DEBRE BIRHAN UNIVERSITY

COLLEGE OF COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING

COURSE TITLE: FUNDAMENTAL OF BIG DATA ANALYTICS AND BUSINESS INTELIGENCE

INDIVIDUAL ASSIGNMENT

NAME: *ABDULAZIZ MUSA*

ID NO: *DBUR/0597/13*

SUBMITTED TO: *DERBEW FELASMAN(MSc)*

SUBMITION DATA: *2/13/2025*

Documentation for ETL Pipeline for Commerce Data

This documentation provides a detailed explanation of the ETL (Extract, Transform, Load) pipeline implementation for the commerce data table. The project is based on the assignment requirements provided and the code you implemented.

Project Objective

The goal of this project is to extract data from a large e-commerce dataset, clean and transform it, store it in a PostgreSQL database, and create visualizations in Power BI to derive meaningful insights.

Data Extraction

Data Source

- The data was downloaded from an external source Kaggle and loaded into a Pandas DataFrame.

Code Snippet for Extraction:

```
import pandas as pd

# Load data into a Pandas DataFrame
data = pd.read_csv(r"C:\Users\Edu\Downloads\archive (4)\commerce_data.csv")

# View top and bottom rows of the data
data.head()
data.tail()

```

Purpose:

- The data was loaded to inspect its structure and identify any potential issues like missing or duplicate values.

Data Transformation

The data transformation phase involved:

Removing Duplicates:

- Checked for and removed duplicate rows to avoid redundant entries.

```
data.duplicated().sum()  # Check for duplicates
data.drop_duplicates(keep='first', inplace=True)  # Remove duplicates
```

Handling Missing Data:

- Identified missing values to plan appropriate handling strategies:

```
data.isnull().sum()
```

- In the code provided, missing value treatment has been inspected but didn't find any missing data.

Data Loading

The cleaned data was loaded into a PostgreSQL database using SQLAlchemy for efficient connection handling.

Database Connection

- PostgreSQL credentials:

Username: postgres

Password: postgres123

Host: localhost

Port: 5432

Database: postgres

Code for Database Connection:

```
from sqlalchemy import create_engine
```

Create a database connection

```
username = 'postgres'
password = 'postgres123'
host = 'localhost'
port = 5432
db_name = 'postgres'

engine = create_engine(f'postgresql://{username}:{password}@{host}:{port}/{db_name}')

```

Loading Data to PostgreSQL:

Load data into PostgreSQL

```
data.to_sql('commerce_Data', engine, if_exists='replace', index=False)
```

Close the connection

```
engine.dispose()
```

Data was loaded into the `commerce_Data` table, replacing any existing data.

Data Visualization in Power BI

Key Visualizations Suggested:

Based on the data:

1. Sales Trends Over Time:

- Line chart to display sales trends over time using the `time` and `price` columns.

2. Brand Performance:

- Bar chart showing total sales per brand using the `brand` and `price` columns.

3. User Behavior:

- Analyze user purchase frequency with bar charts using the `user id` and `session` columns.

4. Event Analysis:

- Use `event name` and `price` to understand how events impact sales.

Follow the link below to view visualized data

https://app.powerbi.com/links/VNT8zhC-UA?ctid=1695066a-e388-40d1-8ed5-5d0b28ba9f80&pbi_source=linkShare

Design Choices

Database Schema:

- A single relational table (`commerce_Data`) was used to store all the cleaned e-commerce data.

Cleaning Choices:

- Removed duplicates to ensure data quality.
- Inspected missing values.

Conclusion

The ETL pipeline successfully extracted, transformed, and loaded the data into a PostgreSQL database. Power BI was used to generate meaningful visualizations and insights, providing a comprehensive view of the commerce data for analysis and decision-making.