Aiji C. Umemori Ad Hoc Approach

ITMGT 25.03-C

1. Imported the necessary libraries and opened the json file as a DataFrame.
2. Obtained the month of each transaction by using DatetimeIndex.
3. PPI FORK: Obtained the unit prices of each item by isolating transactions with only one quantity of one unique item. (Final Product: ppi_df)
4. ITEM COUNT FORK:
   a. Separated transactions with multiple item types into rows containing one item type each (using ';' as delimiter and explode() function).
   b. Obtained the quantity of each item type per month by summing the quantity, grouped by item type and transaction month.
   c. Appended a new column containing the total quantity sold per item. (Final Product: monthly_count_copy)
5. TOTAL SALES FORK:
   a. Merged PPI DataFrame and and Item Count DataFrame to be able to create a new column that will display the product of their respective values (qty * ppi).
   b. Created a new DataFrame containing the numerical products, sorted by item type and transaction month.
   c. Appended a new column containing the total sales per item and a new row containing total sales per month. (Final Product: new2_copy)
6. CUSTOMER ENGAGEMENTS FORK:
   a. Obtained list of unique customers per month. (Product: transaction_count)
   b. REPEATERS - Starting Month 2 (up to 6), count the customer if they made a transaction in the previous month and current month.
   c. ENGAGED - Starting Month 1, count the customer if they made a transaction in the current month and ALL previous months.
   d. INACTIVE - Starting Month 2, count the customer if they made a transaction in ANY (minimum of one) of the previous months AND DID NOT make a transaction in the current month.
   e. Created a new DataFrame containing the following based on month: unique customer count, repeaters, inactive, and engaged. (FP: customer_metrics3)
7. Generated charts (pie, line, and bar) and other pertinent statistics from the produced DataFrames for better visualization.