Retail Sales Trend Monitoring Project

# 📌 Project Overview

This project aims to help a retail chain monitor stores with increasing sales trends.  
We use time-series sales data to detect upward trends using linear regression and visualize them through a Streamlit dashboard.

# 📘 Sample Data

Sample of the sales\_data.csv file:

store\_id date sales  
 101 2023-01-01 1500  
 101 2023-01-02 1600  
 101 2023-01-03 1700  
 101 2023-01-04 1800  
 101 2023-01-05 1900

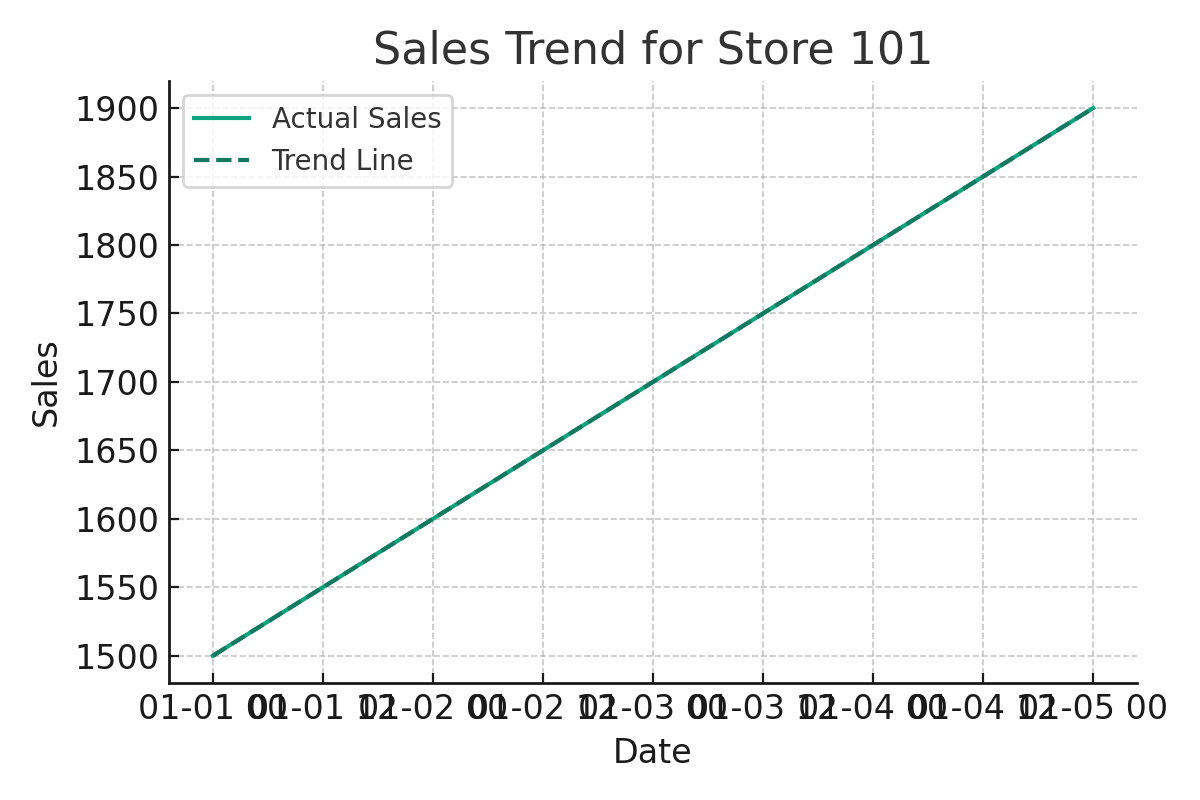
# 📓 Trend Detection Using Linear Regression

We calculate the sales trend for each store using linear regression:

Trend detection result:

store\_id slope trend  
 101 100.0 Increasing  
 102 -50.0 Decreasing

# 📊 Example Trend Plot



# 🚀 Streamlit Dashboard Code

Below is the code for the Streamlit dashboard:

import streamlit as st  
import pandas as pd  
import matplotlib.pyplot as plt  
from sklearn.linear\_model import LinearRegression  
  
df = pd.read\_csv('sales\_data.csv', parse\_dates=['date'])  
trends = pd.read\_csv('store\_trends.csv')  
  
st.title("📈 Store Sales Trend Monitor")  
store\_ids = df['store\_id'].unique()  
selected\_store = st.selectbox("Select a store", store\_ids)  
  
store\_data = df[df['store\_id'] == selected\_store].copy()  
store\_data['day\_num'] = (store\_data['date'] - store\_data['date'].min()).dt.days  
  
X = store\_data[['day\_num']]  
y = store\_data['sales']  
model = LinearRegression().fit(X, y)  
predicted = model.predict(X)  
  
fig, ax = plt.subplots()  
ax.plot(store\_data['date'], y, label='Actual Sales')  
ax.plot(store\_data['date'], predicted, label='Trend Line', linestyle='--')  
ax.set\_title(f"Sales Trend for Store {selected\_store}")  
ax.set\_xlabel("Date")  
ax.set\_ylabel("Sales")  
ax.legend()  
st.pyplot(fig)  
  
trend\_info = trends[trends['store\_id'] == selected\_store]  
if not trend\_info.empty:  
 st.markdown(f"### 📊 Trend: \*\*{trend\_info.iloc[0]['trend']}\*\*")