

Bazaar Case Study Analytics

بازار

BAZAAR

Presented by
Muhammad Hamza

 The buttons/arrows in this presentation are clickable

Agenda

- 01. Introduction →
- 02. Data Exploration →
- 03. Analytical Questions →
- 04. Data Visualization & Insights →
- 05. Recommendations →





Back to Agenda

Introduction

This presentation showcases findings and recommendations derived from an analysis of sales transactions data from a retail company. Through SQL querying, analytical techniques, and data visualization tools, we are able to observe various aspects of business performance.



Back to Agenda

Note for the viewer

```
create table analytics_data (
    order_date DATE,
    order_number BIGINT,
    order_warehouse_id INTEGER,
    store_id VARCHAR,
    order_status VARCHAR,
    item_id VARCHAR,
    amount_per_unit NUMERIC,
    ordered_quantity INTEGER,
    item_discount NUMERIC,
    expected_delivery_date DATE
);
```

- The following table is accessed as public.analytics_data in further queries.
- Through data exploration I identified “order_warehouse_id” had 5 rows which had null values, but I decided not to remove them since they were part of a huge number of orders, hence, removal would have affected the calculations made.
- For every visualization/query insights are provided in that section .



Back to Agenda

Data Exploration

- 01. Total number of orders →
- 02. Total Sales Revenue →
- 03. Average order Quantity →
- 04. Distribution of orders by warehouse & store →
- 05. Top selling items →
- 06. Insights →





Main Section

Insights



Key findings

Task 1:

Total number
of orders

```
SELECT SUM(ordered_quantity)  
AS total_orders  
FROM public.analytics_data;
```

571K+ orders

Task 2:

Total Sales
Revenue

```
SELECT  
SUM(amount_per_unit * ordered_quantity)  
AS total_sales_revenue  
FROM public.analytics_data;
```

278M sales revenue

Task 3:

Average order
Quantity

```
SELECT AVG(ordered_quantity)  
AS avg_order_quantity  
FROM public.analytics_data;
```

22.8

Distribution of orders by warehouse & store

Task 4:

```
SELECT order_warehouse_id, store_id, COUNT(*) AS order_count
FROM public.analytics_data
GROUP BY order_warehouse_id, store_id
ORDER BY order_warehouse_id, store_id;
```

For the following Query the output included unordered ID's therefore the top 10 rows are shown in descending order of order_count.

```
ORDER BY order_count DESC;
```

order_warehouse_id integer	store_id character varying	order_count bigint
15	5qQncdfXpBQ6cwiFP9nd87	78
2	FOxy6fUZZUGqy6yqejuUa	71
9	70rkqCk86iNc8FedKQ4M...	56
8	1TrqGzT37l31lsrYN5Kyn8	55
3	190iW3Fd0OqHa0RnzTh...	53
2	3gdPnWstu13FnBWyUuzjEn	51
4	1yCyaUMSjUNaMPA4dBM...	51
2	2nDuQzdvEGeSPB6yjzJJf8	51
9	2P5euTl4oSJGtqxb8LE5Y	48
9	6GCJ8YtCRtPVFOz7dxhqkJ	48



Main Section

Insights



Top selling items

Task 5:

```
SELECT item_id, SUM(ordered_quantity) AS top_selling_items
FROM public.analytics_data
GROUP BY item_id
ORDER BY top_selling_items DESC;
```

	item_id character varying	top_selling_items bigint
1	P2RwFPjdKJ0oFr9OoVmrD...	400000
2	P5914112688284	20926
3	P227644615428796458	12108
4	P5678188101788	7379
5	P193053596797216954	5715
6	P881311747325487679	5006
7	P037963480594898300	2874
8	P5665770111132	2569
9	P358836009799511708	2372
10	P343784657338674125	2001



Insights

Task 1:

- High order volume, indicating a strong demand of products and efficient operational capacity to handle large volumes.

Task 2:

- Reflects strong market presence and successful sales strategies. This translates to an average revenue of approximately \$487.72 per order.

Task 3:

- Indicating customer's reliance on bulk sales rather than small, frequent purchases.



Insights

Task 4:

- Reflects customer preferences and purchasing behavior in specific areas where the stores are located.
- Best performing warehouses and stores can be identified.

Task 5:

- Gain insight about popular items which enhance brand visibility and recognition.
- Gain insight about items which increase the company's market share.



Back to Agenda

Analytical Questions

- 01. Overall Discount rate →
- 02. Warehouse with the highest avg order value →
- 03. Total revenue generated by each store →
- 04. Top 5 customers based on total amount spent →
- 05. Month over Month growth rate →
- 06. % of Orders that got cancelled →
- 07. Insights →





Overall Discount rate (Avg discount per item sold)

Task 1:

```
SELECT
COALESCE(
AVG(
NULLIF(
ABS(item_discount) / NULLIF(amount_per_unit, 0), 0)),0)
AS avg_discount_rate
FROM public.analytics_data;
```

`ABS()` function is used on “`item_discount`” column in order to deal with the negative values in the dataset.



70.6% is the average discount rate on every item that sold

Warehouse with the highest avg order value

Task 2:

```
SELECT
    order_warehouse_id,
    SUM(amount_per_unit * ordered_quantity) AS total_sales_revenue,
    COUNT(*) AS number_of_orders,
    SUM(amount_per_unit * ordered_quantity) / COUNT(*) AS avg_order_value
FROM
    public.analytics_data
WHERE
    order_warehouse_id IS NOT NULL
GROUP BY
    order_warehouse_id
ORDER BY
    avg_order_value DESC
LIMIT
    1;
```

Warehouse ID: 14

Avg order value: 76K+

The condition of “order_warehouse_id IS NOT NULL” is applied in order to deal with 5 rows that were empty in the column of “order_warehouse_id”.

