Data Intake Report

Name: **G2M Insights for Cab Investments**
Report date: **13th April 2025**
Internship Batch: **LISUM44**
Version:
Data intake by: **Cynthia Onyia**
Data intake reviewer:
Data storage location: https://github.com/MelCyn/Go-2-Market-Analysis/tree/main

Tabular data details: For this project, I was given four csv files, and the details of each of the files are presented in the table below.

1. Cab_Data.csv

Total number of observations	359392
Total number of files	-
Total number of features	7 (Transaction ID, Date of Travel, Company, City, KM travelled, Price Charged, Cost of Trip)
Base format of the file	.csv
Size of the data	20.7MB

2. City.csv

Total number of observations	20
Total number of files	-
Total number of features	3 (City, Population, Users)
Base format of the file	.csv
Size of the data	0.001MB

3. Customer _ID.csv

Total number of observations	49171
Total number of files	-
Total number of features	4 (Customer ID, Gender, Age, Income (USD/Month))
Base format of the file	.csv
Size of the data	1.03 MB

4. Transaction_ID.csv

Total number of observations	440098
Total number of files	-
Total number of features	3 (Transaction ID, Customer ID, Payment_Mode)
Base format of the file	.csv
Size of the data	8.8MB

Understanding the Data:

- This data contains information from two cab companies: Pink Cab and Yellow Cab, over two years (2016 - 2018). It has four different sections, each with its primary key and giving insights on daily transactions in terms of travel or rides each customer embarks on or books for.
- After merging all four different data sections into one uniform data set, a bigger picture emerges from the data. The insights derived from understanding this data give the Go-to-Market (G2M) strategy for making informed decisions.
- A good look was taken into the basic statistics of the data, such as checking for missing/null (NAN) values using the code: **merged.isna().sum**(). This gave a total of 80706 rows with missing values in 9 columns. I decided to drop these rows with null values from the data, as key pieces of information were missing and filling them with values might not give a clearer picture. I also dropped these rows because there would be no significant impact on the data, as it just makes up 18.3% of the data, compared to 81.7% (359392 rows) that is left, which also gives a very good amount needed for the analysis.
- After dropping the null values, I changed my dataframe name from merged to cleaned_df. I went ahead to identify if each record or row is unique by checking for duplicates using this code: **len(cleaned_df[cleaned_df.duplicated()])**. The outcome was zero (0). So for this data, each observation (359392 rows) is unique.