

End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System



Project Overview:

<https://github.com/anandjha90/MASTER-THE-ART-OF-EXTRACT-TRANSFORM-LOAD-WITH-ALTERYX/blob/main/End%20To%20End%20Project/Alteryx%20Mini%20Project.docx>

This project simulates a **Swiggy-like food ordering system**, where customers place orders from various states in India. The goal is to **build a complete ETL pipeline using Alteryx** that reads transactional data from **AWS S3**, processes it within **Alteryx**, and loads it into **Snowflake**, ensuring **incremental data updates**. A **Power BI dashboard** will visualize key **KPIs** based on the final tables.

Project Components:

- Data Source:** AWS S3 (CSV files for Customers, Transactions, Coupons, and State data)
- ETL Tool:** Alteryx (for all data processing, cleaning, transformations, and incremental loads)
- Target Database:** Snowflake (storing the final cleaned tables)
- Dashboarding Tool:** Power BI (for visualizing KPIs)
- Data Volume:** ~500-600K records (majority in Transactions)

Key Highlights of the Project:

- End-to-end pipeline fully designed in Alteryx** – No manual processing in Snowflake
- Incremental loading from AWS S3** – Only new records are appended
- Full-fledged data cleaning & transformation** using maximum Alteryx tools
- Realistic Indian dataset for a Swiggy-like platform**
- Interactive Power BI dashboard showcasing business KPIs**

Architecture Overview:

1. **Data Source (AWS S3)**
 - Store raw CSV files for Customers, Transactions, Coupons, and States in S3 under structured folders.
 - Incremental transaction files are also stored in S3.
2. **ETL Processing (Alteryx)**
 - Read data from S3 using Input Data components.
 - **Perform transformations:** filtering, joins, sorting, aggregations, masking, and deduplication.

End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System



- **Apply Business Rules** (Joins between tables, data validation).
- Handle incremental loads using File Iterator and Append Operations.
- Load cleaned data into Snowflake using the Output Data component.

3. Data Warehouse (Snowflake)

- Stores final transformed tables (Customers, Transactions, Coupons, Restaurants, States).
- Enforces relationships via foreign keys.
- Data is available for Power BI reporting.

4. Visualization (Power BI)

- Connects to Snowflake for real-time reporting.
- Generates KPIs on revenue, top restaurants, coupon usage, and state-wise order distribution.

Project Data Files :

https://github.com/anandjha90/MASTER-THE-ART-OF-EXTRACT-TRANSFORM-LOAD-WITH-ALTERYX/tree/main/End%20To%20End%20Project/Project_Datasets

Step-by-Step Implementation:

1. Generate and Upload Data to AWS S3

Create CSV files (~500K+ records) for the following tables:

- **customers.csv** – Customer details (ID, Name, Phone, Email, Address, State Code)
- **restaurants.csv** – Restaurant details (ID, Rest_Name, Address, State Code)
- **coupons.csv** – Available and used coupons (Coupon Code, Discount %, Validity, Usage Count)
- **states.csv** – Indian states reference table (State Code, State Name, Region)
- **transactions.csv** – Order transactions (Txn ID, Customer ID, Restaurant ID, Amount, Date, Payment Mode, Coupon Used)

🚀 End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System 🔒



📌 Relationships for the following tables:

State_Code is present in **Customers**, **Restaurants**, and **States** for state-level analysis.

Customer_ID in **Transactions** links to **Customers**.

Restaurant_ID in **Transactions** links to **Restaurants**.

Coupon_Used in **Transactions** can be matched with **Coupon_Code** in **Coupons**.

📌 DDL Scripts :

-- Customers Table

```
CREATE TABLE Customers (
    Customer_ID STRING PRIMARY KEY,
    Name STRING,
    Phone STRING,
    Email STRING,
    Address STRING,
    State_Code STRING REFERENCES States(State_Code)
);
```

-- Transactions Table

```
CREATE TABLE Transactions (
    Transaction_ID STRING PRIMARY KEY,
    Customer_ID STRING REFERENCES Customers(Customer_ID),
    Restaurant_ID STRING REFERENCES Restaurants(Restaurant_ID),
    Amount FLOAT,
    Date DATE,
    Payment_Mode STRING,
```

End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System



```
Coupon_Used STRING REFERENCES Coupons(Coupon_Code)
```

```
);
```

-- Coupons Table

```
CREATE TABLE Coupons (
```

```
    Coupon_Code STRING PRIMARY KEY,
```

```
    Discount_Percentage STRING,
```

```
    Validity DATE,
```

```
    Usage_Count INT
```

```
);
```

-- States Table

```
CREATE TABLE States (
```

```
    State_Code STRING PRIMARY KEY,
```

```
    State_Name STRING
```

```
);
```

-- Restaurants Table

```
CREATE TABLE Restaurants (
```

```
    Restaurant_ID STRING PRIMARY KEY,
```

```
    Restaurant_Name STRING,
```

```
    Address STRING,
```

```
    State_Code STRING REFERENCES States(State_Code)
```

```
);
```

