



BSIS 3320 DATABASE SYSTEMS

E-COMMERCE CLOTHING DATABASE SYSTEM

Instructor: Dr. **Turki AlThaqafi**

Prepared by: Ayesha Taneem & Rihab Asif

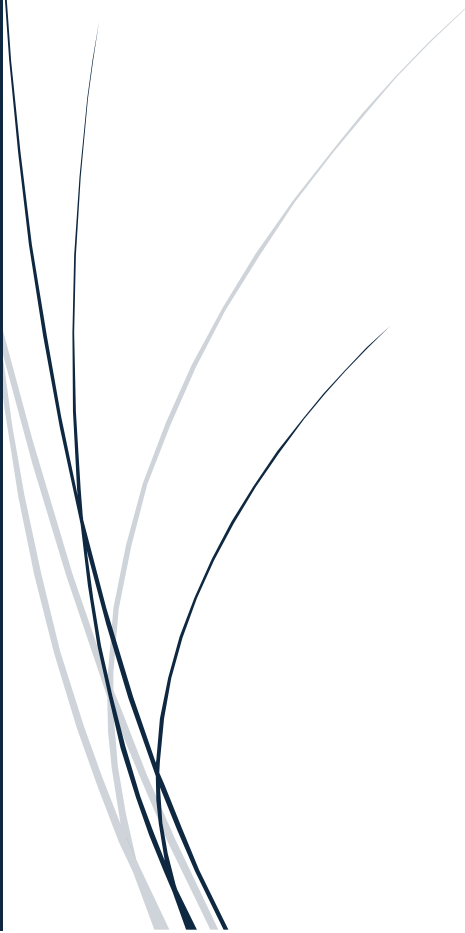


TABLE OF CONTENT

TOPIC	INDEX	PAGE
Phase 1	Data requirements	3
Phase 2	ERD	6
Phase 3	Relational schema Nomalization Final logical model	10
Phase 4	Implementation of sql	20

Phase 1: Requirement Analysis for E-Commerce Management System

1. Introduction

The requirement analysis phase is crucial in the development of any software system as it lays the foundation for understanding what the system needs to accomplish. In this phase, we will outline the user specification for an E-Commerce Management System, focusing on data and transaction requirements. This report will highlight the types of data that need to be stored in the database and identify important and frequent database operations for eTicient system functionality.

2. User Specification

2.1 Data Requirements

The E-Commerce Management System will need to store various types of data to facilitate its operations. These data requirements include:

Entity Name	Description	Aliases	Occurrences
Customer	Information about customers including their identification, contact details, and shipping address.	Clients, Users	1
Seller	Details about sellers, including their identification, contact information, and company details.	Vendors, Merchants	1
Product	Attributes and characteristics of products available	Items, Goods	1

	for purchase, such as name, description, and price.		
Order	Records of customer orders, including order ID, customer ID, seller ID, order date, and total cost.	Purchase Orders, Sales Orders	1
Transaction	Information regarding financial transactions associated with orders, including payment details and status.	Financial Transactions	1
Inventory	Records of available stock for each product managed by sellers.	Stock, Inventory	1
Customer Interaction	Details of interactions between customers and the platform, such as feedback, inquiries, or support queries.	Customer Engagement, Feedback	1

2.2 Transaction Requirements

The E-Commerce Management System will perform various database operations to facilitate the smooth functioning of the platform. These transaction requirements include:

Transaction Type	Tables(Entities) Involved	Actions/Requirements
Customer Registration	Customer Table	<ul style="list-style-type: none"> - Insert a new record with auto-generated CustomerID, Name, Email, Password, ContactNumber, ShippingAddress provided
Customer Login	Customer Table	<ul style="list-style-type: none"> - Verify credentials: Email and hashed Password - Retrieve customer info based on matching Email
Placing an Order	Customer, Order, Inventory, Transaction	<ul style="list-style-type: none"> - Insert a new record in Order table with auto-generated OrderID - OrderDate, TotalCost, ShippingDetails provided - Update Inventory: Decrease StockLevel for purchased products - Insert a new record in Transaction table with auto-generated TransactionID
Seller-related transactions	Seller Table	<ul style="list-style-type: none"> - Insert a new record with auto-generated SellerID - CompanyName, ContactName, Email, Password, ContactNumber, CompanyAddress provided
Adding Products	Product, Inventory, Seller	<ul style="list-style-type: none"> - Insert a new record in Product table with auto-generated ProductID, Name, Description, Price, Category, Brand, Size, Color provided - Insert a new record in Inventory table referencing Product and Seller
Fulfilling Orders	Order Table	<ul style="list-style-type: none"> - Update OrderStatus to "Shipped" or "Delivered"

Updating Customer Info	Customer Table	- Update existing customer record based on CustomerID, -Modify Name, Address, PhoneNumber, etc.
Order Tracking	Customer, Order	- Retrieve order details based on CustomerID from Order table, OrderID, OrderDate, TotalCost, ShippingDetails, OrderStatus
Processing Payments	Transaction Table	- Validate payment details - Insert a new record in Transaction table for payment processing

3. Conclusion

In this phase, we have outlined the user specification for the E-Commerce Management System, focusing on data and transaction requirements. Understanding these requirements is essential for the successful design and implementation of the system. The next phase will involve system design based on these specifications, ensuring that the final product meets the needs of both the business and its users.

This report provides a comprehensive overview of the key aspects to consider during the requirement phase of the E-Commerce Management System development process.

Phase 2: Conceptual Design - Entity-Relationship Model (ER Model)

1. Introduction

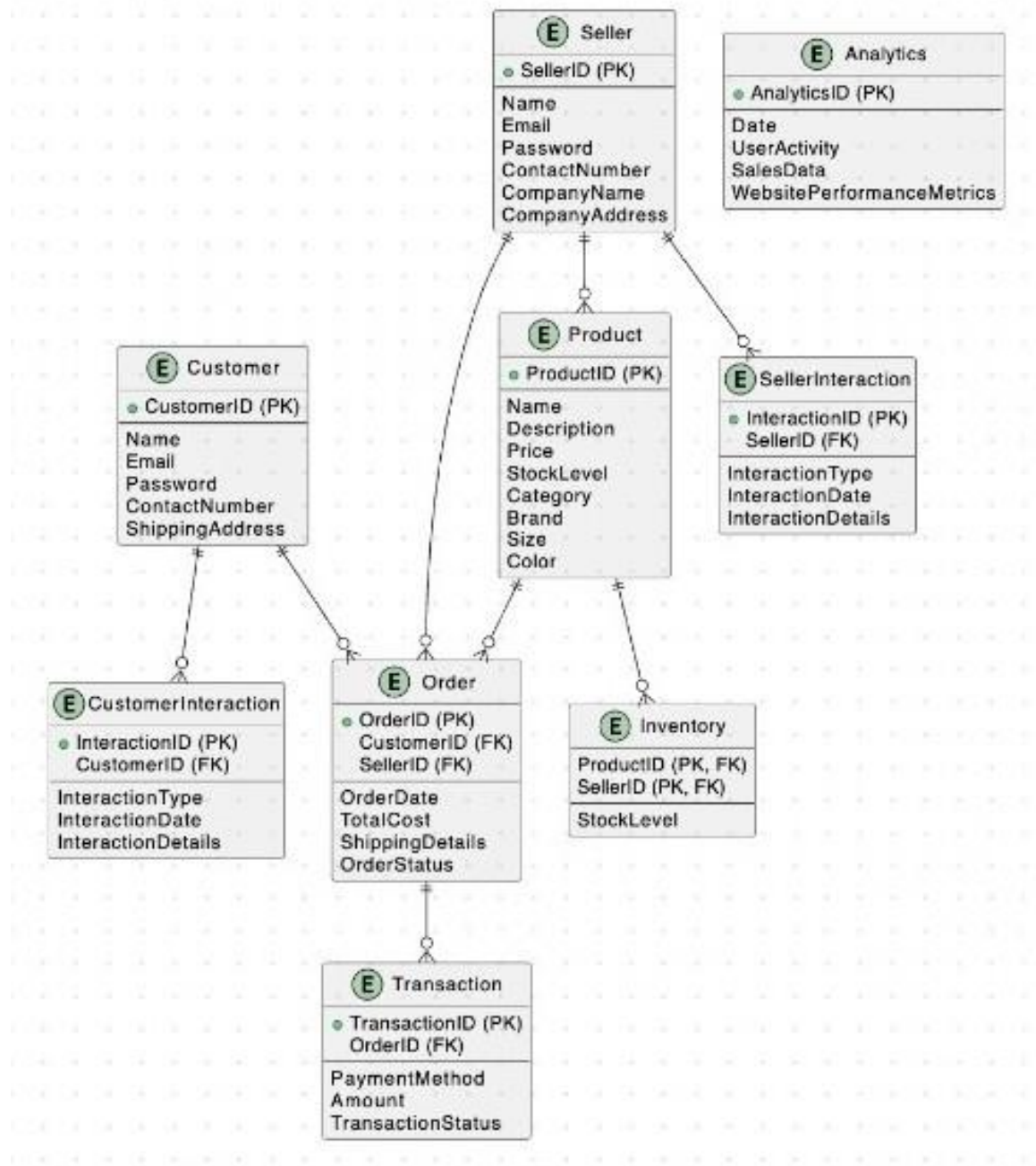
The conceptual design phase is pivotal in the development lifecycle of the E-Commerce Management System. In this phase, we will construct an Entity-Relationship (ER) model to visualize the relationships between different entities in the system. This model will serve as a blueprint for database design, helping to organize and represent the data requirements identified in the previous phase.

