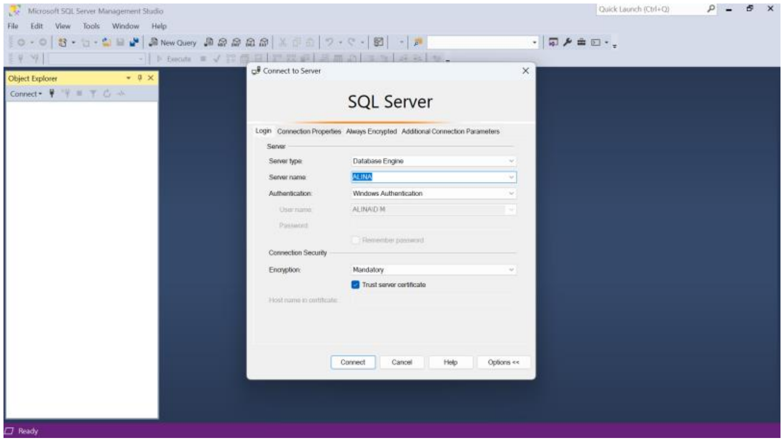
BIKE STORE DATA ANALYSIS CHALLENGE

1. SETUP PHASE

Connect to SSMS



Connect to Server

SQL Server

Login | Connection Properties | Always Encrypted | Additional Connection Parameters

Server

Server type: Database Engine

Server name: ALINA

Authentication: Windows Authentication

User name: ALINA\D M

Password:

☐ Remember password

Connection Security

Encryption: Mandatory

☑ Trust server certificate

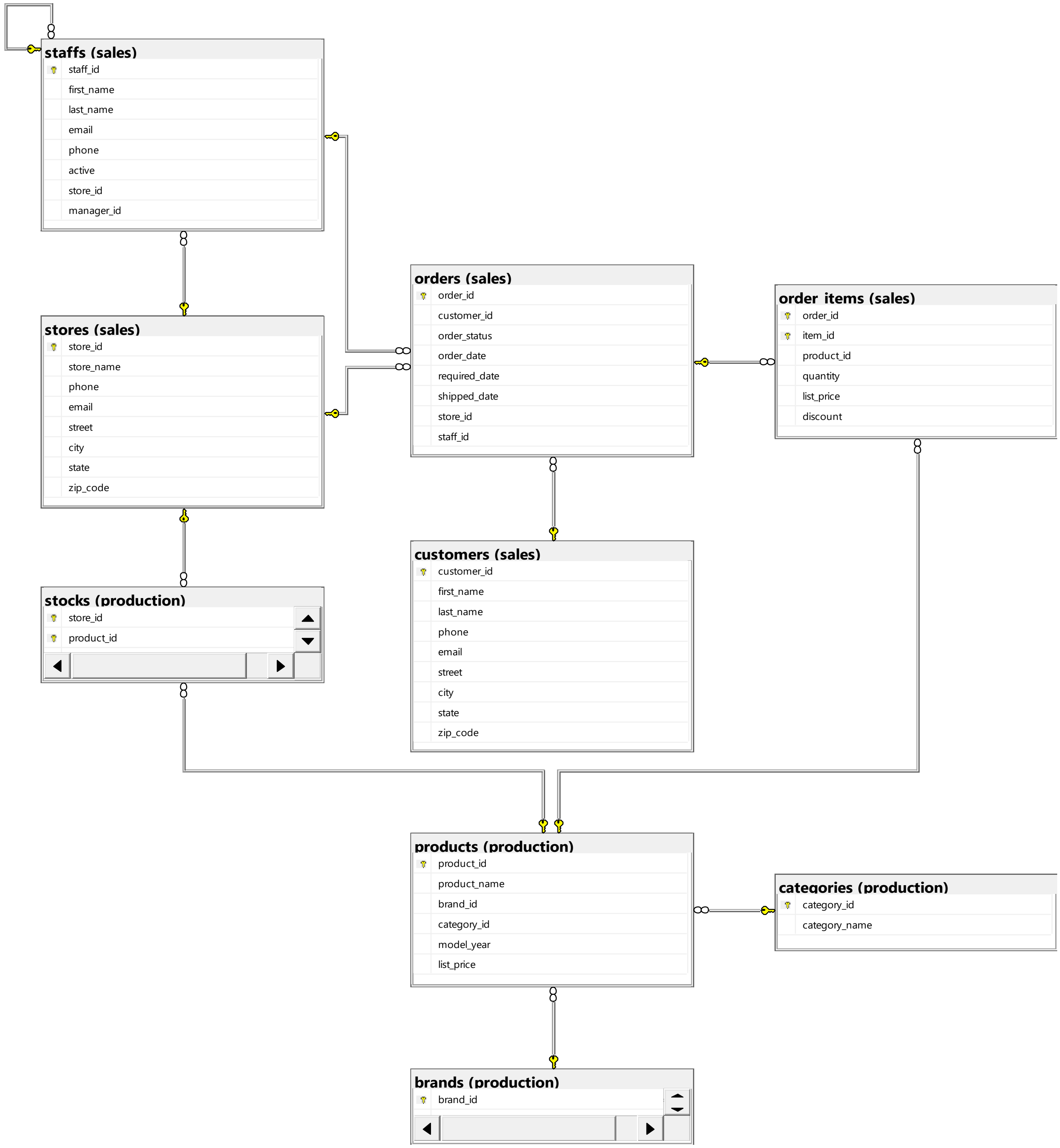
Host name in certificate:

Connect | Cancel | Help | Options <<

From the above BikeStores, we can understand the relationship between the tables in BikeStores database.

Table Name	Description	Primary Key	Foreign Keys
Staffs (sales)	Information about staff members.	staff_id	store_id, manager_id
Stores (sales)	Details of stores in the bike shop.	store_id	None
Customers (sales)	Details of customers placing orders.	customer_id	None
Orders (sales)	Information about customer orders.	order_id	customer_id, store_id, staff_id
Order Items (sales)	Items included in each order.	order_id, item_id	order_id, product_id
Products (production)	Details of products available for sale.	product_id	brand_id, category_id
Stocks (production)	Inventory data for each store.	store_id, product_id	store_id, product_id

BikeStores ERD diagram



Relationships

We can also interpret relationships between the tables in ERD.

Production Relationships

1. categories and products
    - Relationship: One-to-Many
    - Explanation: A category can include multiple products, and a product belongs to a single category.
2. brands and products
    - Relationship: One-to-Many
    - Explanation: A brand can have multiple products, and each product belongs to one brand.
3. products and stocks
    - Relationship: One-to-Many
    - Explanation: A product can be stocked in multiple stores, and each stock record relates to a single product.

Sales Relationships

1. customers and orders
    - Relationship: One-to-Many
    - Explanation: A customer can place multiple orders, and each order is placed by a single customer.
2. orders and order_items
    - Relationship: One-to-Many
    - Explanation: An order can contain multiple order items, and each order item belongs to a single order.
3. order_items and products
    - Relationship: Many-to-One
    - Explanation: Multiple order items can refer to a single product, and each order item corresponds to one product.
4. staffs and orders

   - Relationship: One-to-Many
   - Explanation: A staff member can handle multiple orders, and each order is processed by a single staff member.