Total revenue generated by each store

Task 3:

```
SELECT
store_id,
SUM(amount_per_unit * ordered_quantity
    - ABS(item_discount)) AS total_revenue
FROM
public.analytics_data
GROUP BY
store_id;
```

The total revenue is calculated by subtracting the item discount as it focuses on the total income that is received.

For the following Query the output included unordered ID's therefore the top 10 rows are shown in descending order of total_revenue.

```
ORDER BY total_revenue DESC;
```

	store_id character varying	total_revenue numeric
1	7ldsLx849ptRljYZTK8tl8	3360070
2	2Z9CsLeVlPvooYgQOUjcRI	2616020
3	4PdVF4u8WXhO4xnSMCY...	2560000
4	49WcD3UqaC72tOQ6tood...	2030000
5	3ydAgE2Bkjbp6CvTfZjg7P	1880500
6	6FFhVY8Ujlw7dpqveE2UTh	1836788
7	6nEbvZQMYLEkxxN4JwV...	1819875
8	2IB2SBi7xRUgcwRUQUQfz6	1419946
9	HZ1kJk0NnaMJsWMcDsi...	1243620
10	4yMY1r3kwuZCmTyZiDK5...	1170520



Top 5 customers based on total amount spent

Task 4:

```
SELECT
store_id,
SUM(amount_per_unit * ordered_quantity)
    AS total_amount_spent
FROM
public.analytics_data
GROUP BY
store_id
ORDER BY
total_amount_spent DESC
LIMIT 5;
```

	store_id character varying	total_amount_spent numeric
1	5ZjIBaxM8y0r8SrbiLCnbD	160000000
2	3ydAgE2Bkjbp6CvTfZjg7P	3538028
3	7ldsLx849ptRljYZTK8tl8	3360070
4	2Z9CsLeVlPwooYgQOUjcRI	2616020
5	4PdVF4u8WXh04xnSMCYIzs	2560000

In the following case, according to our dataset, the customers are identified as stores.



Month over Month growth rate

Task 5:

```
WITH monthly_revenue AS (
  SELECT
    DATE_TRUNC('month', order_date) AS month,
    SUM(amount_per_unit * ordered_quantity) AS sales_revenue
  FROM
    public.analytics_data
  GROUP BY
    DATE_TRUNC('month', order_date)
  ORDER BY
    month
)
SELECT
  TO_CHAR(month, 'Month') AS month,
  sales_revenue,
  COALESCE(LAG(sales_revenue) OVER (ORDER BY month), 0) AS previous_month_revenue,
  CASE
    WHEN COALESCE(LAG(sales_revenue) OVER (ORDER BY month), 0) = 0 THEN 0
    ELSE COALESCE((sales_revenue - LAG(sales_revenue) OVER (ORDER BY month)) / COALESCE(LAG(sales_revenue) OVER (ORDER BY month), 0) * 100, 0)
  END AS growth_rate
FROM
  monthly_revenue;
```

	month text	sales_revenue numeric	previous_month_revenue numeric	growth_rate numeric
1	January	3965941.016	0	0
2	February	74260892.862	3965941.016	1772.4658930227519000
3	March	164541223.122	74260892.862	121.5718351619729800
4	April	3913002.43	164541223.122	-97.62187106929508922800
5	May	962576.04	3913002.43	-75.40057648264736702500
6	June	3961525.39	962576.04	311.5545396288899900
7	July	8279539.08	3961525.39	108.9987634788325800
8	August	7485046.98	8279539.08	-9.59584938634047730100
9	September	3021258.57	7485046.98	-59.63607739440000148100
10	October	2697012.952	3021258.57	-10.73213730263411383600
11	November	3672301.212	2697012.952	36.16179370873113997600
12	December	1702727.696	3672301.212	-53.63322348297610179800