2. Entity-Relationship Model

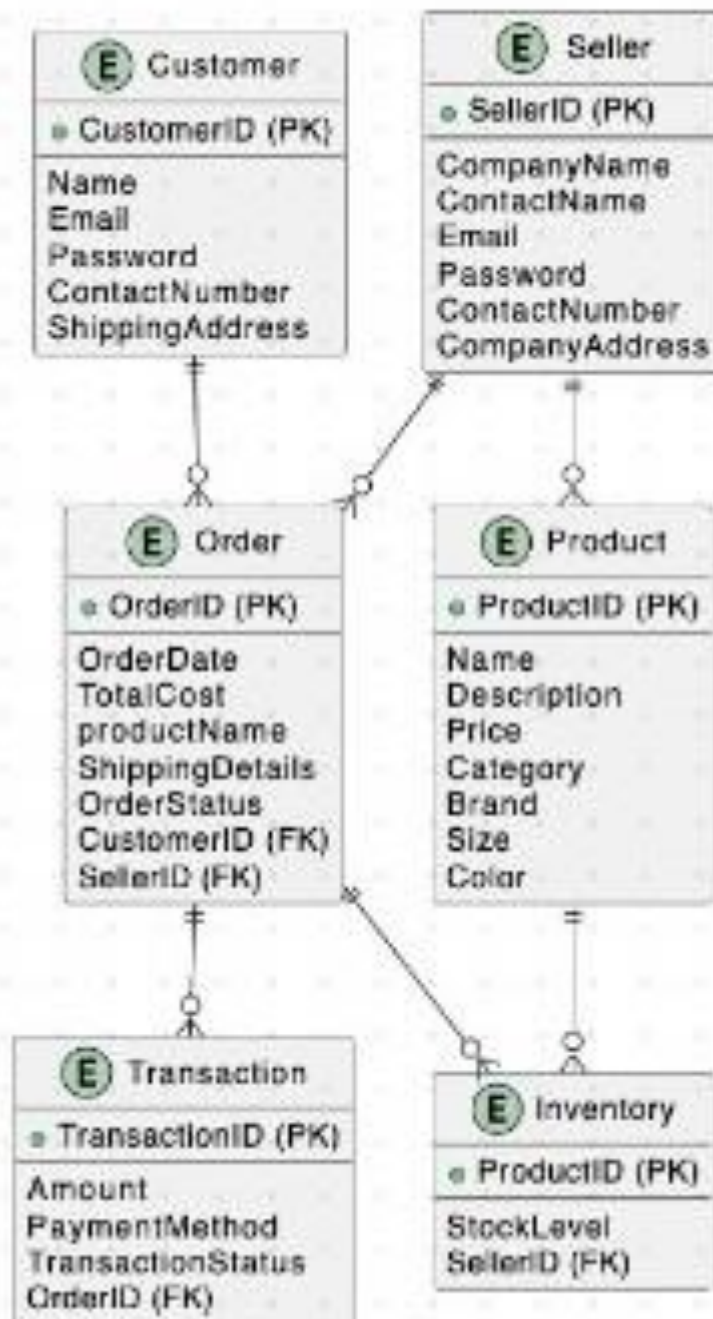
Entities represent the main objects or concepts in the system. Based on the identified data requirements, the following entities and their relationships have been identified:

Entity Name	Multiplicity	Relationship	Multiplicity	Entity Name
Customer	1	(1:1)	1	Customer Login
Customer	1	(1:1)	*	Order
Customer	*	(1:1)	*	Order Tracking
Order	1	(1:1)	*	Transaction
Order	1	(1:*)	*	Order Tracking
Order	1	(1:*)	1	Inventory
Seller	1	(1:*)	*	Order
Seller	1	(1:*)	1	Product
Product	1	(1:*)	1	Inventory
Order	1	(1:*)	1	Seller
Order	1	(1:*)	1	Customer
Transaction	1	(1:1)	1	Order

Data Requirements ERD



Transactional requirements ERD



3. Conclusion

In this phase, we have developed a conceptual Entity-Relationship (ER) model for the E-Commerce Management System. This model provides a clear visualization of the entities, attributes, and relationships within the system, laying the groundwork for database design in the subsequent phases of development. By establishing a solid foundation through conceptual design, we ensure that the final system meets the data requirements and transactional needs outlined in the initial phase.

Phase 3: Relational Schema, Normalization, Final Logical Model

1. Introduction

In this phase of the project, we continue our work on the Database System focusing on the logical design. The aim is to convert the Enhanced Entity-Relationship (EER) model developed in Phase 2 into a relational schema, normalize it to ensure compliance with Boyce-Codd Normal Form (BCNF), and finally present the final Entity-Relationship Diagram (ERD) depicting the logical model.

2. Relational Schema

The Relational Schema is derived from the EER model established in Phase 2. Each entity in the EER model corresponds to a relation in the relational schema, with attributes becoming the attributes of the relation. Below is the relational schema with specified primary keys, foreign keys, and candidate keys:

Customer entity:

Customer ID (PK)	Name	Email	Passowrd	Shipping Address
---------------------	------	-------	----------	------------------

Order:

Order ID (PK)	Customer ID(FK)	ProductID (FK)	SellerID (FK)	Total Cost	Shipping Details	Order status
------------------	--------------------	-------------------	------------------	---------------	---------------------	-----------------

Seller:

SellerID(PK)	Name	Email	Password	ContactNumber	CompanyName	CompanyAddress
--------------	------	-------	----------	---------------	-------------	----------------

Transaction:

Transaction ID (PK)	Order ID (FK)	Payement Method	Amount	Transaction status
------------------------	------------------	--------------------	--------	-----------------------

Product entity:

Product ID (PK)	Product Name	Description	Price	Stock Level	Category	Brand	Size	Color
--------------------	-----------------	-------------	-------	----------------	----------	-------	------	-------

Inventory:

SellerID (PK, FK)	ProductID (PK, FK)	StockLevel
-------------------	--------------------	------------

CustomerInteraction:

InteractionID (PK)	CustomerID (FK)	SellerID (FK)	InteractionType	InteractionDate	InteractionDetails
-----------------------	--------------------	------------------	-----------------	-----------------	--------------------

Analytics:

AnalyticsID (PK)	SellerID (FK)	Date	UserActivity	SalesData	WebsitePerformanceMetrics
---------------------	------------------	------	--------------	-----------	---------------------------

Seller Interaction:

InteractionID (PK)	SellerID (FK)	InteractionType	InteractionDate	InteractionDetails
-----------------------	------------------	-----------------	-----------------	--------------------

Here's an explanation of the relationships between the tables:

- A Customer can place many Orders. (One-to-Many relationship between Customer and Order tables; CustomerID is a foreign key in the Order table)
- A Seller can have many Orders and many entries in Inventory. (One-to-Many relationship between Seller and Order and Inventory tables; SellerID is a foreign key in both Order and Inventory tables)
- A Product can be in the Inventory of many Sellers and can be part of many Orders. (Many-to-Many relationship between Product and Seller tables resolved with the Inventory table; ProductID is a foreign key in both Inventory and Order tables; SellerID is a foreign key in Inventory table)
- An Order is placed by one Customer and involves one Seller. (Many-to-One relationship between Customer and Order tables, and Many-to-One relationship between Seller and Order tables; CustomerID and SellerID are foreign keys in the Order table)
- An Order can have one Transaction. (One-to-One relationship between Order and Transaction tables; OrderID is a foreign key in the Transaction table)
- A Customer can have many CustomerInteractions with a Seller. (Many-to-Many relationship between Customer and Seller tables resolved with the CustomerInteraction table; CustomerID and SellerID are foreign keys in the CustomerInteraction table)
- A Seller can have many SellerInteractions with a Customer. (Many-to-Many relationship between Customer and Seller tables resolved with the CustomerInteraction table; refer to the explanation above)
- An Analytics record belongs to one Seller. (One-to-Many relationship between Seller and Analytics tables; SellerID is a foreign key in the Analytics table)

- Transactional requirements:

Customer:

Customer ID (PK)	Name	Email	Passowrd	Shipping Address
------------------	------	-------	----------	------------------

Seller:

SellerID(PK)	Name	Email	Password	ContactNumber	CompanyName	CompanyAddress
--------------	------	-------	----------	---------------	-------------	----------------

Product:

Product ID (PK)	Product Name	Description	Price	Stock Level	Category	Brand	Size	Color
-----------------	--------------	-------------	-------	-------------	----------	-------	------	-------

Inventory:

