

## Project Report

### Supply-Demand KPI Analytics Using MSSQL, MySQL, and Power BI

---

#### Overview

This project analyzes inventory demand, availability, and performance KPIs across **Test and Production environments** using **Microsoft SQL Server (MSSQL), MySQL, and Power BI**.

The focus is on evaluating supply-demand balance, identifying shortages or surpluses, and comparing environments to ensure data consistency, integrity, and readiness for production-level analytics.

---

#### Environment Definitions

- **Test Environment:**  
A sandbox setup used for development, query testing, and validating logic. The data is isolated and used to build and verify KPIs before applying them in production. Implemented in MSSQL using test datasets.
  - **Production Environment:**  
A live environment used for finalized, real-world business analysis. It hosts cleaned, standardized, and verified datasets. Implemented in both MSSQL and MySQL for cross-platform verification.
- 

#### Data Sources & Processing

##### MSSQL - Test Environment

- **Database:** test\_env
- **Tables Used:**
  - Products (Product\_ID, Product\_Name, Unit\_Price)
  - Test Environment Inventory Dataset (Order\_Date, Product\_ID, Availability, Demand)
- **Transformation Query:**
- SELECT
- a.Order\_Date\_DD\_MM\_YYYY AS OrderDate,
- a.Product\_ID, a.Availability, a.Demand,
- b.Product\_Name, b.Unit\_Price
- INTO new\_table
- FROM [dbo].[Test Environment Inventory Dataset] a
- LEFT JOIN [dbo].[Products] b
- ON a.Product\_ID = b.Product\_ID;

---

## MSSQL - Production Environment

- **Database:** PROD
- **Issues Identified:**
  - Incorrect Product\_ID mappings (e.g., 21 and 22)
- **Data Cleaning:**
- UPDATE prod.[dbo].[Prod Env Inventory Dataset]
- SET Product\_ID = 7 WHERE Product\_ID = 21;
- 
- UPDATE prod.[dbo].[Prod Env Inventory Dataset]
- SET Product\_ID = 11 WHERE Product\_ID = 22;
- **ETL Query:**
- SELECT
- a.Order\_Date\_DD\_MM\_YYYY AS OrderDate,
- a.Product\_ID, a.Availability, a.Demand,
- b.Product\_Name, b.Unit\_Price
- INTO new\_prod\_table
- FROM [dbo].[Prod Env Inventory Dataset] a
- LEFT JOIN [dbo].[Products] b
- ON a.Product\_ID = b.Product\_ID;

---

## MySQL - Production Environment

- **Database:** prod
- **Tables Used:**
  - Prod\_Env\_Inventory\_Dataset
  - Products
- **Data Cleaning:**
- UPDATE prod.Prod\_Env\_Inventory\_Dataset
- SET Product\_ID = 7 WHERE Product\_ID = 21;
- 
- UPDATE prod.Prod\_Env\_Inventory\_Dataset

- SET Product\_ID = 11 WHERE Product\_ID = 22;
- **New Table:** prod\_inventory\_combined  
Consolidated after joining with Products table for unit price and name.

---

### KPI Formulas (Used in Power BI)

KPI	Formula
Total Availability	SUM(Availability)
Total Demand	SUM(Demand)
Days Count	DISTINCTCOUNT(OrderDate)
Avg Availability/Day	DIVIDE([Total Availability], [Days Count])
Avg Demand/Day	DIVIDE([Total Demand], [Days Count])
Loss	SUMX(FILTER(..., [Availability]-[Demand]<0), ([Availability]-[Demand]*Unit_Price)*-1)
Profit	SUMX(FILTER(..., [Availability]-[Demand]>0), ([Availability]-[Demand]*Unit_Price))
Supply Shortage Total	Total Demand - Total Availability

---

### Dashboard Layout (Conceptual)

#### Page 1: Test Environment

- KPI Cards:
  - Average Demand
  - Average Availability
  - Total Supply Shortage

#### Page 2: Production Environment

- KPI Cards:
  - Total Profit
  - Total Loss
  - Average Loss per Day

#### Page 3: Combined View

- Comparison Points:

- Side-by-side metrics for Avg Availability and Demand between test and production
- Evaluation of consistency across both environments
- Drill-down comparison by Product\_ID and Order Date
- Analysis of whether shortages and profitability trends remain similar across both environments

---

### Key Observations

- **Corrected Inconsistencies:** Product IDs 21 and 22 were corrected to their appropriate mapped values 7 and 11 in both MSSQL and MySQL production datasets.
- **Shortage Patterns:** Some products consistently showed higher demand than supply, indicating frequent stockouts.
- **Profit Trends:** Products with better availability ratios showed better profit margins.
- **Data Consistency Achieved:** After corrections and proper transformation, all environments showed **identical KPI outputs**.

---

### Final Conclusion

The same dataset was analyzed across:

- **MSSQL Test Environment**
- **MSSQL Production Environment**
- **MySQL Production Environment**

Post-transformation, all three environments produced **consistent results**, ensuring the **accuracy of cleaning, logic, and KPI calculations**. This consistency is crucial in real-world applications for confidence in test-to-production deployment.

---

### Project Structure

Supply\_KPI\_Analysis/

```
├── SQL_Scripts/
|   ├── test_env_transform.sql
|   ├── prod_env_cleanup.sql
|   └── mysql_prod_transform.sql
├── Data/
|   ├── test_inventory.csv
|   └── prod_inventory.csv
```

```
| └─ products.csv
|
| └─ PowerBI/
|
| └─ KPI_Analysis.pbix
|
└─ Reports/
    └─ Final_Report.docx
    └─ Readme.md
```

---

### ✅ Skills Gained from Working with MSSQL and MySQL Environments

- **Cross-Platform SQL Expertise:** Gained hands-on experience in writing and optimizing queries across both MySQL and MSSQL environments, enhancing adaptability in real-world database systems.
- **Data Cleaning & Transformation:** Developed proficiency in handling inconsistent and raw datasets using SQL techniques like JOIN, CASE, GROUP BY, CAST, and conditional filtering.
- **KPI Computation & Analysis:** Calculated key performance metrics (like average demand, availability, profit/loss) to support business decision-making using SQL aggregation and subqueries.
- **Environment-Based Testing & Validation:** Learned how to differentiate between test and production data environments, validate outputs across both, and ensure consistency of results for production readiness.