**Bike store dataset**

Link: <https://www.kaggle.com/datasets/dillonmyrick/bike-store-sample-database>?

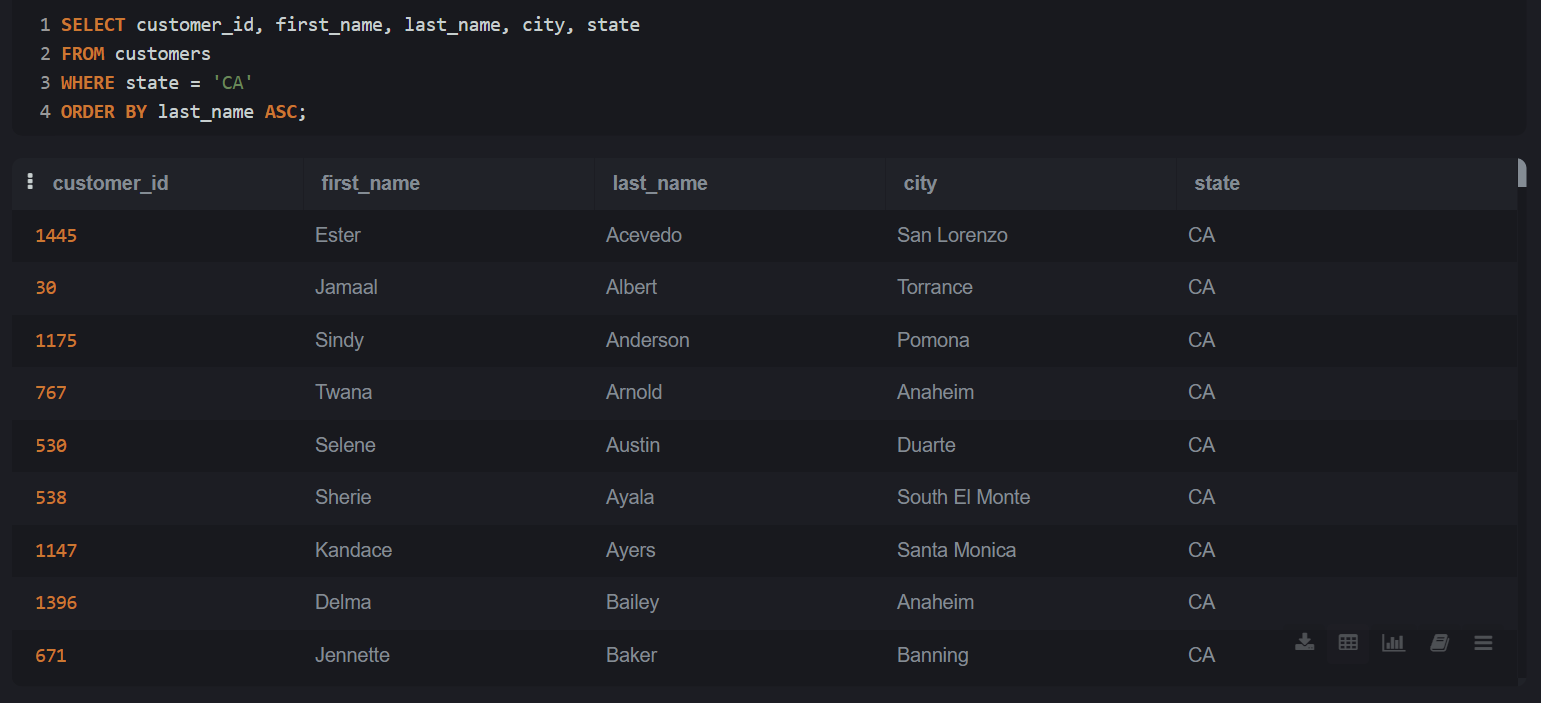
1. List all customers from California, ordered by last name

SELECT customer\_id, first\_name, last\_name, city, state

FROM customers

WHERE state = 'CA'

ORDER BY last\_name ASC;



