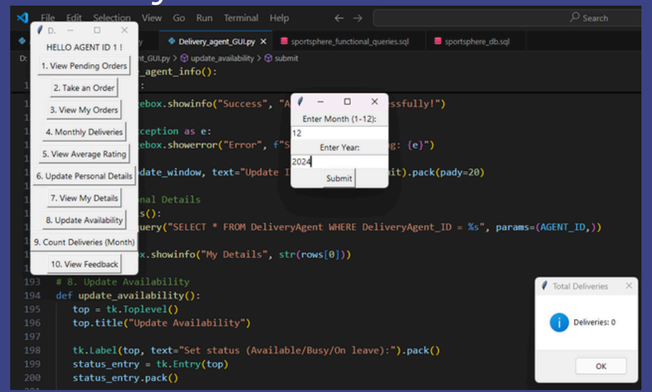
Average Rating



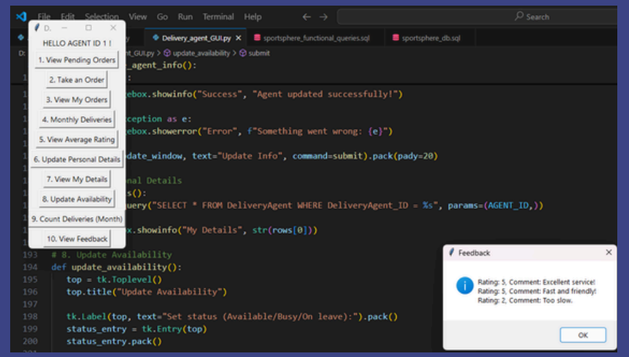
Status Update



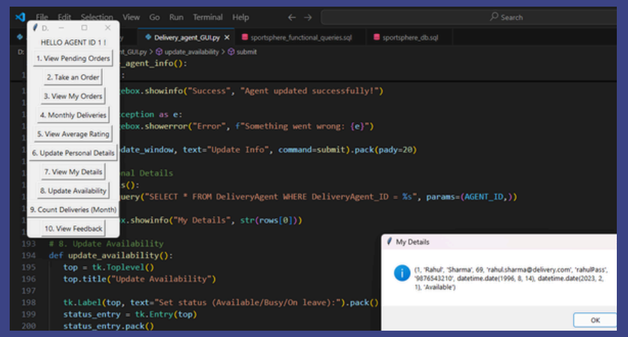
Monthly Deliveries



View Feedback

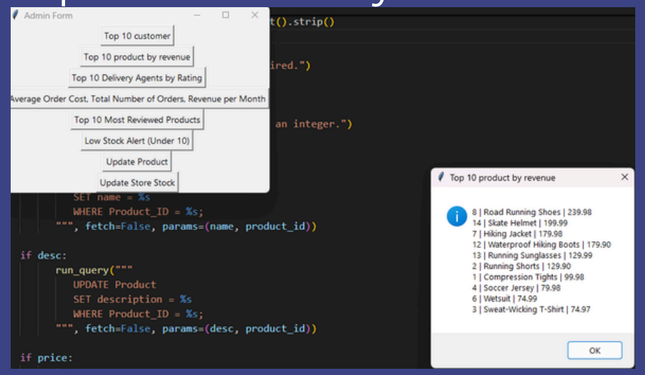


View Personal Details

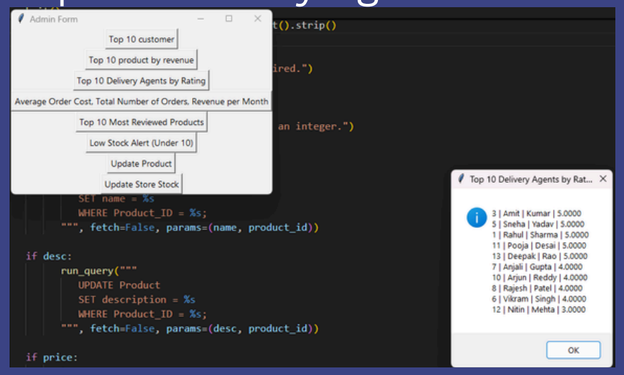


Deadline 4: Admin UI

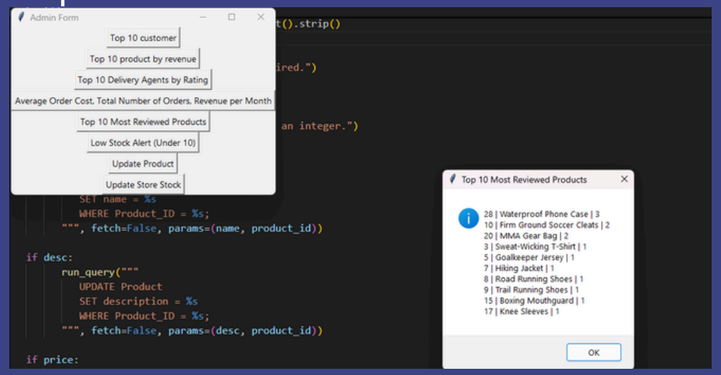
Top 10 Products by revenue



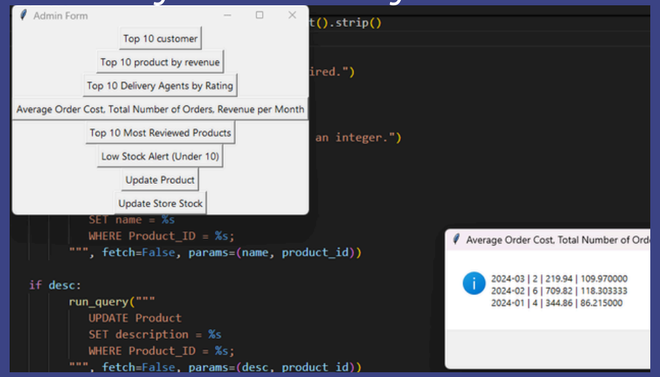
