RETAIL STORE INVENTORY OPTIMIZATION

Using SQL and Power BI

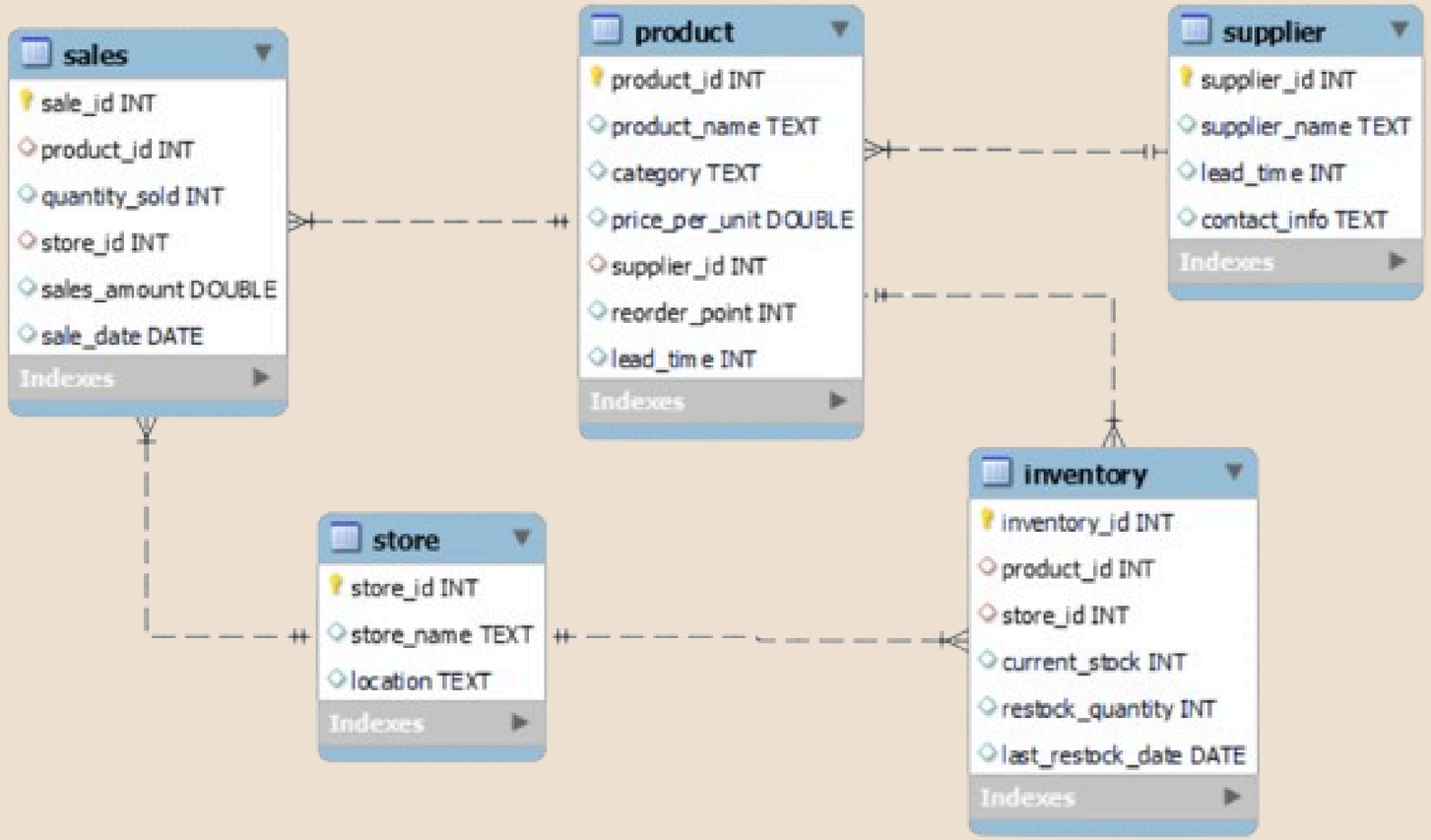
Presented By : Monika Dangwal

OVERVIEW

This project focuses on optimizing retail store inventory management to address common challenges like overstocking, stockouts, and inefficient supplier management. By writing SQL queries to analyze data and using Power BI for visualization, the goal is to ensure balanced stock levels, improve sales performance, and enhance supplier efficiency, ultimately maximizing profitability and customer satisfaction.