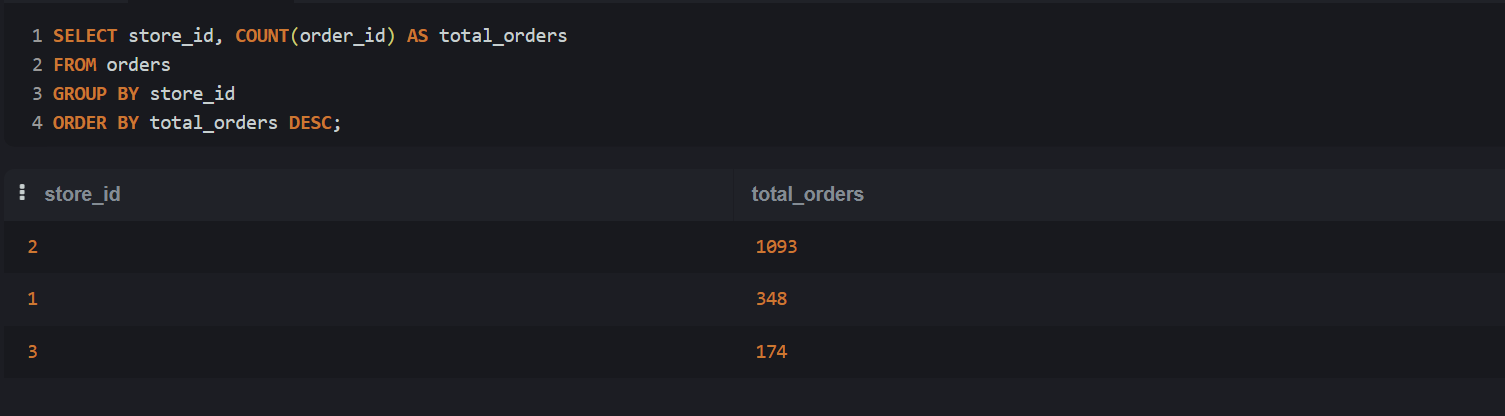
1. Total number of orders per store

SELECT store\_id, COUNT(order\_id) AS total\_orders

FROM orders

GROUP BY store\_id

ORDER BY total\_orders DESC;



