Coffee Shop DB

CS-GY 6083 Databases | Spring ‘25

Final Project

Presented By: Esteban Lopez

May 2025

Agenda

Section

01

NYU | TANDON

Agenda

- Project Goal & Overview
- Database Design & Key Features (ERD, Objects)
- Application Technology & Structure
- Live Application Demonstration (CRUD, Orders, Reports)
- Database Concepts Discussion (Normalization, Integrity, Isolation)
- Conclusion

NYU

Database Overview

Section

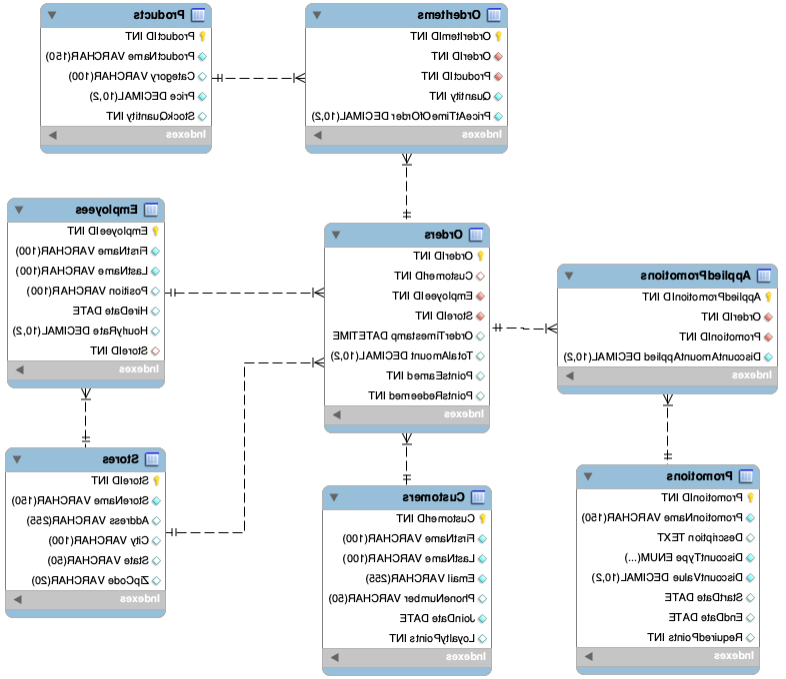
02

NYU | TANDON

MySQL DB Foundation

01. Structure

- Backend powered by a MySQL relational database (coffee_shop schema).
- Designed to reliably store and manage all shop data.
- Features 8 tables,: (Stores, Employees, Customers, Products, Promotions, Orders, OrderItems, AppliedPromotions).
- Relationships enforced via Primary and Foreign Keys.



NYU

DB Objects

Section

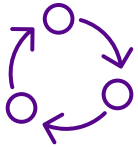
03

Database Object Highlights

1. Views

Simplifying data access for reporting.

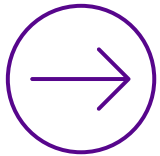
- vw_CustomerOrderSummary
- vw_ProductSalesPerformance



2. Functions

Encapsulating reusable calculations.

- fn_GetCustomerLoyaltyPoints
- fn_CalculatePointsEarned



3. Stored Procedures

Encapsulating business processes..

- sp_AddCustomer
- sp_ProcessOrder

4. Triggers

Automating actions for consistency..

