Phase 4: Development Part 2 -

ETL Processes:

Implement ETL Processes to extract, transform, and load data into the data warehouse.

The ETL (Extract, Transform, Load) process is a fundamental data integration methodology used to collect, transform, and load data from various sources into a target database or data warehouse. Here's a brief explanation of each phase:

1. Extract: In the "Extract" phase, data is retrieved from source systems such as databases, files, or external APIs. This extraction can be full (all data) or incremental (only new or changed data since the last extraction). The goal is to gather the required data from diverse sources.
2. Transform: During the "Transform" phase, the extracted data is cleaned, structured, and converted into a suitable format for analysis and storage. Transformations may involve data cleansing, validation, aggregation, and the application of business rules to ensure data quality and consistency.
3. Load: In the "Load" phase, the transformed data is loaded into the target database or data warehouse. This phase involves populating tables, ensuring data integrity, and maintaining referential integrity, if applicable. Data loading can be batch-based or real-time, depending on the specific requirements of the system.

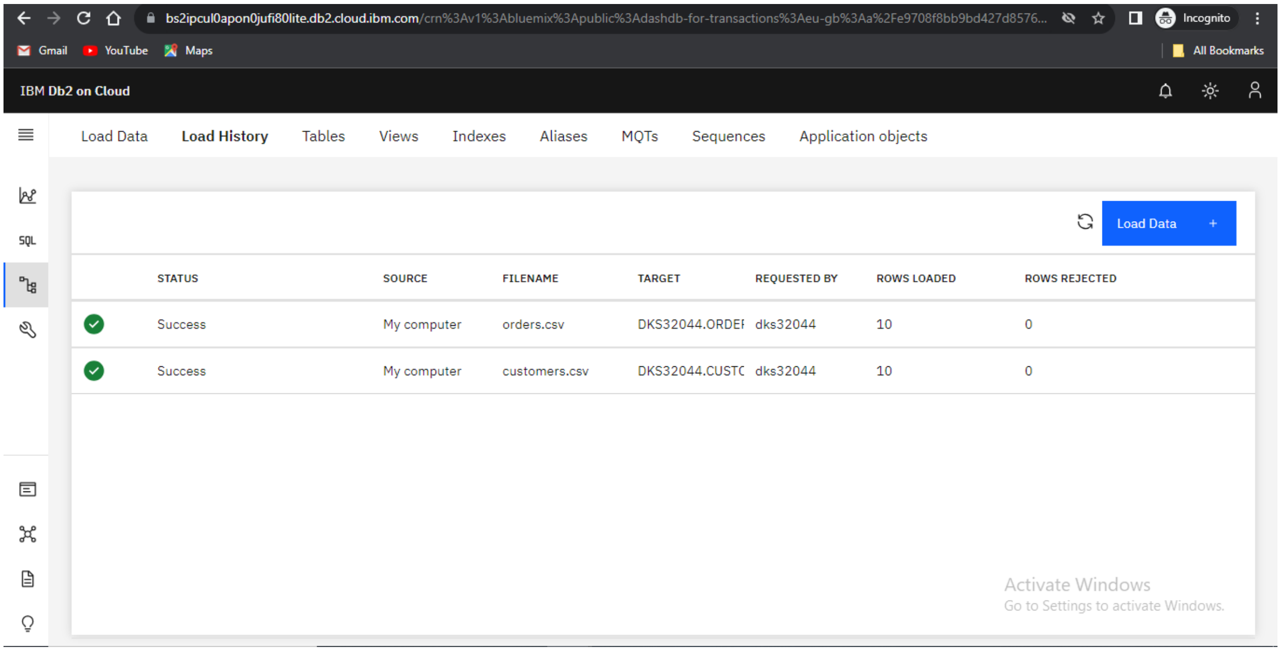
The ETL process is crucial for ensuring that data in the target system is accurate, consistent, and ready for analysis. It plays a vital role in business intelligence, reporting, and decision-making by providing a reliable and well-structured data foundation for organizations

CODE:

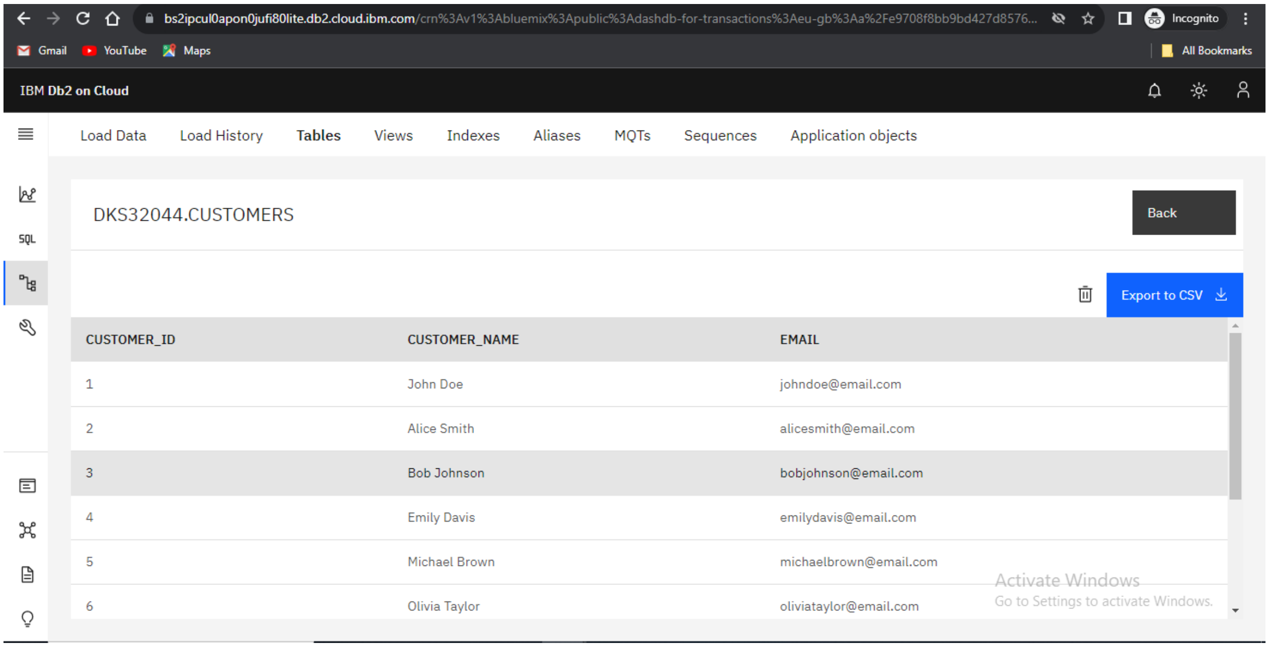
1. Extract (E): Extract data from an external source (e.g., a CSV file) into a staging table.

```
# Read customer and orders data from CSV files

        customer_data = pd.read_csv('customer.csv')

        orders_data = pd.read_csv('orders.csv')
```

```
# ETL Process

# Transform: Merging customer and orders data

        merged_data = pd.merge(customer_data, orders_data, on='CustomerID')


# Add a new 'TotalAmount' column by summing 'Amount' for each customer

        total_amount = orders_data.groupby('CustomerID')['Amount'].sum().reset_index()

        merged_data = pd.merge(merged_data, total_amount, on='CustomerID', how='left')

        merged_data.rename(columns={'Amount': 'TotalAmount'}, inplace=True)


# Save the transformed data to a CSV file

        output_csv_filename = 'output_data_with_total_amount.csv'

        merged_data.to_csv(output_csv_filename, index=False)
```