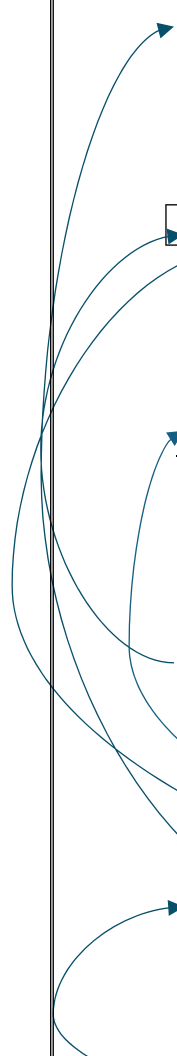
SellerID (PK, FK)	ProductID (PK, FK)	StockLevel
-------------------	--------------------	------------

Order:

Order ID (PK)	Customer ID(FK)	ProductID (FK)	SellerID (FK)	Total Cost	Shipping Details	Order status
---------------	-----------------	----------------	---------------	------------	------------------	--------------

Transaction:

Transaction ID (PK)	Order ID (FK)	Payement Method	Amount	Transaction status
---------------------	---------------	-----------------	--------	--------------------



Here's an explanation of the relationships between the tables:

- Customer and Order: A Customer can place many Orders (One-to-Many). CustomerID is a foreign key in the Order table.
- Seller: A Seller can have many Orders and many Inventory entries (One-to-Many). SellerID is a foreign key in both Order and Inventory tables.
- Product and Order: A Product can be in the Inventory of many Sellers and can be part of many Orders (Many-to-Many).
 - ProductID is a foreign key in both Inventory and Order tables.
 - SellerID is a foreign key in the Inventory table.
- Order: An Order is placed by one Customer and involves one Seller (Many-to-One). CustomerID and SellerID are foreign keys in the Order table.
- Order and Transaction: An Order can have one Transaction (One-to-One). OrderID is a foreign key in the Transaction table.

These relationships ensure data consistency and allow for efficient retrieval of information. For instance, you can easily find all Orders placed by a specific Customer or view the current stock level of a product managed by a particular Seller.

3. Normalization

Normalization ensures that the relational schema is in BCNF, which eliminates redundancy and anomalies in the database design. We performed normalization on the relational schema derived from the EER model to achieve BCNF.

To normalize the data into 3NF (Third Normal Form), we need to ensure that:

1. There are no repeating groups in the data.
2. Data is stored in separate tables based on the functional dependencies.

From the relational schema given above, let's normalize it step by step:

Step 1: Identify functional dependencies.

- Customer ID → Name, Email, Password, Shipping Address
- Order ID → Customer ID, Product ID, Seller ID, Total Cost, Shipping Details, Order status.
- Seller ID → Name, Email, Password, Contact Number, Company Name, Company Address
- Order ID → Transaction ID, Payment Method, Amount, Transaction status.

- Product ID → Product Name, Description, Price, Stock Level, Category, Brand, Size, Color
- (Seller ID, Product ID) → Stock Level
- Interaction ID → Customer ID, Seller ID, Interaction Type, Interaction Date, Interaction Details
- Analytics ID → Seller ID, Date, User Activity, Sales Data, Website Performance Metrics

Step 2: Remove partial dependencies (2NF)

- There are no partial dependencies in this schema.

Step 3: Remove transitive dependencies (3NF)

- Order ID → Customer ID, Total Cost, Shipping Details, Order status (Customer ID is transitively dependent on Order ID)
- To remove this transitive dependency, we need to create a new table for orders and remove Customer ID from the Order table.

Normalized Tables:

1. Customer

CustomerID [PK]	Name	Email	Password	Shipping Address
--------------------	------	-------	----------	------------------

2. Order

OrderID [PK]	ProductID [FK]	SellerID [FK]	Total Cost	Shipping Details	Order status
-----------------	-------------------	------------------	------------	------------------	--------------

3. Seller

SellerID [PK]	Name	Email	Password	Contact Number	Company Name	Company Address
------------------	------	-------	----------	----------------	--------------	-----------------

4. Transaction

TransactionID [PK]	OrderID [FK]	Payment Method	Amount	Transaction status
-----------------------	--------------	----------------	--------	--------------------

5. Product

ProductID [PK]	Product Name	Description	Price	Stock Level	Category	Brand	Size	Color)
-------------------	--------------	-------------	-------	-------------	----------	-------	------	--------

6. Inventory

SellerID [PK,FK]	ProductID [P,FK]	Stock Level)
------------------	------------------	--------------

7. CustomerInteraction

InteractionID [PK]	CustomerID [FK]	SellerID [FK]	Interaction Type	Interaction Date	Interaction Details)
--------------------	-----------------	---------------	------------------	------------------	----------------------

8. Analytics

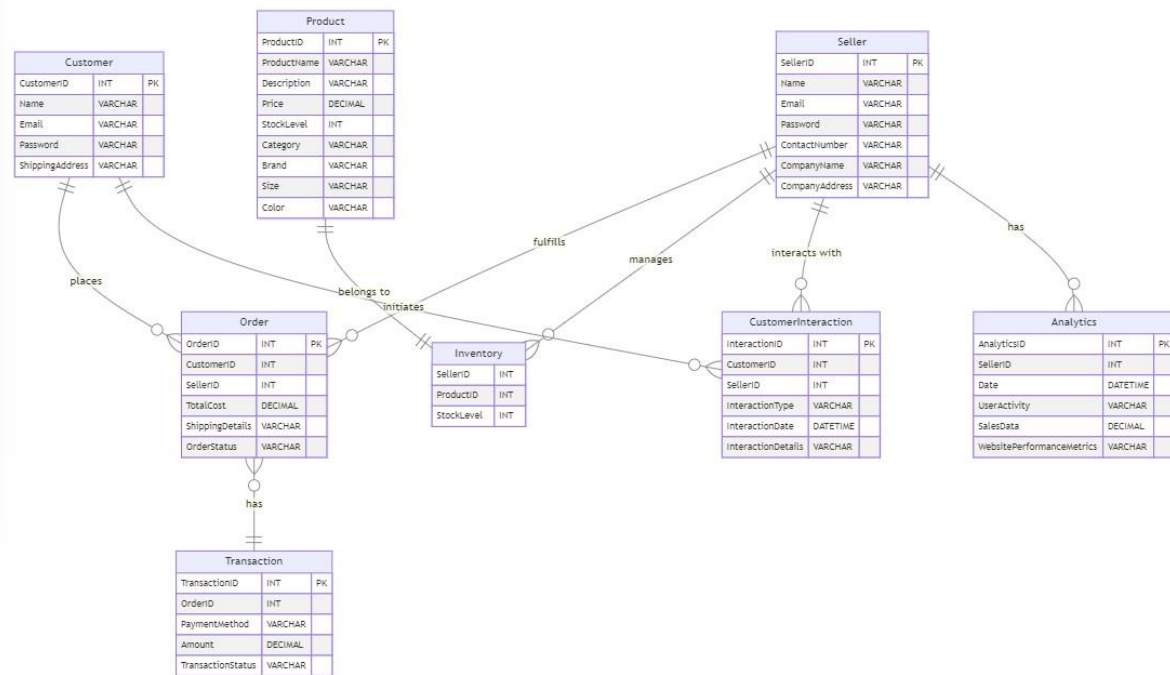
AnalyticsID [PK]	SellerID [FK]	Date	User Activity	Sales Data	Website Performance Metrics)
------------------	---------------	------	---------------	------------	------------------------------

These tables are now in 3NF, with each table containing only attributes that are functionally dependent on the primary key.

4. Final Logical Model

The Final Logical Model represents the Entity-Relationship Diagram (ERD) derived from the normalized relational schema. It depicts the relationships between entities and attributes in the database system, presenting a clear and comprehensive view of the logical structure.

Final Logical ERD:



5. Conclusion

In this phase, we successfully transformed the EER model into a relational schema, ensuring compliance with BCNF through normalization. The final logical model, depicted in the ERD, provides a solid foundation for the implementation phase of the Database System project.

Phase 4: Implementation

1. Database Creation and Schema Implementation

We have successfully translated our Entity-Relationship model (EER) into a relational schema, and now we are implementing it in our chosen database management system using SQL. The database system is designed to meet the requirements specified in the previous phases, incorporating all necessary constraints and business rules. Below are the details of the database creation and schema implementation:

1.1 Database Creation

We have created a new database named `ecommerce_clothing` to host our system's tables and data. This database will serve as the foundation for our application, facilitating efficient data management and retrieval.

1.2 Table Definitions

We have implemented the following tables within our database, each representing a specific entity or relationship in our system:

Table 1: Customer

- Description: Stores information about customers registered in the system.
 - Attributes:
 - CustomerID: INT (Primary Key)
 - Name: VARCHAR
 - Email: VARCHAR
 - Password: VARCHAR
 - ContactNumber: VARCHAR
 - ShippingAddress: VARCHAR
 - Constraints:
 - CustomerID: PRIMARY KEY

Table 2: Seller

- Description: Stores information about sellers registered in the system.
 - Attributes:
 - SellerID: INT (Primary Key)
 - Name: VARCHAR
 - Email: VARCHAR
 - Password: VARCHAR
 - ContactNumber: VARCHAR
 - CompanyName: VARCHAR
 - CompanyAddress: VARCHAR
 - Constraints:
 - SellerID: PRIMARY KEY

Table 3: Product

- Description: Stores information about products available in the system.