ABOUT DATA

The dataset is generated using Python's random library and consists of 5 CSV files: inventory, sales, products, suppliers, and stores. The ER diagram illustrates the relationships between these tables, allowing for effective analysis of inventory optimization and sales trends.



SQL QUERIES

(written some queries using aggregate functions, joins, group by clauses)

Inventory Management

WHAT IS THE TOTAL VALUE OF THE CURRENT INVENTORY?

```
select round(sum((i.current_stock * p.price_per_unit)), 2) as inventory_value
from inventory i
join product p on i.product_id = p.product_id;
```

	inventory_value
▶	5122057.97

Inventory Management

WHICH PRODUCTS NEED IMMEDIATE RESTOCKING BASED ON CURRENT STOCK AND REORDER POINT?

```sql
select p.product_name, i.current_stock, p.reorder_point
from inventory as i
join product as p on i.product_id = p.product_id
where i.current_stock < p.reorder_point;
```

product_name	current_stock	reorder_point
T-shirt	4	9
T-shirt	0	9
Smartphone	9	16
Action Figure	2	16
Action Figure	13	16
Toy Car	2	7
Jeans	12	13

Inventory Management

HOW QUICKLY ARE PRODUCTS BEING SOLD AND REPLACED?

```sql
select p.product_name, round(SUM(s.quantity_sold) / AVG(i.current_stock), 2) as "turnover_ratio"
from sales as s
join inventory as i on s.product_id = i.product_id
join product as p on p.product_id = s.product_id
group by p.product_name
order by turnover_ratio desc;
```

product_name	turnover_ratio
Smart Home Device	162.51
Spices	147.78
Pillow	141.08
Yogurt	130.24
Juice	119.66
Stuffed Animal	114.19
Honey	111.92

Sales Performance

WHICH PRODUCTS GENERATE THE MOST SALES REVENUE?

```sql
select s.product_id, p.product_name,
round(sum(s.sales_amount), 2) as "Total_sales"
from sales as s
join product as p
on s.product_id = p.product_id
group by product_id
order by Total_sales desc;
```

product_id	product_name	Total_sales
6	Jeans	7982.09
73	Smart Home Device	7446.93
77	Blazer	7367.84
18	Tablet	7269.88
37	Router	7236.85
69	Spices	7147.24
15	Table	7121.49

Sales Performance

WHICH ARE THE BEST-SELLING PRODUCTS ACROSS STORES?

```
select st.store_name, p.product_name ,
sum(s.quantity_sold) as "Total_Quantity_Sold"
from sales as s
join product as p ON s.product_id= p.product_id
join store as st ON s.store_id= st.store_id
group by st.store_name, p.product_name
order by Total_Quantity_Sold desc
limit 10;
```

store_name	product_name	Total_Quantity_Sold
Ultimate Retail Hub	Smart Home Device	388
One-Stop Shop	Fitness Tracker	368
All-In-One Mart	Bed	358
All-In-One Mart	Water Gun	346
Everthing and More Store	Doll	337
Everthing and More Store	Router	307
Everthing and More Store	Table	304

Sales Performance

WHICH STORES ARE GENERATING THE MOST SALES REVENUE?

```sql
select st.store_name, round(SUM(s.sales_amount),2) as "total_sales"
from sales as s
join store as st on s.store_id = st.store_id
group by st.store_name
order by total_sales desc;
```

store_name	total_sales
All-In-One Mart	132025.62
One-Stop Shop	127241.62
Everthing and More Store	124581.78
Ultimate Retail Hub	121456.98