Main Section

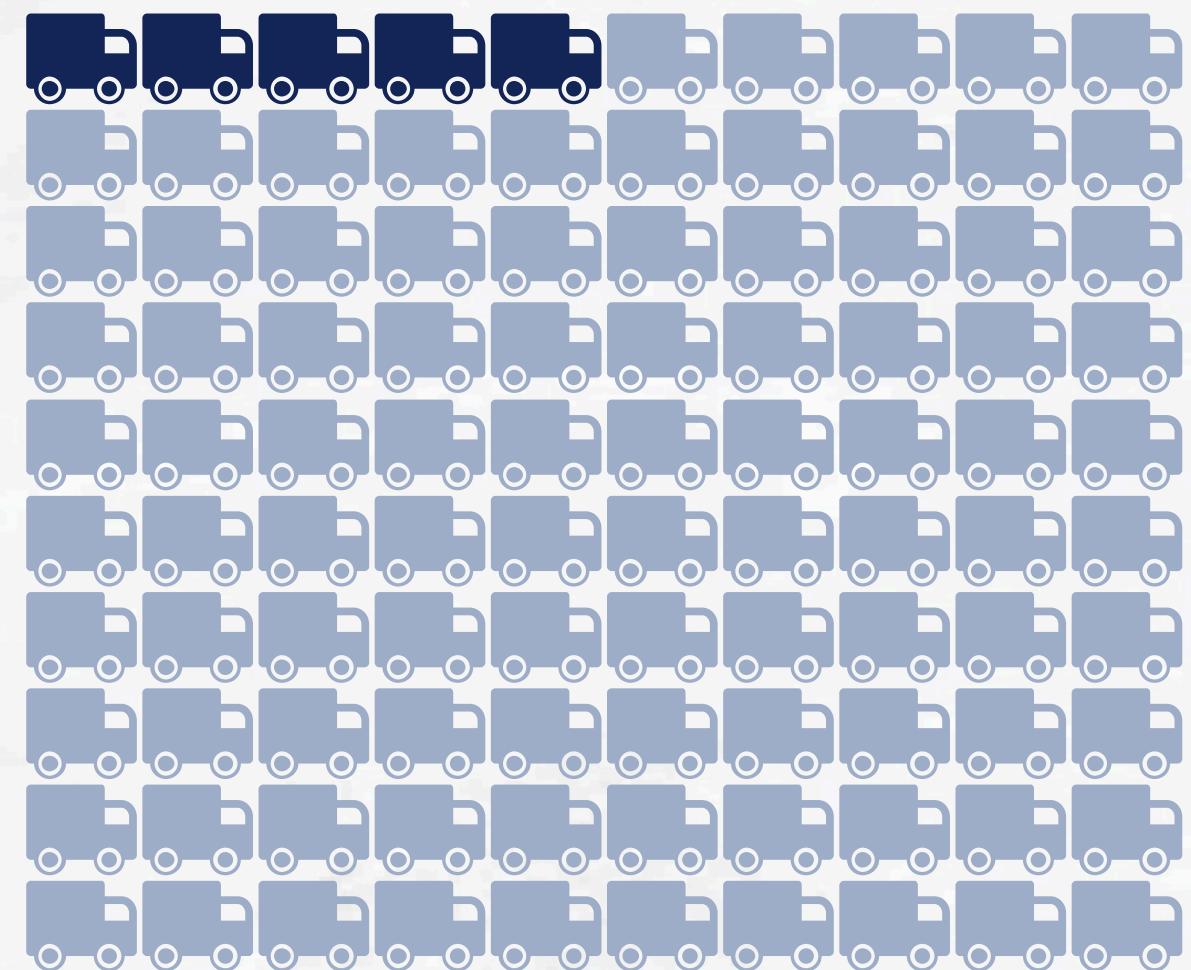
Insights



% of Orders that got cancelled

Task 6:

```
SELECT  
    (COUNT(CASE WHEN order_status = 'CANCELLED' THEN 1 END)  
     ::decimal / COUNT(*)) * 100 AS percentage_canceled_orders  
FROM  
    public.analytics_data;
```



5.52% orders got cancelled



Insights

Task 1:

- reflects that discounting is a significant component of the company's pricing strategy, which reflects the effort to attract customers.

Task 2:

- It reflects that this warehouse is leading in following the most efficient operational processes .
- Moreover the types of items that are driving high order value can provide insights into market demand and trend.

Task 3:

- This data helps reflect on the stores that contribute the most in overall revenue.
- It also highlights the discrepancies in revenue generation amongst stores, which helps identify potential areas for improvement.



Insights

Task 4:

- Helps identify the most significant contributors to overall sales.
- Indicates the strength of company's relationship with these customers.

Task 5:

- The data suggests a seasonal pattern with the months feb, march, june, july, and nov, having a positive growth rate. Whereas the rest of the months have a negative growth rate, reflecting fluctuation in sales revenue throughout the year.

Task 6:

- A low cancellation rate indicates a higher satisfaction level from the customers.



Back to Agenda

Data Visualization & Insights

- 01. Time Series plot of sales revenue over time ➔
- 02. Forecast on Time Series plot of sales revenue over time ➔
- 03. Top 10 stores with the most revenue ➔
- 04. Distribution of orders by warehouse ➔
- 05. Order Status based on quantity bought ➔
- 06. Quarterly Order Status ➔
- 07. Delivery Times over the year ➔