5. stores and stocks
   - Relationship: One-to-Many
   - Explanation: A store can have stocks of multiple products, and each stock record is for a specific product in a store.
6. stores and orders
   - Relationship: One-to-Many
   - Explanation: A store can process multiple orders, and each order is placed at a specific store.

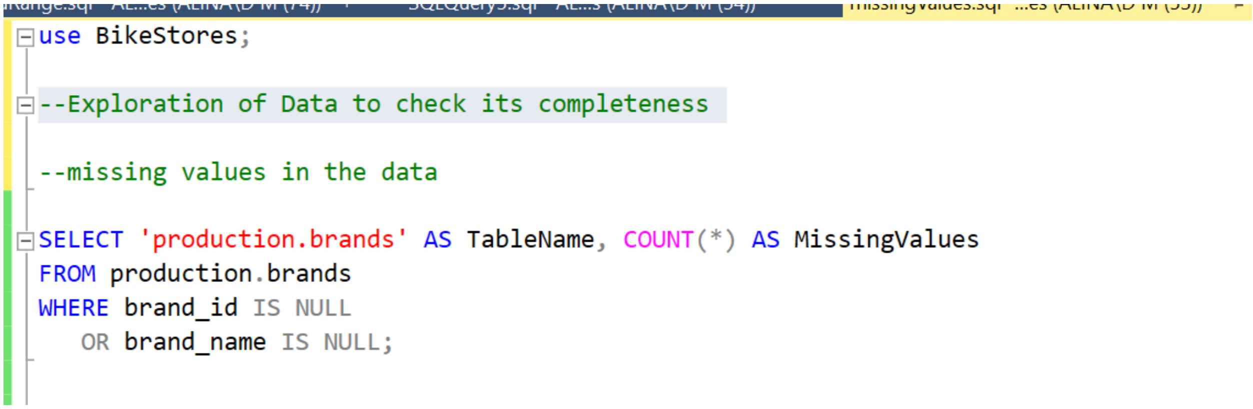
2. Exploration

How complete is my dataset?

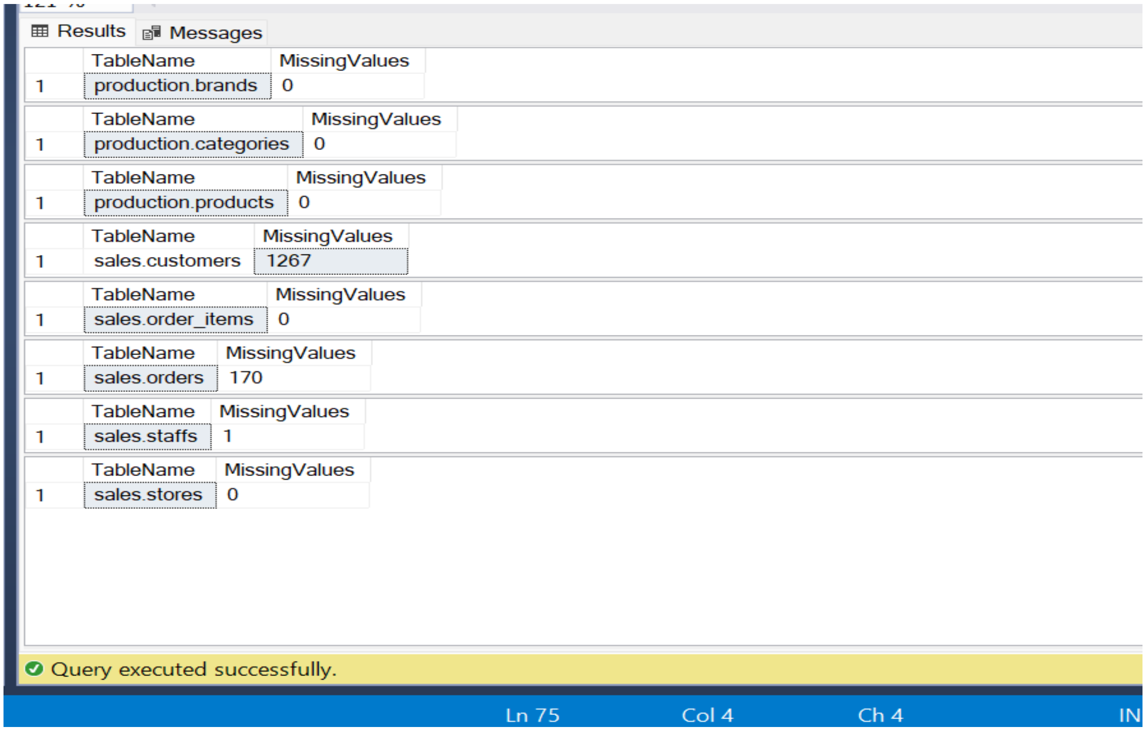
To assess the completeness of the dataset, we need to check for:

- Missing records
- Consistency
- Outliers

To check the missing records:



```
use BikeStores;

--Exploration of Data to check its completeness

--missing values in the data

SELECT 'production.brands' AS TableName, COUNT(*) AS MissingValues
FROM production.brands
WHERE brand_id IS NULL
   OR brand_name IS NULL;
```