- trg_UpdateStockAfterOrder

NYU

App Tech Stack & Structure

Section

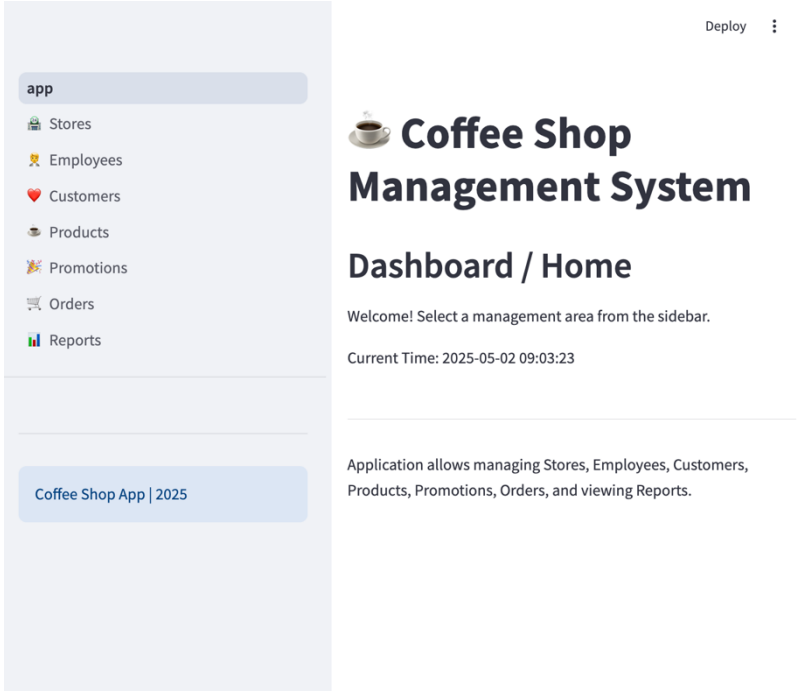
04

NYU | TANDON

App Tech Stack & Structure

- Stack:
  - Python, Streamlit (Web UI), PyMySQL (DB Connector)
- Execution:
  - Runs locally.
- Structure:

```
|-- App/
    |-- pages/ # Streamlit page scripts
    |     |-- 01_🏪_Stores.py
    |     |-- 02_👨💼_Employees.py
    |     |-- 03_👤_Customers.py
    |     |-- 04_☕_Products.py
    |     |-- 05_🎉_Promotions.py
    |     |-- 06_📋_Orders.py
    |     |-- 07_📊_Reports.py
    |-- app.py # Main Streamlit app file (Home page)
    |-- database.py # DB connection & helper functions
```



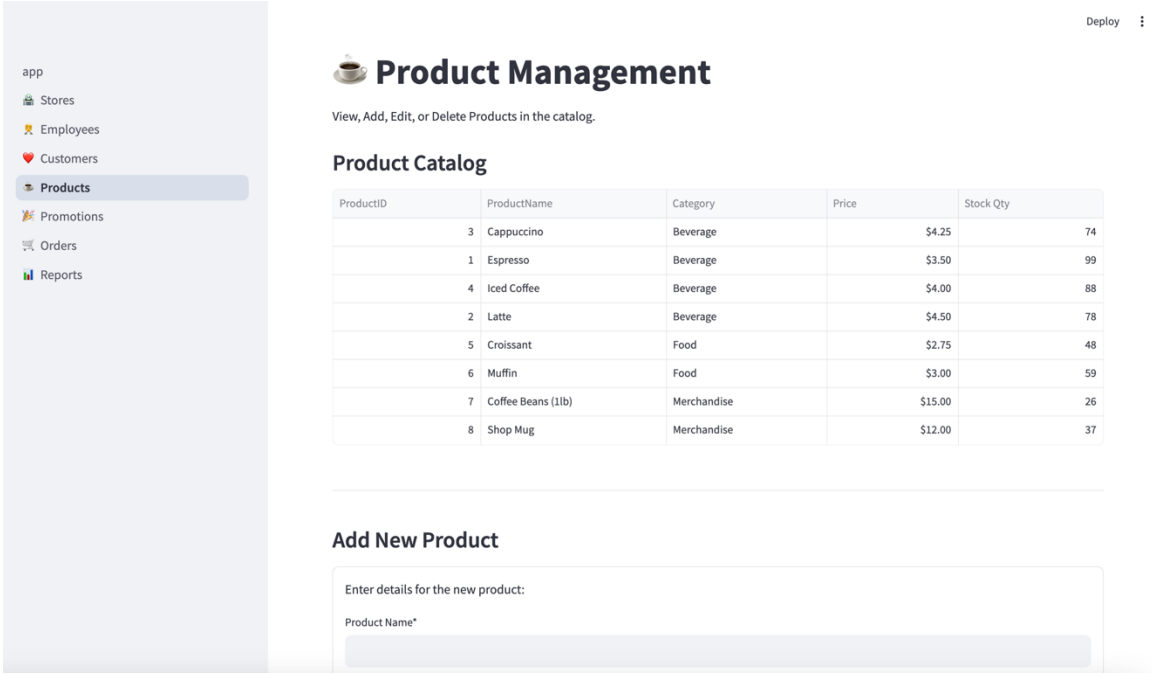
Live Demo

Section

05

Live Demo

- CRUD Operations (Products Example)
- Order Creation Workflow (Items, Points, Promotions)
- Order Viewing
- Reports Dashboard



