Project Report
Solving Inventory Inefficiencies Using SQL

Introduction

Urban Retail Co. is a rapidly growing mid-sized retail chain with a widespread presence across both physical stores and online platforms. The company offers a diverse portfolio of over 5,000 stock-keeping units (SKUs), ranging from everyday groceries and home essentials to electronics and personal care items. **With its expanding operations across multiple cities, the company depends on a network of regional warehouses that supply inventory to local retail outlets.**

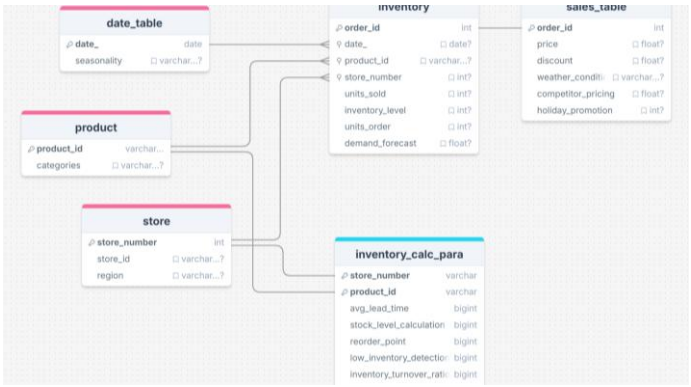
Despite having access to extensive sales and inventory data, Urban Retail Co. faces challenges in effectively utilising this data for decision-making. This project aims to design a SQL-driven analytics solution to monitor inventory trends, identify inefficiencies, and recommend strategies for improving inventory performance using actionable insights.

Problem Faced By Urban Retail Co.

As operations scale, Urban Retail Co. is facing significant challenges in inventory management due to reactive planning and underutilised data analytics. The key problems identified include:

- Frequent **stockouts** of high-demand items lead to missed sales and a poor customer experience.
- **Overstocking** of slow-moving SKUs, locking working capital and increasing warehousing costs.
- Lack of **real-time visibility** into SKU performance, reorder thresholds, and supplier delays.
- Inefficient manual processes and the absence of predictive tools to guide replenishment decisions.

Schema Design / ERD



This ERD outlines the data model for Urban Retail Co.'s inventory analytics system:

- **date_table**: Stores dates and seasonality info for time-based trend analysis.
- **product**: Contains product_id and category for grouping and classification.
- **store**: Captures store details (store_number, region) for location-based insights.
- **Inventory** *(central fact table)*: Tracks daily inventory data — units sold, inventory levels, orders, and forecasts — linked to product, store, and date.
- **sales_table**: Enriches inventory data with pricing, discounts, competitor, and promotion info using order_id.
- **inventory_calc_para**: Stores derived metrics (e.g. avg_lead_time) used in inventory planning and reorder point logic.

Relationships:

- The **inventory** table acts as a hub, connecting to the date_table, product, and store tables through foreign keys, ensuring that every inventory record is tied to a specific product, store, and date.
- The **sales_table** is linked to inventory via the order_id, allowing for integrated analysis of sales and inventory data.
- The **inventory_calc_para** table connects to both product and store, supporting advanced calculations for inventory optimisation.

Table Structure

Main Table

```
CREATE TABLE inventory_combined (
    order_data DATE,
    store_id  varchar(255),
    product_id varchar(255),
    categories varchar(255),
    region varchar(255),
    inventory_level int ,
    units_sold int,
    units_order int,
    demand_forecast int,
    price float,
    discount int,
    weather_condition varchar(255),
    holiday_promotion int,
    competitor_pricing float,
    seasonality varchar(255)
);
```

Product Table

```
CREATE TABLE product (
    product_id varchar(255) primary key,
    categories varchar(255)
);
```

Store Table

```
CREATE TABLE store (
    store_number varchar(255) primary key,
    store_id varchar(255) ,
    region varchar(255)
);
```

Date Table

```
CREATE TABLE date_table (
    data_ date primary key ,
    seasonality varchar(255)
);
```

Parameter Table

```
CREATE TABLE inventory_calc_para(
    store_id varchar(255),
    region varchar(255),
    product_id varchar(255),
    avg_lead_time_days FLOAT
);
```