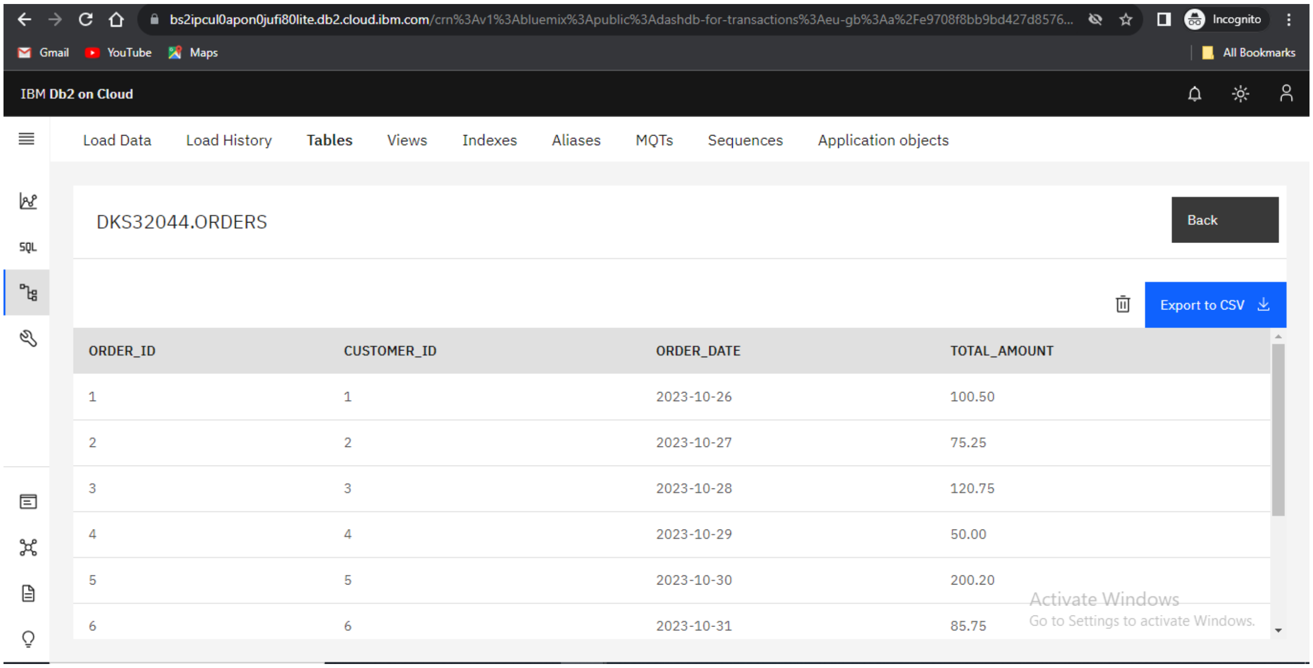

bs2ipcul0apon0jufi80lite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aeu-gb%3Aa%2Fe9708f8bb9bd427d8576...
Incognito
Gmail
YouTube
Maps
All Bookmarks
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
Load Data +
Activate Windows
Go to Settings to activate Windows.

	STATUS	SOURCE	FILENAME	TARGET	REQUESTED BY	ROWS LOADED	ROWS REJECTED
✓	Success	My computer	orders.csv	DKS32044.ORDEI	dks32044	10	0
✓	Success	My computer	customers.csv	DKS32044.CUSTC	dks32044	10	0


bs2ipcul0apon0jufi80lite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aeu-gb%3Aa%2Fe9708f8bb9bd427d8576...
Incognito
Gmail
YouTube
Maps
All Bookmarks
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
Back
Export to CSV
Activate Windows
Go to Settings to activate Windows.

DKS32044.CUSTOMERS

CUSTOMER_ID	CUSTOMER_NAME	EMAIL
1	John Doe	johndoe@email.com
2	Alice Smith	alicesmith@email.com
3	Bob Johnson	bobjohnson@email.com
4	Emily Davis	emilydavis@email.com
5	Michael Brown	michaelbrown@email.com
6	Olivia Taylor	oliviataylor@email.com


bs2ipcul0apon0jufi80lite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aeu-gb%3Aa%2Fe9708f8bb9bd427d8576...
Incognito
Gmail
YouTube
Maps
All Bookmarks
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
Back
Export to CSV
Activate Windows
Go to Settings to activate Windows.

DKS32044.ORDERS

ORDER_ID	CUSTOMER_ID	ORDER_DATE	TOTAL_AMOUNT
1	1	2023-10-26	100.50
2	2	2023-10-27	75.25
3	3	2023-10-28	120.75
4	4	2023-10-29	50.00
5	5	2023-10-30	200.20
6	6	2023-10-31	85.75