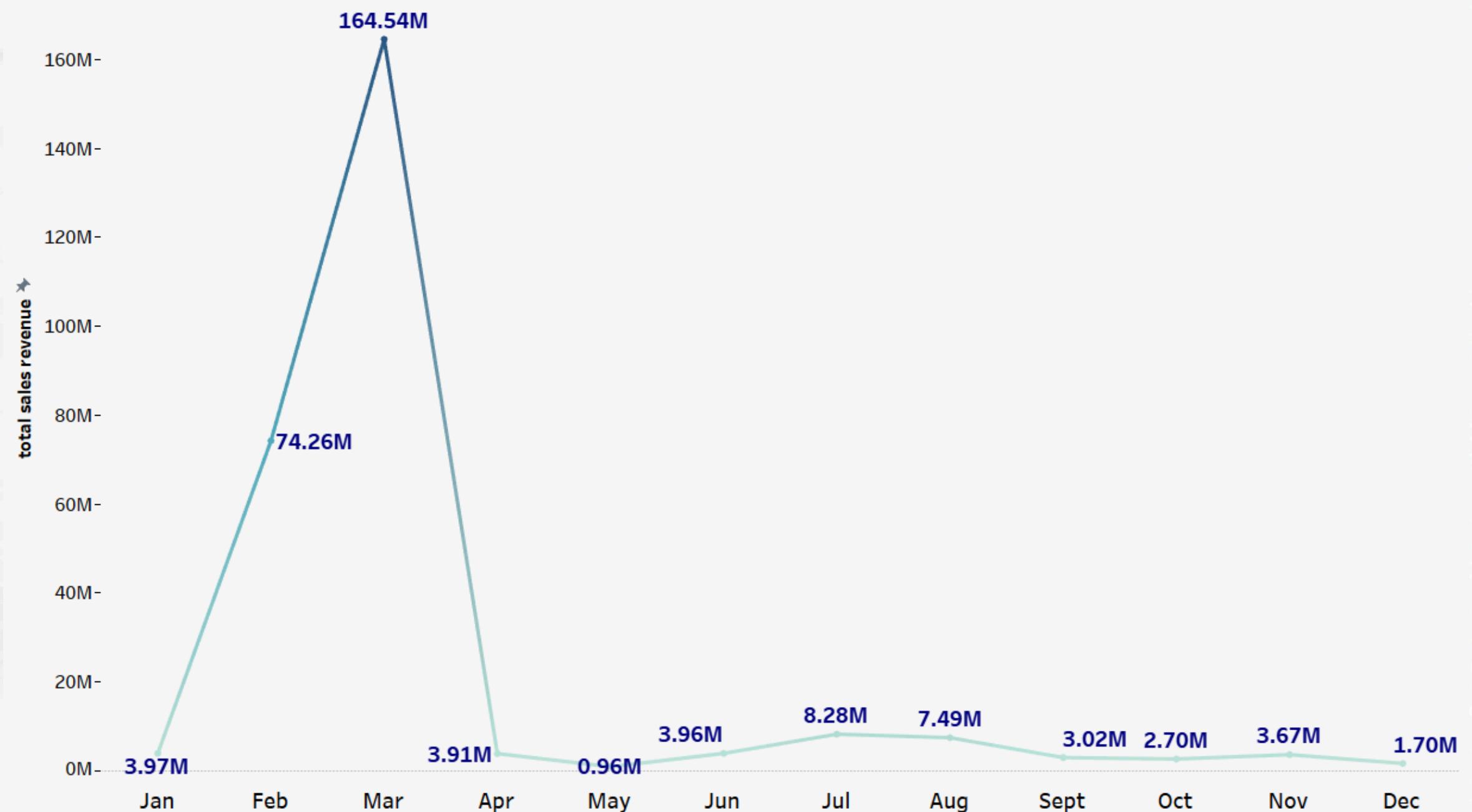




Main Section

Time Series plot of sales revenue over time

- Reflects a massive increase in sales revenue from January to March; March having the highest number of Sales throughout the year, and May having the least.

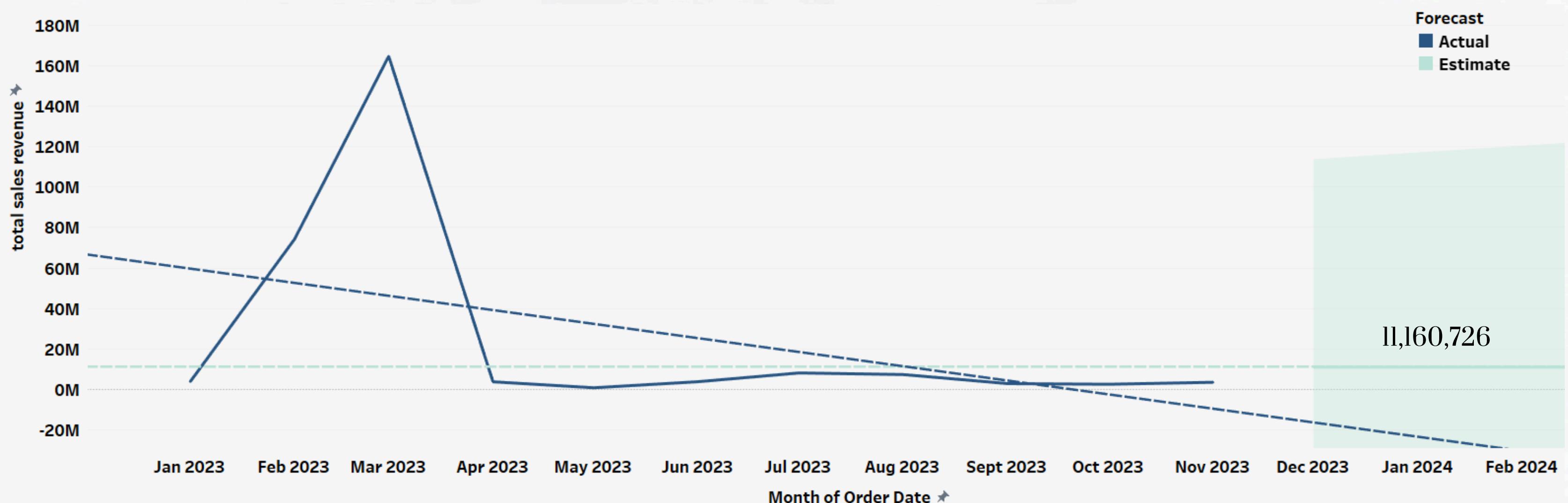




Main Section

Forecast on Time Series plot of sales revenue over time

- Reflects a decrease in the actual trend of the Sales revenue over time and the forecasting performed indicates Sales revenue of '11 Million' in the coming months.