Sales Table

```
CREATE TABLE sales (
    order_id INT AUTO_INCREMENT ,
    price float ,
    discount float ,
    competitor_pricing float ,
    weather_condition varchar(255),
    holiday_promotion int ,
    data_ date ,

    FOREIGN KEY (data_) REFERENCES date_table(data_),
    PRIMARY KEY (order_id),
    FOREIGN KEY (order_id) REFERENCES inventory_table(order_id)
) AUTO_INCREMENT=1001;
```

Inventory Table

```
CREATE TABLE inventory_table (
    order_id INT PRIMARY KEY AUTO_INCREMENT  ,
    units_sold INT ,
    inventory_level INT ,
    units_order INT ,
    demand_forecast FLOAT ,
    product_id varchar(255) ,
    store_number varchar(255),
    data_ date ,

    FOREIGN KEY (product_id) REFERENCES product(product_id),
    FOREIGN KEY (store_number) REFERENCES store(store_number),
    FOREIGN KEY (data_) REFERENCES date_table(data_)
) AUTO_INCREMENT = 1001;
```

Data Exploration

1. Dataset Description:

This dataset comprises detailed transactional, categorical, and environmental records spanning **two full years (730 days)**. It captures product-level activity across a multi-store, multi-region retail environment, providing comprehensive insights suitable for sales forecasting, pricing analysis, regional performance evaluation, and weather-impact studies.

2. Entities and Hierarchy

- **Stores**
  - Total Stores: **5**
  - Store Identifiers: S001 to S005
- **Regions**
  - Each store operates in **4 regions**:
    - North, South, East, West
- **Products**
  - Total Distinct Products: **30**
  - Product Categories:
    - Toys: 3 products
    - Clothing: 11 products
    - Electronics: 8 products
    - Groceries: 6 products
    - Furniture: 2 products
  - Product Availability: All 30 products are available in **every store**.

3. Temporal Coverage

- **Duration**: 730 consecutive days (2 years)
- **Granularity**: Daily records per store-region-product combination

4. Weather Information

- **Weather Conditions Tracked**: Sunny, Rainy, Snowy, Cloudy
- **Weather Variability**: Weather conditions may **vary by region** on the same calendar date.

5. Pricing and Sales Metrics

- **Product Pricing**: Prices are **dynamic**; they differ **by store and by day** for the same product.
- **Turnover Consistency**: All stores show **similar overall turnover** levels.
- **Average Order Value (AOV)**: AOV is **uniform across all weather conditions** and stores.

6. Promotional and Holiday Indicators

- **Holiday/Promotion Flags**: Each data entry includes binary indicators for holidays and promotions.

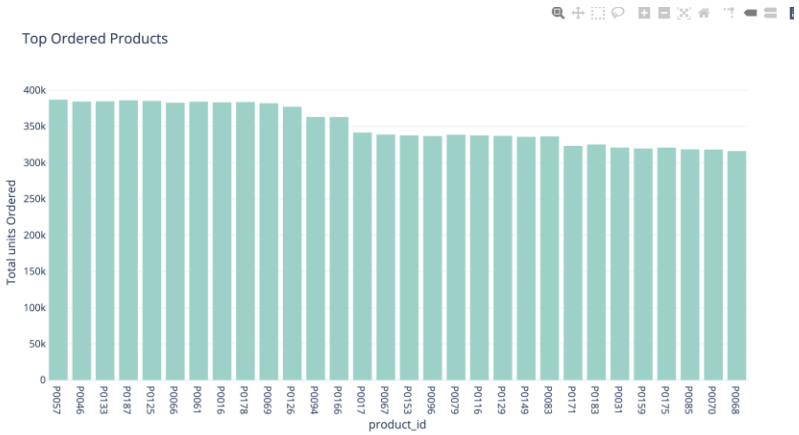
7. Data Integrity

- **Missing Data**: The dataset is **complete** with **no missing values** across any feature.

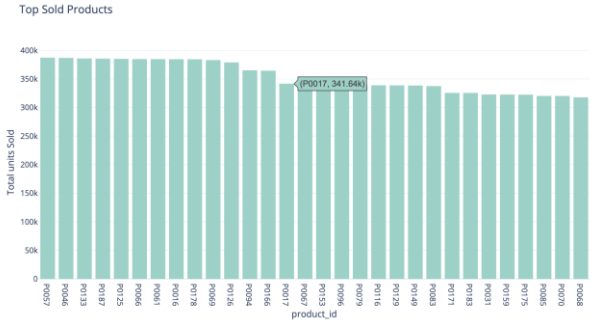
Summary

This dataset is structured and clean, supporting advanced analytics use cases including regional sales analysis, impact of promotions, pricing strategies, and demand forecasting under varying weather and promotional conditions.

