**Hexaware Technologies**

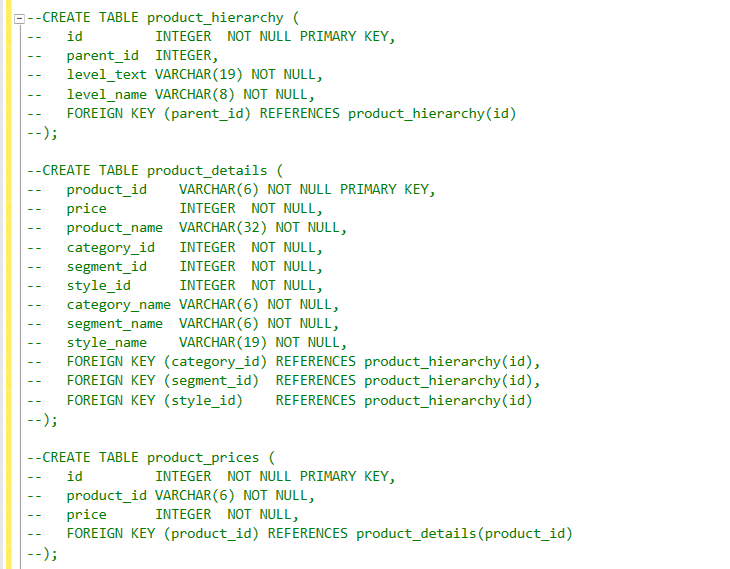
SQL Case Study

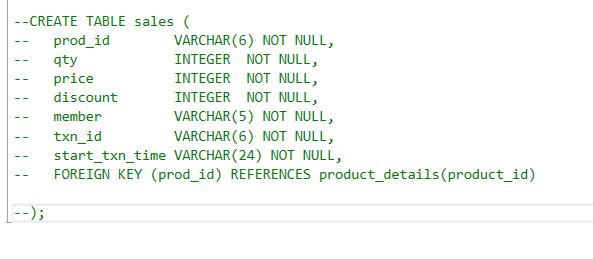
Texture Tales (Case-2)

**Step 1:** Database and Table Creation

In this step, we set up the database by creating the necessary tables for the sales and product information. The tables include:

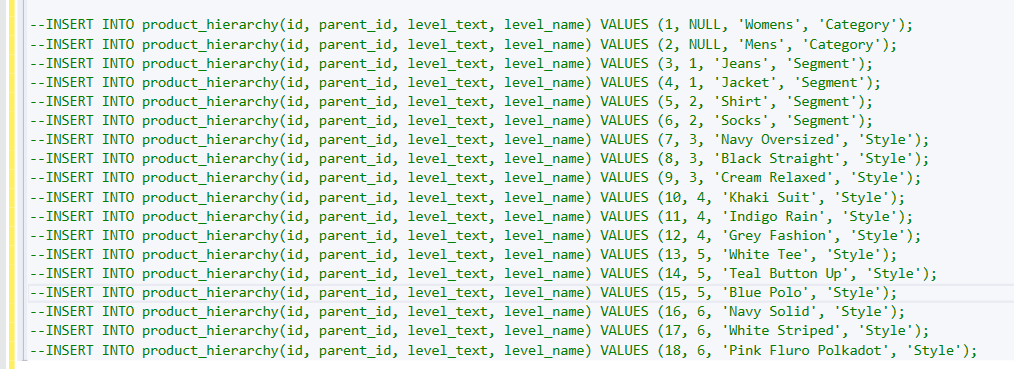
* **product\_hierarchy**: Stores hierarchical information about product categories, segments, and styles.
* **product\_details**: Contains detailed information for each product, including the category, segment, style, and price.
* **product\_prices**: Tracks the price details for each product.
* **sales**: Records each transaction with details such as product ID, quantity, price, discount, membership status, and transaction time.

