# Facilitating a Relational Database for Data-Driven Decision Making

**LUCENO, CRISTOPHER JOHN**
DATA CONCEPTS, DATA 037-008
Continuing Education and Professional Studies
Southern Alberta Institute of Technology

**TABLE OF CONTENTS**

## I. INTRODUCTION

The modern business landscape is increasingly shaped by data-driven decision-making, offering both opportunities and challenges for enterprises. A well-structured relational database serves as the foundation for managing and analyzing business data effectively. Success in this competitive environment requires a strategic approach that focuses on data integrity, operational efficiency, and technological adaptability. This article outlines a comprehensive framework for designing a relational database that enables organizations to make informed decisions, emphasizing key database principles, data analytics strategies, and essential technological considerations.

## II. MISSION AND OBJECTIVES

A well-defined mission establishes the foundation for effective database implementation. The mission of this study is:

*"To develop a scalable, efficient, and secure relational database that enhances business intelligence and data-driven decision-making."*

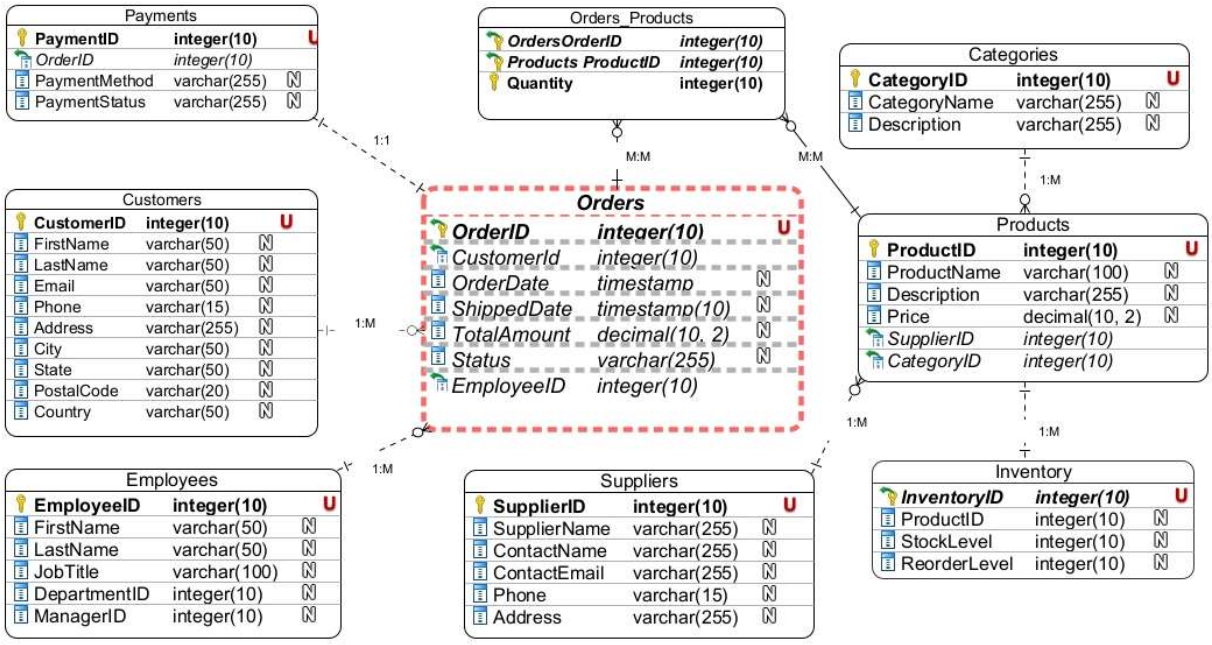
This mission translates into three core strategic objectives. The first objective is to optimize data organization and accessibility, ensuring that information is structured efficiently for ease of retrieval and analysis. The second objective is to leverage data analytics to provide meaningful insights that inform business strategies, enabling organizations to identify trends and optimize performance. Lastly, achieving operational excellence through well-designed database management enhances data security, streamlines workflows, and improves overall efficiency.

