

# Sales Analysis

using sql

By Kavya Parihar





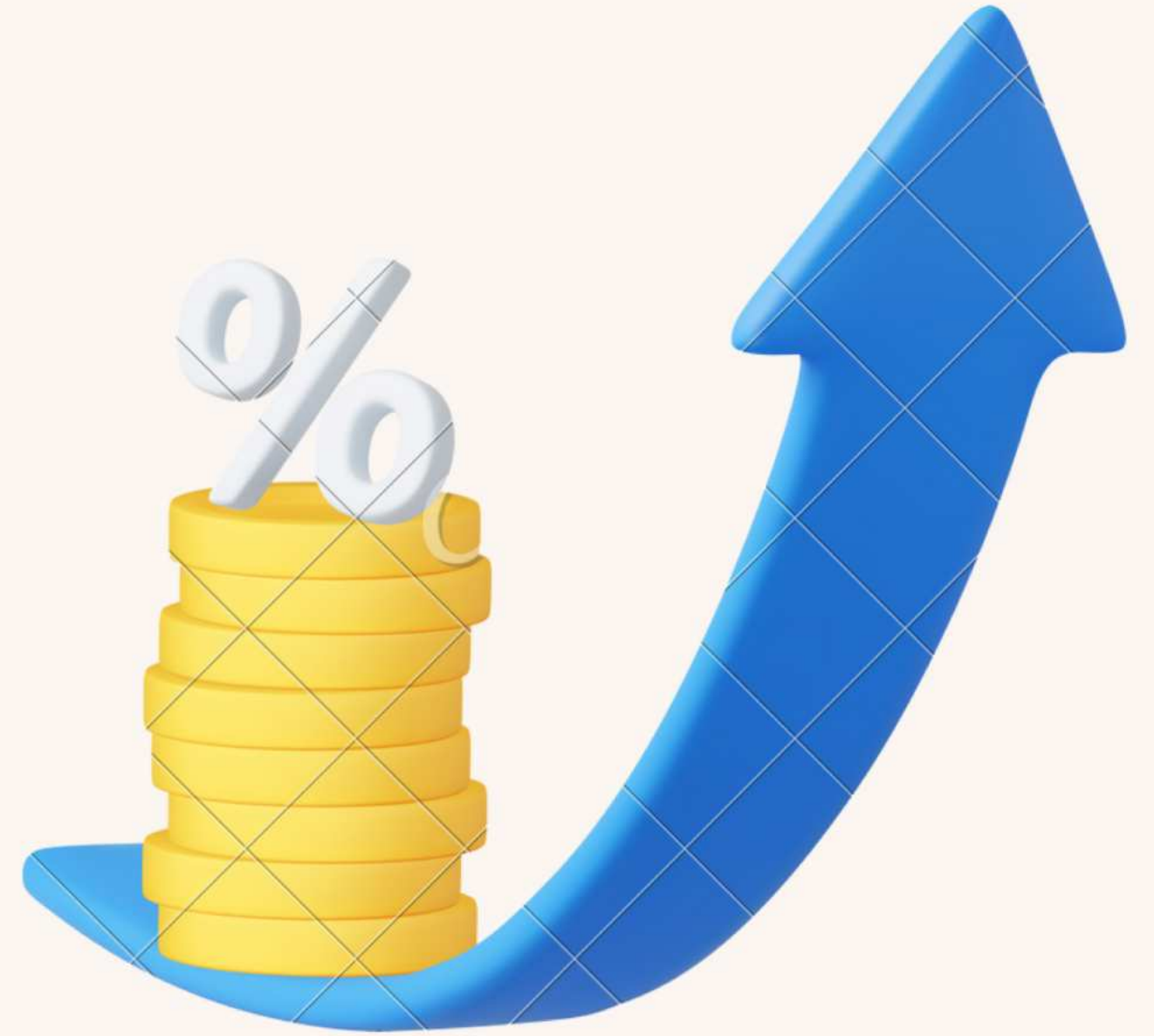
# Project Overview

The project focuses on sales data analysis and customer relationship management(CRM) analysis over a one-year period to derive actionable insights using SQL. The project goal is to derive insights of sales patterns, customer behaviors and category trends.



# Objectives of this project

- Analyze sales performance
- Understand customer spending habits.
- Evaluate regional sales trends
- assess category performance.