1. INNER JOIN: Orders with customer and staff details

SELECT

o.order\_id,

c.first\_name AS customer\_name,

s.first\_name AS staff\_name,

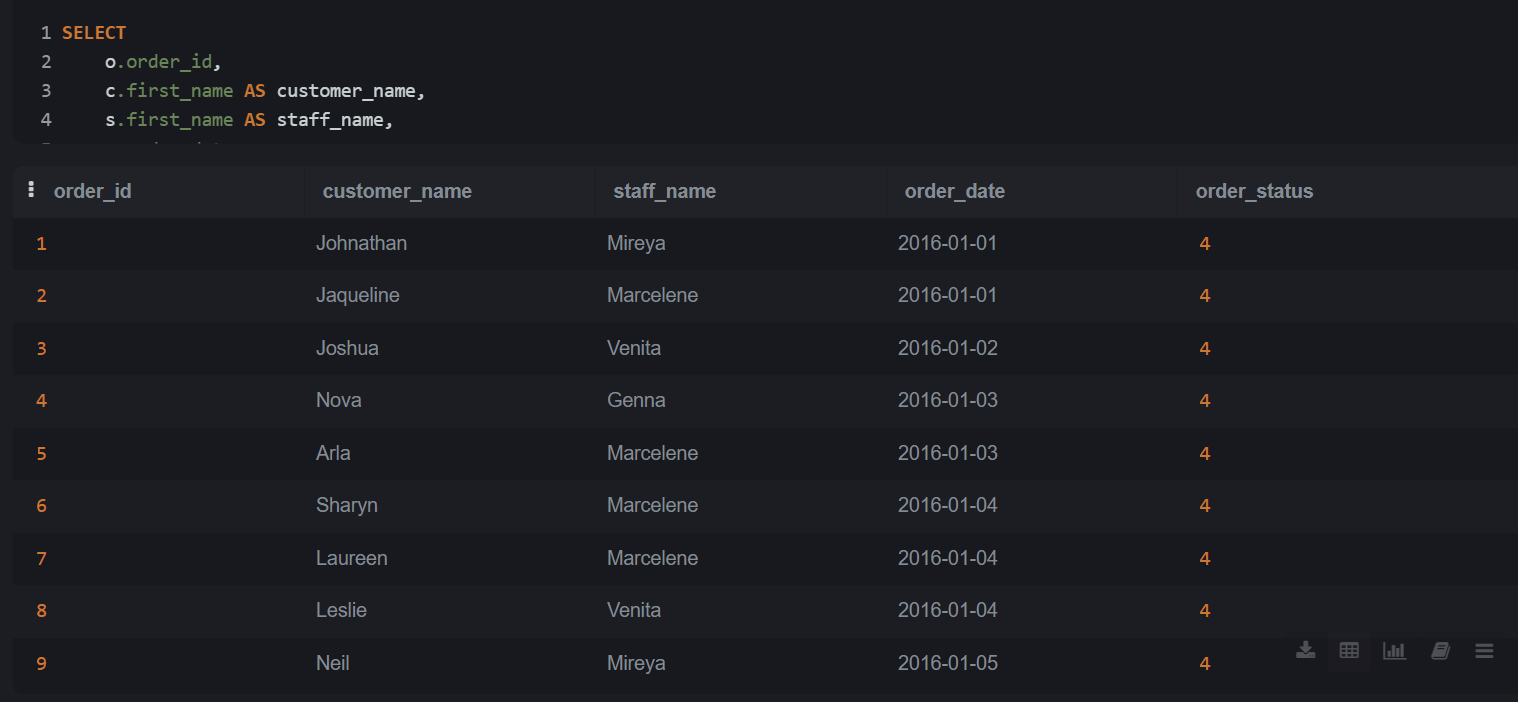
o.order\_date,

o.order\_status

FROM orders o

INNER JOIN customers c ON o.customer\_id = c.customer\_id

INNER JOIN staffs s ON o.staff\_id = s.staff\_id;