## III. FOUNDATIONAL DATA STRUCTURES

A well-structured relational database is essential for efficient data management. The key entities and their relationships include:

Customers represent individuals who interact with the organization and generate data through transactions. Orders capture records of customer transactions, including purchase details and fulfillment status. Employees manage various operational processes, such as customer service and logistics. Products encompass items or services offered, with attributes including descriptions, categories, and pricing. Categories provide a structured classification of products based on common characteristics. Suppliers contribute raw materials or finished goods essential to operations. Inventory management ensures real-time tracking of stock levels for better decision-making. Payments store financial transaction records, tracking payment methods and transaction details.

## IV. ENTITY RELATIONSHIP DIAGRAM




## ENTITY RELATIONSHIP TABLE

| Entities | Relationship | Details |
|---|---|---|
| Customers ↔ Orders | One-to-Many | Multiple orders per customers |
| Orders → Payments | One-to-One | Single payment for single order |
| Orders ↔ Employees | One-to-Many | Multiple order processing per employee |
| Orders ↔ Orders Products | One-to-Many | Multiple products per order |
| Products ↔ Orders Products | One-to-Many | Multiple orders for single product |
| Products ↔ Categories | One-to-Many | Multiple products belong to single category |
| Products ↔ Suppliers | One-to-Many | Multiple products per supplier |
| Products → Inventory | One-to-One | Products have unique identifier for inventory |

## SAMPLE DATABASE

A sample database for this case study is available at: GitHub Repository.

## SAMPLE QUERIES

```
mysql> -- Revenue trend from Jan 2024
mysql>     SELECT
    ->         cat.CategoryName,
    ->         SUM(o.TotalAmount) AS TotalRevenue,
    ->         COUNT(DISTINCT o.OrderID) AS TotalOrders
    ->     FROM Orders o
    ->     JOIN Orders_Products op ON o.OrderID = op.OrderID
    ->     JOIN Products p ON op.ProductID = p.ProductID
    ->     JOIN Categories cat ON p.CategoryID = cat.CategoryID
    ->     WHERE o.OrderDate >= '2024-01-01'
    ->     GROUP BY cat.CategoryName
    ->     ORDER BY TotalRevenue DESC;
+-----------------+-------------------+-------------+
| CategoryName    | TotalRevenue      | TotalOrders |
+-----------------+-------------------+-------------+
| Male_Shirts     |             135.5 |           2 |
| Male_Tshirts    | 123.4900016784668 |           2 |
| Female_Trousers |  95.9900016784668 |           2 |
| Male_Trousers   |                80 |           1 |
| Female_Shirts   |  76.9900016784668 |           2 |
| Female_Tshirts  |  55.9900016784668 |           1 |
+-----------------+-------------------+-------------+
6 rows in set (0.00 sec)
```

*Revenue trend from January 2024*

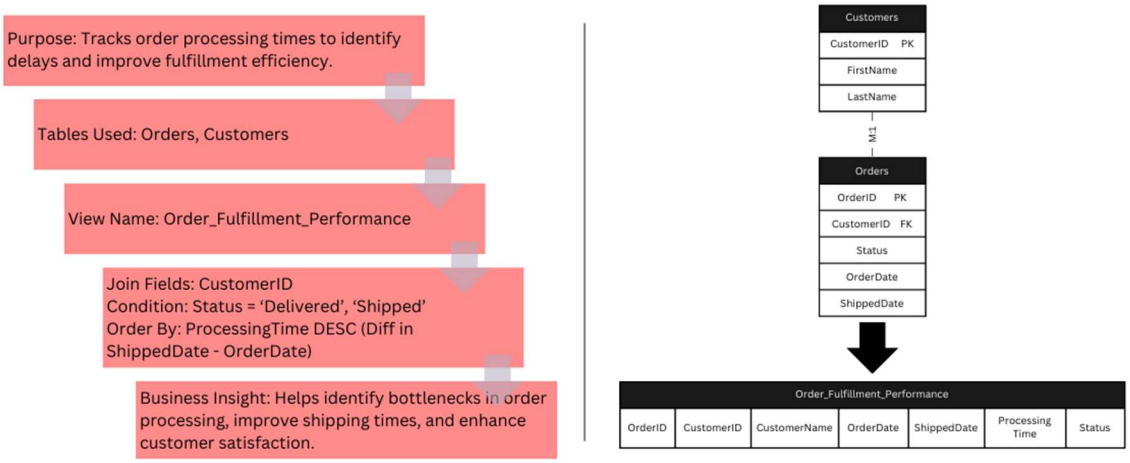
```
mysql> -- Returning Customers vs. One-Time Buyers
mysql> SELECT
    ->     CASE
    ->         WHEN order_count > 1 THEN 'Returning Customer'
    ->         ELSE 'One-Time Buyer'
    ->     END AS CustomerType,
    ->     COUNT(*) AS TotalCustomers
    -> FROM (
    ->     SELECT CustomerID, COUNT(OrderID) AS order_count
    ->     FROM Orders
    ->     GROUP BY CustomerID
    -> ) AS OrderSummary
    -> GROUP BY CustomerType;
+--------------------+----------------+
| CustomerType       | TotalCustomers |
+--------------------+----------------+
| Returning Customer |              2 |
| One-Time Buyer     |              3 |
+--------------------+----------------+
2 rows in set (0.00 sec)
```