	TableName	MissingValues
1	production.brands	0

	TableName	MissingValues
1	production.categories	0

	TableName	MissingValues
1	production.products	0

	TableName	MissingValues
1	sales.customers	1267

	TableName	MissingValues
1	sales.order_items	0

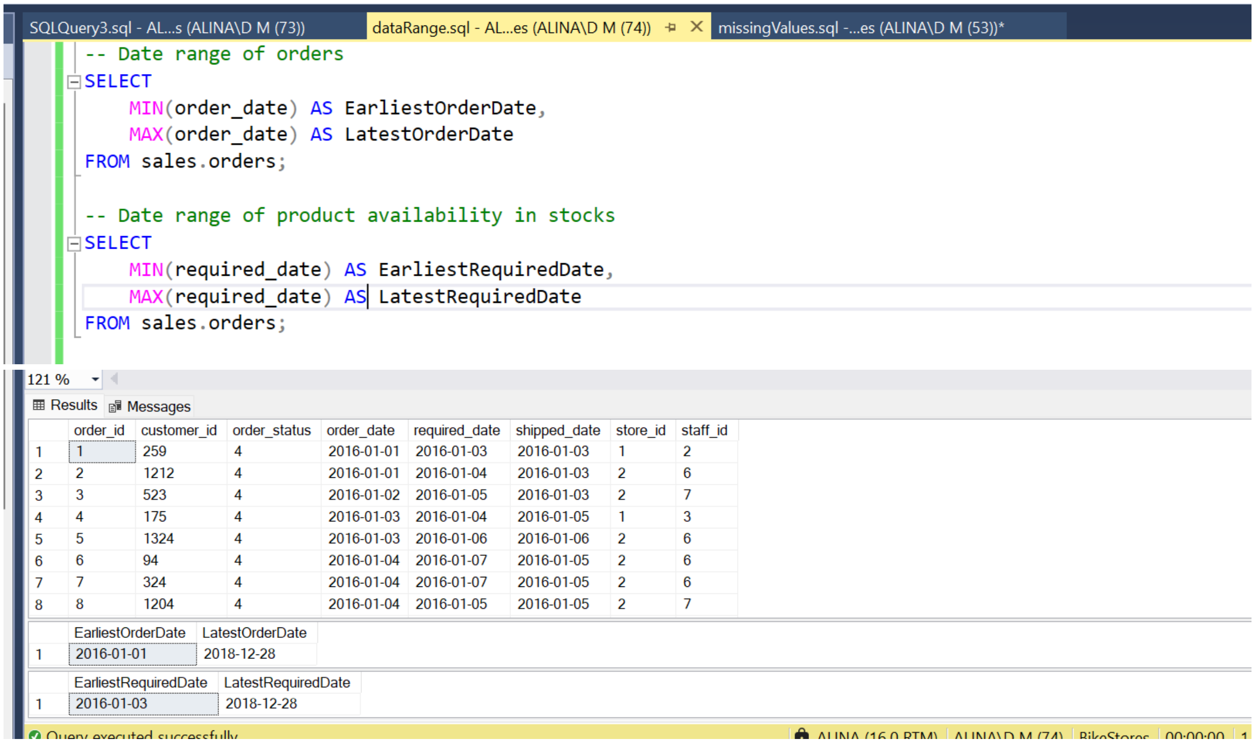
	TableName	MissingValues
1	sales.orders	170

	TableName	MissingValues
1	sales.staffs	1

	TableName	MissingValues
1	sales.stores	0

Query executed successfully.

To check the date range of our data:


SQLQuery3.sql - AL...s (ALINA\D M (73))
dataRange.sql - AL...es (ALINA\D M (74))
missingValues.sql -...es (ALINA\D M (53))*

```sql
-- Date range of orders
SELECT
    MIN(order_date) AS EarliestOrderDate,
    MAX(order_date) AS LatestOrderDate
FROM sales.orders;

-- Date range of product availability in stocks
SELECT
    MIN(required_date) AS EarliestRequiredDate,
    MAX(required_date) AS LatestRequiredDate
FROM sales.orders;
```

	order_id	customer_id	order_status	order_date	required_date	shipped_date	store_id	staff_id
1	1	259	4	2016-01-01	2016-01-03	2016-01-03	1	2
2	2	1212	4	2016-01-01	2016-01-04	2016-01-03	2	6
3	3	523	4	2016-01-02	2016-01-05	2016-01-03	2	7
4	4	175	4	2016-01-03	2016-01-04	2016-01-05	1	3
5	5	1324	4	2016-01-03	2016-01-06	2016-01-06	2	6
6	6	94	4	2016-01-04	2016-01-07	2016-01-05	2	6
7	7	324	4	2016-01-04	2016-01-07	2016-01-05	2	6
8	8	1204	4	2016-01-04	2016-01-05	2016-01-05	2	7

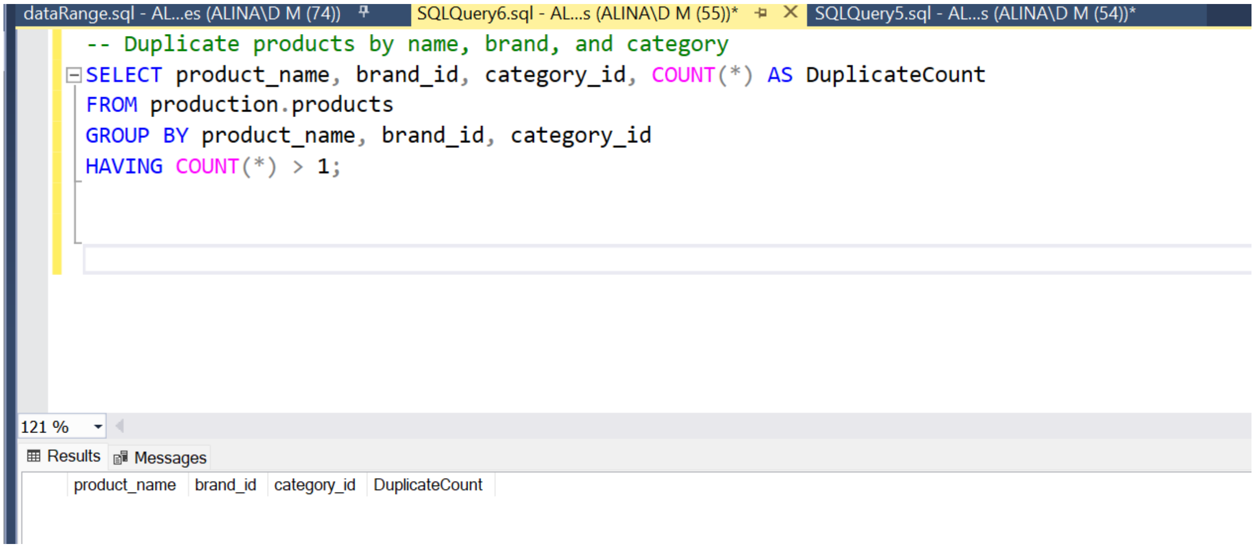
	EarliestOrderDate	LatestOrderDate
1	2016-01-01	2018-12-28

	EarliestRequiredDate	LatestRequiredDate
1	2016-01-03	2018-12-28

Query executed successfully.