# Methodology

## 1. Data Collection and Preparation

### Datasets Used:

- **Order\_Details:** Contains detailed order information (order ID, quantity, amount, , category, subcategory)
- **Order\_Summary:** Summarizes orders (order ID, order date, customer name, state, city ).
- **Sales\_Target:** Includes sales targets for each month and each category(date, category, target)

### Data Cleaning:

- **Tools Used:** Excel

## 2. Sql analysis

- **Analyze sales performance-**Calculated key metrics such as total revenue and average order value
- **Evaluate regional and customer sales trends-**Analyzed sales data to identify trends by region and customer spending habits
- **Assess category performance-**Evaluated sales performance across different product categories .

## 3. Documenting

Compiled and summarized key findings.

Created comprehensive documentation detailing the analysis process, findings, and actionable insights.

# Sales analysis

## Order Per Day:

```
select os.order_date, sum(od.amount*od.quantity) from  
order_summary os inner join order_details od  
on os.order_id=od.order_id  
group by os.order_date  
order by os.order_date;
```

## Total Sum of Quantities Sold Under Each Order ID:

```
select a.order_id ,sum(a.quantity) from  
order_details a inner join order_summary b on  
a.order_id=b.order_id  
group by a.order_id  
order by order_id;
```

## highest and lowest quantity sold

```
(select count(quantity), category from order_details
```

```
group by category
```

```
order by count(quantity) desc
```

```
limit 1)
```

```
union all(
```

```
select count(quantity), category from order_details
```

```
group by category
```

```
order by count(quantity)
```

```
limit 1);
```



# Difference Between Monthly Target and Monthly Sales by Each Category

```
select st.order_date,sq.months,st.category,sq.monthly_total_sales,st.target,st.target-sq.monthly_total_sales as difference
from sales_target st left join (
select a.category, sum(a.amount*a.quantity) as monthly_total_sales ,extract (month from b.order_date) as months
from order_details a inner join order_summary b
on a.order_id=b.order_id
where extract(month from b.order_date) in (select extract(month from st.order_date) from sales_target st)
group by extract (year from b.order_date),months,a.category )sq on extract(month from st.order_date)=sq.months
and st.category =sq.category
order by extract (year from st.order_date), sq.months,st.category;
```

## Identify Orders Where the Target Wasn't Met

```
select sbq.months,st.target,sbq.monthly_sales,sbq.category from sales_target st left join (
select sum(a.amount*a.quantity) as monthly_sales, extract(month from b.order_date) as months,a.category
from order_summary b inner join order_details a
on a.order_id=b.order_id
where extract(month from order_date) in (select extract(month from st.order_date) from sales_target st)
group by months,a.category) as sbq on extract(month from st.order_date) = sbq.months and st.category=sbq.category
where st.target>sbq.monthly_sales
order by sbq.months;
```

calculate the percentage of the highest monthly sales compared to the total yearly sales

```
select (bc.total_sales/ab.total_sales_yearly)*100 from (select sum(amount*quantity) as  
total_sales_yearly  
from order_details) ab cross join  
(select sum(a.amount*a.quantity) as total_sales, extract(month from b.order_date) as months  
from order_details a inner join order_summary b on a.order_id=b.order_id  
group by months  
order by total_sales desc  
limit 1) bc ;
```



# CRM Analysis

List the top 3 customers with the highest total purchase amount for each month

```
select customer_name,total_sales,years,months, sales_rank from(
select os.customer_name , sum(od.amount*od.quantity) as total_sales,extract(year from os.order_date)
as years ,extract(month from os.order_date)
as months,rank()
over(partition by extract(month from os.order_date) order by sum(od.amount*od.quantity) desc ) as
sales_rank
from order_summary os inner join order_details od
on os.order_id=od.order_id
group by years,months,os.customer_name,os.state,os.city)
where sales_rank <=3
order by years,months,sales_rank;
```

how many customers purchased more than once

```
select a.customer_name, count( distinct (b.order_id)) as frq
from order_details b inner join order_summary a on a.order_id=b.order_id
group by a.customer_name,a.state,a.city
having count( distinct (b.order_id))>1
order by frq desc;
```

top 3 customers based on their purchases

```
select os.customer_name, sum(od.amount*od.quantity) as total_order
from order_summary os inner join order_details od on os.order_id=od.order_id
group by os.customer_name,os.state,os.city
order by total_order desc
limit 3
```