*Returning Customers vs. One-Time Buyers*

```
mysql> -- Employee Order Management from Jan 2024
mysql> SELECT
    ->     e.EmployeeID,
    ->     CONCAT(e.FirstName, ' ', e.LastName) AS EmployeeName,
    ->     COUNT(o.OrderID) AS TotalOrdersHandled
    -> FROM Orders o
    -> JOIN Employees e ON o.EmployeeID = e.EmployeeID
    -> WHERE o.OrderDate >= '2024-01-01'
    -> GROUP BY e.EmployeeID, EmployeeName
    -> ORDER BY TotalOrdersHandled DESC;
+------------+--------------+--------------------+
| EmployeeID | EmployeeName | TotalOrdersHandled |
+------------+--------------+--------------------+
|        601 | Liam Brown   |                  3 |
|        604 | Ava Miller   |                  2 |
|        603 | Noah Wilson  |                  2 |
+------------+--------------+--------------------+
3 rows in set (0.00 sec)
```

*Employee Order Management from January 2024*

## VIEW – ORDER FULFILLMENT PERFORMANCE




```
mysql> CREATE VIEW Order_Fulfillment_Performance_View AS
    -> SELECT
    ->     o.OrderID,
    ->     c.CustomerID,
    ->     CONCAT(c.FirstName, ' ', c.LastName) AS customer_name,
    ->     o.OrderDate,
    ->     o.ShippedDate,
    ->     DATEDIFF(o.ShippedDate, o.OrderDate) AS processing_time_days,
    ->     o.Status
    -> FROM Orders o
    -> JOIN Customers c ON o.CustomerID = c.CustomerID
    -> WHERE o.Status IN ('Delivered', 'Shipped')
    -> ORDER BY processing_time_days DESC;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from Order_Fulfillment_Performance_View;
+---------+------------+--------------+---------------------+---------------------+---------------------+-----------+
| OrderID | CustomerID | customer_name | OrderDate           | ShippedDate         | processing_time_days | Status    |
+---------+------------+--------------+---------------------+---------------------+---------------------+-----------+
|     700 |     500100 | Alex Roberts  | 2024-01-10 14:30:00 | 2024-01-11 10:00:00 |                   1 | Shipped   |
|     701 |     500101 | Jennifer Kirk | 2024-01-11 10:15:00 | 2024-01-12 12:00:00 |                   1 | Delivered |
|     703 |     500103 | Sophia Grace  | 2024-01-13 13:10:00 | 2024-01-14 09:30:00 |                   1 | Shipped   |
|     704 |     500104 | Henry Powell  | 2024-01-14 11:55:00 | 2024-01-15 14:20:00 |                   1 | Delivered |
+---------+------------+--------------+---------------------+---------------------+---------------------+-----------+
4 rows in set (0.00 sec)
```

## VIEW – TOP SELLING PRODUCT

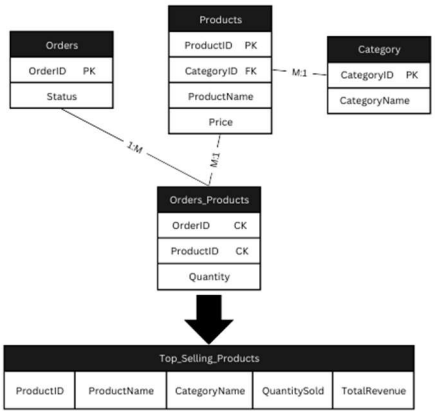
Purpose: Identifies the best-selling products based on order quantity and revenue.

