Milestone 2: Data Ingestion Pipeline

Objective:

The objective of Milestone 2 is to design and implement a structured data ingestion pipeline that can automatically process daily incoming data. This ensures that raw sales and inventory data are properly cleaned, validated, and stored before being used for further analysis and model development.

In this milestone, a batch-based data ingestion system was developed using Python and Pandas. The pipeline simulates real-world daily data processing used in retail and e-commerce systems.

The ingestion workflow performs the following operations:

1. Reads raw dataset files from the data/raw/ directory.
2. Validates dataset structure.
3. Removes duplicate records.
4. Checks and handles missing values.
5. Adds structured time-based features (already generated in dataset).
6. Stores the cleaned dataset in the `data/processed/` directory with a timestamped filename.
7. Ensures successful execution without runtime errors.

Project Structure Used:

PythonProject1/

```
│
├── data/
│   ├── raw/        → Incoming raw dataset
│   ├── processed/  → Cleaned and validated dataset
│
├── dynamic_pricing.py  → Dataset generation & feature engineering
├── ingestion.py        → Data ingestion pipeline
```

This structure separates raw and processed data, following industry best practices.

Tools & Technologies Used

- Python
- Pandas
- NumPy
- PyCharm IDE

Results:

- Successfully processed 300,000 records.
- No duplicate or missing data detected.
- Cleaned dataset saved with date-based naming convention.
- ingestion pipeline executed successfully with exit code 0.
- System structured to simulate real-world daily batch processing.

Conclusion

Milestone 2 was successfully completed by implementing a structured and automated data ingestion pipeline. The system ensures data reliability, consistency, and readiness for further analytical and machine learning tasks.

->The pipeline is scalable and can be scheduled for daily execution in a production environment.

code used in ingestion pipeline:

```
import os
import pandas as pd
from datetime import datetime

RAW_FOLDER = "data/raw"
PROCESSED_FOLDER = "data/processed"
os.makedirs(PROCESSED_FOLDER, exist_ok=True)
for file in os.listdir(RAW_FOLDER):
  if file.endswith(".csv"):
    print(f"\nProcessing file: {file}")

    file_path = os.path.join(RAW_FOLDER, file)
    df = pd.read_csv(file_path)

    print("Original rows:", len(df))
    print("Columns:", df.columns.tolist())
```

 # Cleaning

df = df.drop_duplicates()

df = df.dropna()

print("Rows after cleaning:", len(df))

 # Save cleaned version with timestamp

today = datetime.now().strftime("%Y_%m_%d")

new_file_name = f"processed_{today}_{file}"

processed_path = os.path.join(PROCESSED_FOLDER, new_file_name)

df.to_csv(processed_path, index=False)

print(f"Saved cleaned file: {new_file_name}")

print("\n Daily ingestion completed successfully")

output:

simulated 2 days of data ingestion pipeline.