# Commerce System Database

# Team3

# Project Scope and Objectives:

**Scope:**

This database aims to store and manage data for an E-commerce platform. It will facilitate the storage of customer information, product details, shopping carts, orders, and order details.

**Objectives:**

**Efficient Customer Management**: Store and manage customer information for efficient order processing and tracking.

**Accurate Product Inventory:** Maintain accurate product details including stock quantities and pricing.

**Smooth Order Processing:** Enable the creation, management, and tracking of orders, ensuring accurate billing and timely delivery.

**Detailed Order History:** Store comprehensive order details to analyze customer preferences and sales trends.

# Data Model ERD diagram:

# 

# Admin 1:

**Name**: Admin1

**Responsibilities**: Database maintenance, user access management, and ensuring data integrity.

# Admin 2:

**Name**: Admin2

**Responsibilities**: Overseeing database security measures, backups, and troubleshooting issues related to the database system.

## Name of Database will be (EcommerceDB).

# The tables are:

* Customer
* Product
* Cart
* Order
* OrderDetails

# Column Detail (Attributes) in Tables:

## Customer Table:

* CustomerID (Primary Key)
* Name
* Email
* Address
* Phone

## Product Table:

* ProductID (Primary Key)
* Name
* Description
* Price
* StockQuantity

## Cart Table:

* CartID (Primary Key)
* CustomerID (Foreign Key referencing Customer)
* CreationDate

## Order:

OrderID (Primary Key)

CustomerID (Foreign Key referencing Customer)

OrderDate

TotalAmount

## OrderDetails:

OrderDetailID (Primary Key)

OrderID (Foreign Key referencing Order)

ProductID (Foreign Key referencing Product)

Quantity

UnitPrice

# Database Schema:

**Customer Table**

| Attribute | Data Type | Constraints |
| --- | --- | --- |
| CustomerID | Primary Key, Integer | Not Null, Unique |
| Name | Varchar | Not Null |
| Email | Varchar | Not Null, Unique |
| Address | Varchar |  |
| Phone | Varchar |  |

**Product Table**

| Attribute | Data Type | Constraints |
| --- | --- | --- |
| ProductID | Primary Key, Integer | Not Null, Unique |
| Name | Varchar | Not Null |
| Description | Varchar |  |
| Price | Decimal | Not Null, >= 0 |
| StockQuantity | Integer | Not Null, >= 0 |

**Cart Table**

| Attribute | Data Type | Constraints |
| --- | --- | --- |
| CartID | Primary Key, Integer | Not Null, Unique |
| CustomerID | Foreign Key (Customer.CustomerID) | Not Null, References Customer(CustomerID) |
| CreationDate | Date | Not Null |

**Order Table**

| Attribute | Data Type | Constraints |
| --- | --- | --- |
| OrderID | Primary Key, Integer | Not Null, Unique |
| CustomerID | Foreign Key (Customer.CustomerID) | Not Null, References Customer(CustomerID) |
| OrderDate | Date | Not Null |
| TotalAmount | Decimal | Not Null, >= 0 |

**OrderDetails Table**

| Attribute | Data Type | Constraints |
| --- | --- | --- |
| OrderDetailID | Primary Key, Integer | Not Null, Unique |
| OrderID | Foreign Key (Order.OrderID) | Not Null, References Order(OrderID) |
| ProductID | Foreign Key (Product.ProductID) | Not Null, References Product(ProductID) |
| Quantity | Integer | Not Null, >= 0 |
| UnitPrice | Decimal | Not Null, >= 0 |

**Relationships**:

**Customer - Cart:** one-to-one (A customer can have one cart).

**Customer - Orde**r: one-to-many (A customer can have multiple orders).

**Product - OrderDetails**: one-to-many(A product can be in multiple order details).

**Order - OrderDetails**: one-to-many (An order can have multiple order details).

# Data Integrity:

**Task Description**: Ensure data integrity through the use of constraints

(e.g., primary keys, foreign keys, unique constraints).

Primary Key Constraint: Ensures each ProductID is unique.

Data Type Constraints: Define the types of data allowed in each column (**ProductName**, **Price**, **StockQuantity**).