- Attributes:
 - ProductID: INT (Primary Key)
 - Name: VARCHAR
 - Description: VARCHAR
 - Price: DECIMAL
 - StockLevel: INT
 - Category: VARCHAR
 - Brand: VARCHAR
 - Size: VARCHAR
 - Color: VARCHAR
- Constraints:
 - ProductID: PRIMARY KEY

Table 4: Order

- Description: Stores information about orders placed by customers.
 - Attributes:
 - OrderID: INT (Primary Key)
 - CustomerID: INT (Foreign Key)
 - SellerID: INT (Foreign Key)
 - OrderDate: DATE
 - TotalCost: DECIMAL
 - ShippingDetails: VARCHAR
 - OrderStatus: VARCHAR
 - Constraints:
 - OrderID: PRIMARY KEY
 - CustomerID: FOREIGN KEY
 - SellerID: FOREIGN KEY

Table 5: Inventory

- Description: Stores information about product inventory managed by sellers.
 - Attributes:
 - ProductID: INT (Primary Key, Foreign Key)
 - SellerID: INT (Primary Key, Foreign Key)
 - StockLevel: INT
 - Constraints:
 - (ProductID, SellerID): PRIMARY KEY
 - ProductID: FOREIGN KEY
 - SellerID: FOREIGN KEY

Table 6: Transactional

- Description: Stores information about transactions related to orders.
 - Attributes:
 - TransactionID: INT (Primary Key)
 - OrderID: INT (Foreign Key)
 - PaymentMethod: VARCHAR
 - Amount: DECIMAL
 - TransactionStatus: VARCHAR
 - Constraints:
 - TransactionID: PRIMARY KEY
 - OrderID: FOREIGN KEY

1.3 Data Population

We have populated the tables with sample data to ensure the functionality and integrity of the database system. This sample data will be used for testing and validation purposes during the development phase.

Customer table:

Instance 3306 - Warning - not supported

Query 1

Limit to 1000 rows

230 ('2024-05-10', 'Logged in, viewed product Y, added product Z to cart, checked out', 'Total sales: \$15000', 'Average page load
231
232 show tables;
233
234 select * from Customer;
235
236

100% 1:235

Result Grid

Filter Rows: Search Edit: Export/Import:

CustomerID	Name	Email	Password	ContactNumb...	ShippingAddress
1	John Doe	john@gmail.com	password123	1234567890	123 Main St, City, Country
2	Alice Smith	alicesm2@yahoo.com	securepass	9876543210	456 Oak Ave, Town, Country
3	Bob Johnson	bobby@gmail.com	bobpass	5551234567	789 Elm St, Village, Country
4	Emma Brown	emma@gmail.com	emmapass	7778899999	321 Pine Rd, Hamlet, Country
5	Michael Davis	michael@yahoo.com	michaelpass	1112223333	654 Cedar St, Suburb, Country
6	Sarah Wilson	sarah@gmail.com	sarahpass	4445556666	987 Birch Ave, City, Country
7	David Lee	david@gmail.com	davidpass	9990001111	147 Willow Rd, Town, Country
8	Jennifer Taylor	jennifer@gmail.com	jenniferpas	3334445555	258 Maple St, Village, Country
9	Chris Evans	chris@gmail.com	chrispass	6667778888	369 Oak Ln, Hamlet, Country
10	Laura Martinez	laura@yahoo.com	laurapass	8889990000	741 Elm Ave, Suburb, Country
100%	100%	100%	100%	100%	100%

Customer 4 Apply

Seller table:

Instance 3306 - Warning - not supported

Query 1

Limit to 1000 rows

232 show tables;
233
234 select * from Customer;
235
236 select * from Seller;
237
238

100% 1:237

Result Grid

Filter Rows: Search Edit: Export/Import:

SellerID	CompanyName	ContactName	Email	Password	ContactNumb...	CompanyAddress
1	ABC Clothing	John Smith	abc@example.com	abcpass	1234567890	123 Main St, City, Country
2	Chinatown Fashion	Alice Johnson	xyz@example.com	xyzpass	9876543210	456 Oak Ave, Town, Country
3	Fashion World	Bob Brown	fashion@example.com	fashionpass	5551234567	789 Elm St, Village, Country
4	Trendy Apparel	Emma Davis	trendy@example.com	trendypass	7778899999	321 Pine Rd, Hamlet, Country
5	StyleMart	Michael Wilson	style@example.com	stylepass	1112223333	654 Cedar St, Suburb, Country
6	Chic Boutique	Sarah Lee	chic@example.com	chicpass	4445556666	987 Birch Ave, City, Country
100%	100%	100%	100%	100%	100%	100%

Seller 6 Apply

Product table:

Instance 3306 - Warning - not supported

Query 1

Limit to 1000 rows

234 • select * from Customer;

235

236 • select * from Seller;

237

238 • select * from Product;

239

240

100% 1:239

ProductID	Name	Description	Price	StockLevel	Category	Brand	Size	Color
1	Classic White T-Shirt	A comfortable cotton t-shirt in classic white color.	19.99	50	Clothing	ABC Apparel	M	White
2	Slim Fit Blue Jeans	Stylish slim-fit jeans made from durable denim...	39.99	30	Clothing	XYZ Denim	30W x 32L	Blue
3	Sporty Sneakers	Trendy sneakers perfect for sports and casual w...	49.99	40	Footwear	Trendy Kicks	10	Black
4	Cozy Hooded Sweatshirt	Warm and comfortable hooded sweatshirt for ch...	29.99	20	Clothing	Fashion Co.	L	Gray
5	Classic Leather Belt	Genuine leather belt with a timeless design.	24.99	60	Accessories	StyleMart	One Size	Brown
6	Denim Jacket	Fashionable denim jacket with a modern fit.	59.99	25	Clothing	XYZ Denim	XL	Blue
7	Casual Plaid Shirt	Stylish plaid shirt suitable for everyday casual w...	34.99	35	Clothing	ABC Apparel	L	Red
8	Formal Dress Shirt	Elegant dress shirt suitable for formal occasions.	44.99	30	Clothing	Fashion Co.	M	White
9	Classic Leather Handbag	Chic leather handbag with multiple compartments.	79.99	15	Accessories	Chic Boutique	One Size	Black
10	Running Shoes	High-performance running shoes with advanced...	69.99	20	Footwear	Trendy Kicks	9	Gray
11	Retro Sunglasses	Vintage-style sunglasses with UV protection len...	29.99	45	Accessories	StyleMart	One Size	Black
12	Cozy Flannel Pajamas	Soft flannel pajamas for a comfortable night's sl...	49.99	10	Clothing	XYZ Sleepw...	XL	Plaid
13	Leather Jacket	Slim and durable leather jacket with multiple ca...	39.99	30	Accessories	Chic Boutique	One Size	Brown
14	Stylish Fedora Hat	Classic fedora hat made from high-quality wool f...	34.99	25	Accessories	StyleMart	One Size	Black
15	Formal Necktie	Silk necktie perfect for adding a touch of elegan...	19.99	50	Accessories	Fashion Co.	One Size	Navy
16	Slim Fit Chinos	Modern slim-fit chinos crafted from breathable c...	49.99	20	Clothing	ABC Apparel	34W x 32L	Khaki
17	Canvas Backpack	Sturdy canvas backpack with multiple pockets f...	59.99	15	Accessories	Chic Boutique	One Size	Gray
18	Cozy Knit Sweater	Warm and soft knit sweater perfect for layering l...	54.99	20	Clothing	XYZ Knitwear	S	Navy
19	Elegant Pearl Necklace	Timeless pearl necklace with a delicate silver ch...	89.99	10	Accessories	Chic Boutique	One Size	White
20	Classic Leather Jacket	Sophisticated leather jacket suitable for both ca...	79.99	15	Footwear	StyleMart	11	Brown

Product 6

Apply

Inventory table:

Instance 3306 - Warning - not supported

Query 1

Limit to 1000 rows

239

240

241 • select * from Inventory;

242

243

244

245

100% 25:241

ProductID	SellerID	StockLevel
1	1	100
2	2	80
3	3	120
4	4	50
5	5	70
6	6	90
7	1	60
8	2	40
9	3	110
10	4	30
11	5	80
12	6	20
13	1	70
14	2	90
15	3	40
16	4	60
17	5	100
18	6	25
19	1	45
20	2	75

_Order 7

Result 13

Customer 14

Seller 15

Product 16

Inventory 17

Result 18

Apply

Transaction table:

L80 xDatabase Server Tools Scripting Help

SQL File 2" xRihab Muhammad Asif (rmasif@dah.edu.sa) is signed in

Limit to 1000 rows

-- Inserting information into the Transaction table

INSERT INTO _Transaction (OrderID, PaymentMethod, Amount, TransactionStatus) VALUES