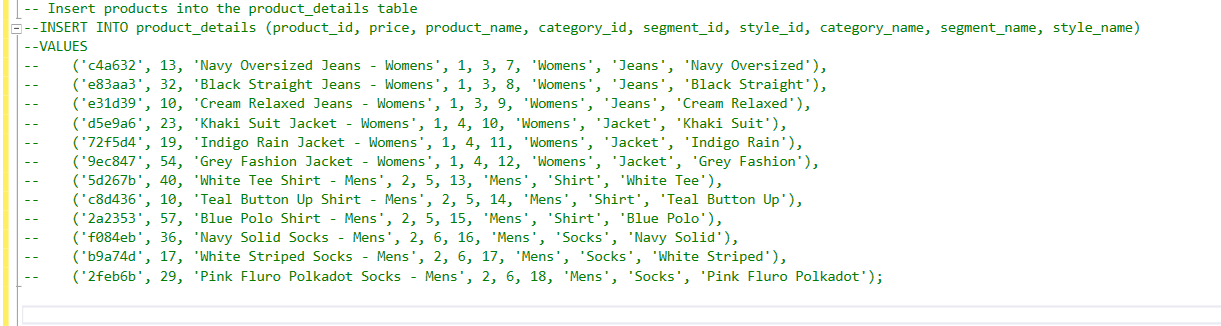


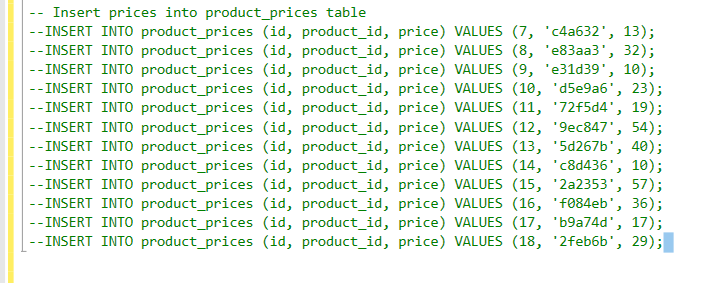


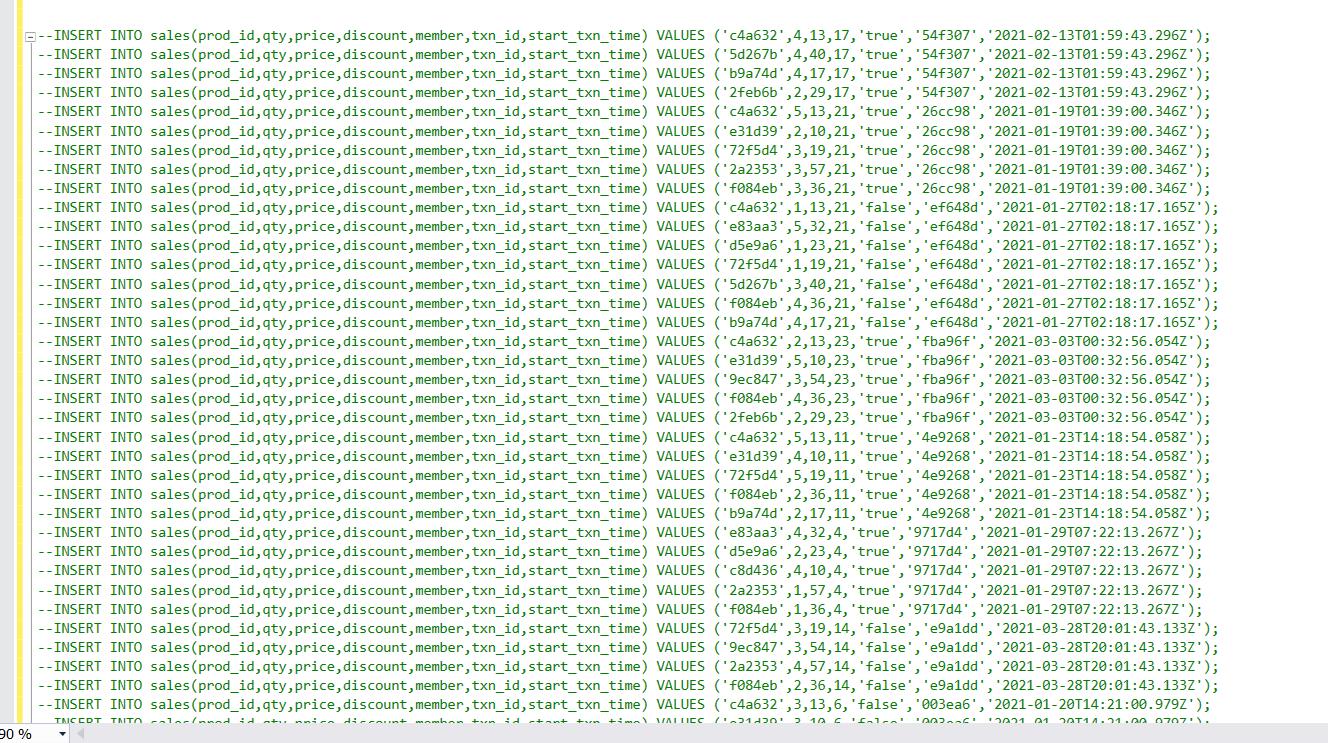
**Step 2:** Data Insertion

We inserted sample data into each table to populate the database for analysis. This includes adding entries for product categories, product details, product prices, and sales transactions. The inserted data will help us calculate totals and perform aggregations for the sales reports.





