

# AI Project

Week 2 – Retail Analysis with Cursor AI  
(E-commerce Dashboard)

# Guideline

- Project Overview
- Tool Used
- Part 1 – KPI Monitoring
- Part 2 – RFM Customer Segmentation
- Part 3 – Return Analysis
- Insights & Findings
- Next steps
- Cursor Interface

# Project Overview

Transforming raw e-commerce data into actionable business insights through an interactive **dashboard** using **Cursor AI**.

## Objectives:

- Build a data-driven **dashboard** to monitor key business metrics (Revenue, Return Rate, AOV, ARPU).
- Perform customer segmentation using RFM analysis.
- Identify high-return-risk products and customer behavior patterns.

## Key Results:

- Processed **500k+ order records**, automated data cleaning and segmentation by **Cursor AI**.
- Produced 2 Python scripts to show KPI dashboard and visualization result on Streamlit webpage.
- Enabled **real-time decision support** through visual insights.

# Tool Used



# Part 1 – KPI Monitoring



## E-commerce Dashboard - 2011年11月

數據期間: 2011年11月

Revenue  
\$1,493,710  
↑ 30.7% MoM

AOV  
\$18.50

Return Amount  
\$46,391

Orders  
80,725

ARPU  
\$896.58

Return Orders  
1,178

Customers  
1,666

Return Rate  
1.44%



### Metrics(KPI Card):

**Revenue/Orders/Customer:** Tracks month-over-month performance and seasonal trends.

**AOV / ARPU:** Evaluates purchasing efficiency and customer value.

**Return:** Detects product quality or logistics issues.

### Visualization Layout:

MoM trend charts.

# Part 2 – RFM Customer Segmentation



**Method:**

Recency (last purchase date),  
Frequency (purchase count),  
Monetary (spending).

Segmented into: **Champions, Loyal, Potential Loyal, At Risk, Lost.**

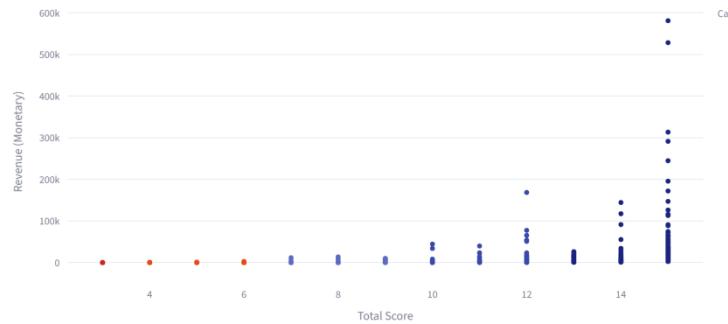
**Techniques:**

Quartile-based scoring (R, F, M → 1–15).  
Combined segmentation logic with Streamlit filters.

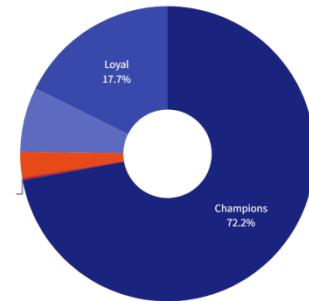
# Part 2 – RFM Customer Segmentation

RFM Scatter Plot (Total Score vs Revenue)

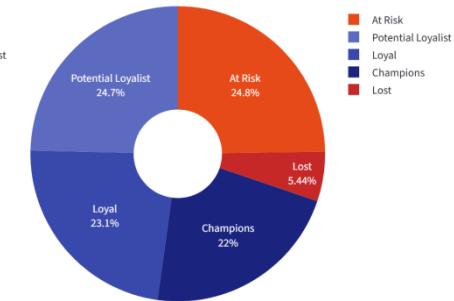
RFM Scatter Plot (Total Score vs Revenue)