Potential Data quality issues can be

- Checking for duplicate products

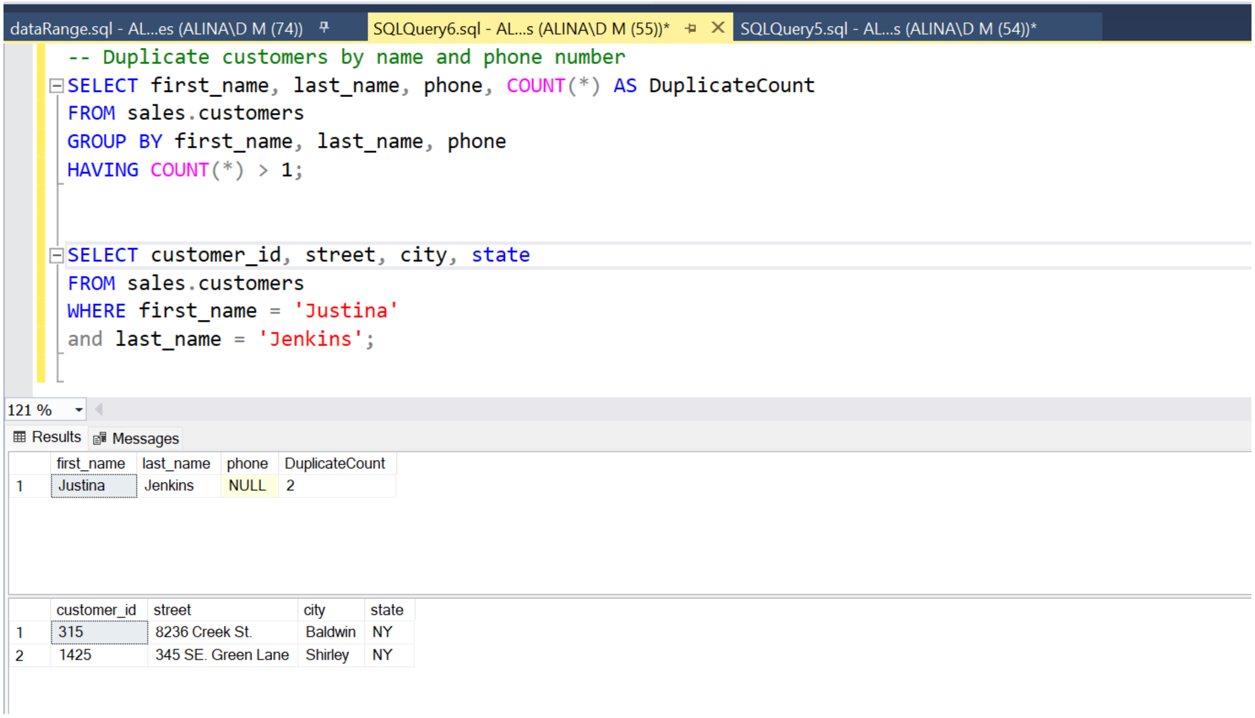

dataRange.sql - AL...es (ALINA\D M (74))
SQLQuery6.sql - AL...s (ALINA\D M (55))*
SQLQuery5.sql - AL...s (ALINA\D M (54))*
121 %
Results
Messages

```sql
-- Duplicate products by name, brand, and category
SELECT product_name, brand_id, category_id, COUNT(*) AS DuplicateCount
FROM production.products
GROUP BY product_name, brand_id, category_id
HAVING COUNT(*) > 1;
```

product_name	brand_id	category_id	DuplicateCount

Since the output is nothing, there are duplicate products.

- Checking for duplicate customers


dataRange.sql - AL...es (ALINA\D M (74))
SQLQuery6.sql - AL...s (ALINA\D M (55))*
SQLQuery5.sql - AL...s (ALINA\D M (54))*
121 %
Results
Messages

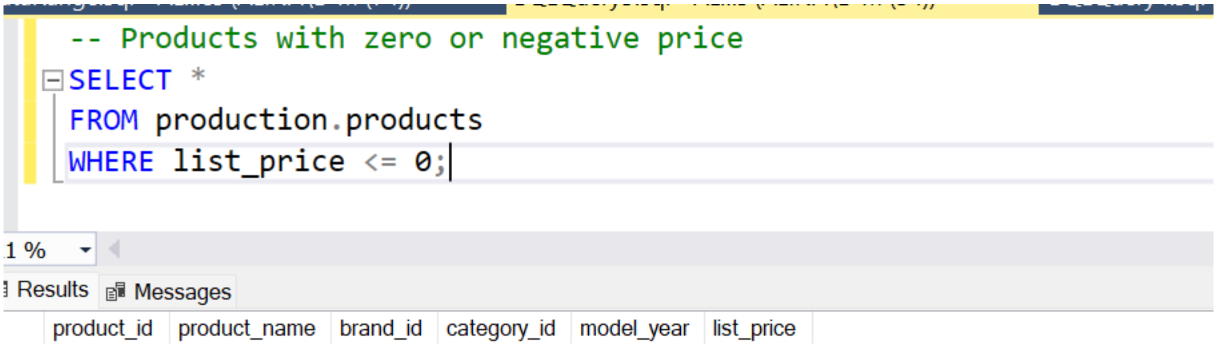
```sql
-- Duplicate customers by name and phone number
SELECT first_name, last_name, phone, COUNT(*) AS DuplicateCount
FROM sales.customers
GROUP BY first_name, last_name, phone
HAVING COUNT(*) > 1;


SELECT customer_id, street, city, state
FROM sales.customers
WHERE first_name = 'Justina'
and last_name = 'Jenkins';
```

	first_name	last_name	phone	DuplicateCount
1	Justina	Jenkins	NULL	2

	customer_id	street	city	state
1	315	8236 Creek St.	Baldwin	NY
2	1425	345 SE. Green Lane	Shirley	NY

Since the output is Justina Jenkins with DuplicateCount of 2, it means the data is duplicated once. As we can see, it has two customer_id of same name but with different street, city and state which means they are two different people.

- Negative or zero pricing in products.

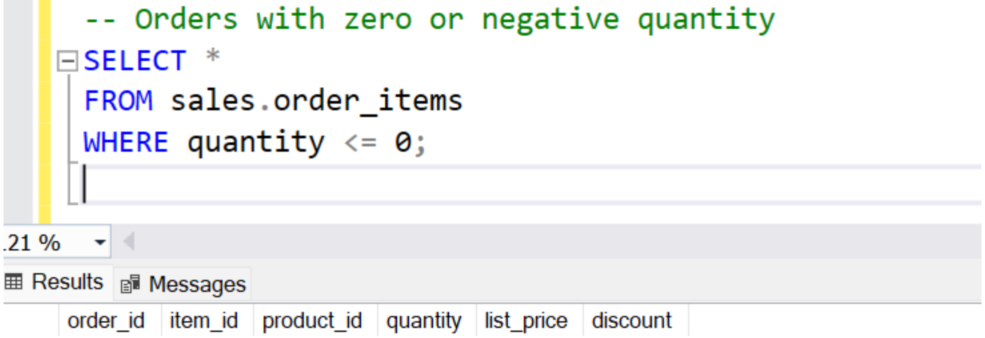


```sql
-- Products with zero or negative price
SELECT *
FROM production.products
WHERE list_price <= 0;
```

product_id	product_name	brand_id	category_id	model_year	list_price

There was no data for pricing of the product with negative or zero.

- Orders with quantity = 0.