(1, 'Credit Card', 99.99, 'Completed'),

(2, 'PayPal', 149.99, 'Completed'),

(3, 'Cash on Delivery', 79.99, 'Processing'),

(4, 'Credit Card', 199.99, 'Completed'),

(5, 'Credit Card', 59.99, 'Completed'),

(6, 'PayPal', 129.99, 'Processing'),

(7, 'Credit Card', 89.99, 'Completed'),

(8, 'PayPal', 179.99, 'Processing'),

Result Grid

Filter Rows:

TransactionID OrderID PaymentMethod Amount TransactionStatus

51 1 Credit Card 99.99 Completed

52 2 PayPal 149.99 Completed

53 3 Cash on Delivery 79.99 Processing

54 4 Credit Card 199.99 Completed

55 5 Credit Card 59.99 Completed

56 6 PayPal 129.99 Processing

57 7 Credit Card 89.99 Completed

58 8 PayPal 179.99 Processing

Customer 2 _Transaction 3 x

Apply Revert

CustomerInteraction table:

L80 xDatabase Server Tools Scripting Help

SQL File 2" x

Limit to 1000 rows

('2024-05-09', 'Logged in, viewed product V, added product W to cart, viewed product X', 'Total sales: \$12500', 'Average page load time: 2.7 seconds,

('2024-05-10', 'Logged in, viewed product Y, added product Z to cart, checked out', 'Total sales: \$15000', 'Average page load time: 2.4 seconds, Conve

select * from Customer;

select * from _Transaction;

select * from _Order;

Result Grid

Filter Rows:

InteractionID CustomerID InteractionType InteractionDate InteractionDetails

1 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

11 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

21 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

31 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

41 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

51 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

61 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

71 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

81 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

91 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

101 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

111 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

121 1 Feedback 2024-05-15 10:00:00 Provided positive feedback on recent purchase.

131 1 Support 2024-05-25 18:20:00 Asked for assistance with account login.

Order 2 _Order 3 Seller 4 Inventory 5 CustomerInteraction 6 SellerInteraction 7 Analytics 8 Product 9 CustomerInteraction 10 x Result 11 Result 12 Apply Revert

SellerInteraction table:

InteractionID	SellerID	InteractionType	InteractionDate	InteractionDetails
1	1	Feedback	2024-05-15 10:00:00	Received positive feedback from a customer.
2	2	Support	2024-05-16 11:30:00	Assisted a customer with product inquiries.
3	3	Review	2024-05-17 12:45:00	Received a 5-star review for a product.
4	4	Feedback	2024-05-18 09:15:00	Addressed a customer complaint.
5	5	Support	2024-05-19 08:00:00	Provided guidance on order fulfillment.
6	6	Feedback	2024-05-20 13:20:00	Suggested improvements for website navigation.
7	1	Support	2024-05-21 14:30:00	Resolved a customer's issue with their order.
8	2	Feedback	2024-05-22 15:45:00	Received suggestions for new product ideas.
9	3	Review	2024-05-23 16:00:00	Received a detailed review with photos.
10	4	Feedback	2024-05-24 17:10:00	Addressed concerns about shipping times.
11	5	Support	2024-05-25 18:20:00	Assisted with returns and exchanges.
12	6	Feedback	2024-05-26 19:30:00	Received feedback on product packaging.
13	1	Support	2024-05-27 20:45:00	Provided technical support for a product.
14	2	Feedback	2024-05-28 21:00:00	Received feedback on customer service experience.

Analytics:

AnalyticsID	Date	UserActivity	SalesData	WebsitePerformanceMetrics
1	2024-05-01	Logged in, viewed product A, added product B to cart	Total sales: \$10000	Average page load time: 2.5 seconds, Conversion rate: 3.2%
2	2024-05-02	Logged in, viewed product C, added product D to cart	Total sales: \$12000	Average page load time: 2.6 seconds, Conversion rate: 3.5%
3	2024-05-03	Logged in, searched for product E, viewed product F	Total sales: \$9500	Average page load time: 2.7 seconds, Conversion rate: 3.1%
4	2024-05-04	Logged in, viewed product G, added product H to cart	Total sales: \$11000	Average page load time: 2.4 seconds, Conversion rate: 3.3%
5	2024-05-05	Logged in, viewed product I, added product J to cart	Total sales: \$10500	Average page load time: 2.8 seconds, Conversion rate: 3.0%
6	2024-05-06	Logged in, viewed product K, added product L to cart	Total sales: \$13000	Average page load time: 2.3 seconds, Conversion rate: 3.6%
7	2024-05-07	Logged in, viewed product M, added product N to cart	Total sales: \$11500	Average page load time: 2.6 seconds, Conversion rate: 3.2%
8	2024-05-08	Logged in, viewed product O, added product P to cart	Total sales: \$14000	Average page load time: 2.5 seconds, Conversion rate: 3.4%
9	2024-05-09	Logged in, viewed product Q, added product R to cart	Total sales: \$12500	Average page load time: 2.7 seconds, Conversion rate: 3.1%
10	2024-05-10	Logged in, viewed product S, added product T to cart	Total sales: \$13500	Average page load time: 2.4 seconds, Conversion rate: 3.3%
11	2024-05-01	Logged in, viewed product A, added product B to cart	Total sales: \$10000	Average page load time: 2.5 seconds, Conversion rate: 3.2%
12	2024-05-02	Logged in, viewed product C, added product D to cart	Total sales: \$12000	Average page load time: 2.6 seconds, Conversion rate: 3.5%
13	2024-05-03	Logged in, searched for product E, viewed product F	Total sales: \$9500	Average page load time: 2.7 seconds, Conversion rate: 3.1%
14	2024-05-04	Logged in, viewed product G, added product H to cart	Total sales: \$11000	Average page load time: 2.4 seconds, Conversion rate: 3.3%

2. Transaction Requirements

Our system supports various transactions to enable users to interact with the database effectively. These transactions are designed to fulfill specific business needs and ensure data consistency and integrity. Below are the transaction requirements identified for our system:

2.1 Retrieve Products with Stock Level Greater Than 0

Description: Retrieve all products from the database where the stock level is greater than 0.

SQL Code: `SELECT * FROM Product WHERE StockLevel > 0;`

Outcome:

SQL File 2'

```

256
257 * SELECT * FROM Product WHERE StockLevel = 0;
258 * SELECT * FROM CustomerInteraction WHERE CustomerID = 1;
259 * SELECT COUNT(*) AS TotalOrders FROM _Order;
260 * SELECT AVG(TotalCost) AS AverageTotalCost FROM _Order;

```

ProductID	Name	Description	Price	StockLevel	Category	Brand	Size	Color
5	Classic Leather Belt	Genuine leather belt with a timeless design.	24.99	60	Accessories	StyleMart	One Size	Brown
6	Denim Jacket	Fashionable denim jacket with a modern fit.	59.99	25	Clothing	XYZ Denim	XL	Blue
7	Casual Plaid Shirt	Stylish plaid shirt suitable for everyday casual wear.	34.99	35	Clothing	ABC Apparel	L	Red
8	Formal Dress Shirt	Elegant dress shirt suitable for formal occasions.	44.99	30	Clothing	Fashion Co.	M	White
9	Classic Leather Handbag	Chic leather handbag with multiple compartments.	79.99	15	Accessories	Chic Boutique	One Size	Black
10	Running Shoes	High-performance running shoes with advanced cushioning.	69.99	20	Footwear	Trendy Kids	9	Gray
11	Retro Sunglasses	Vintage-style sunglasses with UV protection lenses.	29.99	45	Accessories	StyleMart	One Size	Black
12	Cozy Flannel Pajamas	Soft flannel pajamas for a comfortable night's sleep.	49.99	10	Clothing	XYZ Sleepwear	XL	Plaid
13	Leather Jacket	Slim and durable leather jacket with multiple pockets.	39.99	30	Accessories	Chic Boutique	One Size	Brown
14	Stylish Fedora Hat	Classic fedora hat made from high-quality wool.	34.99	25	Accessories	StyleMart	One Size	Black
15	Formal Necktie	Silk necktie perfect for adding a touch of elegance.	19.99	50	Accessories	Fashion Co.	One Size	Navy
16	Slim Fit Chinos	Modern slim-fit chinos crafted from breathable cotton.	49.99	20	Clothing	ABC Apparel	34W x 32L	Khaki
17	Canvas Backpack	Durable canvas backpack with multiple pockets and straps.	59.99	15	Accessories	Chic Boutique	One Size	Gray
18	Cozy Knit Sweater	Warm and soft knit sweater perfect for layering.	54.99	20	Clothing	XYZ Knitwear	S	Navy
19	Elegant Pearl Necklace	Timeless pearl necklace with a delicate silver chain.	89.99	10	Accessories	Chic Boutique	One Size	White

2.2 Retrieve Customer Interactions for a Specific Customer

Description: Retrieve all customer interactions for a specific customer identified by their CustomerID.