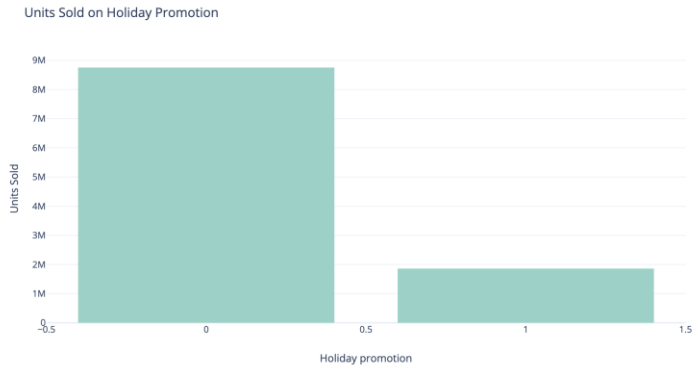
Urban Retail Co. offers over **109,000 products** across five categories. **Clothing (40,150)** and **Electronics (29,200)** dominate the catalogue, followed by Furniture, Toys, and Groceries. **Together, Clothing and Electronics make up over 60% of total SKUs.**



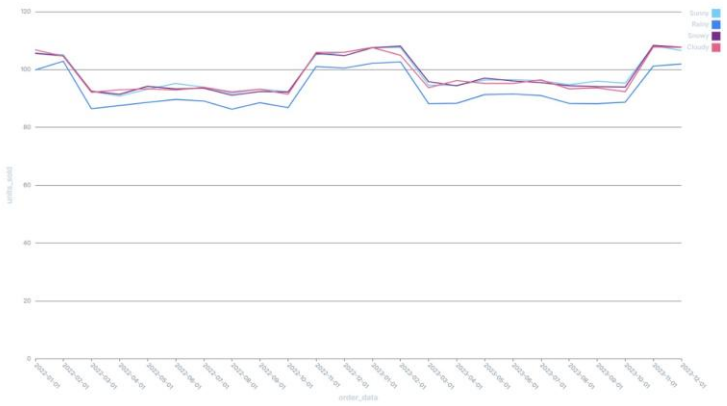
The chart highlights the top **30** most ordered products, each exceeding **320,000 units in total orders.** Products like **P0057, P0046, and P0133** lead the demand, indicating consistently high customer preference across the board.



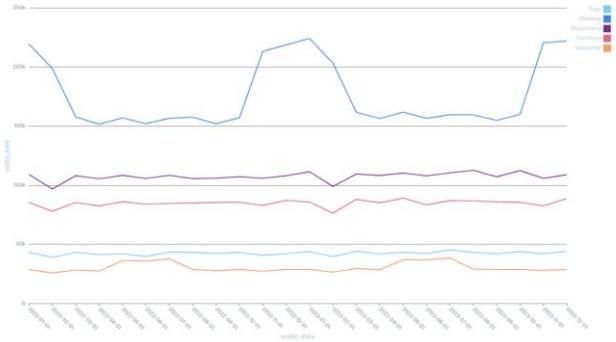
The chart shows the top **30 best-selling products**, all with sales exceeding 320,000 units. Products like **P0057, P0046, and P0133 lead in performance, indicating strong and consistent customer demand.**



Insight: **Units were sold more on the dates when there was a holiday promotion**. This indicates a strong sales uplift during promotions, despite fewer promotional days.



Insight: **For the same date, sales of products decrease due to rainy weather**



Insight: The chart indicates a significant surge in sales for the **Clothing** category **during winter months** (especially around December and January). This trend suggests higher seasonal demand for apparel, likely driven by colder weather and holiday shopping behaviour.

Approach

Inventory Metrics & Analytics – Logic and Calculations

1. Inventory Gap Analysis

Objective: To identify mismatches between expected and actual inventory levels.
Logic: Expected Inventory = Previous Inventory - Previous Sales + Previous Orders
Inventory Gap = Actual Inventory - Expected Inventory
Use case: Helps in identifying anomalies like data entry errors, delivery issues, or stock losses.