Top 10 Delivery Agents



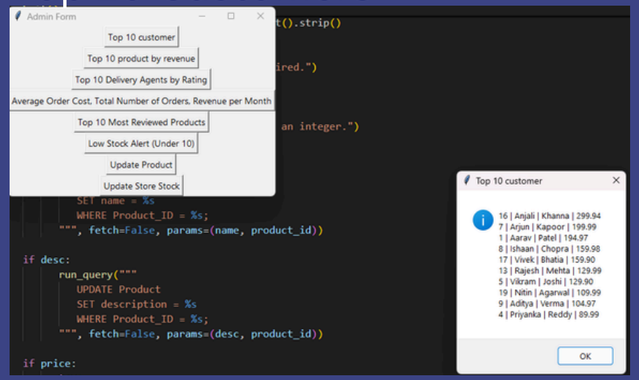
Top 10 Most Reviewed Products



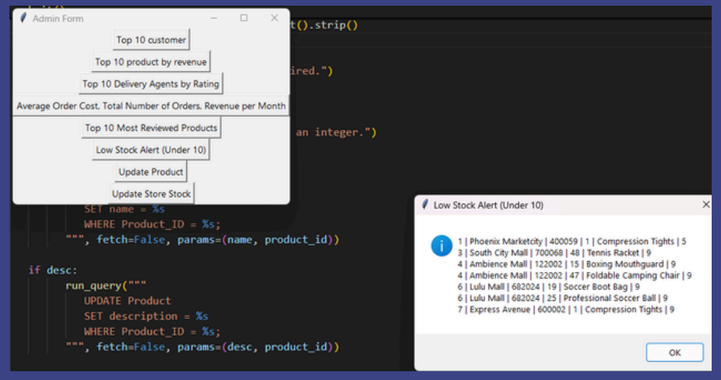
Monthly Sales Analytic



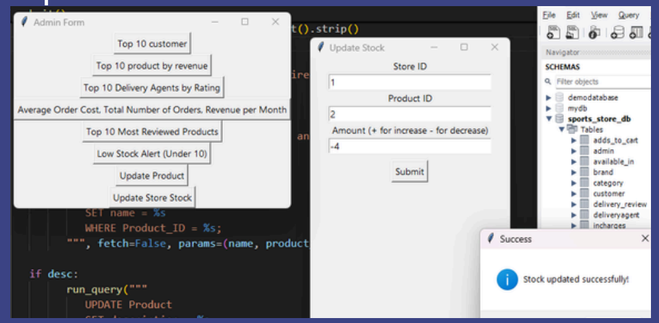
Top 10 Customers



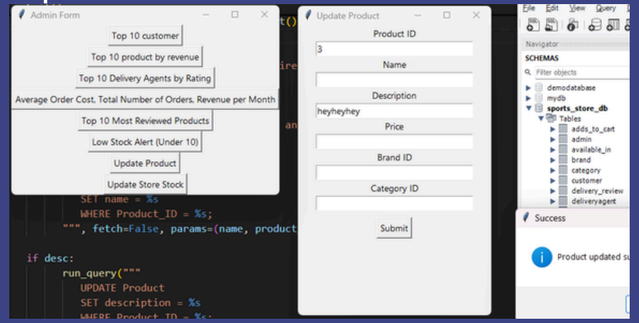
Low Stock Alert



Update Store Stock



Update Product Detail



Thank You For Watching

Tung Duc Vu (2023558), Gaurav Shankar (2023219), Aditya Chaudhary (2023042), Turbold Amarbat (2023559)