SQL Query: `SELECT * FROM CustomerInteraction WHERE CustomerID = 1;`

Outcome:

SQL File 2'

```

256
257 * SELECT * FROM Product WHERE StockLevel = 0;
258 * SELECT * FROM CustomerInteraction WHERE CustomerID = 1;
259 * SELECT COUNT(*) AS TotalOrders FROM _Order;
260 * SELECT AVG(TotalCost) AS AverageTotalCost FROM _Order;

```

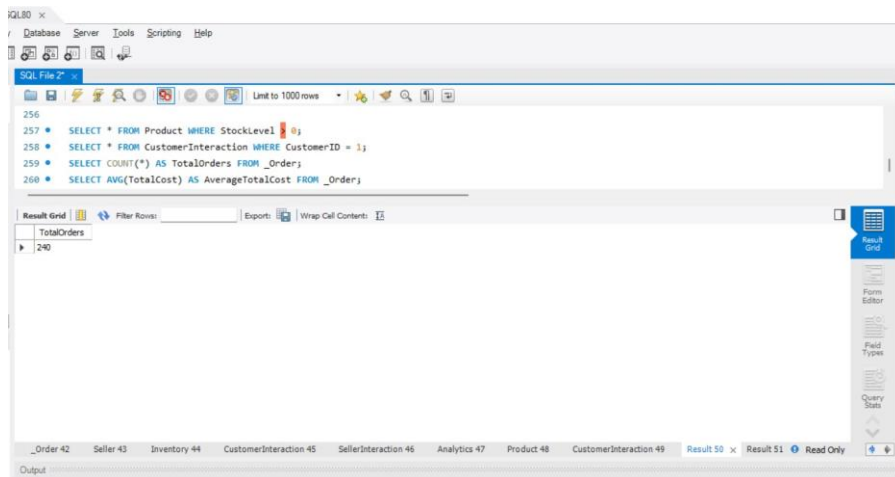
InteractionID	CustomerID	InteractionType	InteractionDate	InteractionDetails
1	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
11	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
21	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
31	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
41	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
51	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
61	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
71	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
81	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
91	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
101	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
111	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
121	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.
131	1	Support	2024-05-25 18:20:00	Asked for assistance with account login.
141	1	Feedback	2024-05-15 10:00:00	Provided positive feedback on recent purchase.

2.3 Count Total Orders

Description: Count the total number of orders in the system.

SQL Query: `SELECT COUNT(*) AS TotalOrders FROM _Order;`

Outcome:

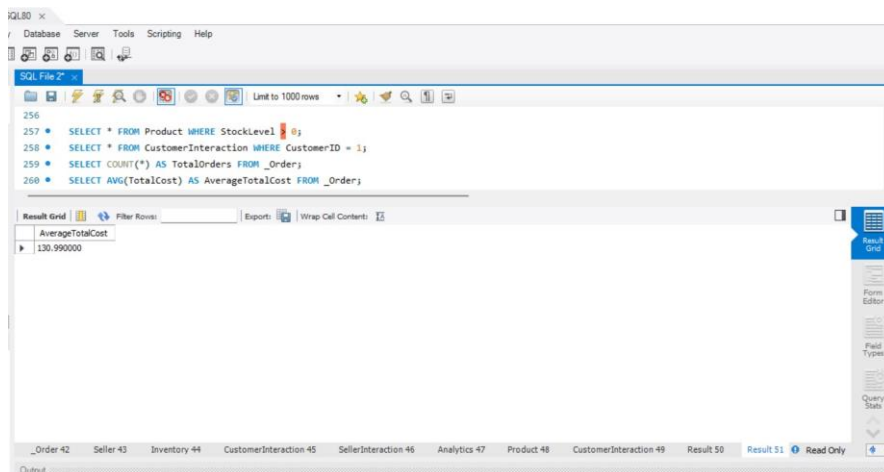


2.4 Calculate Average Total Cost of Orders

Description: Calculate the average total cost of all orders in the system.

SQL Query: `SELECT AVG(TotalCost) AS AverageTotalCost FROM _Order;`

Outcome:



2.5 Retrieve Order Details with Customer Information

Description: Retrieve order details along with customer information for each order.

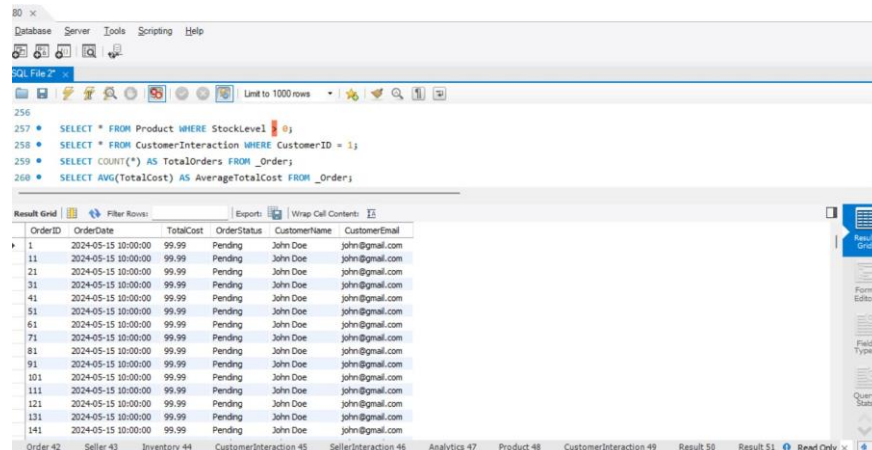
SQL Query:

```
SELECT _Order.OrderID, _Order.OrderDate, _Order.TotalCost, _Order.OrderStatus,  
Customer.Name AS CustomerName, Customer.Email AS CustomerEmail
```

```
FROM _Order
```

```
JOIN Customer ON _Order.CustomerID = Customer.CustomerID;
```

Outcome:



The screenshot shows a SQL IDE window with a query editor and a result grid. The query editor contains the following SQL code:

```
256  
257 * SELECT * FROM Product WHERE StockLevel < 0;  
258 * SELECT * FROM CustomerInteraction WHERE CustomerID = 1;  
259 * SELECT COUNT(*) AS TotalOrders FROM _Orders;  
260 * SELECT AVG(TotalCost) AS AverageTotalCost FROM _Orders;
```

The result grid displays the following data:

OrderID	OrderDate	TotalCost	OrderStatus	CustomerName	CustomerEmail
1	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
11	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
21	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
31	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
41	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
51	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
61	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
71	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
81	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
91	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
101	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
111	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
121	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
131	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com
141	2024-05-15 10:00:00	99.99	Pending	John Doe	john@gmail.com

MySQL Full Code:

```
create DATABASE ecommerce_clothing;
```

```
use ecommerce_clothing;
```

```
CREATE TABLE Customer (
```

```
    CustomerID INT AUTO_INCREMENT PRIMARY KEY,
```

```
    Name VARCHAR(255) NOT NULL,
```

```
    Email VARCHAR(255) UNIQUE NOT NULL,
```

```
    Password VARCHAR(255) NOT NULL,
```

```
    ContactNumber VARCHAR(20),
```

```
    ShippingAddress VARCHAR(255)
```

```
);
```

```
CREATE TABLE Seller (
```

```
    SellerID INT AUTO_INCREMENT PRIMARY KEY,
```

```
    CompanyName VARCHAR(255) NOT NULL,
```

```
    ContactName VARCHAR(255) NOT NULL,
```

```
    Email VARCHAR(255) UNIQUE NOT NULL,
```

```
    Password VARCHAR(255) NOT NULL,
    ContactNumber VARCHAR(20),
    CompanyAddress VARCHAR(255)
);

CREATE TABLE Product (
    ProductID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Description TEXT,
    Price DECIMAL(10, 2) NOT NULL,
    StockLevel INT NOT NULL,
    Category VARCHAR(255),
    Brand VARCHAR(255),
    Size VARCHAR(20),
    Color VARCHAR(50)
);

CREATE TABLE _Order (
    OrderID INT AUTO_INCREMENT PRIMARY KEY,
    CustomerID INT,
    SellerID INT,
    OrderDate DATETIME NOT NULL,
    TotalCost DECIMAL(10, 2) NOT NULL,
    ShippingDetails VARCHAR(255),
    OrderStatus VARCHAR(50) NOT NULL,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID),
    FOREIGN KEY (SellerID) REFERENCES Seller(SellerID)
);

CREATE TABLE _Transaction (
    TransactionID INT AUTO_INCREMENT PRIMARY KEY,
    OrderID INT UNIQUE,
    PaymentMethod VARCHAR(50) NOT NULL,
```

```
Amount DECIMAL(10, 2) NOT NULL,
TransactionStatus VARCHAR(50) NOT NULL,
FOREIGN KEY (OrderID) REFERENCES _Order(OrderID)
);
CREATE TABLE Inventory (
    ProductID INT,
    SellerID INT,
    StockLevel INT NOT NULL,
    PRIMARY KEY (ProductID, SellerID),
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID),
    FOREIGN KEY (SellerID) REFERENCES Seller(SellerID)
);
CREATE TABLE CustomerInteraction (
    InteractionID INT AUTO_INCREMENT PRIMARY KEY,
    CustomerID INT,
    InteractionType VARCHAR(50) NOT NULL,
    InteractionDate DATETIME NOT NULL,
    InteractionDetails TEXT,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)
);
CREATE TABLE SellerInteraction (
    InteractionID INT AUTO_INCREMENT PRIMARY KEY,
    SellerID INT,
    InteractionType VARCHAR(50) NOT NULL,
    InteractionDate DATETIME NOT NULL,
    InteractionDetails TEXT,
    FOREIGN KEY (SellerID) REFERENCES Seller(SellerID)
);
CREATE TABLE Analytics (
    AnalyticsID INT AUTO_INCREMENT PRIMARY KEY,
```

```
Date DATE NOT NULL,  
UserActivity TEXT,  
SalesData TEXT,  
WebsitePerformanceMetrics TEXT  
);
```

```
INSERT INTO Customer (Name, Email, Password, ContactNumber, ShippingAddress) VALUES  
(  
  ('John Doe', 'john@gmail.com', 'password123', '1234567890', '123 Main St, City, Country'),  
  ('Alice Smith', 'alicesm2@yahoo.com', 'securepass', '9876543210', '456 Oak Ave, Town,  
Country'),  
  ('Bob Johnson', 'bobby@gmail.com', 'bobpass', '5551234567', '789 Elm St, Village, Country'),  
  ('Emma Brown', 'emma@gmail.com.com', 'emmapass', '7778889999', '321 Pine Rd, Hamlet,  
Country'),  
  ('Michael Davis', 'michael@yahoo.com', 'michaelpass', '1112223333', '654 Cedar St, Suburb,  
Country'),  
  ('Sarah Wilson', 'sarah@gmail.com', 'sarahpass', '4445556666', '987 Birch Ave, City, Country'),  
  ('David Lee', 'david@gmail.com', 'davidpass', '9990001111', '147 Willow Rd, Town, Country'),  
  ('Jennifer Taylor', 'jennifer@gmail.com', 'jenniferpass', '3334445555', '258 Maple St, Village,  
Country'),  
  ('Chris Evans', 'chris@gmail.com', 'chrisspass', '6667778888', '369 Oak Ln, Hamlet, Country'),  
  ('Laura Martinez', 'laura@yahoo.com', 'laurapass', '8889990000', '741 Elm Ave, Suburb, Country');
```

```
INSERT INTO Seller (CompanyName, ContactName, Email, Password, ContactNumber,  
CompanyAddress) VALUES  
(  
  ('ABC Clothing', 'John Smith', 'abc@example.com', 'abcpass', '1234567890', '123 Main St, City,  
Country'),  
  ('chintown Fashion', 'Alice Johnson', 'xyz@example.com', 'xyzpass', '9876543210', '456 Oak Ave,  
Town, Country'),  
  ('Fashion World', 'Bob Brown', 'fashion@example.com', 'fashionpass', '5551234567', '789 Elm St,  
Village, Country'),  
  ('Trendy Apparel', 'Emma Davis', 'trendy@example.com', 'trendypass', '7778889999', '321 Pine  
Rd, Hamlet, Country'),
```