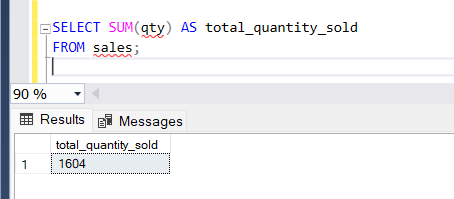




**Step 3:** Queries and Explanations

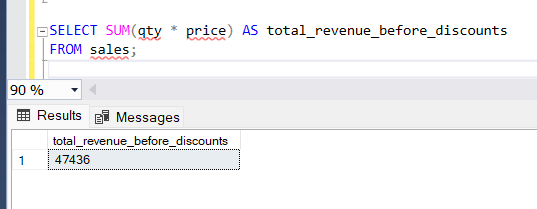
1. What was the total quantity sold for all products?

**Total Quantity Sold for All Products**: Aggregates the total quantity of all products sold across all transactions.



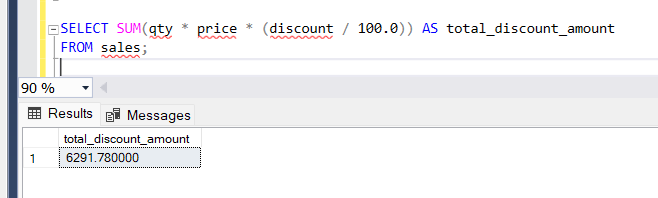
1. What is the total generated revenue for all products before discounts?

**Total Revenue for All Products Before Discounts**: Calculates total revenue generated from sales before applying any discounts.



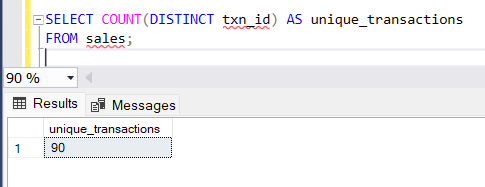
1. What was the total discount amount for all products?

**Total Discount Amount for All Products**: Computes the total discount amount applied across all transactions.



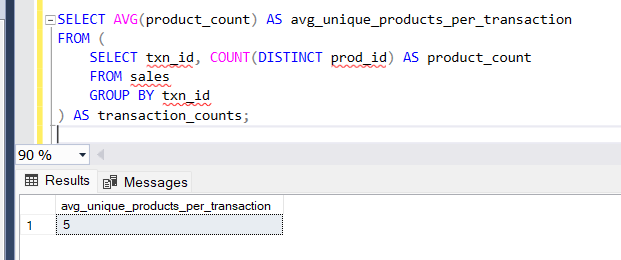
1. How many unique transactions were there?

**Unique Transaction Count**: Retrieves the total number of unique transactions recorded in the sales table.



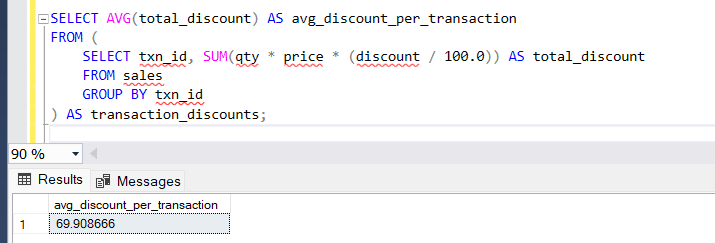
1. What are the average unique products purchased in each transaction?

**Average Unique Products per Transaction**: Uses **GROUP BY** with **COUNT(DISTINCT)** to calculate the average number of unique products per transaction.



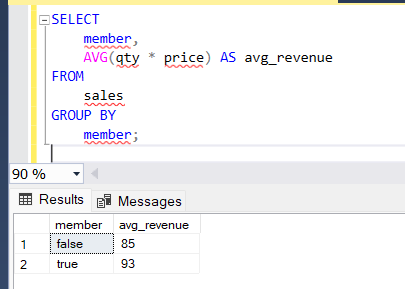
1. What is the average discount value per transaction?

