Full Stack Engineer Data Tech Task

XYZ plc has an internal marketing team that collects retail transactions data from their online sales. They receive the data in the form of an Excel file which contains three sheets:

- Transactions
- Products
- Customers

The data is received ad-hoc multiple times a week. There could be 3 files each week or if there is less data there could even be 2 files per week.

The marketing team have been carrying out a manual task of identifying the top 5 customers, they are keen to automate this task in an ETL pipeline.

Furthermore, they require some data transformations as part of this ETL pipeline which delivers the data to a data lake for dashboard analysis. Due to the sheer volume of data, the data should be delivered as three separate Parquet files.

One for transactions, one for products and one for customers.

The marketing team are open to the ETL pipeline being built in either Python or JavaScript or both if the solution requires it, in addition to using libraries from either language.

Manual Task Automation 1:

The marketing team wants to reward their top 5 purchasing customers by total revenue. What are the full names of these five customers? Output the results into a one column CSV report which will be stored in the data lake.

As we receive multiple files per week, we want to understand what days we take the top 5 customers for. This should be clearly indicated in the naming of the file.

Data Transformation 1:

The customers that do not fill in their email purchased online as a guest and did not sign up with an account. When the marketing team create their dashboards, they like to carry out analysis on individual emails however, due to missing values they cannot get a clear indication of how many customers are not signing up.

Implement a solution so when the marketing team use the email field in their analysis, they have a clear indication of the number of missing email records.

Data Transformation 2:

The ETL pipeline for this data will load the data into the final data lake. The source data does not give any indication into what day this data is received. As we can receive multiple ad-hoc files each week, when the marketing team carry out their analysis, they will not know what days the files correspond to.

How can we go about displaying this data in the final data lake? The result needs to be clear and concise to the marketing team, they should be able to identify what day the file came in both in their analytics tool ***and*** in the data lake.

Data Transformation 3:

As we receive a large volume of data, we must store the data in the data lake as Parquet files. Convert the CSV's to parquet files and validate the results.