2. Top-Selling Products

Query: Aggregates units_sold per product_id.
Purpose: Identifies high-volume SKUs for prioritisation in replenishment, marketing, or procurement.

3. Lead Time Calculation

Objective: Estimate the **average number of days** between order placement and stock replenishment.
Method:

- Uses LAG() window function to access inventory, sales, and order data from the **previous day**.
- Filters all rows where units_order > 0.
- Finds when inventory increases significantly (≥ 80% of ordered units) after an order.
- Calculates DATEDIFF(delivery_date, order_date) and averages it..

4. Safety Stock

Formula: $\text{Safety Stock} = Z \times \sigma(\text{Demand}) \times \sqrt{(\text{Lead Time})}$

- Assumes Z = 1.65 for 95% service level.
- Uses STDDEV(units_sold) as demand volatility.

5. Reorder Point (ROP)

Formula: ROP = (Avg Daily Demand × Lead Time) + Safety Stock

- Combines expected consumption during lead time with a buffer.

6. Low Inventory Detection

Logic: Compares Avg Inventory Level to ROP.

- If below threshold → Status = "REORDER NOW", else → "SUFFICIENT STOCK".

7. Inventory Turnover Ratio (ITR) & Days of Inventory (DOI)

Formulas: ITR = Total Units Sold / Average Inventory Level
DOI = 365 / ITR
Meaning:

- ITR indicates how efficiently inventory is sold and replaced.
- DOI shows how long the inventory stays unsold.

8. Stockout Rate Analysis

Definition: Stockout Rate (%) = (Days inventory < forecast) / (Days with forecasted demand) × 100
Interpretation: Percentage of time a product faced demand without adequate stock.
Application: Performance measure for supply reliability and customer satisfaction.

9. Lost Revenue Estimation

Formula:Lost Units = Max(0, Demand Forecast - Inventory)
Lost Revenue = Lost Units × Price
Logic:

- For each day a product is understocked, it calculates the missed sales opportunity.
- **Insight**: Quantifies the monetary impact of poor inventory planning.

10. Inventory Age (AOI)

Formula:
AOI = Average Inventory Level / Average Daily Sales
Use: Detects slow-moving inventory and potential obsolescence.

11. Day+1 Inventory Forecast

Steps:

1. Identify the **last known inventory** per product-store on Day 730.
2. Subtract **units sold** on that day.
3. Add **expected delivery** due on Day 731 (based on lead time and past orders).

Output: Projected inventory for Day 731 under assumptions of delivery reliability.

Business Insights

Monthly Sales

	product_id	store_number	year	MONTH	monthly_sales
1	P0066	2002	2023	12	2488
2	P0057	2004	2023	12	2337
3	P0066	3004	2023	1	2226
4	P0046	2002	2023	2	2046
5	P0061	3001	2023	12	2042
6	P0178	3003	2022	12	2029
7	P0016	4001	2022	11	2011
8	P0126	4002	2023	1	1984
9	P0057	4001	2022	12	1977
10	P0069	5003	2023	11	1962
11	P0125	2002	2022	11	1935
12	P0178	1001	2023	11	1916
13	P0126	1002	2022	2	1915
14	P0166	3004	2022	6	1891
15	P0126	5002	2022	12	1889

Peak Demand in December: Products like **P0066**, **P0057**, and **P0061** consistently show high monthly sales in **December**, indicating strong seasonal demand during year-end, likely due to holidays and winter needs.

Top Performing Products: **P0066**, **P0057**, and **P0046** appear multiple times across different stores and months with high sales, suggesting they are **core revenue drivers** and should be prioritised for inventory planning

Geographic Spread of Demand: High sales are spread across **multiple store numbers** (2002, 3004, 4001, etc.), indicating **broad demand coverage** and the importance of maintaining sufficient stock across regions.