Tables Used: Orders, Orders_Products, Products, Categories

View Name: Top_Selling_Products

Join Fields: ProductID, CategoryID, OrderID
Condition: Status = 'Delivered'
Group By: ProductID, ProductName, CategoryName
Order By: TotalRevenue DESC

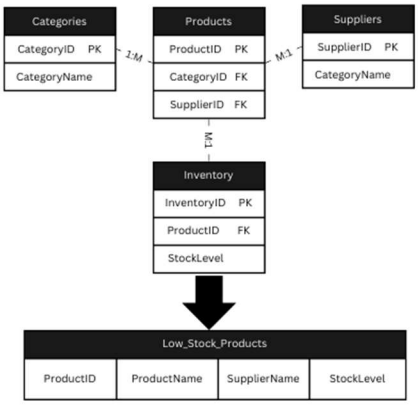
Business Insight: Helps in understanding which products drive the most revenue and sales volume, guiding restocking and marketing decisions.




```
mysql> CREATE VIEW Top_Selling_Products_View AS
    -> SELECT
    ->     p.ProductID,
    ->     p.ProductName,
    ->     c.CategoryName,
    ->     SUM(op.Quantity) AS total_quantity_sold,
    ->     SUM(op.Quantity * p.Price) AS total_revenue
    -> FROM Orders_Products op
    -> JOIN Products p ON op.ProductID = p.ProductID
    -> JOIN Categories c ON p.CategoryID = c.CategoryID
    -> JOIN Orders o ON op.OrderID = o.OrderID
    -> WHERE o.Status = 'Delivered'
    -> GROUP BY p.ProductID, p.ProductName, c.CategoryName
    -> ORDER BY total_revenue DESC;
Query OK, 0 rows affected (0.03 sec)

mysql> select * from Top_Selling_Products_View;
+-----------+---------------+--------------+---------------------+---------------+
| ProductID | ProductName   | CategoryName | total_quantity_sold | total_revenue |
+-----------+---------------+--------------+---------------------+---------------+
|       302 | Men Blue Jeans | Male_Trousers |                   2 |            80 |
|       305 | Women Blouse   | Female_Shirts |                   1 |          28.5 |
+-----------+---------------+--------------+---------------------+---------------+
2 rows in set (0.04 sec)
```

## VIEW – LOW STOCK PRODUCT


```
mysql> CREATE VIEW Low_Stock_Products_View AS
    -> SELECT
    ->     p.ProductID,
    ->     p.ProductName,
    ->     c.CategoryName,
    ->     i.StockLevel,
    ->     s.SupplierName
    -> FROM Inventory i
    -> JOIN Products p ON i.ProductID = p.ProductID
    -> JOIN Categories c ON p.CategoryID = c.CategoryID
    -> JOIN Suppliers s ON p.SupplierID = s.SupplierID
    -> WHERE i.StockLevel < 50
    -> ORDER BY i.StockLevel ASC;
Query OK, 0 rows affected (0.05 sec)

mysql> select * from Low_Stock_Products_View;
+-----------+------------------+----------------+------------+------------------+
| ProductID | ProductName      | CategoryName   | StockLevel | SupplierName     |
+-----------+------------------+----------------+------------+------------------+
|       303 | Women Black Jeans | Female_Trousers |         30 | Trendy Wear Inc. |
|       305 | Women Blouse      | Female_Shirts   |         35 | Trendy Wear Inc. |
|       302 | Men Blue Jeans    | Male_Trousers   |         40 | ABC Clothing Co. |
|       304 | Men Formal Shirt  | Male_Shirts     |         45 | ABC Clothing Co. |
+-----------+------------------+----------------+------------+------------------+
4 rows in set (0.00 sec)
```

## V. CONCLUSION

Success in the competitive e-commerce landscape requires a strategic and data-driven approach. By prioritizing customer experience, leveraging data analytics, and embracing technological innovation, apparel retailers can build a thriving online presence and achieve sustainable growth. This framework provides a roadmap for navigating the complexities of e-commerce and establishing a brand that resonates with target customers.