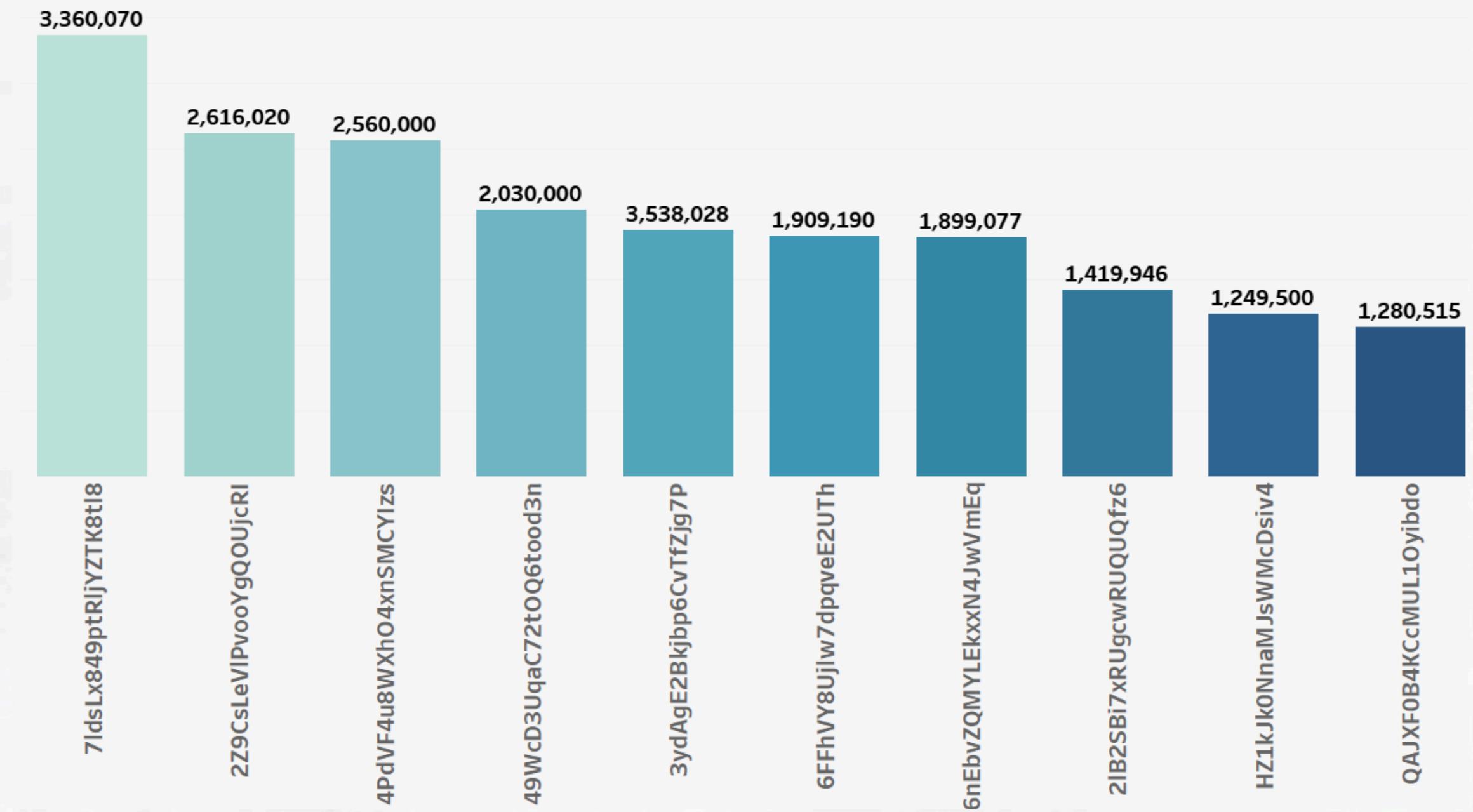




Main Section

Top 10 stores with the most revenue

- Visualizes the comparison of revenue performance among top stores, among which stores with the highest and lowest revenue can be seen on the graph.