🚀 End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System 🔒



📌 Store these CSVs in structured S3 folders:

- s3://swiggy-data/customers/customers.csv
- s3://swiggy-data/transactions/transactions.csv
- s3://swiggy-data/coupons/coupons.csv
- s3://swiggy-data/states/states.csv

2. Extract Data from AWS S3 using Alteryx and copy into above tables

📌 Use the following Alteryx tools/components to read data:

- ◆ **Input Data Tool** – Connect to AWS S3 and read CSV files
- ◆ **AWS S3 Download Tool** – Fetch files dynamically from the bucket

3. Data Cleaning & Transformation in Alteryx

📌 Apply various Alteryx transformations before loading into Snowflake:

- ◆ **Remove duplicates** – Ensure no redundant data
- ◆ **Filter invalid records** – Remove null/missing data
- ◆ **Data sorting & aggregation** – Organize transactions by date, filter active coupons, etc.
- ◆ **Use Joins & Unions:**
 - **Left Join** – Get all customers with their latest transaction details
 - **Right Join** – Identify missing coupon usage data
 - **Inner Join** – Match valid transactions with state information
 - **Full Outer Join** – Merge customer and transaction data

📌 Alteryx Components Used(Not limited to below):

- ✓ **Filter Tool** – Remove unnecessary rows
- ✓ **Sort Tool** – Arrange data for efficient loading
- ✓ **Summarize Tool** – Aggregate sales & customer insights
- ✓ **Join Tool** – Merge data across different tables
- ✓ **Formula Tool** – Add calculated fields (Discount Applied, Final Bill Amount)

4. Load Processed Data into Snowflake via Alteryx

📌 Create final tables in Snowflake using Alteryx:

- ◆ **Output Data Tool** – Write final transformed data to Snowflake
- ◆ **Incremental Load Setup** – Append only new records using File Iteration & Timestamp filters

End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System



Final Tables in Snowflake:

- 1.**customers_final** – Processed customer details
- 2.**transactions_final** – Cleaned transaction data
- 3.**coupons_final** – Valid & used coupons
- 4.**states_final** – Reference state information

5. Power BI Dashboard – KPIs & Insights

 Connect Power BI directly to Snowflake & build interactive dashboards with these KPIs along with any additional which you think might be useful. Use your creativity

-  **Total Revenue by State**
-  **Most Frequently Used Coupons**
-  **Top Customers by Order Volume**

-  **Repeat Customers**
-  **Order Trends Over Time**
-  **Popular Payment Modes**

6. Data Validation Steps :

1. Source Data Validation (AWS S3)

- File Integrity Check:** Ensure CSV files are not corrupt and can be read.
- Schema Validation:** Verify that column names and data types match the expected format.
- Row Count Check:** Ensure the expected number of records exist in each file.
- Duplicate Records:** Identify and remove duplicate rows before processing.

2. Data Transformation Validation (Alteryx)

- Null & Missing Values Handling:** Detect NULL values and replace them with defaults or remove invalid rows.
- Data Type Consistency:** Ensure numerical fields contain numbers, dates are valid, and text fields do not contain unexpected characters.
- Business Rule Validation:**
 - Transactions must have a **valid customer ID** from the customers table.
 - Coupon usage must have a **matching coupon code** from the coupons table.

End-to-End ETL Pipeline with Alteryx, AWS S3 & Snowflake for a Swiggy-Like Food Ordering System



- State codes in transactions should exist in the states table.
 - ✓ **Referential Integrity Check:** Ensure foreign keys (like customer_id, coupon_id) exist in parent tables.
 - ✓ **Join Validations:** Ensure expected record counts match before and after applying joins.
 - ✓ **Incremental Data Handling:** Ensure only new records are inserted while avoiding duplicates.
-

3. Snowflake Data Validation (Target System)

- ✓ **Data Count Comparison:** Compare source vs. Snowflake row counts after ingestion.
 - ✓ **Primary Key Uniqueness:** Ensure no duplicate primary keys exist in tables.
 - ✓ **Masked Data Validation** (if applicable): Ensure PII data is masked based on user roles.
 - ✓ **Data Completeness Check:** Ensure all required fields have valid values post-load.
 - ✓ **Aggregation Check:** Verify totals and counts before and after transformation.
-

4. Power BI KPI Validation

- ✓ **Cross-check Calculations:** Validate KPIs (total orders, revenue by state, coupon usage) against source data.
 - ✓ **Data Freshness Check:** Ensure the latest data is available in reports.
 - ✓ **Data Drill-down Accuracy:** Verify that drill-downs in Power BI match the transactional data.
-

Automated Testing Suggestions (Feel free to explore else will do it in further sessions)

- **Unit Testing:** Validate individual transformations in Alteryx.
- **Regression Testing:** Ensure changes do not break existing workflows.
- **Threshold Alerts:** Set alerts for missing or unexpectedly high/low records.
- **ETL Pipeline Logging:** Capture logs for failures or anomalies during execution.

*****Thank You*****