## Top 3 Highest Sales Cities

```
select city from (  
select b.city,sum(a.amount*a.quantity) as monthly_sales from order_details a inner join order_summary  
b  
on a.order_id=b.order_id  
group by b.state,b.city  
order by monthly_sales desc)  
limit 3;
```

## Cities where these categories are performing well

```
select category,city,sales from (  
select a.category,b.city,sum(a.amount*a.quantity) as sales  
from order_details a inner join order_summary b  
on a.order_id=b.order_id  
group by a.category,b.state,b.city  
order by a.category,sales desc);
```

lowest sale city

```
select b.city,sum(a.amount*a.quantity) as monthly_sales  
from order_details a inner join order_summary b  
on a.order_id=b.order_id  
group by b.state,b.city  
order by monthly_sales )  
limit 1;
```



# Category Analysis

monthly sales by each category

```
select a.category, sum(a.amount*a.quantity) as ts,  
extract (year from b.order_date) as years ,  
extract(month from b.order_date) as months from order_details a inner join order_summary b  
on a.order_id=b.order_id  
group by rollup (years, months,a.category)  
order by years, months,a.category;
```

which category has generated the highest revenue

```
select sum(amount*quantity) as total_sales,  
category from order_details  
group by category  
order by total_sales desc  
limit 1;
```

which category has highest sales in each month

```
select category, total_sales, years, months from (  
select a.category, sum(a.amount*a.quantity) total_sales,  
extract (year from b.order_date) as years ,  
extract(month from b.order_date) as months,  
rank() over(partition by extract(month from b.order_date) order by sum(a.amount*a.quantity) desc)as  
ranks  
from order_details a inner join order_summary b  
on a.order_id=b.order_id  
group by(years, months,a.category))  
where ranks=1  
order by years, months ;
```



Identify the month with the highest sales for each category in the year

```
select category,months from(  
select extract(month from b.order_date)as months,  
a.category,sum(a.amount*a.quantity) as total_sales_pm,  
rank() over(partition by category order by sum(a.amount*a.quantity)desc ) as ranks  
from order_details a inner join order_summary b on a.order_id=b.order_id  
group by a.category,months)  
where ranks=1;
```

Determine the category with the highest growth in sales compared to the previous month

```
select category,max(next_ms-ts)as growth from(
select category, ts,lead(ts) over( partition by category order by category ) as next_ms,months from(
select a.category,
sum(a.amount*a.quantity) as ts,extract (year from b.order_date) as years,
extract(month from b.order_date) as months from order_details a inner join order_summary b
on a.order_id=b.order_id
group by years, months,a.category
order by a.category,years, months) )
group by category
order by growth desc
limit 1;
```



# KEY FINDINGS

## SALES OVERVIEW:

- Total sales for the period Apr 2018 to Mar 2019 - Rs 2146870.00
- Highest revenue month -Jan 2019 of Rs 337229.00 (15% of total sales)
- The month and category in which sales target was not met - clothing in 7th month of 2018.
- Highest growth in sales compared to last month - furniture category in Jan 2019 (growth -2.8 times)

## CATEGORY PERFORMANCE:

- Highest revenue category- Electronics  
amount generated-Rs 816583 (38% of total sales. )
- Highest quantity sold - clothing
- Lowest quantity sold - furniture
- Highest revenue generated (month and city)  
Clothing - march 2019, Indore  
Electronics - Jan 2019 ,Indore  
Furniture -March 2019.,Indore

## CUSTOMERS AND GEOGRAPHIC INSIGHTS:

- Top 3 cities with highest sales  
Indore  
Mumbai  
Pune
- City with lowest sale  
Delhi
- Top 3 customers according to their purchases  
Yaanvi  
Seema  
Saumya
- No. of customers who purchased more than once-70

# CONCLUSION

In conclusion, the sales data from April 2018 to March 2019 highlights a decent performance of the business as sales targets were surpassed in almost every month with Jan month witnessing the highest sales. Key highlights include significant revenue contributions from the Electronics category. Cities like Indore, Mumbai, Pune has the highest sales and attention should be given to cities with lower sales performance like Delhi to identify opportunities for growth.

Business also has a loyal customer base which is beneficial for the business and to sustain this performance or to enhance it addressing underperforming areas, and improving customer retention strategies would be beneficial.



Thank  
you very  
much!