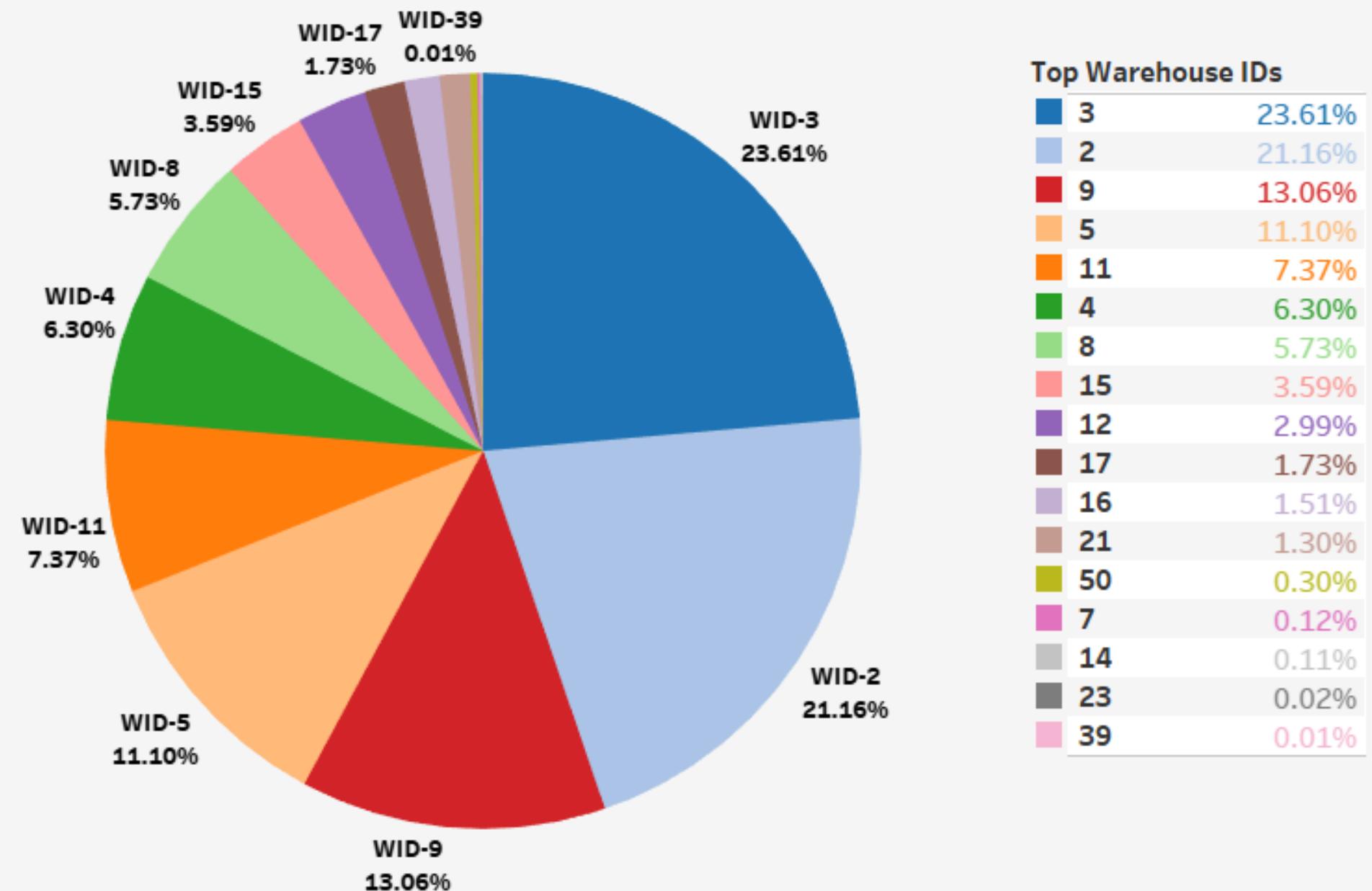




Main Section

Distribution of orders by warehouse

- Visualizes the proportion of total orders handled by each warehouse, showing how orders are distributed.
- Warehouse ID-3 has highest percentage of orders, and Warehouse ID-39 has lowest percentage of orders.