('StyleMart', 'Michael Wilson', 'style@example.com', 'stylepass', '1112223333', '654 Cedar St, Suburb, Country'),

('Chica Boutique', 'Sarah Lee', 'chic@example.com', 'chicpass', '4445556666', '987 Birch Ave, City, Country');

-- Inserting information into the Product table

INSERT INTO Product (Name, Description, Price, StockLevel, Category, Brand, Size, Color)
VALUES

('Classic White T-Shirt', 'A comfortable cotton t-shirt in classic white color.', 19.99, 50, 'Clothing', 'ABC Apparel', 'M', 'White'),

('Slim Fit Blue Jeans', 'Stylish slim-fit jeans made from durable denim material.', 39.99, 30, 'Clothing', 'XYZ Denim', '32W x 32L', 'Blue'),

('Sporty Sneakers', 'Trendy sneakers perfect for sports and casual wear.', 49.99, 40, 'Footwear', 'Trendy Kicks', '10', 'Black'),

('Cozy Hooded Sweatshirt', 'Warm and comfortable hooded sweatshirt for chilly days.', 29.99, 20, 'Clothing', 'Fashion Co.', 'L', 'Gray'),

('Classic Leather Belt', 'Genuine leather belt with a timeless design.', 24.99, 60, 'Accessories', 'StyleMart', 'One Size', 'Brown'),

('Denim Jacket', 'Fashionable denim jacket with a modern fit.', 59.99, 25, 'Clothing', 'XYZ Denim', 'XL', 'Blue'),

('Casual Plaid Shirt', 'Stylish plaid shirt suitable for everyday casual wear.', 34.99, 35, 'Clothing', 'ABC Apparel', 'L', 'Red'),

('Formal Dress Shirt', 'Elegant dress shirt suitable for formal occasions.', 44.99, 30, 'Clothing', 'Fashion Co.', 'M', 'White'),

('Classic Leather Handbag', 'Chic leather handbag with multiple compartments.', 79.99, 15, 'Accessories', 'Chic Boutique', 'One Size', 'Black'),

('Running Shoes', 'High-performance running shoes with advanced cushioning.', 69.99, 20, 'Footwear', 'Trendy Kicks', '9', 'Gray'),

('Retro Sunglasses', 'Vintage-style sunglasses with UV protection lenses.', 29.99, 45, 'Accessories', 'StyleMart', 'One Size', 'Black'),

('Cozy Flannel Pajamas', 'Soft flannel pajamas for a comfortable night's sleep.', 49.99, 10, 'Clothing', 'XYZ Sleepwear', 'XL', 'Plaid'),

('Leather jacket', 'Slim and durable leather jacket with multiple card slots.', 39.99, 30, 'Accessories', 'Chic Boutique', 'One Size', 'Brown'),

('Stylish Fedora Hat', 'Classic fedora hat made from high-quality wool felt.', 34.99, 25, 'Accessories', 'StyleMart', 'One Size', 'Black'),

('Formal Necktie', 'Silk necktie perfect for adding a touch of elegance to any outfit.', 19.99, 50, 'Accessories', 'Fashion Co.', 'One Size', 'Navy'),

('Slim Fit Chinos', 'Modern slim-fit chinos crafted from breathable cotton.', 49.99, 20, 'Clothing', 'ABC Apparel', '34W x 32L', 'Khaki'),

('Canvas Backpack', 'Sturdy canvas backpack with multiple pockets for organization.', 59.99, 15, 'Accessories', 'Chic Boutique', 'One Size', 'Gray'),

('Cozy Knit Sweater', 'Warm and soft knit sweater perfect for layering in colder weather.', 54.99, 20, 'Clothing', 'XYZ Knitwear', 'S', 'Navy'),

('Elegant Pearl Necklace', 'Timeless pearl necklace with a delicate silver chain.', 89.99, 10, 'Accessories', 'Chic Boutique', 'One Size', 'White'),

('Classic Leather jacket', 'Sophisticated leather jacket suitable for both casual and formal occasions.', 79.99, 15, 'Footwear', 'StyleMart', '11', 'Brown');

-- Inserting information into the _Order table

```
INSERT INTO _Order (CustomerID, SellerID, OrderDate, TotalCost, ShippingDetails, OrderStatus) VALUES
```

(1, 1, '2024-05-15 10:00:00', 99.99, '123 Main St, City, Country', 'Pending'),

(2, 2, '2024-05-16 11:30:00', 149.99, '456 Oak Ave, Town, Country', 'Processing'),

(3, 3, '2024-05-17 12:45:00', 79.99, '789 Elm St, Village, Country', 'Shipped'),

(4, 4, '2024-05-18 09:15:00', 199.99, '321 Pine Rd, Hamlet, Country', 'Delivered'),

(5, 5, '2024-05-19 08:00:00', 59.99, '654 Cedar St, Suburb, Country', 'Pending'),

(6, 6, '2024-05-20 13:20:00', 129.99, '987 Birch Ave, City, Country', 'Processing'),

(7, 1, '2024-05-21 14:30:00', 89.99, '123 Main St, City, Country', 'Shipped'),

(8, 2, '2024-05-22 15:45:00', 179.99, '456 Oak Ave, Town, Country', 'Delivered'),

(9, 3, '2024-05-23 16:00:00', 69.99, '789 Elm St, Village, Country', 'Pending'),

(10, 4, '2024-05-24 17:10:00', 249.99, '321 Pine Rd, Hamlet, Country', 'Processing');

-- Inserting information into the Transaction table

```
INSERT INTO _Transaction (OrderID, PaymentMethod, Amount, TransactionStatus) VALUES
```

(1, 'Credit Card', 99.99, 'Completed'),

(2, 'PayPal', 149.99, 'Completed'),
(3, 'Cash on Delivery', 79.99, 'Processing'),
(4, 'Credit Card', 199.99, 'Completed'),
(5, 'Credit Card', 59.99, 'Completed'),
(6, 'PayPal', 129.99, 'Processing'),
(7, 'Credit Card', 89.99, 'Completed'),
(8, 'PayPal', 179.99, 'Processing'),
(9, 'Credit Card', 69.99, 'Completed'),
(10, 'Cash on Delivery', 249.99, 'Processing'),
(11, 'Credit Card', 39.99, 'Completed'),
(12, 'PayPal', 149.99, 'Completed'),
(13, 'Credit Card', 109.99, 'Processing'),
(14, 'Credit Card', 199.99, 'Completed'),
(15, 'PayPal', 59.99, 'Completed'),
(16, 'Credit Card', 279.99, 'Processing'),
(17, 'Credit Card', 79.99, 'Completed'),
(18, 'PayPal', 169.99, 'Processing'),
(19, 'Credit Card', 129.99, 'Completed'),
(20, 'Credit Card', 219.99, 'Completed'),
(21, 'PayPal', 99.99, 'Processing'),
(22, 'Credit Card', 299.99, 'Completed'),
(23, 'Credit Card', 99.99, 'Processing'),
(24, 'PayPal', 189.99, 'Completed'),
(25, 'Credit Card', 149.99, 'Completed');