1. LEFT JOIN: List all products with their stock quantity

SELECT

p.product\_name,

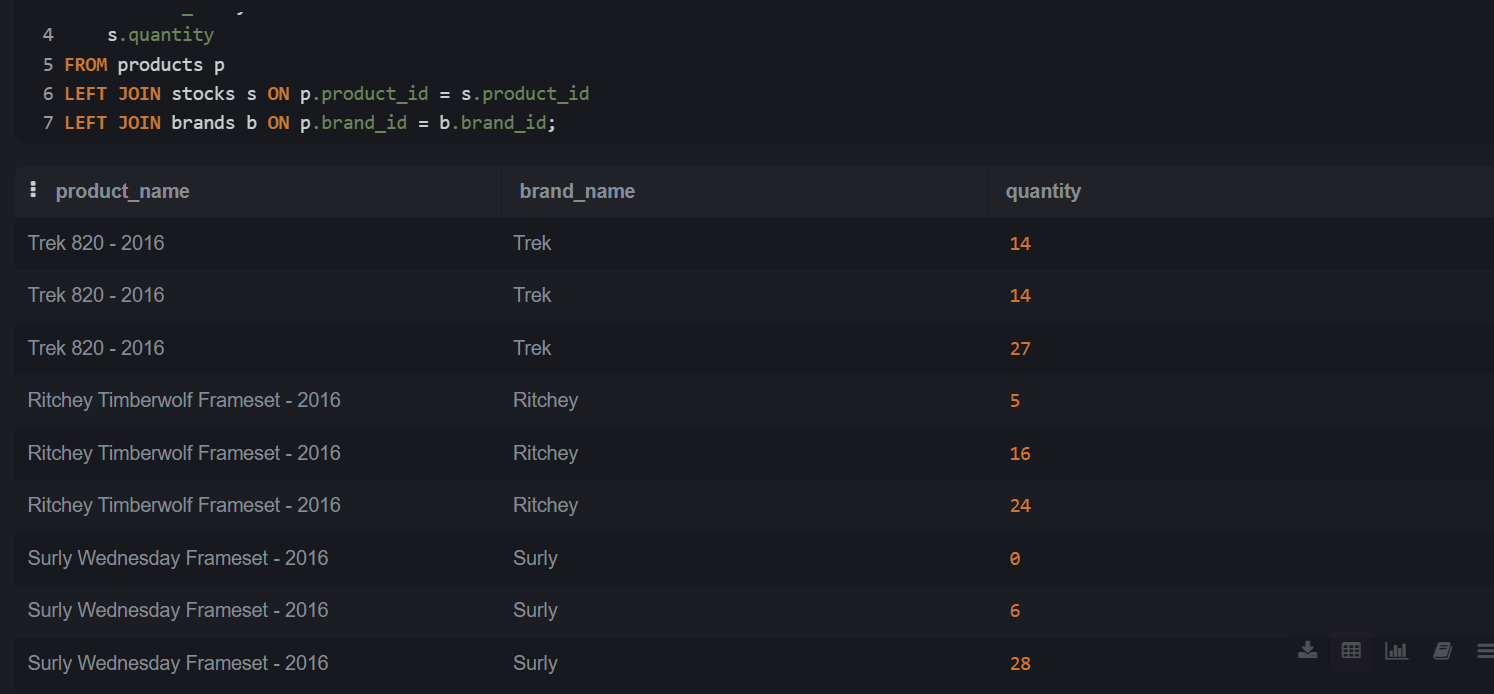
b.brand\_name,

s.quantity

FROM products p

LEFT JOIN stocks s ON p.product\_id = s.product\_id

LEFT JOIN brands b ON p.brand\_id = b.brand\_id;



1. RIGHT JOIN: All brands even if no products exist

SELECT

b.brand\_name,

p.product\_name

FROM products p

RIGHT JOIN brands b ON p.brand\_id = b.brand\_id;