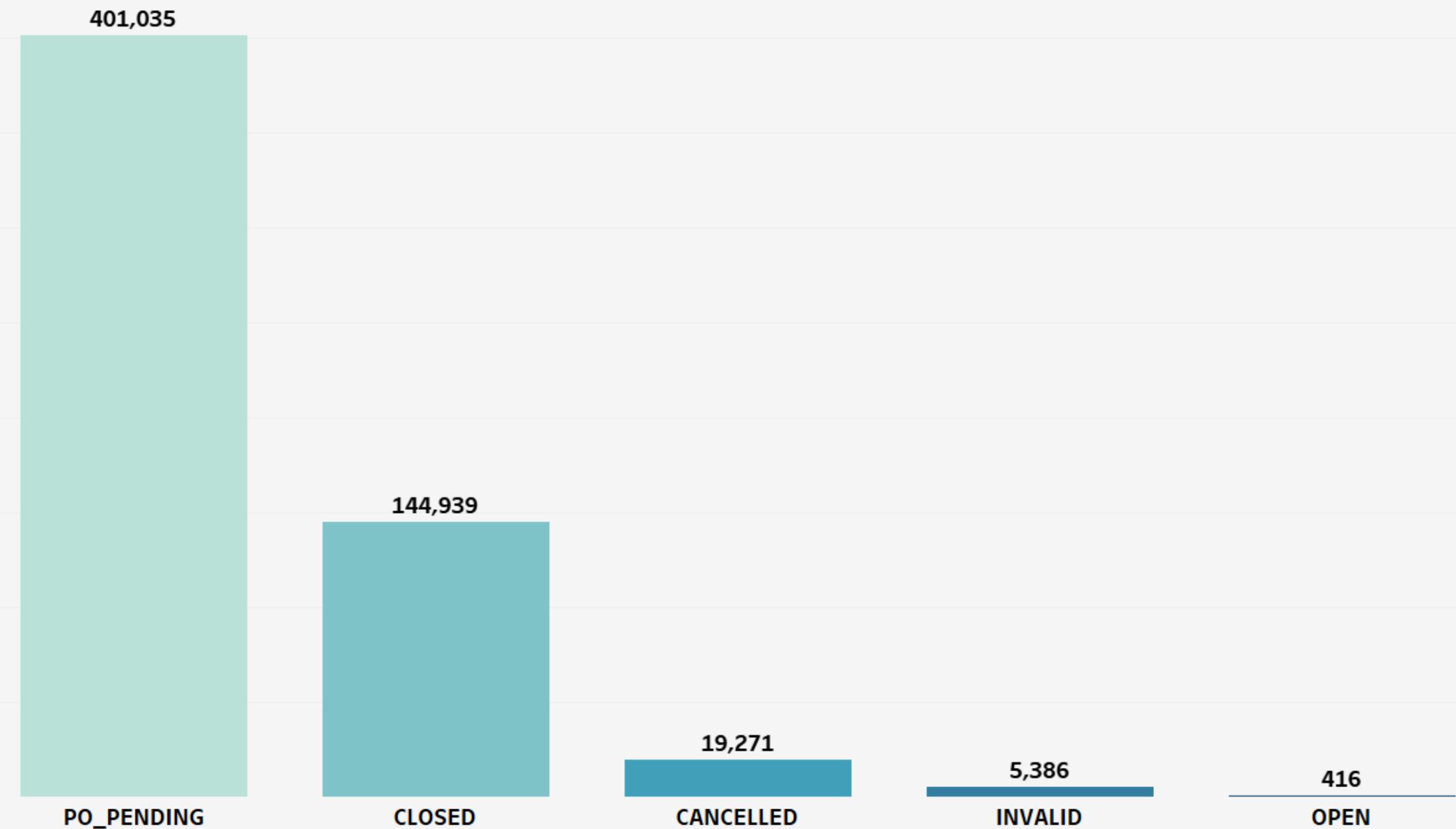




Main Section

Order Status based on quantity bought

- Shows most orders having “Pending” status and least orders having “Open” Status.

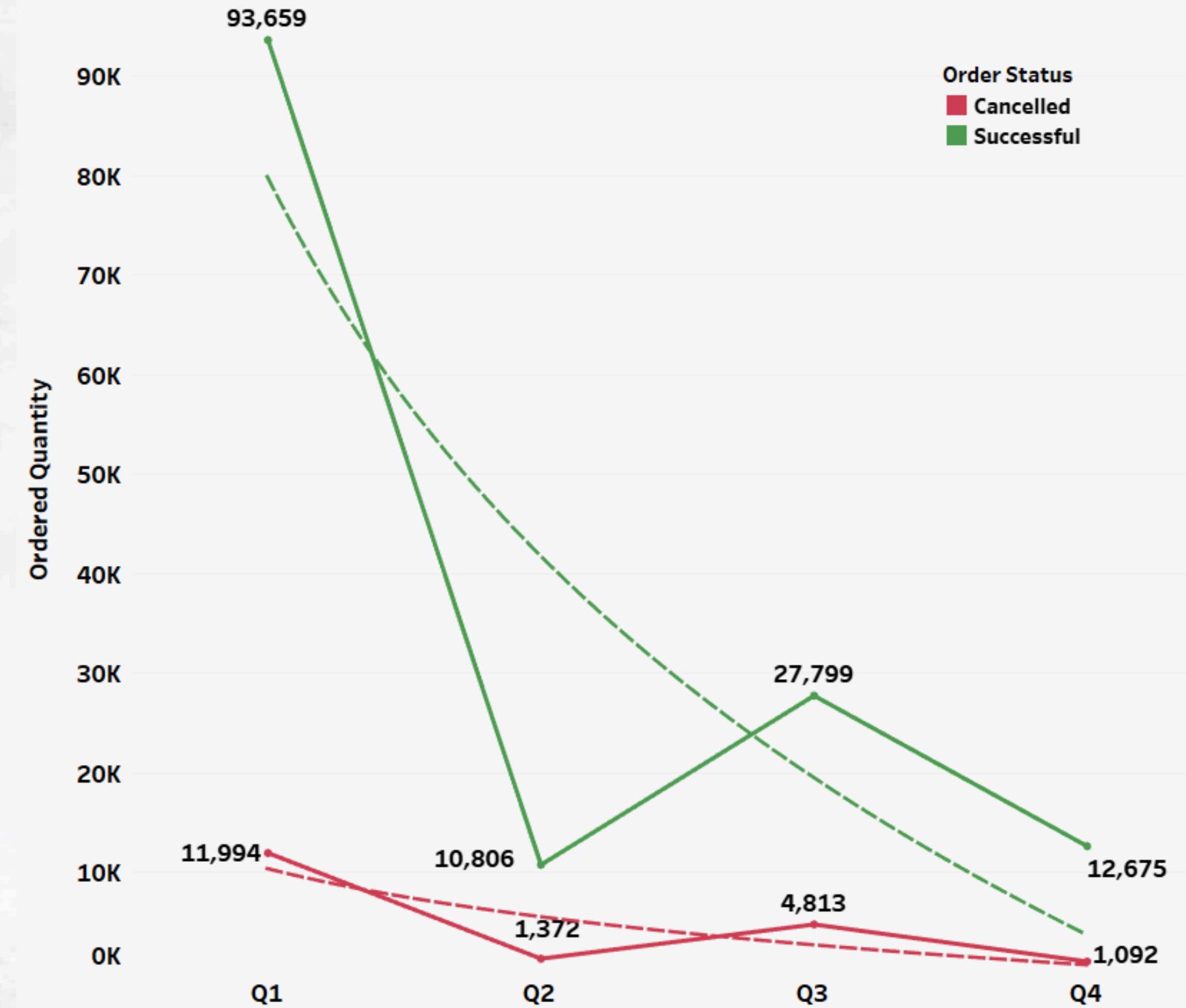




Main Section

Quarterly Order Status

- Shows that the trend of successful orders massively declined over the year.
- Whereas cancellation orders showed a slight consistent decline.