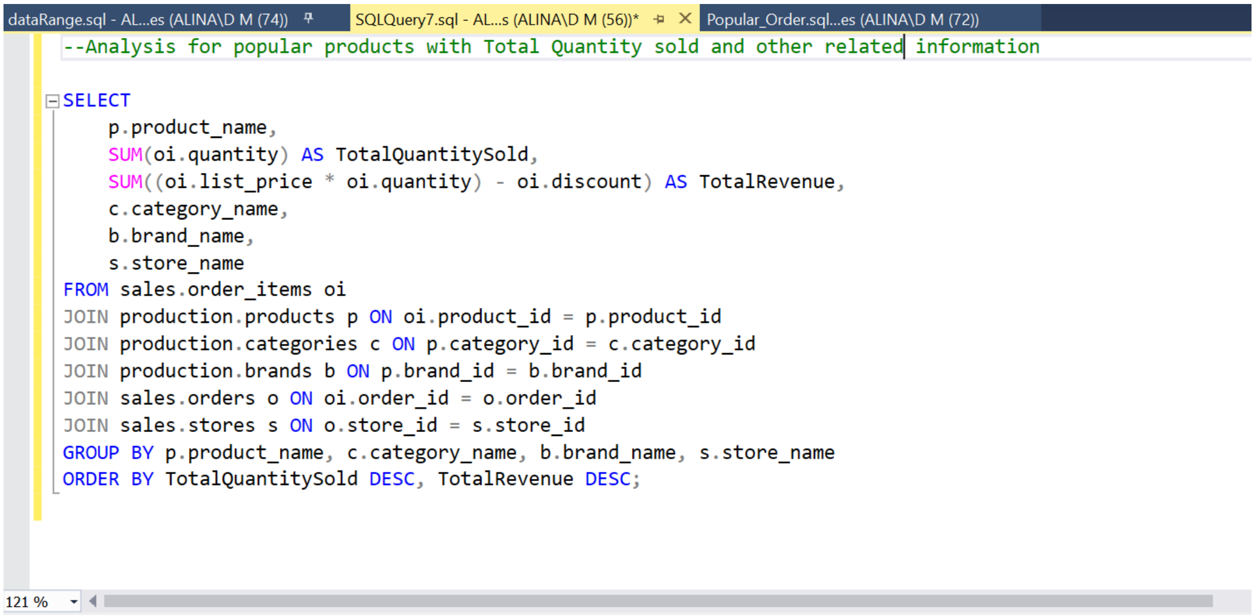
```sql
-- Orders with zero or negative quantity
SELECT *
FROM sales.order_items
WHERE quantity <= 0;
```

order_id	item_id	product_id	quantity	list_price	discount

The orders with quantity below 0 were found none.

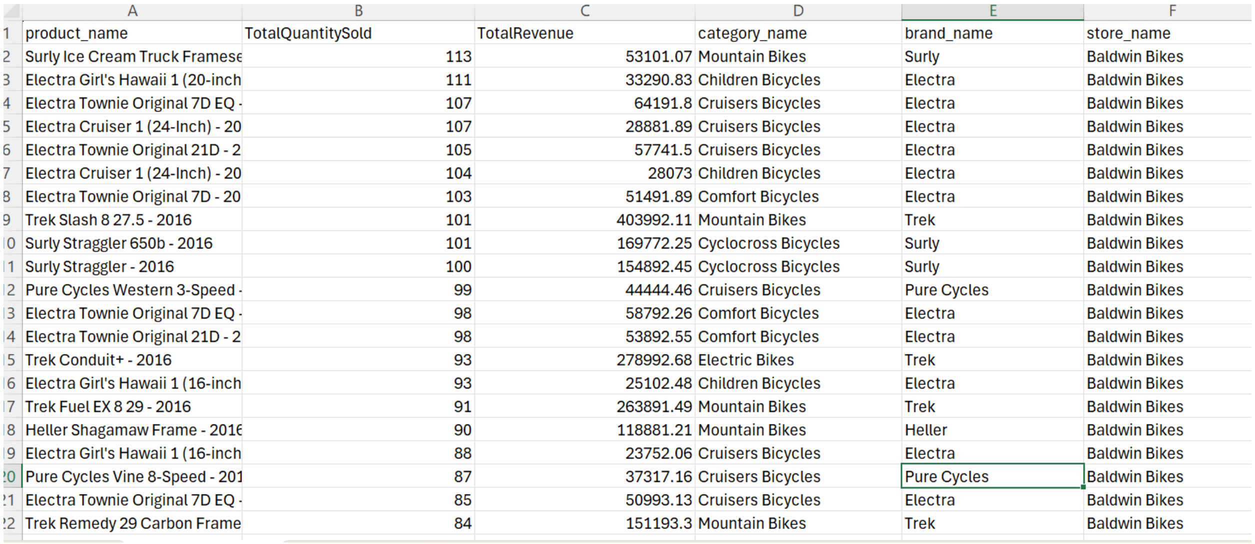
3. Analysis

For analysis, I wanted to analyze the popular product using the total quantity and information related to stock management.


dataRange.sql - AL...es (ALINA\D M (74))
SQLQuery7.sql - AL...s (ALINA\D M (56))*
Popular_Order.sql...es (ALINA\D M (72))
121 %

```sql
--Analysis for popular products with Total Quantity sold and other related information

SELECT
    p.product_name,
    SUM(oi.quantity) AS TotalQuantitySold,
    SUM((oi.list_price * oi.quantity) - oi.discount) AS TotalRevenue,
    c.category_name,
    b.brand_name,
    s.store_name
FROM sales.order_items oi
JOIN production.products p ON oi.product_id = p.product_id
JOIN production.categories c ON p.category_id = c.category_id
JOIN production.brands b ON p.brand_id = b.brand_id
JOIN sales.orders o ON oi.order_id = o.order_id
JOIN sales.stores s ON o.store_id = s.store_id
GROUP BY p.product_name, c.category_name, b.brand_name, s.store_name
ORDER BY TotalQuantitySold DESC, TotalRevenue DESC;
```

Result of the query:



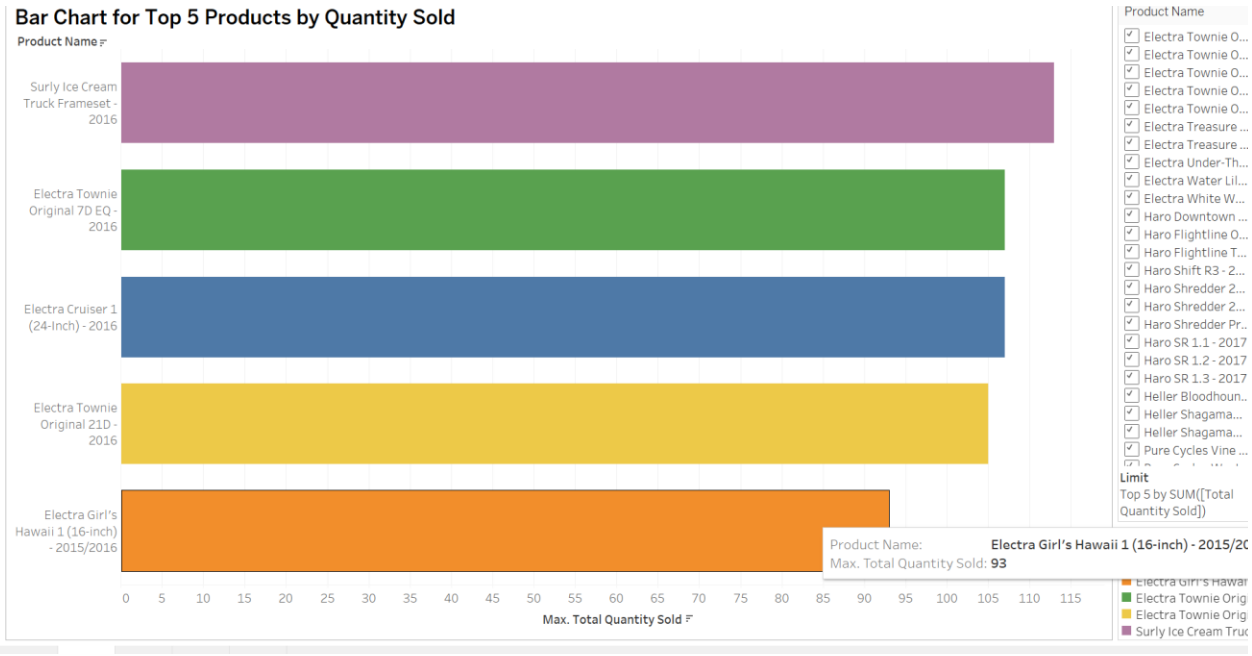
product_name	TotalQuantitySold	TotalRevenue	category_name	brand_name	store_name
Surly Ice Cream Truck Framese	113	53101.07	Mountain Bikes	Surly	Baldwin Bikes
Electra Girl's Hawaii 1 (20-inch	111	33290.83	Children Bicycles	Electra	Baldwin Bikes
Electra Townie Original 7D EQ -	107	64191.8	Cruisers Bicycles	Electra	Baldwin Bikes
Electra Cruiser 1 (24-Inch) - 20	107	28881.89	Cruisers Bicycles	Electra	Baldwin Bikes
Electra Townie Original 21D - 2	105	57741.5	Cruisers Bicycles	Electra	Baldwin Bikes
Electra Cruiser 1 (24-Inch) - 20	104	28073	Children Bicycles	Electra	Baldwin Bikes
Electra Townie Original 7D - 20	103	51491.89	Comfort Bicycles	Electra	Baldwin Bikes
Trek Slash 8 27.5 - 2016	101	403992.11	Mountain Bikes	Trek	Baldwin Bikes
Surly Straggler 650b - 2016	101	169772.25	Cyclocross Bicycles	Surly	Baldwin Bikes
Surly Straggler - 2016	100	154892.45	Cyclocross Bicycles	Surly	Baldwin Bikes
Pure Cycles Western 3-Speed -	99	44444.46	Cruisers Bicycles	Pure Cycles	Baldwin Bikes
Electra Townie Original 7D EQ -	98	58792.26	Comfort Bicycles	Electra	Baldwin Bikes
Electra Townie Original 21D - 2	98	53892.55	Comfort Bicycles	Electra	Baldwin Bikes
Trek Conduit+ - 2016	93	278992.68	Electric Bikes	Trek	Baldwin Bikes
Electra Girl's Hawaii 1 (16-inch	93	25102.48	Children Bicycles	Electra	Baldwin Bikes
Trek Fuel EX 8 29 - 2016	91	263891.49	Mountain Bikes	Trek	Baldwin Bikes
Heller Shagamaw Frame - 2016	90	118881.21	Mountain Bikes	Heller	Baldwin Bikes
Electra Girl's Hawaii 1 (16-inch	88	23752.06	Cruisers Bicycles	Electra	Baldwin Bikes
Pure Cycles Vine 8-Speed - 201	87	37317.16	Cruisers Bicycles	Pure Cycles	Baldwin Bikes
Electra Townie Original 7D EQ -	85	50993.13	Cruisers Bicycles	Electra	Baldwin Bikes
Trek Remedy 29 Carbon Frame	84	151193.3	Mountain Bikes	Trek	Baldwin Bikes

4. Visualization using Tableau

Link to Tableu : [Data Visualization | Tableau Public]

Bar Chart: Top 5 Products by Quantity Sold

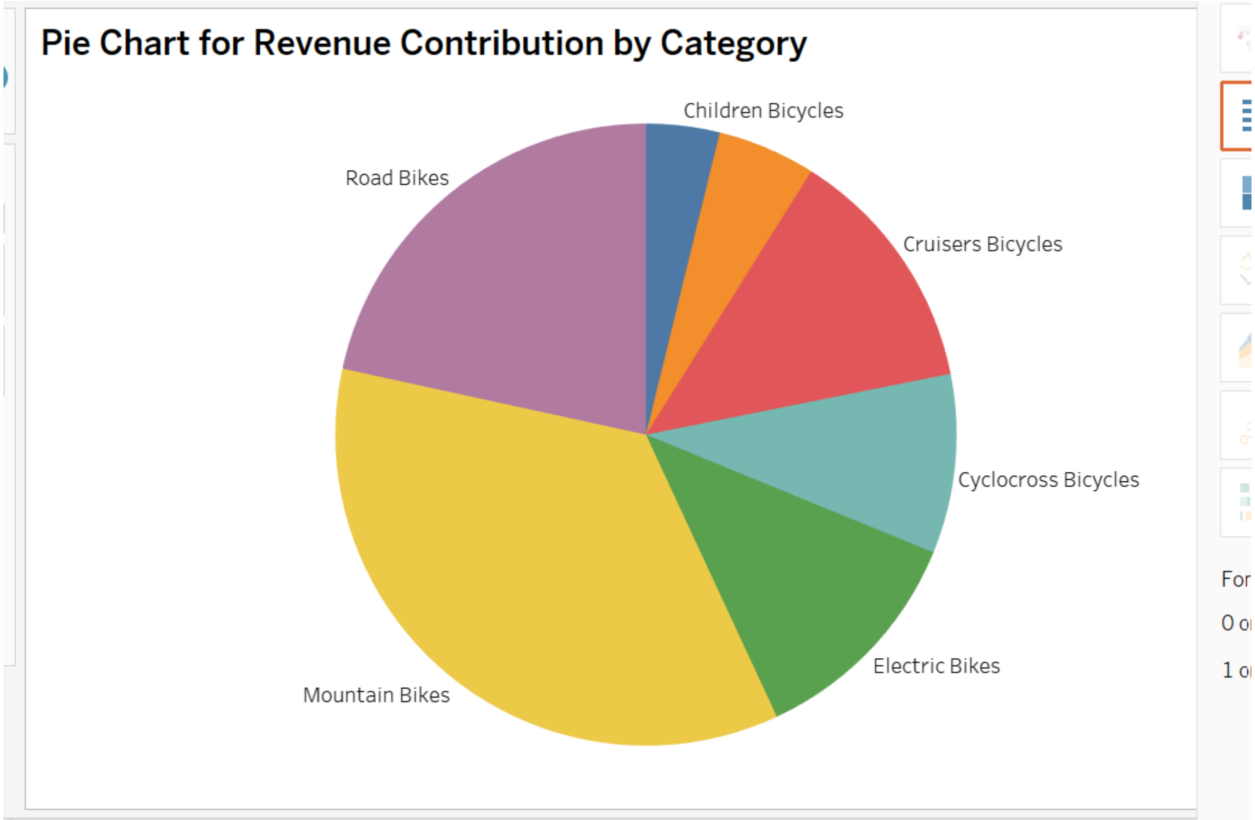
A bar chart is ideal for comparing discrete items like products, showcasing their popularity based on sales volume (Total Quantity Sold). It ranks the products, making it easy to see which products customers prefer the most.