WHICH PRODUCTS PROVIDE THE HIGHEST PROFIT MARGINS?

```sql
select p.product_name,
round((SUM(s.sales_amount) - SUM(p.price_per_unit * s.quantity_sold)),2) as "profit"
from sales s
join product p on s.product_id = p.product_id
group by p.product_name
order by profit desc
limit 3;
```

product_name	profit
Bed	3647.92
Blazer	2241.37
Tablet	1047.83

Seasonality and Trends

ARE THERE SEASONAL PATTERNS IN SALES FOR SPECIFIC PRODUCTS?

```sql
select p.product_name, EXTRACT(month from s.sale_date) as "sale_month",
SUM(s.sales_amount) as "total_sales"
from sales as s
join product as p on s.product_id = p.product_id
group by p.product_name, sale_month
order by p.product_name, sale_month desc;
```

product_name	sale_month	total_sales
Action Figure	12	108.57
Action Figure	11	358.01
Action Figure	10	768.12
Action Figure	9	635.99
Action Figure	7	1099.87
Action Figure	5	181.87

Supplier Performance

WHICH SUPPLIERS' PRODUCTS ARE SELLING THE MOST?

```
select sp.supplier_name, SUM(s.quantity_sold) as "total_sold"
from sales as s
join product as p on s.product_id = p.product_id
join supplier as sp on p.supplier_id = sp.supplier_id
group by sp.supplier_name
order by total_sold desc;
```

supplier_name	total_sold
Rajesh	4689
Shivam	4668
Raju	4143
Ganesh	3291
Rohit	3280
Girish	3061
Manoj	2917

POWER BI REPORT

(contains Dax queries, reports, visuals, and analysis)

DAX QUERIES

Total Inventory

```
= SUMX(inventory,
inventory[current_stock] *
RELATED(product[price_per_unit]))
```

Current Stock Quantity

```
= SUM(inventory[current_stock])
```

Sale-to-Inventory Ratio

```
= DIVIDE(SUM(sales[quantity_sold]),
AVERAGE(inventory[current_stock]))
```

Products Below Reorder Level

```
= CALCULATE( COUNTROWS(inventory),
  FILTER(inventory,
inventory[current_stock] <
related(product[reorder_point])))
```