**Average Discount per Transaction**: **GROUP BY** groups by transaction ID, and **AVG()** calculates the average discount per transaction.



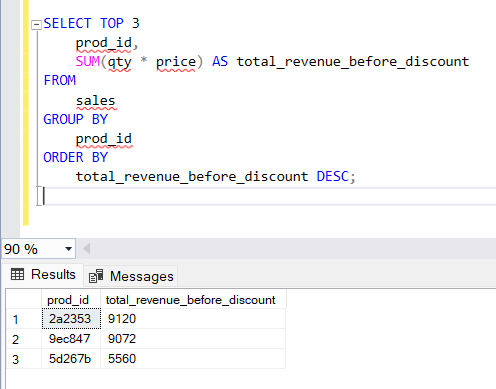
1. What is the average revenue for member transactions and non-member transactions?

**Average Revenue for Member vs. Non-Member Transactions**: Uses **GROUP BY** on membership status to calculate average revenue for both members and non-members.



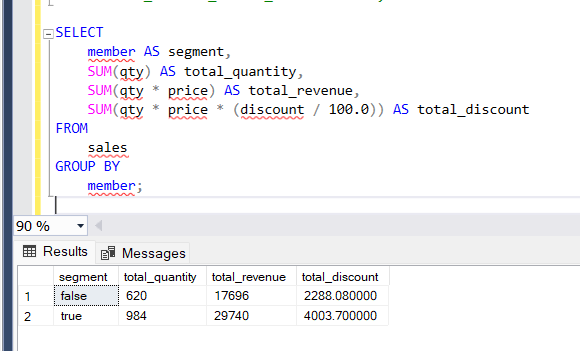
1. What are the top 3 products by total revenue before discount?

**Top 3 Products by Total Revenue Before Discount**: **GROUP BY** groups by product, and **ORDER BY** sorts by revenue, with **TOP** to fetch the top 3.



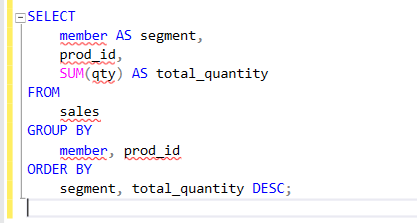
1. What are the total quantity, revenue and discount for each segment?

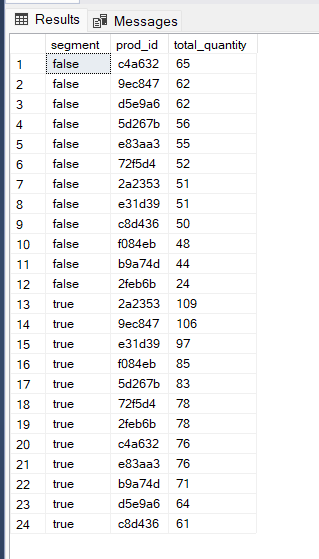
**Total Quantity, Revenue, and Discount for Each Segment**: **GROUP BY** segment ID with **SUM()** to aggregate total quantity, revenue, and discount per segment.



1. What is the top selling product for each segment?

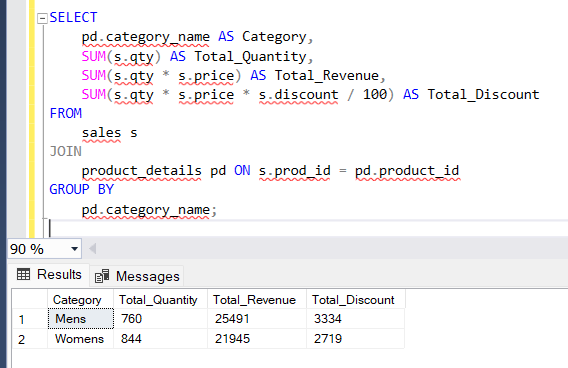
**Top-Selling Product for Each Segment**: **GROUP BY** segment ID, **ORDER BY** quantity, and **TOP** returns the best-selling product in each segment.





1. What are the total quantity, revenue and discount for each category?

**Total Quantity, Revenue, and Discount for Each Category**: **GROUP BY** category ID with **SUM()** aggregates total quantity, revenue, and discount per category.



1. What is the top selling product for each category?

**Top-Selling Product for Each Category**: **GROUP BY** category ID, **ORDER BY** quantity, and **TOP**/**LIMIT** identifies the top-selling product in each category.

