Restaurant Management System: Database Design and Features

Solving Real-World Challenges with SQL

Presented by: Rohit Kumar

Date: 2024-12-12

Overview of the Problem

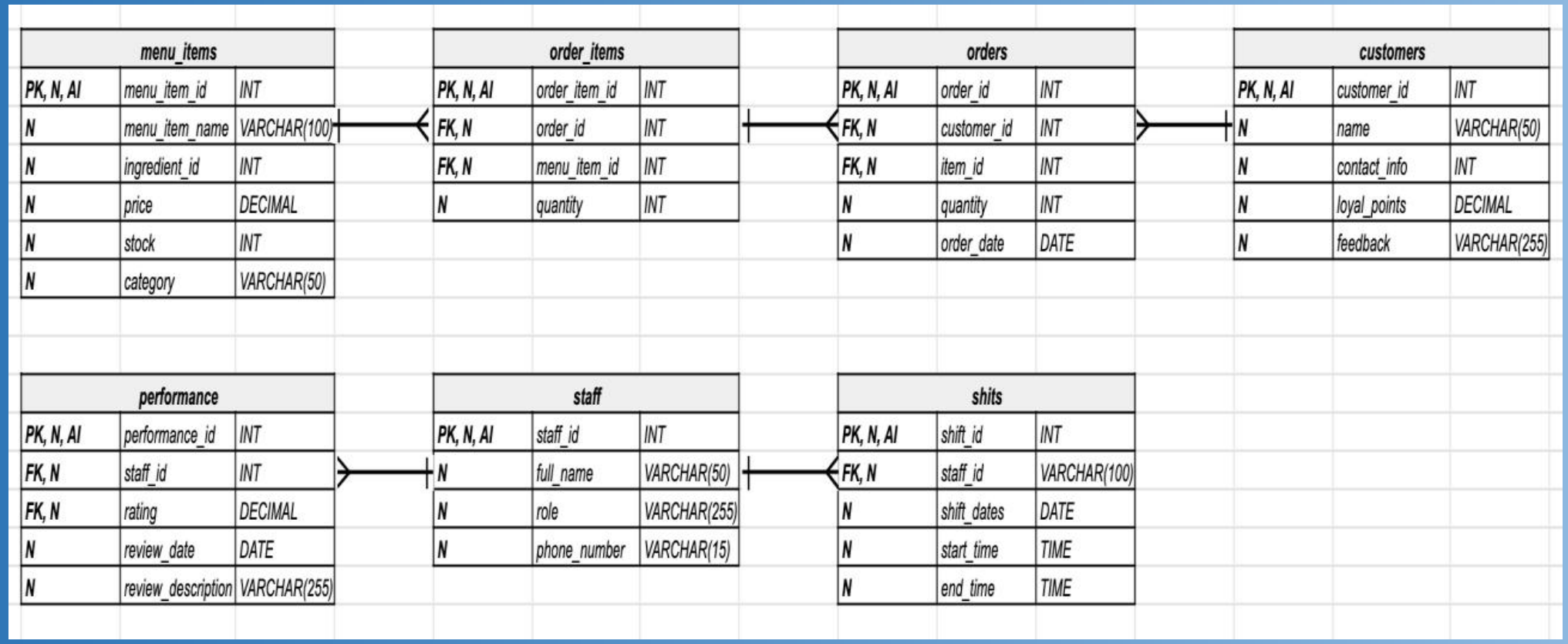
- **Context:** Managing a restaurant involves tracking customer orders, maintaining inventory, and evaluating staff performance.
- **Challenges:**
    1. **Customer and Inventory Management:** Track orders while automatically adjusting stock levels.
    2. **Staff Performance Evaluation:** Generate performance reports to improve service quality.

Objectives

- **Design a normalized database to:**
  - Efficiently store and manage customer, staff, and order data.
  - Automate inventory updates using triggers.
  - Provide staff performance reports using stored procedures.

ER Diagram

- **Content:**
  - Fully normalized Entity-Relationship Diagram showcasing all tables:
    - Customers, Orders, Items, Menu, Staff, Shifts, Performance.
- Visual:



menu_items		
PK, N, AI	menu_item_id	INT
N	menu_item_name	VARCHAR(100)
N	ingredient_id	INT
N	price	DECIMAL
N	stock	INT
N	category	VARCHAR(50)

order_items		
PK, N, AI	order_item_id	INT
FK, N	order_id	INT
FK, N	menu_item_id	INT
N	quantity	INT

orders		
PK, N, AI	order_id	INT
FK, N	customer_id	INT
FK, N	item_id	INT
N	quantity	INT
N	order_date	DATE

customers		
PK, N, AI	customer_id	INT
N	name	VARCHAR(50)
N	contact_info	INT
N	loyal_points	DECIMAL
N	feedback	VARCHAR(255)

performance		
PK, N, AI	performance_id	INT
FK, N	staff_id	INT
FK, N	rating	DECIMAL
N	review_date	DATE
N	review_description	VARCHAR(255)

staff		
PK, N, AI	staff_id	INT
N	full_name	VARCHAR(50)
N	role	VARCHAR(255)
N	phone_number	VARCHAR(15)

shits		
PK, N, AI	shift_id	INT
FK, N	staff_id	VARCHAR(100)
N	shift_dates	DATE
N	start_time	TIME
N	end_time	TIME

Code Demo and Features

Automated Inventory Updates:-

- **Feature:** Trigger to update stock levels dynamically after each order.
- **Code:**

```
DELIMITER //
CREATE TRIGGER update_inventory_trigger
AFTER INSERT ON items
FOR EACH ROW
BEGIN
   UPDATE menu
   SET stock = stock - NEW.quantity
   WHERE id = NEW.menu_item_id;
END //
DELIMITER ;
```

- **Explanation:** Prevents manual stock updates, ensuring real-time inventory accuracy.

Code Demo and Features

Staff Performance Reporting:-

- **Feature:** Generate performance reports using a stored procedure
- **Code:**

```
DELIMITER //
CREATE PROCEDURE generate_staff_performance_report(IN start_date
DATE, IN end_date DATE)
BEGIN
   SELECT s.name, p.rating, p.review_date, p.review_description
   FROM staff s
   JOIN performance p ON s.id = p.staff_id
   WHERE p.review_date BETWEEN start_date AND end_date;
END //
DELIMITER ;
```

- **Explanation:** Retrieves performance ratings and reviews for a specified date range.

Conclusion and Questions

- **Challenge 1 Solved:** Automated inventory management.
- **Challenge 2 Solved:** Staff performance evaluation via reports.
- **Key Features:**
  - Normalized database structure.
  - Automation through triggers and procedures.