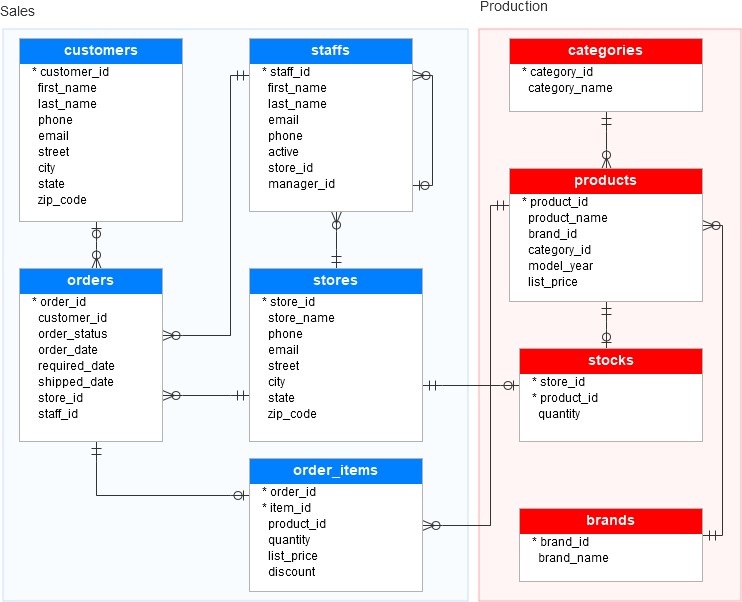
**Customer Sales and Analysis**

**Business problem:** A bike store management aims to maximize its product inventory and promotions in order to improve their sales performance. The management team also wants to examine previous sales data in order to find ways to enhance inventory control, advertise products that are well-liked, increase sales, and improve business strategy.

**Approach:** To enhance sales performance, I will conduct various analyses using SQL queries to gain a better insight into sales based on store, product, staff, and brand performance.



**Find the top 5 best-selling products based its total revenue**

SELECT

p.product\_id,

p.product\_name, oi.list\_price,

SUM(oi.quantity) AS total\_units\_sold,

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

order\_items oi INNER JOIN

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

GROUP BY

p.product\_id,

p.product\_name ORDER BY

total\_revenue DESC

LIMIT 5;

A screenshot of a list

Description automatically generated

**-Total sales based on stores**

SELECT

s.store\_id,

s.store\_name,

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

FROM stores s

JOIN

orders o ON s.store\_id = o.store\_id

JOIN

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

GROUP BY

s.store\_id, s.store\_name

A screenshot of a phone

Description automatically generated

**- Average units sold for each month of the year, for each product category**

SELECT

c.category\_name,

AVG(oi.quantity) AS avg\_units\_sold,

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

FROM

order\_items oi

INNER JOIN

orders o ON oi.order\_id = o.order\_id

INNER JOIN

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

INNER JOIN

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

GROUP BY

c.category\_name

order by total\_revenue desc

ORDER BY

year,

month,

category\_name;

A screenshot of a graph

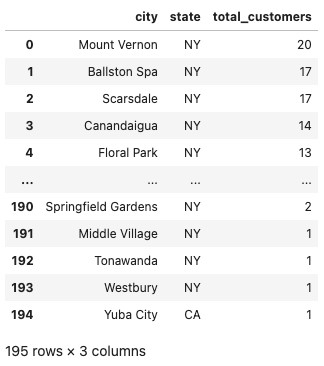
Description automatically generated

**- What city and state have the highest customer concentration?**

SELECT city, state,

COUNT (DISTINCT customer\_id) AS total\_customers FROM customers GROUP BY city, state ORDER BY

total\_customers DESC;



**Top 5 staffs with the highest sales performance**

SELECT

s.staff\_id,

s.first\_name || ' ' || s.last\_name AS staff\_name,

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

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

FROM staffs s

JOIN orders o ON s.staff\_id = o.staff\_id

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

GROUP BY s.staff\_id, staff\_name

ORDER BY total\_sales DESC;

A screenshot of a table

Description automatically generated

**Total Units Sold based on each Brand.**

SELECT

b.brand\_id,

b.brand\_name,

SUM(oi.quantity) AS total\_units\_sold

FROM brands b

JOIN products p ON b.brand\_id = p.brand\_id

JOIN order\_items oi ON p.product\_id = oi.product\_id

GROUP BY b.brand\_id, b.brand\_name

ORDER BY total\_units\_sold DESC;

A screenshot of a cell phone

Description automatically generated

**Yearly Total Sales trend**

SELECT

Extract(year from o.order\_date) AS year,

ROUND(SUM(oi.list\_price \* oi.quantity \* (1 - oi.discount)), 2) AS yearly\_revenue

FROM

orders o

JOIN

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

GROUP BY

year

ORDER BY

yearly\_revenue desc;

A screenshot of a graph

Description automatically generated

**Visualization**

[A screenshot of a graph

Description automatically generated](https://public.tableau.com/app/profile/fathi.a4478/viz/Bikestore_17122724016570/Dashboard2#1)

[Click to view.](https://public.tableau.com/app/profile/fathi.a4478/viz/Bikestore_17122724016570/Dashboard2#1)