INVENTORY

Sales Report

2023

Total Sales	Quantities Sold	Average sales
$505.3K	52K	$252.7

Bengaluru | Chennai | Delhi-NCR | Pune

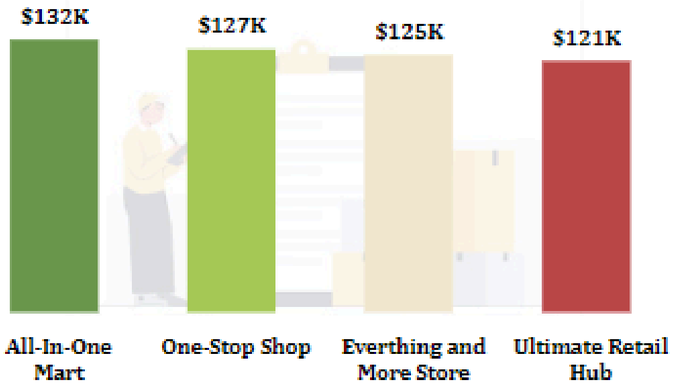
Date

Quarter: All

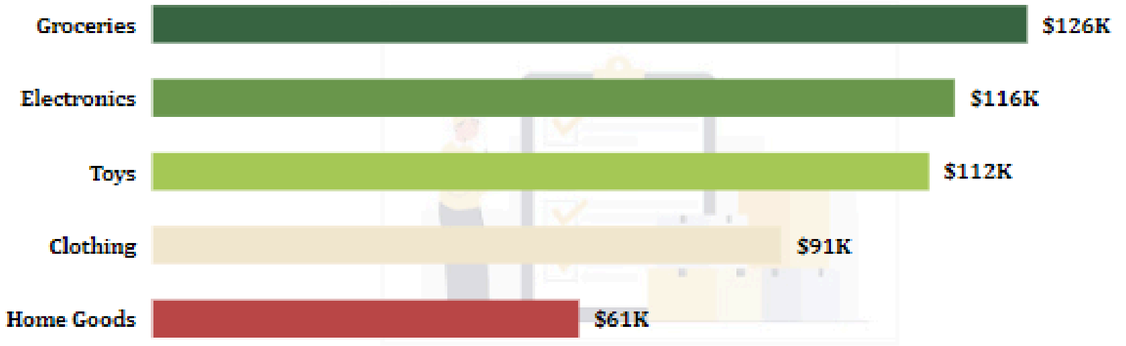
Category: All

Supplier Name: All

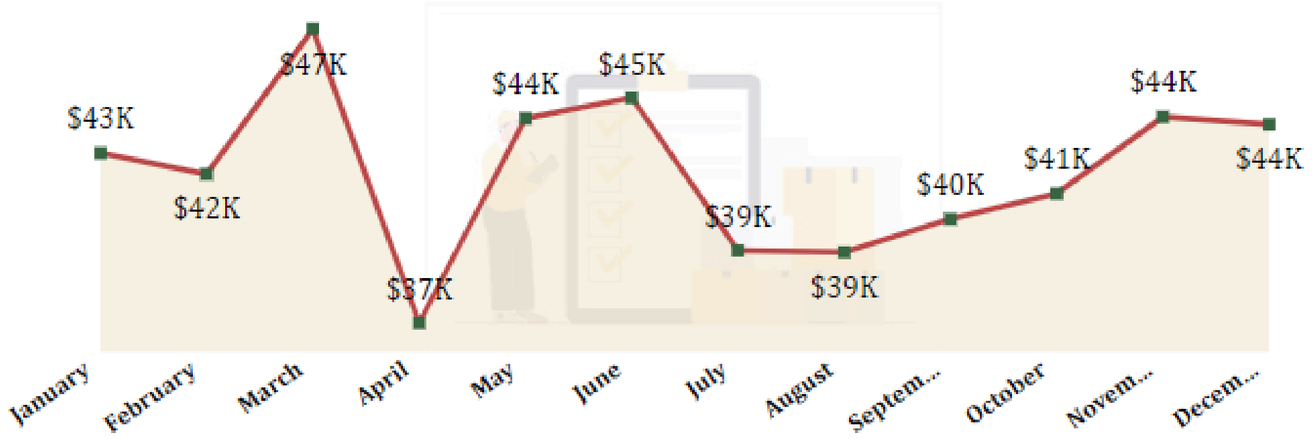
SALES BY STORE



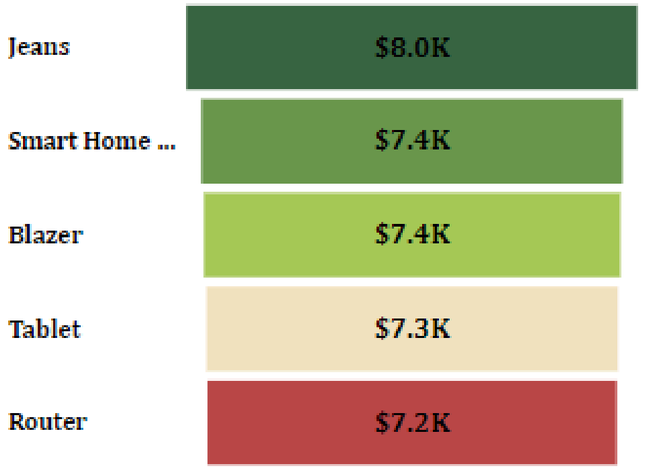
SALES BY CATEGORY



SALES BY MONTH



SALES BY TOP PRODUCTS



Inventory Report

2023

INVENTORY

Date

Quarter

All

Products

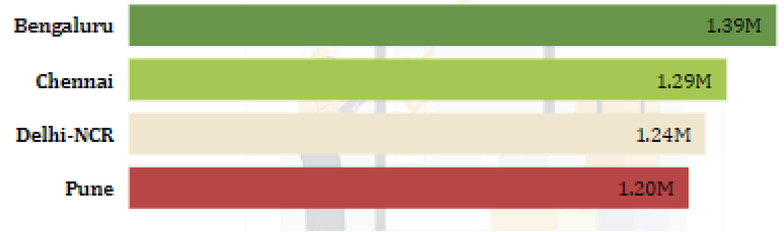
All

Suppliers

All

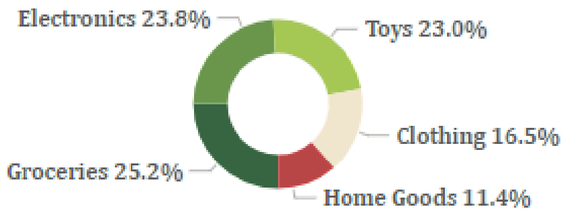
Total Inventory	Current Stock Quantity	Sales- to-Inventory Ratio	Average Lead Time	Products Reorder Level
5.12M	96K	537.85	5.40	83