- The top product, "Surly Ice Cream Truck," sold the highest quantity (167 units), followed by "Electra Cruiser 1" (157 units).
- The chart helps businesses identify which products to stock more of and which to promote further.
- Products with higher quantities might be budget-friendly or have higher demand among a broader customer base.

Pie Chart: Revenue Contribution by Category

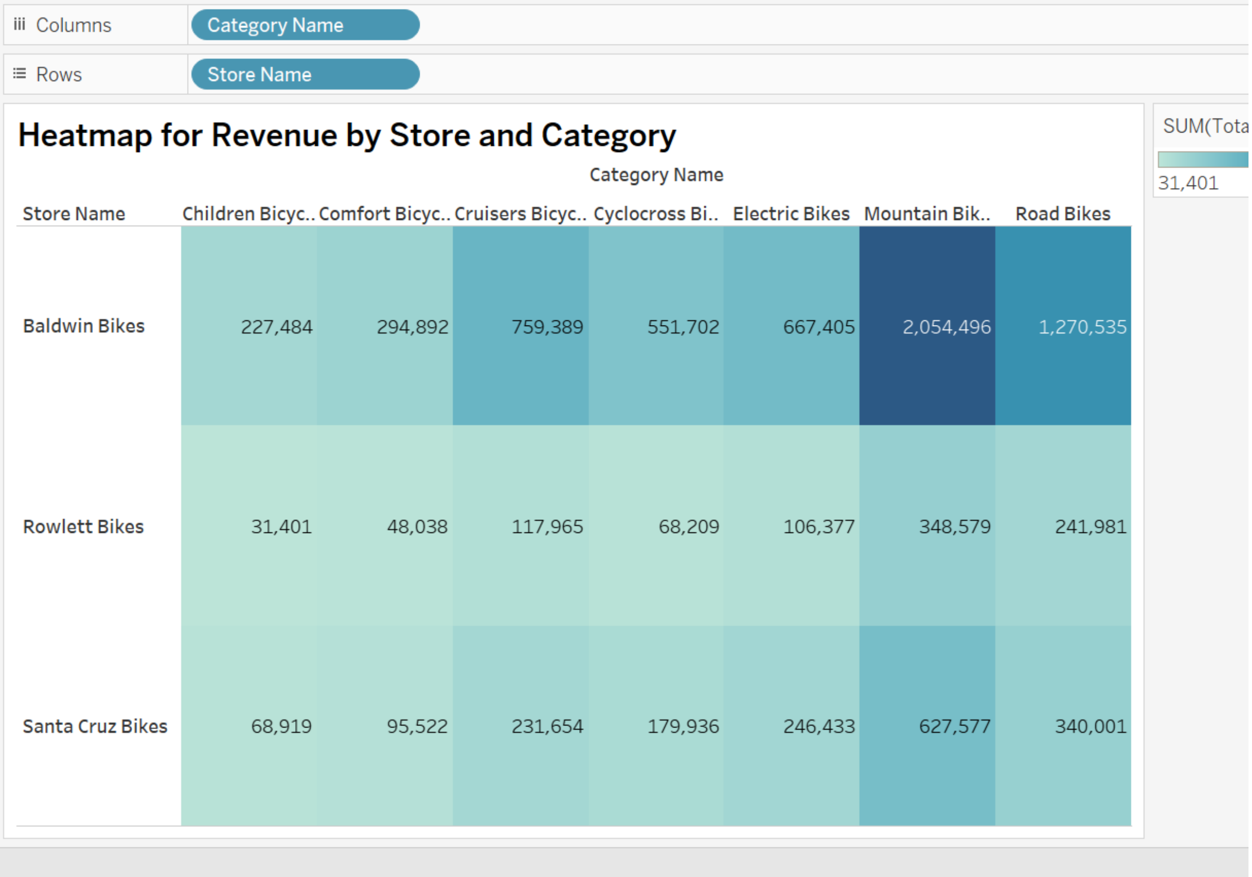
Pie charts are great for showing proportional data. In this case, it illustrates the share of each category in the total revenue. The visual focus is on understanding category dominance in revenue generation.



- "Mountain Bikes" contribute the highest revenue (3,030,651), followed by "Road Bikes" (1,852,516).
- Categories like "Children Bicycles" (327,804) and "Comfort Bicycles" (438,452) contribute significantly less.
- Businesses can use this to allocate resources and marketing budgets to higher-revenue categories and explore ways to boost sales in underperforming categories.

Heat Map: Revenue by Store and Category

Heat maps visually represent data intensity using colors, making it easy to identify areas of high and low performance across multiple dimensions. This chart shows revenue for each category broken down by store.