Revenue Contribution by RFM Category



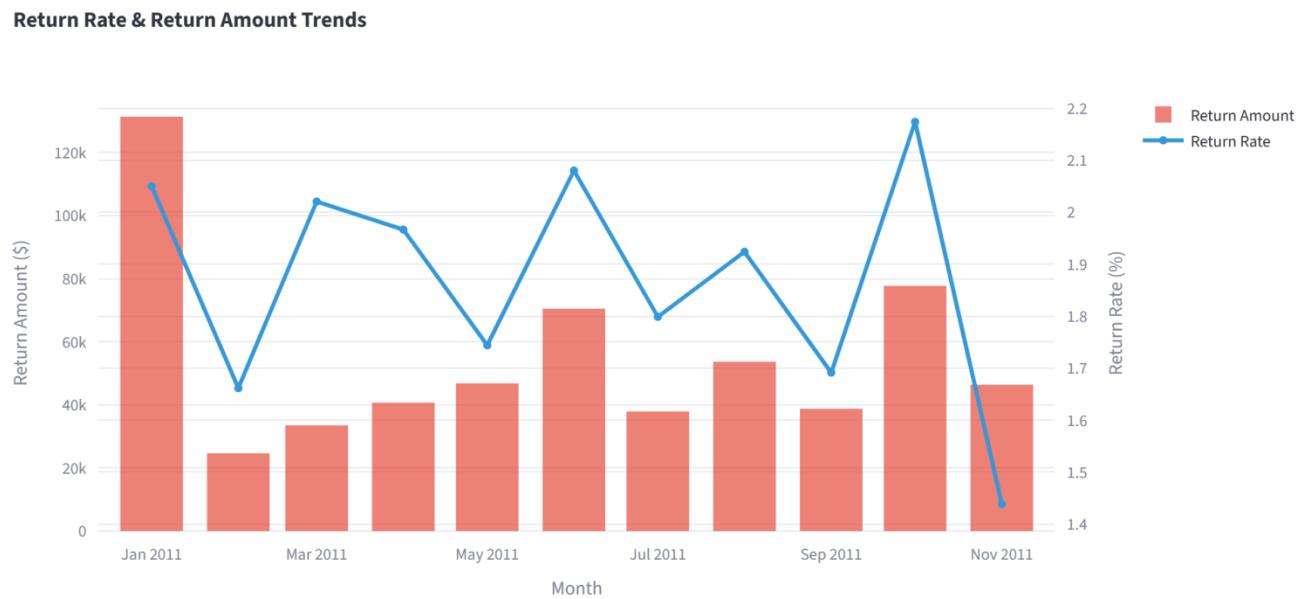
Customer Contribution by RFM Category



## ❑ Visualization Options:

Scatter Plot (Recency vs Frequency, colored by Monetary).  
Bar chart (segment count and revenue contribution).

# Part 3 – Return Analysis



## Method:

Analyze return amount and return rate by showing trend

## Techniques:

Categorized by return rate and return amount.

Combined segmentation logic with Streamlit.

# Part 3 – Return Analysis



## ✓ Visualization Options:

Scatter Plot (Return Rate vs Return Amount, colored by Category)  
In **Product** and **Customer** level.

# Insights & Findings

## Customer Insights:

- Top 20% of customers contribute **over 70%** of total revenue.
- “Loyalty” and “Potential Loyalty” customers suggesting promoting program.
- “At Risk” customers suggesting retention opportunities.

## Operational Insights:

- Revenue spikes observed in **September - November**, indicating rising order and customers.
- AOV decreased, ARPU increased, indicating that customers are purchasing smaller baskets more frequently. This suggests improved customer engagement and retention.
- Return spikes observed in **October**, indicating possible fulfillment or product issues.
- **GUEST** customer which CustomerID = blank unidentified customer system issue suggesting gather other data for investigation.

# Next Steps

## **Analytical Expansion(Advanced Analytics):**

- Add **Anomaly Detection** (Isolation Forest / Z-score) for revenue and return patterns.
- Implement **Customer Lifetime Value (CLV)** prediction using lifetimes package.
- Conduct **Cohort Analysis** to visualize customer retention.
- Analyze **Segment Migration Flow** (Loyal → At Risk → Lost) using Sankey chart.

## **Visualization Upgrade:**

- Integrate with Power BI or Streamlit Cloud for dynamic sharing.
- Add slicer for filtering.
- Add “Actionable Insights Panel” for auto-generated business suggestions.

# Cursor AI Interface

The screenshot shows a code editor interface with the following details:

- File Explorer:** On the left, it lists several files including `visualization_dashboard.py`, `README_Visualization.md`, and `online_retail_ll.xlsx`.
- Code Editor:** The main area displays Python code for a `visualization_dashboard.py` script. The code handles data loading from Excel files and performs various analyses like RFM and AOV-ARPU.
- Terminal:** Below the code editor, the terminal window shows a warning message about the deprecation of the `use_container_width` parameter in the `pandas` library.
- Output:** The output panel shows the results of the code execution, including generated charts and analysis reports.
- Search/Replace:** A search bar at the top right is set to "Online Retail".
- Help/Information:** A floating help card provides information about the `generate_kpi()` function.
- Status Bar:** The bottom status bar shows the cursor position (Ln 847, Col 1), file encoding (UTF-8), and Python version (3.14.0 64-bit).