Customer Data Analysis

Activities:
Clean, process, and visualize customer data using Python and Pandas to identify patterns, improve data quality, and generate meaningful business insights.

Project Title:
Customer Data Analysis for Business Insights

Business Problem Statement:
A mid-sized Indian retail company wants to analyze its customer base to improve targeted marketing, retain valuable customers, and detect patterns in customer behavior. You have been hired as a data analyst to clean, process, and analyze the customer transaction data to generate actionable insights.

Your goal is to:

- Assess customer demographics and transaction behavior
- Clean and prepare the data for analysis
- Conduct RFM (Recency, Frequency, Monetary) analysis
- Visualize key patterns using charts

Key Goals:

- Detect incomplete, inconsistent, or duplicate customer records and improve data quality, if applicable
- Segment customers based on demographics and purchase behavior
- Identify high-value customers using Recency-Frequency-Monetary (RFM) analysis
- Visualize customer distribution across geographies, age groups, or income brackets
- Generate actionable insights for marketing department

Dataset Description
You are provided with a synthetic dataset of **23,050 retail transactions** made by **1,000 unique customers**.

There are two key datasets:

1. Customer Master Data

Column Name	Description
CustomerID	Unique identifier for each customer
Name	Customer's full name
Email	Customer's email ID
Gender	Male, Female, or Not Disclosed

Column Name	Description
Age	Customer's age (between 18 and 75)
City	City where the customer resides (Indian metro/tier-2 cities)
MaritalStatus	Marital status: Single, Married, Divorced, Widowed
NumChildren	Number of children in the household
JoinDate	Date the customer first registered with the company

2. Transaction Data

Column Name	Description
CustomerID	Links back to the customer master dataset
TransactionDate	Date on which the transaction occurred
TransactionAmount	Amount spent by the customer in that transaction (₹)

RFM Analysis Instructions (Conceptual Explanation)

RFM (Recency, Frequency, Monetary) is a customer segmentation method that helps identify high-value customers based on:

Metric	Description
Recency	How recently a customer made a purchase
Frequency	How often a customer purchases
Monetary	How much money a customer spends

Steps to Compute RFM Scores

Step 1: Set Reference Date

Choose a reference "today" date. Typically, this is the most recent TransactionDate + 1 day.

Step 2: Calculate Individual RFM Metrics

For each CustomerID:

- Recency = Days between "today" and last transaction date
- Frequency = Total number of transactions
- Monetary = Sum of all TransactionAmount values

Store this in a new RFM table with columns: CustomerID, Recency, Frequency, Monetary

Step 3: Assign RFM Scores

- Use quantiles or quintiles (e.g., 1–5 scale) to rank customers
- Example logic:
    - Recency: Lower days = higher score (recent buyers)
    - Frequency: Higher count = higher score
    - Monetary: Higher spend = higher score

Metric	1 (Low)	5 (High)
Recency	Stale	Recent
Frequency	Rare	Frequent
Monetary	Low	High

Step 4: Combine Scores
Create a combined RFM segment by joining the 3 scores. Example:

- R=5, F=5, M=5 → "555" Champion
- R=1, F=1, M=1 → "111" Lost

Step 5: Define Segments (Optional)
Use the RFM score combinations to define:

- Champions (e.g., RFM 555)
- Loyal Customers
- At Risk
- Hibernating
- Potential Loyalists, etc.

Python Execution Instructions

Step 1: Load the Data

- Load both CSVs into Pandas DataFrames
- Check shape, structure, and preview

Step 2: Clean the Data

- Convert JoinDate and TransactionDate columns to datetime
- Ensure no nulls or bad types
- Validate uniqueness of CustomerID in master dataset
- Ensure all transaction CustomerIDs exist in master data

Step 3: Merge if Needed

- Join Customer_Master_Data with Customer_Transactions on CustomerID (if demographic data is required)

Step 4: Perform RFM Calculation

- Use groupby on CustomerID for:
  - max(TransactionDate) → Recency
  - count(TransactionDate) → Frequency
  - sum(TransactionAmount) → Monetary
- Use a reference date to compute Recency in number of days
- Store result in a new DataFrame called df_rfm

Step 5: Score RFM

- Use quantile-based scoring using pd.qcut() or rank() and cut()
- Create three new columns: R_Score, F_Score, M_Score

Step 6: Create Combined RFM Segment

- Concatenate the R, F, M scores into a string like "555", "432", etc.

Step 7: Assign Segment Labels

- Use business rules to define segment labels for selected score combinations
- Example:
  - RFM 555 → Champion
  - RFM 111 → Lost
    (Refer RFM explanation above)

Step 8: Visualize

- Count of customers in each segment
- Revenue contribution per segment
- Recency vs Monetary scatter plot colored by segment
- Pareto Analysis: Show how top 20% customers contribute to 80% revenue

Additional Instructions:
Real companies don't memorize all 125 RFM codes. They define 8–10 business-focused segments using threshold rules based on Recency, Frequency, and Monetary value. These segments power email campaigns, discount logic, and loyalty rewards.

Example 1: E-commerce
Goal: Identify customers to target with premium discounts, loyalty programs, or re-engagement

Segment Name	Rule Logic	Description
Champions	R 4–5, F 4–5, M 4–5	Buy frequently and recently, and spend the most. Send early access offers.
Loyal Customers	F 4–5, R 2–5	Repeat buyers, not necessarily recent. Send reward points, upsell.
Potential Loyalist	R 4–5, F 2–3	Recent buyers, building loyalty. Offer welcome packs.
At Risk	R 1–2, F 3–5	Used to buy frequently but haven't lately. Send reactivation offers.
Lost	R 1, F 1–2, M 1–2	Haven't bought in a long time. Consider exit surveys or cut losses.
Big Spenders	M 4–5, F 2–3, R 3–4	Spend big when they buy. Nurture for loyalty.

Example 2: Subscription SaaS
Goal: Identify churn risk and upsell opportunities

Segment Name	Rule Logic	Strategy
Best Subscribers	R 5, F 5, M 5	Premium plans, upsell integrations
Onboarding	R 5, F 1–2	Train well, reduce churn

Segment Name	Rule Logic	Strategy
At-Risk Loyal	R 1–2, F 4–5	Re-engage before churn
One-Time Users	R 1–2, F 1, M 1–2	Low ROI; consider downgrading support