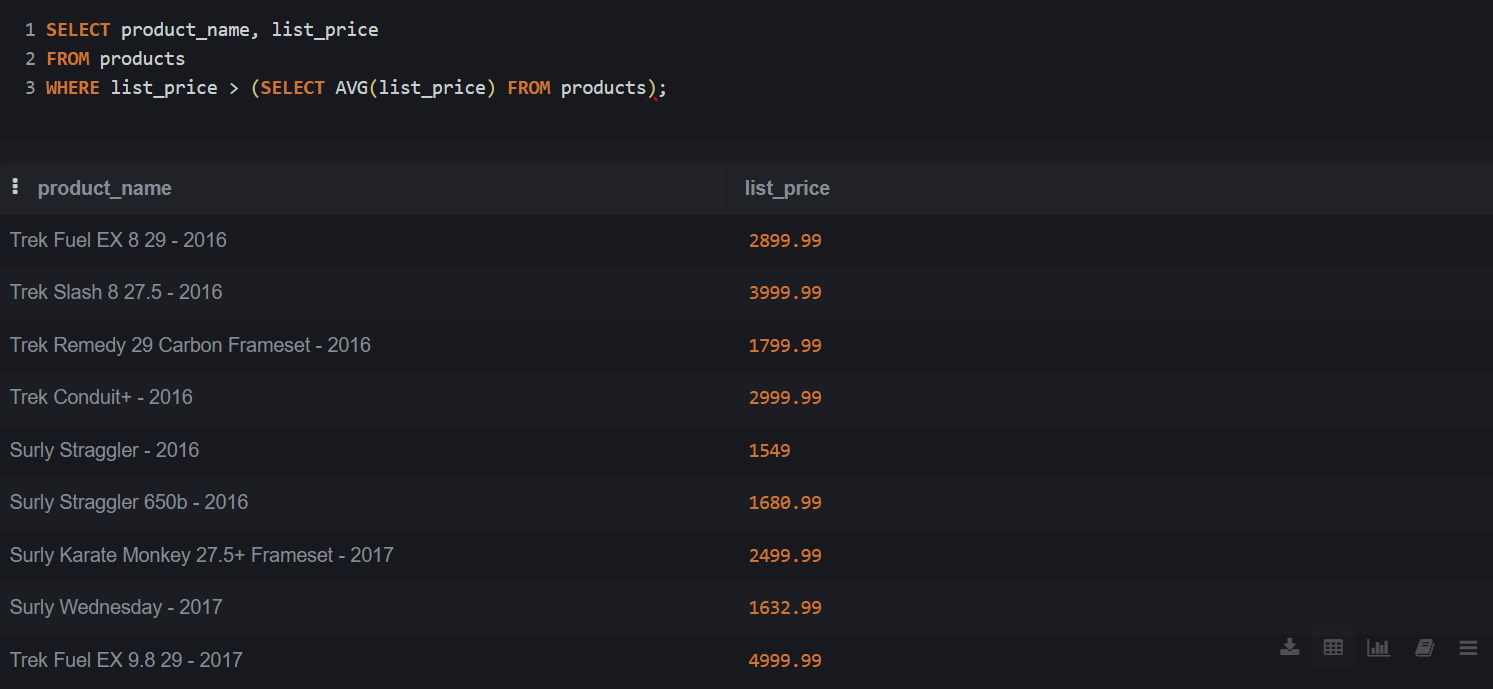


1. Find products that are above average list price

SELECT product\_name, list\_price

FROM products

WHERE list\_price > (SELECT AVG(list\_price) FROM products);



