Extracting and Handling the Data

The 'GarmentFactoryDataset.zip' file provided by Universal Export contained 5 data files in different formats. These files contained information on the transactions, customers, products, logistics, and salespersons of the company. While the 'xlsx' and 'csv' files open quite aptly in PowerBI, it is quite important to extract data from the other files appropriately. Like in the case of 'Salespeople.json' and 'Logistics.txt' files, it was important to extract the data using suitable delimiters.

Key edits: Common for all datasets

- Column names were updated in all tables and set into Capital letters separated by underscores. This helps in having consistency amongst the tables of the dataset and fosters the PowerBI feature that fetches the relationship between the tables automatically.
- The columns representing money, or an amount were set to 'Decimal numbers' datatype having currency format as GBP (£) throughout all tables, as the company is based in UK.
- The columns representing IDs of products, salespersons, logistics, customers, and transactions were all set to 'Text' datatype and format. This made it easier to establish relationships and link tables to each other.

Customising Tables

Customers

Apart from the 'key edits' mentioned above, few changed were necessary. None of the columns had datatypes set to them, so appropriate datatypes were allocated. The 'salespeople_unique_identification' column was renamed to 'SALES_ID' to enable consistency in the dataset. The 'customer_address' column was split up into 'CUSTOMER_CITY' and 'CUSTOMER_COUNTRY' for a more detailed analysis. The column 'newin2023' seemed redundant as the same data could be extracted from the 'customer_since_year' column, although it was not removed, and its datatype was set to 'True/False'.

Logistics

The logistics text file was delimited using semicolon and the dataset overall seemed fine. Similar to the Customers table, the 'LOGISTIC_OFFICE_LOCATION' column was broken down into 'LOGISTIC_CITY' and 'LOGISTIC_COUNTRY' for a deeper analysis. Coincidentally, but appropriately, the datatype of all its columns were set to 'Text'. 5 entries had 'CONTACT_NUMBER' and 1 had 'CONTACT_EMAIL' data missing. Although this was ignored as it was unlikely to influence our analysis in any way. Also, none of the entries had both the details missing, so if needed, a contact detail was still available.

Products

Firstly, the common changes mentioned above were performed for the PRODUCT_ID, PRICE_PER_UNIT, and the COST columns. 6 entries had 'PRICE_PER_UNIT', 4 had 'PRODUCT_COLOUR', and 2 had 'COST' data missing. The 'PRICE_PER_UNIT' had an extreme value for 'PRODUCT_ID' 29 as 7000000, which when mapped with other products of same category, was understood to be £7. The missing values were found in a similar way after mapping them with similar products and a new column 'PRODUCT_PRICE' was created using queries to handle these issues. Also, the missing values of colour were easily drawn into 'PROD_COLOR' from the 'PRODUCT_NAME' column, as all names contained colour of the products as well. Column 'PRODUCT_COST' was created to map the missing values and to handle an erroneous negative value of 5.

Transactions

After making the changes for IDs, amount mentioned in the 'key edits', the dataset appeared quite effective and had no missing values. The format for 'TRANSACTION_DATE' was set to DD/MM/YYYY and the datatype for 'QUANTITY' was kept as 'Whole number'.

Salespeople

The salesperson dataset was quite appropriate by default requiring no major change. Although, as a key edit, the datatype of 'SALES_ID' was converted to 'Text'.

Calculated Measures and Columns & Data Groups

Multiple columns and measures were created using Data Analysis Expressions (DAX) to enable better calculations and smooth analysis of data.

Products

In the products table, the 6 product categories were further characterised into 10 product sub-categories to understand the dynamics of sales and profits in a more structured manner. TOTAL_PROFIT and TOTAL_QUANTITY were created to have profit generated and quantity sold of individual products handy. Also, MARGIN_PERCENTAGE column was created for having margin percentage of each product.

Transactions

A data group was created in Transactions table to filter the transactions before and onwards July, to analyse the effect of company's sustainability program. Profit measure was created in the table to have the total profit from all transactions handy. TRANSACTION_PROFIT column was created to calculate profit from each

transaction, to pair it up with other variables for an exhaustive analysis. WEEK_NUMBER column was created using the TRANSACTION_DATE column to examine the trend of sales and profit through weeks.

Logistics

A sustainability data group was created in logistics column to group exclusive air shipments and all other shipments modes altogether to analyse the influence of the sustainability program.

Customers

A customer increase in 2023 measure was created in the Customers table to measure the acquiring ability of the company.

Relationships

After following proper nomenclature of column names and variables throughout the dataset, relationships were established to have a proper connection between the tables (Figure 1), in order to access data from one another.

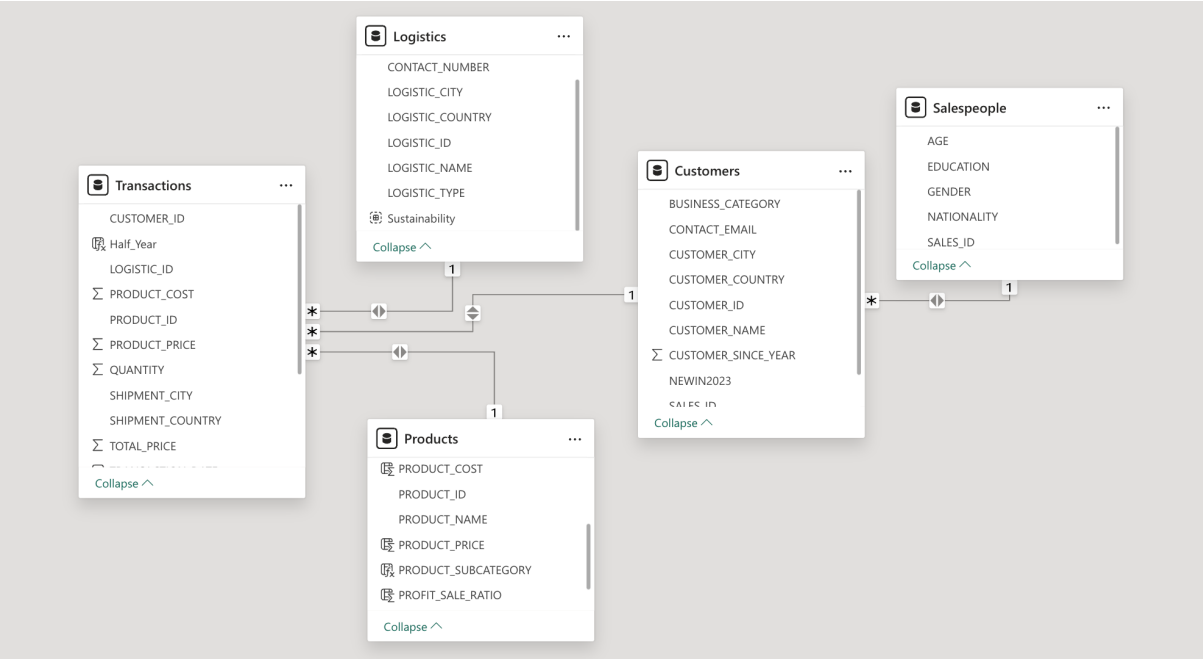

Figure 1: Relationship between tables.

Accessibility

Accessible colour-safe theme has been used to enhance accessibility of the reports, and to provision a better black and white print of the salespeople report.