**DATA CLEANING**

The data was retrieved from the FuelCheck website, covering the price data for the years 2024 and 2025. This data was then combined into a single file to create a consolidated dataset. To ensure the integrity of the data, several functions were utilized, such as `head()` and `shape()`, to observe the structure and dimensions of the dataframe in an initial check for cleaning the dataset. The `shape()` function indicated that the dataset consisted of 1,128,302 rows and 8 columns. The `columns()` function displayed the names of the dataset columns: ‘ServiceStationName', 'Address', 'Suburb', 'Postcode', 'Brand', 'FuelCode', 'PriceUpdatedDate', and 'Price'. Additionally, the `describe()` function was employed to check the distribution of the numerical data, specifically focusing on the 'Price' column. The `info()` function provided a general overview of the datatype of each column and identify any null data.

Subsequently, the presence of missing values was checked using the `isnull` function, which confirmed that there were no missing values in any of the dataset columns. The next step involved checking for duplicate entries to ensure the uniqueness of the dataset. The columns examined for **duplicates** included 'ServiceStationName', 'Address', 'FuelCode', and 'PriceUpdatedDate', given that there might be multiple entries for the same station at the same time. The results revealed that there were 38 rows that were duplicated across these four columns, with some showing different prices. Consequently, **the duplicated rows were eliminated**, and the prices were averaged to maintain consistency.

Lastly, **the datatype of the 'PriceUpdatedDate' column was converted.** The `info()` function indicated that this column was of object type when it should have been formatted as a date. Thus, the column was converted to date format using `pd.to\_datetime`.

**ANOMALY ANALYSIS**

1. Anomaly Price (Outlier)

This anomaly identifies fuel prices at stations that are significantly higher or lower than the standard prices, using z-score analysis on the price column. The z-score measures how many standard deviations a value is from the mean. This method helps to pinpoint stations that may have pricing errors, system glitches, or uncommon pricing strategies such as promo period.

1. Sudden Jump Price (>50%)

This anomaly detects unusual spikes or drops in fuel prices over time. First, the dataset is sorted by Service Station Name, Fuel Code, and Price Updated Date. Then, the percentage change in price for each fuel type at each station is calculated. If a price change exceeds 50% compared to the previous price, it will be flagged. This method captures potential data entry issues or price manipulation.

1. Frequencies Update Anomalies

Service stations usually update their fuel prices several times per day. This anomaly identifies stations that update their prices more frequently than expected, using the Price Updated Date column. Any station that updates its price more than 20 times in a single day is flagged. This method helps to identify indicators of erroneous updates.