1. Calculate total sales and average discount per order

SELECT

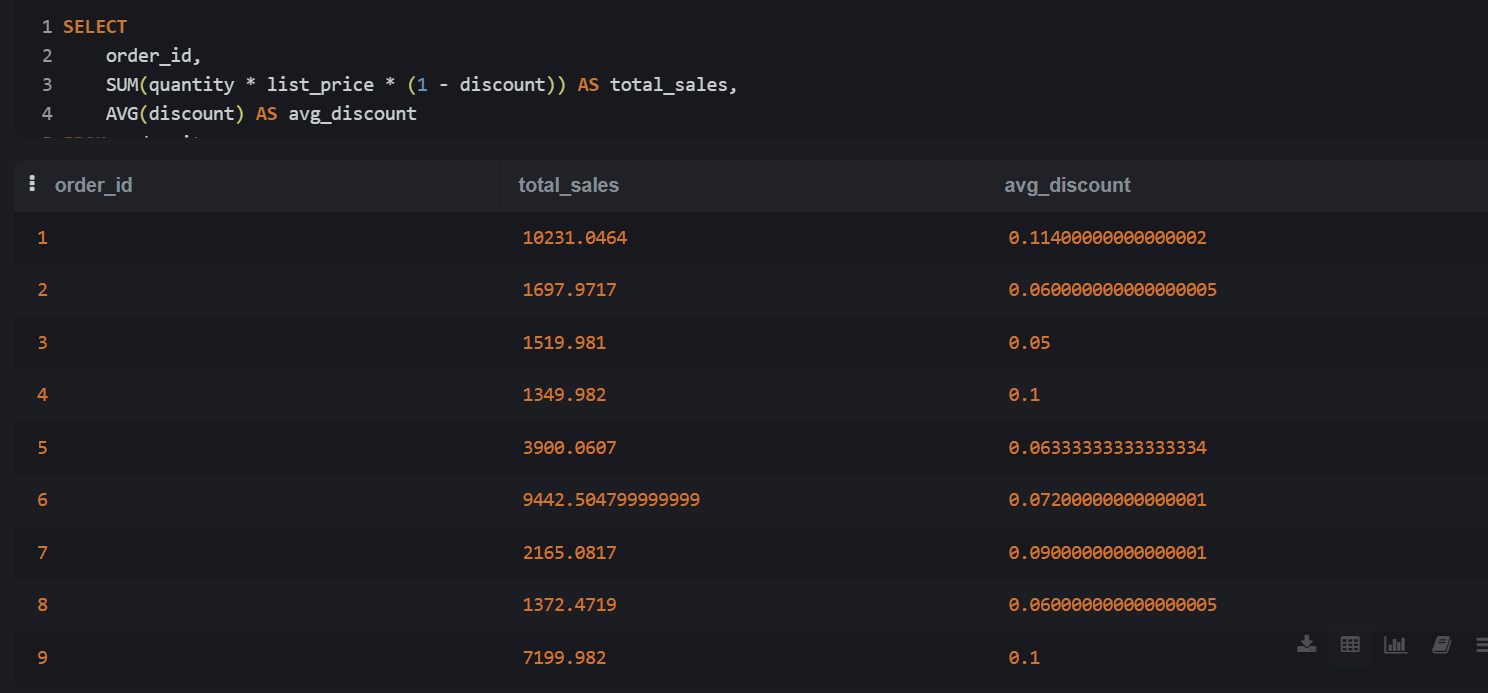
order\_id,

SUM(quantity \* list\_price \* (1 - discount)) AS total\_sales,

AVG(discount) AS avg\_discount

FROM order\_items

GROUP BY order\_id;



1. Average list price per category

SELECT

c.category\_name,

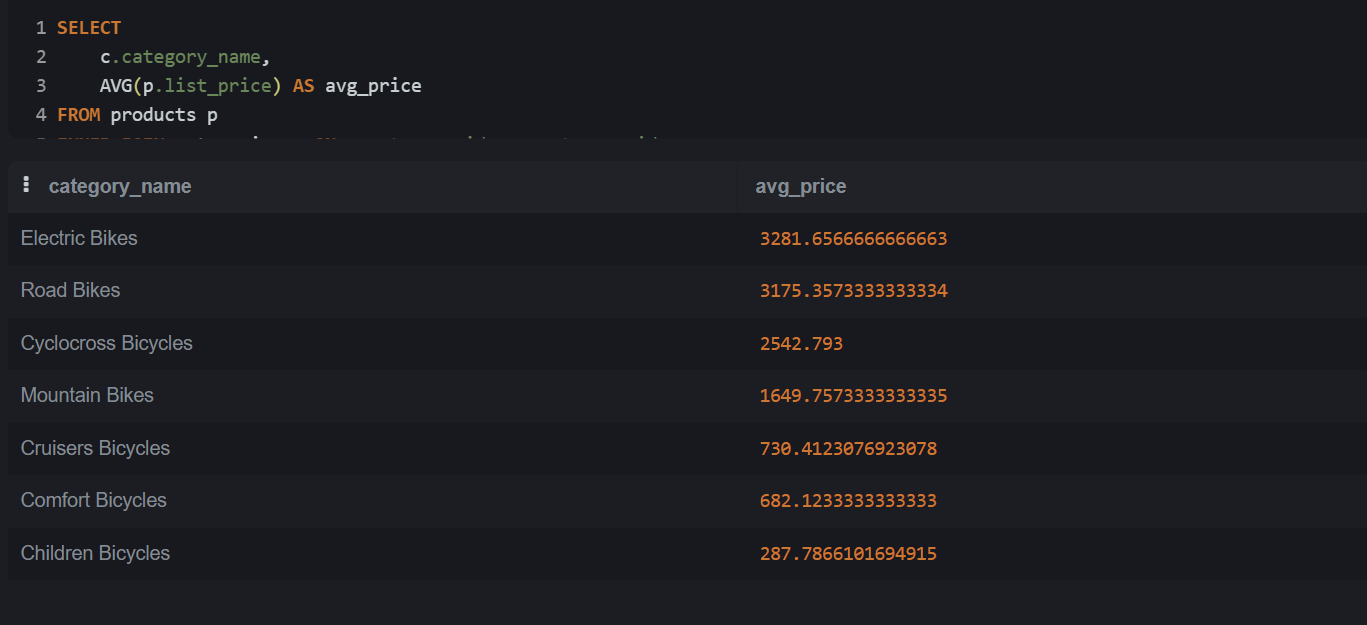
AVG(p.list\_price) AS avg\_price

FROM products p

INNER JOIN categories c ON p.category\_id = c.category\_id

GROUP BY c.category\_name

ORDER BY avg\_price DESC;