Weekly Sales

	product_id	store_number	year	week_number	weekly_sales
1	P0046	2002	2023	8	949
2	P0061	3001	2023	49	940
3	P0125	1004	2023	47	860
4	P0166	2001	2022	20	847
5	P0016	1002	2022	3	846
6	P0057	5003	2023	46	841
7	P0069	4003	2023	49	836
8	P0125	1003	2022	1	828
9	P0133	1004	2022	6	820
10	P0016	4002	2023	2	818
11	P0126	5002	2022	8	816
12	P0066	3004	2023	1	815
13	P0126	5003	2023	47	814
14	P0066	4002	2023	49	805
15	P0057	5002	2023	5	801

Top Weekly Performers: Products like **P0046**, **P0061**, and **P0125** consistently lead in weekly sales, indicating strong short-term demand and potential for frequent restocking.

Late-Year Demand Peaks: Several products (e.g., **P0061**, **P0069**, **P0057**) show high sales in **week 46–49**, suggesting a **pre-holiday or winter rush** in Q4.

Widespread Demand Across Stores: High weekly sales span multiple store numbers (e.g., 2002, 3001, 5003), reinforcing the need for **broad distribution and responsive logistics**.

Age of Inventory

	product_id	store_number	year	week_number	weekly_sales
1	P0046	2002	2023	8	949
2	P0061	3001	2023	49	940
3	P0125	1004	2023	47	860
4	P0166	2001	2022	20	847
5	P0016	1002	2022	3	846
6	P0057	5003	2023	46	841
7	P0069	4003	2023	49	836
8	P0125	1003	2022	1	828
9	P0133	1004	2022	6	820
10	P0016	4002	2023	2	818
11	P0126	5002	2022	8	816
12	P0066	3004	2023	1	815
13	P0126	5003	2023	47	814
14	P0066	4002	2023	49	805
15	P0057	5002	2023	5	801

High Inventory Age Risks: Some products show **significantly aged inventory**, indicating slow movement and potential **capital lock-up or obsolescence risk**. These items may need **promotions or markdowns** to clear stock.
Store-Level Inefficiencies: Inventory ageing patterns differ across **store locations**, suggesting **distribution imbalances**. Optimising stock transfers between stores with lower turnover can improve overall efficiency

- **Consistently Low Inventory Age Indicates Healthy Turnover**

The average inventory age for sampled products is approximately 1.5

days, with values ranging from 1.46 to 1.58 days. This suggests that products are selling quickly and inventory is not remaining idle, minimising the risk of obsolescence and reducing holding costs.

- **Improves Cash Flow and Profitability**
  Managing inventory age effectively enables Urban Retail Co. to avoid unnecessary markdowns and write-offs, improving overall profitability. Quick-moving inventory also ensures that cash is not tied up in unsold goods, supporting better cash flow management..
- **Enhances Demand Forecasting and Planning**
  Inventory age data provides valuable feedback for refining demand forecasts and inventory planning. It helps identify products that may benefit from increased safety stock to prevent stockouts, as well as those that require reduced ordering to avoid excess inventory37.

Lost Revenue

	product_id	store_number	lost_units	lost_revenue
1	P0133	5003	790	47446.61
2	P0125	5001	768	46450.52
3	P0125	2003	801	46242.2
4	P0125	5004	637	41746.17
5	P0046	4002	773	41737.83
6	P0061	3002	753	40362.2
7	P0096	2003	605	40272.28
8	P0178	2002	665	38215.78
9	P0046	1004	599	37789.99
10	P0016	3001	657	37562.18
11	P0016	2003	677	37235.08
12	P0069	4003	680	37011.78
13	P0066	2003	623	36634.13
14	P0046	3004	543	36367.86
15	P0178	2001	695	36362.88

The total lost revenue from stockouts across all products and stores amounts to approximately ₽1.29 crore (12,896,100), with 236,277 units of unmet demand.

The top 10 product-store combinations account for a substantial share of lost revenue, with individual losses ranging from ₽37,500 to ₽47,400 per combination.

The highest single lost revenue was for product P0133 at store 5003, totalling ₽47,446.611.

Actionable Recommendation:
Prioritising inventory optimisation for the top lost revenue products and stores can yield immediate financial benefits and reduce recurring stockouts. Enhanced analytics and targeted interventions in these hotspots should be a key focus for the business

Stock Level Calculation

	store_number	product_id	inventory_731
1	1001	P0096	151
2	1002	P0016	263
3	1001	P0031	102
4	1001	P0159	184
5	1004	P0129	156
6	1002	P0116	158
7	1003	P0070	208
8	1002	P0171	70
9	1002	P0175	142
10	1004	P0046	109
11	1003	P0067	138
12	1001	P0183	235
13	1003	P0166	113
14	1002	P0079	148
15	1004	P0187	207

Stock levels at the end of the period vary significantly by product and store, with **some product-store pairs holding as few as 24 units and others exceeding 374 units1**. This variability highlights the need for more precise inventory balancing to match supply with localised demand.