**Check Constraint:** To ensure that the Price and StockQuantity are within reasonable ranges.

# Data entry:

# ### Customer Table:

1. (1, "Alice Smith", "alice@email.com", "123 Main St", "123-456-7890")

2. (2, "Bob Johnson", "bob@email.com", "456 Oak Ave", "987-654-3210")

3. (3, "Charlie Brown", "charlie@email.com", "789 Elm St", "111-222-3333")

4. (4, "David Lee", "david@email.com", "321 Maple Dr", "444-555-6666")

5. (5, "Eva Garcia", "eva@email.com", "555 Pine Ave", "777-888-9999")

6. (6, "Frank White", "frank@email.com", "678 Walnut Ln", "333-222-1111")

7. (7, "Grace Turner", "grace@email.com", "901 Cedar St", "555-444-3333")

8. (8, "Hannah Baker", "hannah@email.com", "234 Birch Ave", "999-888-7777")

9. (9, "Ian Foster", "ian@email.com", "765 Spruce Dr", "222-333-4444")

10. (10, "Jennifer Hall", "jennifer@email.com", "543 Cherry Rd", "888-999-0000")

11. (11, "Kevin Young", "kevin@email.com", "876 Oakwood Blvd", "777-666-5555")

12. (12, "Lily Adams", "lily@email.com", "432 Elmwood Ave", "666-777-8888")

13. (13, "Mike Clark", "mike@email.com", "789 Pinecone Ln", "444-333-2222")

14. (14, "Nancy Green", "nancy@email.com", "210 Maplewood Dr", "333-444-5555")

15. (15, "Olivia King", "olivia@email.com", "987 Cedarwood Dr", "222-111-0000")

16. (16, "Paul Miller", "paul@email.com", "654 Oakhurst Rd", "111-222-3333")

17. (17, "Quinn Harris", "quinn@email.com", "345 Pine Ridge", "999-888-7777")

18. (18, "Rachel Scott", "rachel@email.com", "543 Birchwood Ave", "888-777-6666")

19. (19, "Sam Taylor", "sam@email.com", "876 Elmwood Blvd", "777-666-5555")

20. (20, "Tina Wright", "tina@email.com", "234 Spruceland Dr", "666-555-4444")

# #**#**# Product Table:

1. (101, "T-Shirt", "Cotton t-shirt, black", 19.99, 50)

2. (102, "Jeans", "Blue denim jeans, slim fit", 39.99, 30)

3. (103, "Sneakers", "White canvas sneakers", 29.99, 25)

4. (104, "Backpack", "Nylon backpack, black", 49.99, 20)

5. (105, "Watch", "Stainless steel watch, analog", 99.99, 15)

6. (106, "Dress", "Floral print dress, knee-length", 59.99, 20)

7. (107, "Jacket", "Leather jacket, brown", 79.99, 10)

8. (108, "Running Shoes", "Mesh running shoes, red", 49.99, 25)

9. (109, "Hoodie", "Cotton hoodie, gray", 34.99, 30)

10. (110, "Skirt", "Pleated skirt, navy blue", 39.99, 25)

11. (111, "Sweater", "Knit sweater, striped", 44.99, 20)

12. (112, "Blouse", "Silk blouse, white", 29.99, 25)

13. (113, "Boots", "Leather boots, black", 69.99, 15)

14. (114, "Scarf", "Cashmere scarf, plaid", 24.99, 40)

15. (115, "Shorts", "Cargo shorts, khaki", 19.99, 35)

16. (116, "Polo Shirt", "Collared polo shirt, blue", 24.99, 30)

17. (117, "Sunglasses", "Polarized sunglasses, aviator", 59.99, 20)

18. (118, "Handbag", "Leather handbag, tan", 89.99, 15)

19. (119, "Earrings", "Diamond stud earrings", 149.99, 10)

20. (120, "Necklace", "Gold-plated necklace, pendant", 199.99, 5)

# ### Cart Table:

1. (501, 1, "2023-11-23")

2. (502, 3, "2023-11-22")

3. (503, 2, "2023-11-21")

4. (504, 5, "2023-11-20")

5. (505, 4, "2023-11-19")

6. (506, 6, "2023-11-18")

7. (507, 8, "2023-11-17")

8. (508, 7, "2023-11-16")

9. (509, 10, "2023-11-15")