1. View: Customer order summary

CREATE VIEW v\_customer\_orders AS

SELECT

c.customer\_id,

c.first\_name,

c.last\_name,

COUNT(o.order\_id) AS total\_orders,

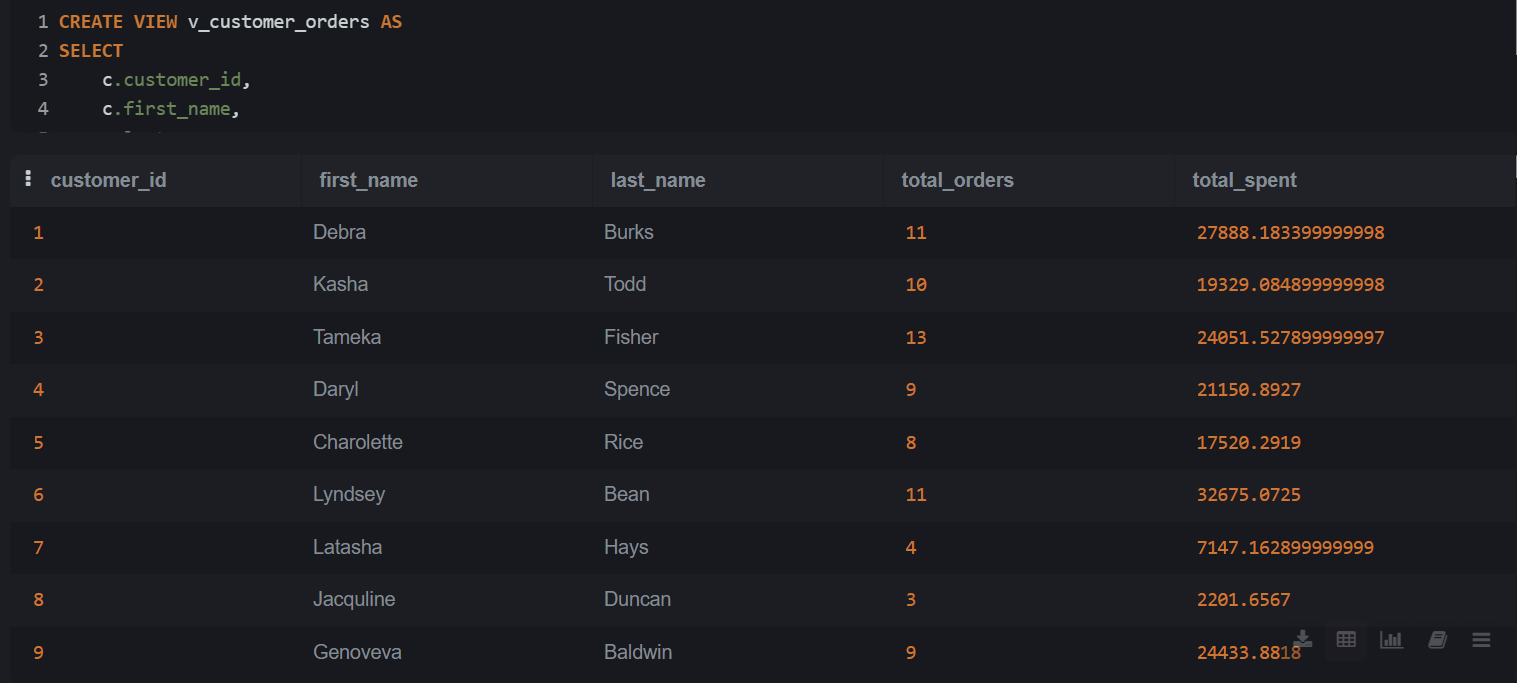
SUM(oi.quantity \* oi.list\_price \* (1 - oi.discount)) AS total\_spent

FROM customers c

JOIN orders o ON c.customer\_id = o.customer\_id

JOIN order\_items oi ON o.order\_id = oi.order\_id

GROUP BY c.customer\_id, c.first\_name, c.last\_name;



1. Top 5 best-selling products (by total quantity)

SELECT

p.product\_name,

SUM(oi.quantity) AS total\_sold

FROM order\_items oi

JOIN products p ON oi.product\_id = p.product\_id

GROUP BY p.product\_name

ORDER BY total\_sold DESC

LIMIT 5;