TOTAL INVENTORY BY LOCATION



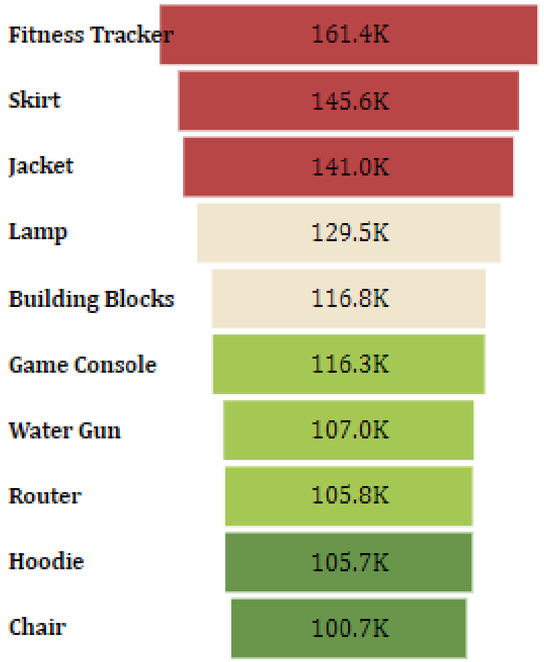
CURRENT STOCK BY STORE

TURNOVER BY CATEGORY

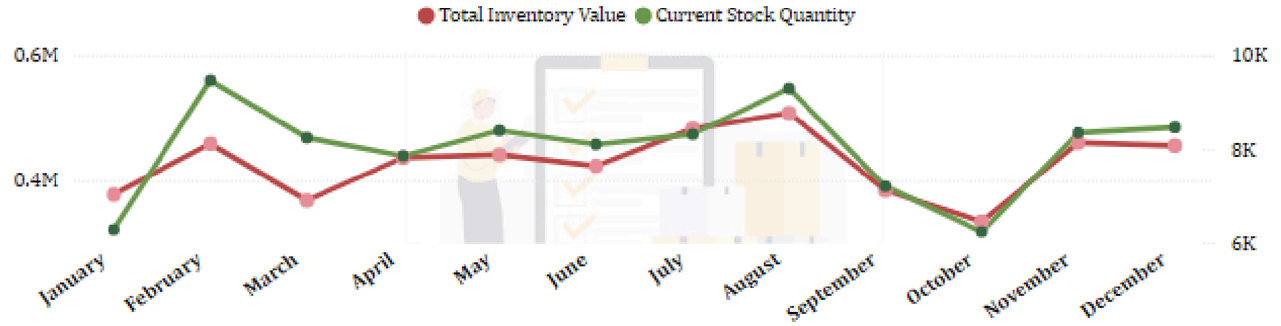


Category	Bengaluru	Chennai	Delhi-NCR	Pune
Clothing	5611	4387	4798	4486
Electronics	6111	5060	5276	4699
Groceries	5839	6889	5548	6680
Home Goods	3018	2941	3192	2011
Toys	4057	5136	5414	4969

TOP PRODUCTS OF INVENTORY



TOTAL INVENTORY AND STOCK BY MONTHS



INSIGHTS AND ANALYSIS

- **Inventory Optimization:** Streamlined stock levels to prevent stockouts and overstock.
- **Sales Trends:** Improved demand forecasting by analyzing sales patterns.
- **Supplier Performance:** Enhanced procurement by evaluating supplier reliability.
- **Product Analysis:** Focused inventory efforts on top-performing and underperforming products.
- **Store Performance:** Implemented targeted strategies based on store-specific data.
- **Cost Reduction:** Reduced holding costs through optimized inventory management.
- **Data-Driven Decisions:** Leveraged Power BI for actionable insights and strategic planning.

THANK YOU

https://github.com/facile29/Retail_Store_Inventory_Optimization