-- Inserting information into the Inventory table

INSERT INTO Inventory (ProductID, SellerID, StockLevel) VALUES

(1, 1, 100),
(2, 2, 80),
(3, 3, 120),

(4, 4, 50),
(5, 5, 70),
(6, 6, 90),
(7, 1, 60),
(8, 2, 40),
(9, 3, 110),
(10, 4, 30),
(11, 5, 80),
(12, 6, 20),
(13, 1, 70),
(14, 2, 90),
(15, 3, 40),
(16, 4, 60),
(17, 5, 100),
(18, 6, 25),
(19, 1, 45),
(20, 2, 75);

-- Inserting information into the CustomerInteraction table

INSERT INTO CustomerInteraction (CustomerID, InteractionType, InteractionDate, InteractionDetails) VALUES

(1, 'Feedback', '2024-05-15 10:00:00', 'Provided positive feedback on recent purchase.'),
(2, 'Support', '2024-05-16 11:30:00', 'Requested assistance with order tracking.'),
(3, 'Review', '2024-05-17 12:45:00', 'Left a 5-star review for the product.'),
(4, 'Feedback', '2024-05-18 09:15:00', 'Expressed satisfaction with customer service.'),
(5, 'Support', '2024-05-19 08:00:00', 'Inquired about product availability.'),
(6, 'Feedback', '2024-05-20 13:20:00', 'Suggested improvement for website layout.'),
(7, 'Review', '2024-05-21 14:30:00', 'Shared detailed review including product pros and cons.'),
(8, 'Feedback', '2024-05-22 15:45:00', 'Requested additional product information.'),
(9, 'Support', '2024-05-23 16:00:00', 'Reported an issue with order delivery.'),

(10, 'Feedback', '2024-05-24 17:10:00', 'Provided feedback on product packaging.'),
(1, 'Support', '2024-05-25 18:20:00', 'Asked for assistance with account login.'),
(2, 'Feedback', '2024-05-26 19:30:00', 'Suggested new product ideas.'),
(3, 'Review', '2024-05-27 20:45:00', 'Left a detailed review with photos.'),
(4, 'Feedback', '2024-05-28 21:00:00', 'Requested faster shipping options.'),
(5, 'Support', '2024-05-29 22:15:00', 'Inquired about return policy.'),
(6, 'Feedback', '2024-05-30 23:30:00', 'Suggested improvement for customer support.'),
(7, 'Review', '2024-05-31 09:45:00', 'Left a review highlighting product quality.'),
(8, 'Feedback', '2024-06-01 10:00:00', 'Requested more color options for a product.'),
(9, 'Support', '2024-06-02 11:15:00', 'Asked for assistance with order modification.'),
(10, 'Feedback', '2024-06-03 12:30:00', 'Provided feedback on website navigation.');

-- Inserting information into the SellerInteraction table

INSERT INTO SellerInteraction (SellerID, InteractionType, InteractionDate, InteractionDetails)
VALUES

(1, 'Feedback', '2024-05-15 10:00:00', 'Received positive feedback from a customer.'),
(2, 'Support', '2024-05-16 11:30:00', 'Assisted a customer with product inquiries.'),
(3, 'Review', '2024-05-17 12:45:00', 'Received a 5-star review for a product.'),
(4, 'Feedback', '2024-05-18 09:15:00', 'Addressed a customer complaint.'),
(5, 'Support', '2024-05-19 08:00:00', 'Provided guidance on order fulfillment.'),
(6, 'Feedback', '2024-05-20 13:20:00', 'Suggested improvements for website navigation.'),
(1, 'Support', '2024-05-21 14:30:00', 'Resolved a customers issue with their order.'),
(2, 'Feedback', '2024-05-22 15:45:00', 'Received suggestions for new product ideas.'),
(3, 'Review', '2024-05-23 16:00:00', 'Received a detailed review with photos.'),
(4, 'Feedback', '2024-05-24 17:10:00', 'Addressed concerns about shipping times.'),
(5, 'Support', '2024-05-25 18:20:00', 'Assisted with returns and exchanges.'),
(6, 'Feedback', '2024-05-26 19:30:00', 'Received feedback on product packaging.'),
(1, 'Support', '2024-05-27 20:45:00', 'Provided technical support for a product.'),
(2, 'Feedback', '2024-05-28 21:00:00', 'Received feedback on customer service experience.'),
(3, 'Review', '2024-05-29 22:15:00', 'Received a review highlighting product quality.');

(4, 'Feedback', '2024-05-30 23:30:00', 'Addressed concerns about product pricing.'),
(5, 'Support', '2024-05-31 09:45:00', 'Assisted with order cancellations.'),
(6, 'Feedback', '2024-06-01 10:00:00', 'Received feedback on website layout.'),
(1, 'Support', '2024-06-02 11:15:00', 'Assisted with account management.'),
(2, 'Feedback', '2024-06-03 12:30:00', 'Received feedback on product descriptions.');

-- Inserting information into the Analytics table

INSERT INTO Analytics (Date, UserActivity, SalesData, WebsitePerformanceMetrics) VALUES

('2024-05-01', 'Logged in, viewed product A, added product B to cart, checked out', 'Total sales: \$10000', 'Average page load time: 2.5 seconds, Conversion rate: 3%'),

('2024-05-02', 'Logged in, viewed product C, added product D to wishlist, viewed product E', 'Total sales: \$12000', 'Average page load time: 2.6 seconds, Conversion rate: 2.8%'),

('2024-05-03', 'Logged in, searched for product F, viewed product G, added product H to cart, checked out', 'Total sales: \$9500', 'Average page load time: 2.7 seconds, Conversion rate: 3.2%'),

('2024-05-04', 'Logged in, viewed product I, added product J to cart, viewed product K', 'Total sales: \$11000', 'Average page load time: 2.4 seconds, Conversion rate: 3.5%'),

('2024-05-05', 'Logged in, viewed product L, added product M to cart, viewed product N', 'Total sales: \$10500', 'Average page load time: 2.8 seconds, Conversion rate: 3.1%'),

('2024-05-06', 'Logged in, viewed product O, added product P to cart, checked out', 'Total sales: \$13000', 'Average page load time: 2.3 seconds, Conversion rate: 3.8%'),

('2024-05-07', 'Logged in, viewed product Q, added product R to cart, viewed product S', 'Total sales: \$11500', 'Average page load time: 2.6 seconds, Conversion rate: 3.4%'),

('2024-05-08', 'Logged in, viewed product T, added product U to cart, checked out', 'Total sales: \$14000', 'Average page load time: 2.5 seconds, Conversion rate: 3.6%'),

('2024-05-09', 'Logged in, viewed product V, added product W to cart, viewed product X', 'Total sales: \$12500', 'Average page load time: 2.7 seconds, Conversion rate: 3.2%'),

('2024-05-10', 'Logged in, viewed product Y, added product Z to cart, checked out', 'Total sales: \$15000', 'Average page load time: 2.4 seconds, Conversion rate: 3.9%');

select * from Customer;

select *from _Transaction;

select *from _Order;

Select*from Seller;

```
select*from Inventory;  
select*from CustomerInteraction;  
select*from SellerInteraction;  
select*from Analytics;
```

```
SELECT * FROM Product WHERE StockLevel > 0;  
SELECT * FROM CustomerInteraction WHERE CustomerID = 1;  
SELECT COUNT(*) AS TotalOrders FROM _Order;  
SELECT AVG(TotalCost) AS AverageTotalCost FROM _Order;  
SELECT SUM(Amount) AS TotalSalesAmount FROM Transaction;  
SELECT _Order.OrderID, _Order.OrderDate, _Order.TotalCost, _Order.OrderStatus,  
Customer.Name AS CustomerName, Customer.Email AS CustomerEmail  
FROM _Order  
JOIN Customer ON _Order.CustomerID = Customer.CustomerID;
```

```
SELECT Customer.CustomerID, Customer.Name, SUM(Transaction.Amount) AS  
TotalSalesAmount  
FROM Customer  
LEFT JOIN _Order ON Customer.CustomerID = _Order.CustomerID  
LEFT JOIN Transaction ON _Order.OrderID = Transaction.OrderID  
GROUP BY Customer.CustomerID, Customer.Name;
```

```
SELECT Product.Name, SUM(OrderProduct.Quantity) AS TotalQuantitySold  
FROM Product  
JOIN OrderProduct ON Product.ProductID = OrderProduct.ProductID  
GROUP BY Product.ProductID, Product.Name  
ORDER BY TotalQuantitySold DESC;
```