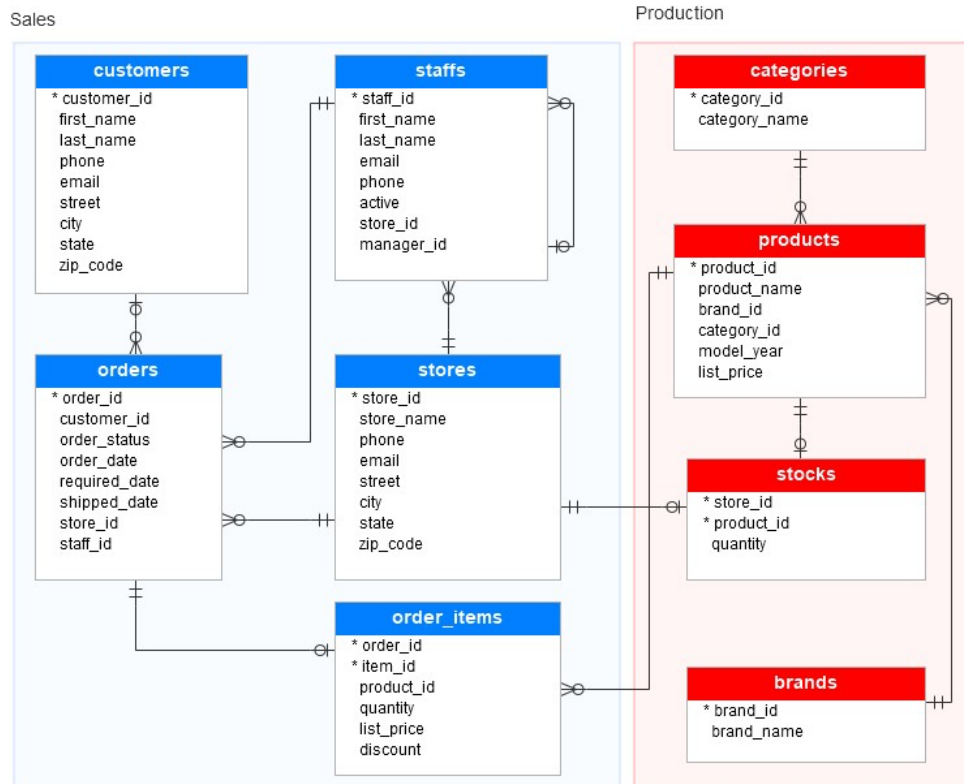


# Customer Sales and Analysis

**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 and 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_units_sold DESC
LIMIT 5;

```

	product_id	product_name	list_price	total_units_sold	total_revenue
0	6	Surly Ice Cream Truck Frameset - 2016	469.99	167	70371.6027
1	13	Electra Cruiser 1 (24-Inch) - 2016	269.99	157	37992.9928
2	16	Electra Townie Original 7D EQ - 2016	599.99	156	82744.6209
3	7	Trek Slash 8 27.5 - 2016	3999.99	154	555558.6111
4	23	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	299.99	154	41011.6329

#### -Rank stores based on their total revenues

```

WITH StoreRevenue AS (
    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
) SELECT
    store_id,
    store_name,
    total_revenue,
    RANK() OVER (ORDER BY total_revenue DESC) AS store_rank
FROM
    StoreRevenue;

```

	store_id	store_name	total_revenue	store_rank
0	2	Baldwin Bikes	5.215751e+06	1
1	1	Santa Cruz Bikes	1.605823e+06	2
2	3	Rowlett Bikes	8.675422e+05	3

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

```

WITH AvgUnitsSold AS (
    SELECT
        strftime('%Y', order_date) AS year,
        strftime('%m',
order_date) AS month,
        c.category_name,
        AVG(oi.quantity) AS avg_units_sold
    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
        strftime('%Y', order_date),
        strftime('%m', order_date),
        c.category_name
)
SELECT
    year,
    month,
    category_name,
    avg_units_sold
FROM
    AvgUnitsSold
ORDER BY
    year, month,
category_name;

```

	year	month	category_name	avg_units_sold
0	2016	01	Children Bicycles	1.500000
1	2016	01	Comfort Bicycles	1.636364
2	2016	01	Cruisers Bicycles	1.518519
3	2016	01	Cyclocross Bicycles	1.636364
4	2016	01	Electric Bikes	1.666667
...	...	...	...	...
202	2018	11	Mountain Bikes	1.000000
203	2018	11	Road Bikes	1.500000
204	2018	12	Cruisers Bicycles	1.000000
205	2018	12	Electric Bikes	2.000000
206	2018	12	Road Bikes	1.000000

207 rows x 4 columns

- What city and state has 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;
```

	city	state	total_customers
0	Mount Vernon	NY	20
1	Ballston Spa	NY	17
2	Scarsdale	NY	17
3	Canandaigua	NY	14
4	Floral Park	NY	13
...	...	...	...
190	Springfield Gardens	NY	2
191	Middle Village	NY	1
192	Tonawanda	NY	1
193	Westbury	NY	1
194	Yuba City	CA	1

195 rows x 3 columns

### 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 sales.staffs s
    JOIN sales.orders o ON s.staff_id = o.staff_id
    JOIN sales.order_items oi ON o.order_id = oi.order_id
GROUP BY s.staff_id, staff_name
ORDER BY total_sales DESC;
```

staff_id	staff_name	total_orders	total_sales
6	Marcelene Boyer	1615	2624120.7
7	Venita Daniel	1580	2591630.6
3	Genna Serrano	544	853287.4
2	Mireya Copeland	462	752535.7
8	Kali Vargas	269	463918.3
9	Layla Terrell	252	403623.9

### 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;
```

brand_id	brand_name	total_units_sold
1	Electra	2612
9	Trek	1839
8	Surly	908
7	Sun Bicycles	731
4	Pure Cycles	376
2	Haro	331
3	Heller	138
5	Ritchey	118
6	Strider	25

#### Yearly Total Sales trend

```

SELECT
    EXTRACT(MONTH FROM o.order_date) AS month,
    Round(SUM(oi.list_price * oi.quantity * (1 - oi.discount)),2) AS
monthly_revenue
FROM sales.orders o
    JOIN sales.order_items oi ON o.order_id = oi.order_id
GROUP BY month ORDER BY month;

```

year	yearly_revenue
2016	2427378.53
2017	3447208.24
2018	1814529.79