Several stores have high closing inventories for certain products (e.g., store 4001 with 374 units of P0069, store 5001 with 327 units of P0125)1. This may indicate overstocking, tying up working capital and increasing holding costs.

Regularly analyse closing stock data to identify opportunities for moving excess inventory from **overstocked to understocked locations, reducing waste and improving availability.**

Reorder Point and Safety Stock

product_id	store_number	avg_lead_time_days	avg_daily_demand	safety_stock	reorder_point
P0046	4001	4.46	106.39	146.27	620.75
P0016	5003	4.54	103.92	143.85	615.63
P0069	5002	4.39	104.23	151.72	609.27
P0069	2003	4.5	103.35	141.54	606.64
P0187	2002	4.18	107.30	150.9	599.4
P0066	5004	4.38	104.21	133.84	590.27
P0061	2001	4.17	106.12	147.13	589.66
P0094	3004	4.58	100.78	127.4	589
P0187	4002	4.18	106.02	141.38	584.55
P0046	2004	4	111.84	129.02	576.37
P0125	4001	4.13	107.17	127.95	570.55
P0046	1004	3.95	107.29	144.32	568.12
P0166	5004	4.32	99.01	138.13	565.86
P0046	5003	4.15	104.49	131.98	565.63
P0061	5003	3.83	110.95	138.39	563.34

Customised Inventory Buffers Across Locations
Safety stock and reorder points vary significantly by product and store, reflecting differences in demand patterns and lead times. For instance, product P0016 has safety stock ranging from 82 to 143 units and reorder points from 295 to 615 units, depending on the store

Some stores have relatively low safety stock and reorder points for the same product, indicating more predictable demand or faster replenishment cycles (e.g., P0016 at store 5001 with a safety stock of 82.36 units and a reorder point of 295.11 units)

Regularly Review and Update Parameters: Continuously monitor demand and lead time data to adjust safety stock and reorder points, ensuring inventory policies remain aligned with real-world conditions.

Inventory Gap

	order_data	store_id	region	product_id	inventory_level	expected_inventory	inventory_gap
1	2022-01-04	S001	East	P0016	93	187	-94
2	2022-01-08	S001	East	P0016	180	76	104
3	2022-01-11	S001	East	P0016	221	191	30
4	2022-01-13	S001	East	P0016	126	210	-84
5	2022-01-16	S001	East	P0016	132	110	22
6	2022-01-17	S001	East	P0016	290	132	158
7	2022-01-19	S001	East	P0016	259	303	-44
8	2022-01-20	S001	East	P0016	167	288	-121
9	2022-01-21	S001	East	P0016	199	188	11
10	2022-01-23	S001	East	P0016	193	183	10
11	2022-01-24	S001	East	P0016	199	164	35
12	2022-02-03	S001	East	P0016	231	210	21
13	2022-02-08	S001	East	P0016	106	243	-137
14	2022-02-09	S001	East	P0016	149	101	48

Future Scope

Urban Retail Co. is well-positioned to advance its inventory management capabilities by leveraging emerging technologies and analytical methodologies. Building upon the foundational insights and infrastructure developed in this project, the following directions offer strategic opportunities for future development:

1. Dynamic Pricing and Customer Personalisation

Using AI-powered dynamic pricing models, Urban Retail Co. can adjust prices in real time based on supply-demand fluctuations, competitor pricing, and customer behaviour. This can be complemented with personalised product recommendations based on inventory availability.

2. Collaborative and Multi-Channel Inventory Systems

The future roadmap includes a unified inventory platform across physical stores, e-commerce, and warehouses. Improved integration with suppliers and logistics partners will ensure accurate visibility and replenishment across all sales channels.

3. Workforce Enablement and Strategic Partnerships

Investing in employee training for data tools and fostering partnerships with technology vendors or academic institutions will be crucial for sustainable implementation. Building internal capability ensures long-term scalability and innovation readiness.