10. (510, 9, "2023-11-14")

11. (511, 12, "2023-11-13")

12. (512, 11, "2023-11-12")

13. (513, 14, "2023-11-11")

14. (514, 13, "2023-11-10")

15. (515, 16, "2023-11-09")

16. (516, 15, "2023-11-08")

17. (517, 18, "2023-11-07")

18. (518, 17, "2023-11-06")

19. (519, 20, "2023-11-05")

20. (520, 19, "2023-11-04")

# ### Order Table:

1. (701, 1, "2023-11-15", 39.98)

2. (702, 2, "2023-11-16", 109.97)

3. (703, 3, "2023-11-17", 59.97)

4. (704, 4, "2023-11-18", 99.99)

5. (705, 5, "2023-11-19", 149.97)

6. (706, 6, "2023-11-20", 119

.98)

7. (707, 7, "2023-11-21", 179.97)

8. (708, 8, "2023-11-22", 89.99)

9. (709, 9, "2023-11-23", 169.97)

10. (710, 10, "2023-11-24", 259.95)

11. (711, 11, "2023-11-25", 219.96)

12. (712, 12, "2023-11-26", 129.98)

13. (713, 13, "2023-11-27", 299.94)

14. (714, 14, "2023-11-28", 69.97)

15. (715, 15, "2023-11-29", 249.95)

16. (716, 16, "2023-11-30", 159.96)

17. (717, 17, "2023-12-01", 119.97)

18. (718, 18, "2023-12-02", 199.95)

19. (719, 19, "2023-12-03", 189.96)

20. (720, 20, "2023-12-04", 279.94)

# ### OrderDetails Table:

1. (901, 701, 101, 2, 19.99)

2. (902, 702, 103, 1, 29.99)

3. (903, 702, 102, 3, 39.99)

4. (904, 703, 105, 1, 99.99)

5. (905, 704, 104, 1, 49.99)

6. (906, 706, 106, 2, 59.99)

7. (907, 706, 107, 1, 79.99)

8. (908, 707, 108, 3, 49.99)

9. (909, 708, 109, 2, 34.99)

10. (910, 709, 110, 3, 119.97)

11. (911, 710, 111, 2, 89.98)

12. (912, 711, 112, 1, 29.99)

13. (913, 711, 113, 1, 69.99)

14. (914, 712, 114, 4, 99.96)

15. (915, 713, 115, 3, 59.97)

16. (916, 714, 116, 1, 24.99)

17. (917, 715, 117, 2, 119.98)

18. (918, 716, 118, 1, 89.99)

19. (919, 717, 119, 2, 299.98)

20. (920, 718, 120, 1, 199.99)

# 6. Data analysis(quirys):

1. What are the distinct product categories available?
2. How many customers do we have in each city?
3. What is the total revenue generated from orders placed in the last month?
4. Which products are out of stock?
5. What is the average price of products in each category?
6. How many orders were placed by each customer?
7. Which customers have placed orders worth more than $1000?
8. What is the minimum and maximum price of products?
9. What are the top 5 most ordered products?
10. Which customers have a name starting with 'A'?
11. How many orders were placed between specific dates?
12. What is the total quantity of products sold in each category?
13. Which customers have placed orders for products with IDs 1, 3, or 5?
14. How many orders have more than 3 products in them?
15. What is the average quantity of products ordered in each city?
16. Which products have a price between $50 and $100?
17. What is the total revenue for each year?
18. How many customers have placed orders in each country?
19. What is the highest quantity of a single product ordered?
20. Which products have been ordered more than 10 times?

# Table of entire tasks

|  |  |  |  |
| --- | --- | --- | --- |
| Member | Task Title | Task Description | Task DeadLine |
| Ezz | Database Queries | [file].SQL | friday 2:00 pm |
| Mosafa | friday 2:00 pm |
| Donia | Documentation | A Documetation File Contatin all files and data about databaseproject | friday 3:00 pm |
| Youseff | ERD + Data Integrity | Choose an appropriate database model based on project requirements.(e.g., primary keys, foreign keys, unique constraints). | friday 2:00 pm |
| Ammar | Database Schima + Upload Files+Queries | • Create a database schema based on the data model. • Specify data types, constraints, and relationships for each table. | friday 2:00 pm |