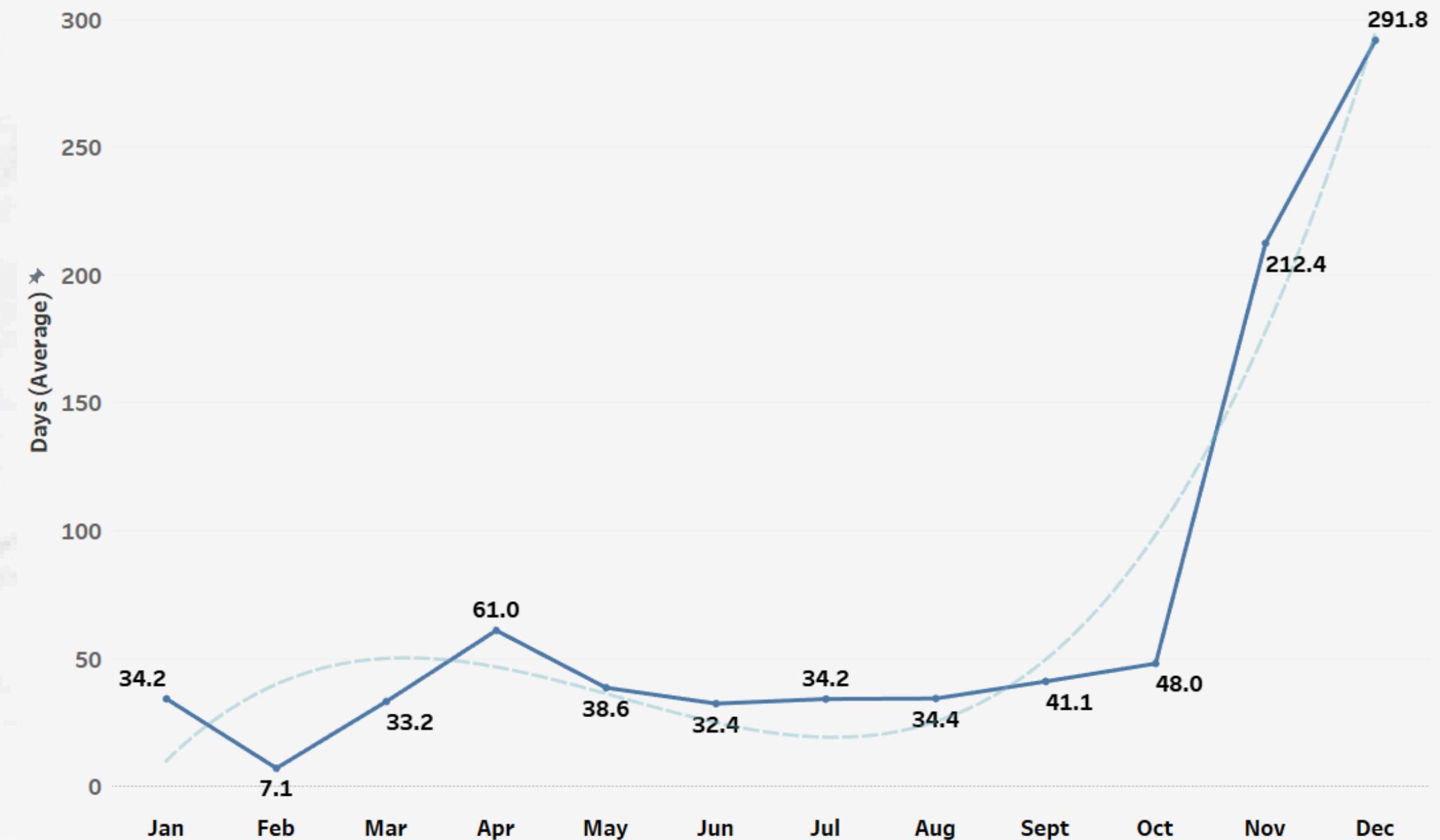




Main Section

Delivery Times over the year

- Shows an increasing trend in delivery time of orders over the year, which indicates a lack of performance in delivering items over the year.





Back to Agenda

Recommendations





Process Optimization:

- Conduct a thorough review of the order processing workflow to identify and eliminate bottlenecks. Streamline processes to ensure faster order processing and reduce pending statuses.

Dynamic Pricing:

- Implement strategies that adjust prices based on factors such as demand, inventory levels, and competitor pricing. This allows for flexible and targeted pricing decisions that maximize revenue and margins.

Item assortment optimization:

- Analyze the item mix at the high-performing warehouse/store to identify best-selling and high-margin items. Consider expanding the assortment of these items across other warehouses/stores to capitalize on their popularity and profitability.



Main Section

Discount optimization:

- An analysis to determine optimal discount rates that balance sales volume with profitability.

Economies of Scale:

- Knowledge of popular items may be used to manage production better, which in turn leads to lower per-unit costs.

Implement cross-selling and upselling strategies:

- Based on the product combinations that are driving high average order values at top-performing warehouse/stores, for other warehouses/stores.



Seasonal Planning:

- Develop a comprehensive seasonal sales strategy to capitalize on peak demand periods, and mitigate revenue declines during off-peak seasons, including targeted marketing campaigns and promotions.

Performance Monitoring:

- Implement regular monitoring of key performance indicators (KPIs) to track sales performance, revenue trends, and growth rates, enabling timely identification of anomalies and proactive response.

Lower Cancellation Rate:

- Conduct a thorough review of the order processing workflow to identify and eliminate bottlenecks, and handle cancellation requests by effectively, offering alternatives such as product substitutions, exchanges, or resolution of issues.

بازار
BAZAAR

Thankyou

Presented by
Muhammad Hamza