Heatmap for Revenue by Store and Category

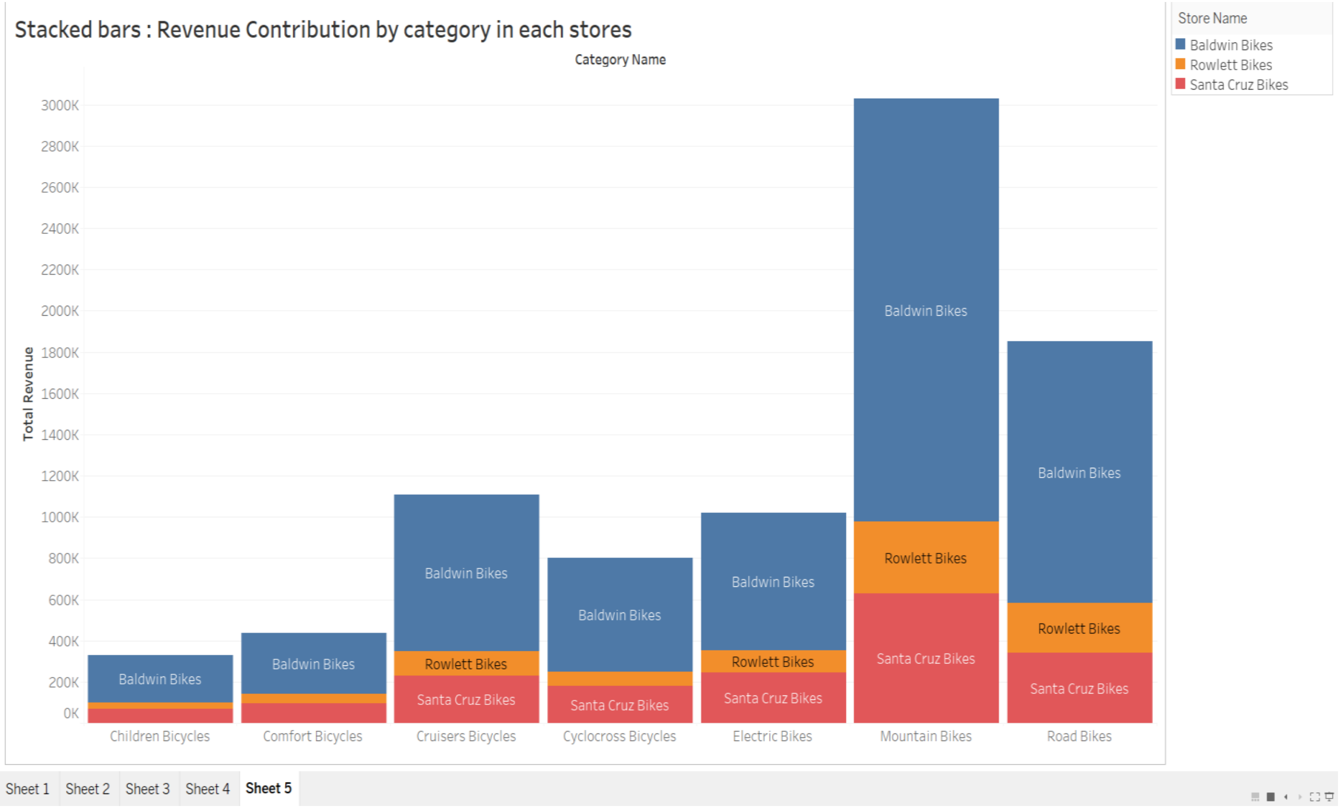
Category Name

Store Name	Children Bicyc..	Comfort Bicyc..	Cruisers Bicyc..	Cyclocross Bi..	Electric Bikes	Mountain Bik..	Road Bikes
Baldwin Bikes	227,484	294,892	759,389	551,702	667,405	2,054,496	1,270,535
Rowlett Bikes	31,401	48,038	117,965	68,209	106,377	348,579	241,981
Santa Cruz Bikes	68,919	95,522	231,654	179,936	246,433	627,577	340,001

- "Mountain Bikes" dominate in all stores, with Baldwin Bikes generating the highest revenue in this category (2,054,496).
- "Children Bicycles" and "Comfort Bicycles" perform poorly across all stores, indicating a need for reevaluation of inventory or marketing strategies for these categories.
- Baldwin Bikes is the highest-performing store overall, suggesting it might have a better location or customer base for premium products.

Stacked Chart: Revenue Contribution by Category with Store Names

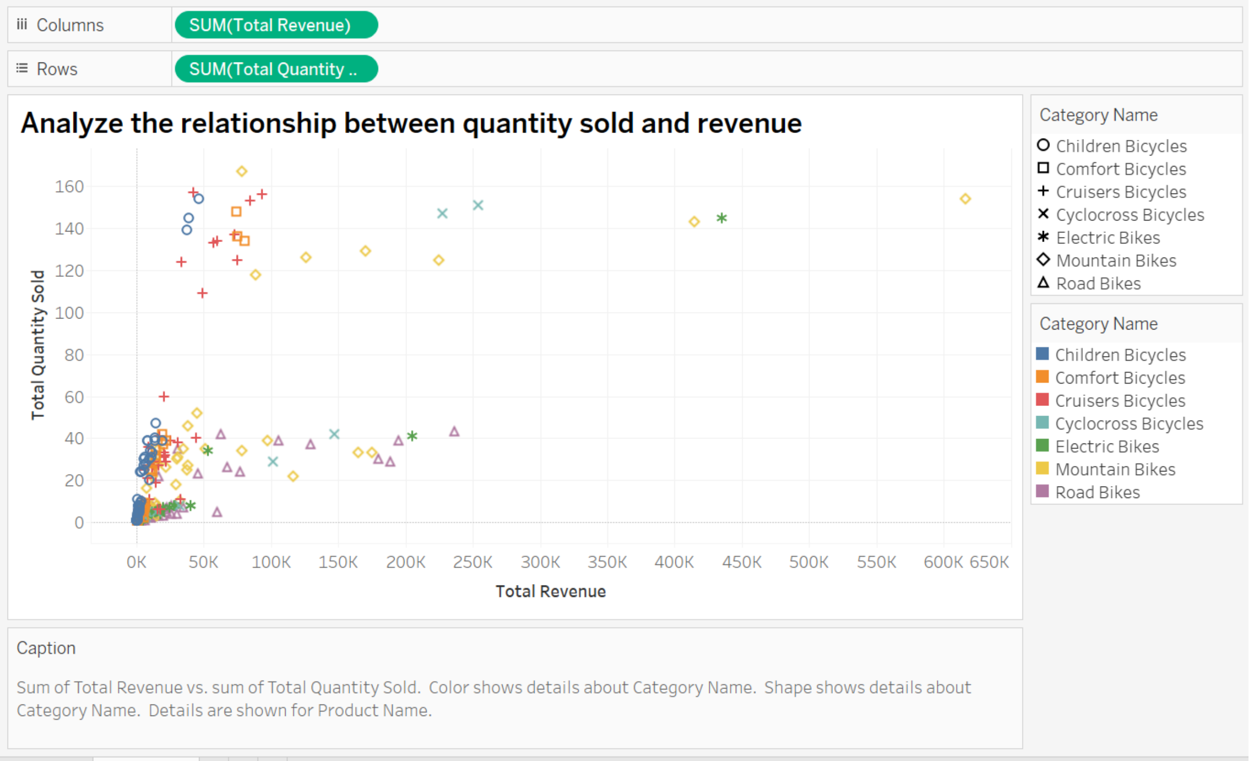
A stacked bar chart provides a clear breakdown of total revenue by category and further divides it by store. It enables businesses to analyze both the category-level performance and the store-specific contributions in one chart.



- "Mountain Bikes" generate the highest total revenue across all categories, with Baldwin Bikes contributing the largest share, followed by Rowlett Bikes and Santa Cruz Bikes.
- This indicates that Baldwin Bikes is a key driver for high-revenue products like Mountain Bikes, and efforts to boost sales at Rowlett Bikes and Santa Cruz Bikes could significantly increase overall revenue.

Scatter Plot: Quantity Sold vs. Revenue

Scatter plots are used to analyze relationships between two variables, here Total Quantity Sold and Total Revenue. The goal is to identify patterns, outliers, and insights about products.



Columns SUM(Total Revenue)

Rows SUM(Total Quantity ..

Analyze the relationship between quantity sold and revenue

Caption

Sum of Total Revenue vs. sum of Total Quantity Sold. Color shows details about Category Name. Shape shows details about Category Name. Details are shown for Product Name.

- Products with high revenue but low quantity sold might be premium items with high margins.
- Products with high quantities but low revenue could be affordable items with thin margins but high demand.
- The scatter plot highlights opportunities to optimize pricing or focus on products that balance revenue and volume.