DB Concepts

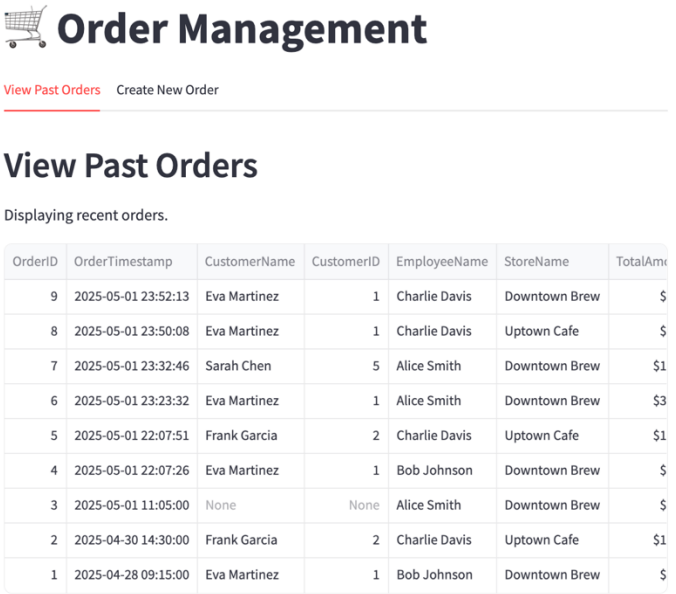
Section

06

NYU | TANDON

DB Concepts – Normalization

- **Level Achieved:** 3NF / BCNF
- **How:** PKs, Full Key Dependency, No Transitive Dependencies
  - **Example:** ProductName in Products, not repeated in OrderItems.
  - **Example:** StoreName in Stores, referenced via StoreID in Orders.
- **Benefits:** Reduced Redundancy, Data Integrity, Update Efficiency.



Order Management

View Past Orders | Create New Order

View Past Orders

Displaying recent orders.

OrderID	OrderTimestamp	CustomerName	CustomerID	EmployeeName	StoreName	TotalAm
9	2025-05-01 23:52:13	Eva Martinez	1	Charlie Davis	Downtown Brew	$
8	2025-05-01 23:50:08	Eva Martinez	1	Charlie Davis	Uptown Cafe	$
7	2025-05-01 23:32:46	Sarah Chen	5	Alice Smith	Downtown Brew	$1
6	2025-05-01 23:23:32	Eva Martinez	1	Alice Smith	Downtown Brew	$3
5	2025-05-01 22:07:51	Frank Garcia	2	Charlie Davis	Uptown Cafe	$1
4	2025-05-01 22:07:26	Eva Martinez	1	Bob Johnson	Downtown Brew	$
3	2025-05-01 11:05:00	None	None	Alice Smith	Downtown Brew	$
2	2025-04-30 14:30:00	Frank Garcia	2	Charlie Davis	Uptown Cafe	$1
1	2025-04-28 09:15:00	Eva Martinez	1	Bob Johnson	Downtown Brew	$

NYU

DB Concepts – Integrity Enf.

- **Primary Keys** (Unique Rows, e.g., OrderID)
- **Foreign Keys** (Relationships, e.g., Orders.CustomerID -> Customers.CustomerID)
    - **Defined Behaviors:** ON DELETE RESTRICT, ON DELETE SET NULL
- **UNIQUE** Constraints (e.g., Customers.Email)
- **NOT NULL** Constraints (e.g., Products.Price)
- **CHECK** Constraints (e.g., StockQuantity >= 0)
- **Procedural Logic** (sp_ProcessOrder stock/point validation)

NYU

DB Concepts – Isolation

- **Environment:** MySQL (InnoDB Engine)
- **Default Level:** REPEATABLE READ
  - Provides: Good consistency (No Dirty/Non-Repeatable Reads).
  - Sufficient For: Most CRUD/Reporting tasks in this app.
- **Explicit Transactions:** sp_ProcessOrder uses START TRANSACTION...COMMIT/ROLLBACK.
  - Ensures: Atomicity for critical multi-step order processing.

NYU

Takeaways

Section

07

NYU | TANDON

Conclusion

- Successfully built a functional Coffee Shop Management System.
- Met project requirements for database objects, CRUD, and reporting.
- Applied core database concepts (Normalization, Integrity, Transactions).
- Demonstrated integration between Python/Streamlit UI and MySQL backend.

NYU