



# Retail Sales Trends Analysis



## Project Overview

This project analyzes a real-world retail sales dataset (Superstore) to uncover trends and patterns in product performance, seasonal sales behavior, regional revenue, and discount effects. It uses Python and data visualization tools to draw business insights from historical sales data.

---



## Objective

To understand:

- Which products and categories generate the most revenue
  - How sales vary over time (by month, year, and day of week)
  - Which regions perform best
  - How discounts affect profits
- 



## Tools & Libraries

- Python
  - pandas
  - matplotlib
  - seaborn
- 



## Key Analysis Performed

## 1. Category & Sub-Category Sales

- **Technology** and **Office Supplies** generated the highest total sales.
- **Phones**, **Chairs**, and **Binders** were among the top-performing sub-categories.

## 2. Time-Based Trends

- Sales peak in **November and December**, likely due to holiday shopping.
- Year-over-year sales show steady growth from 2014 to 2017.
- **Monday and Friday** had higher average sales, showing strong weekday performance.

## 3. Region-Wise Performance




- The **West** and **East** regions contributed the most to total revenue.
- Regional focus could help in targeting marketing or logistics improvements.

## 4. Profit vs. Discount

- Discounts above 20% often lead to **negative profits**, especially in the **Furniture** category.
- Business may need to optimize discount strategies to protect profit margins.

---

## Insights & Recommendations

-  Stock and promote **Technology products** during holiday months for higher sales.
-  Focus efforts in **West and East** regions where demand is highest.
-  Be cautious with heavy discounts, especially on **Furniture**, as they reduce profitability.

-  Consider marketing campaigns on **Fridays and Mondays**, which show higher purchase activity.
- 

## What I Learned

- How to clean and preprocess real retail sales data
- How to extract time-based features from dates (month, year, weekday)
- How to perform grouped aggregations and trend analysis
- How to create effective visualizations using `seaborn` and `matplotlib`
- How to interpret business implications from data patterns