# SSIS Case Study

## Comprehensive ETL Solutions for Retail Data Integration

A retail company, RetailWorks Inc., needs to integrate and manage data from various sources for its reporting and analytics platform. The company maintains sales data in flat files generated from point-of-sale systems and customer data in an SQL Server database. Additionally, they need to extract data from flat files for reporting purposes and transfer some transformed data back into the SQL Server database for further analysis. The operations include:

1. Extracting data from a flat file and loading it into another flat file.
2. Extracting data from the SQL Server database and loading it into a flat file.
3. Extracting data from a flat file and loading it into the SQL Server database.
4. Extracting data from one SQL Server database and loading it into another SQL Server database.

## Objective

The primary objective is to create a comprehensive ETL solution using SSIS that integrates data from flat files and SQL Server databases, ensuring data consistency and availability for the company's reporting and analytical needs. The specific goals include:

- Efficiently transfer and transform data between different formats and storage systems.
- Ensure data integrity and accuracy throughout the ETL processes.
- Automate the ETL workflows to minimize manual intervention and errors.

## Data Description

- **Sales Data (Flat File):** Contains daily sales transactions with fields such as Date, TransactionID, ProductID, Quantity, Price, and StoreID.
- **Customer Data (SQL Server Database):** Contains customer information with fields such as CustomerID, Name, Email, Phone, and Address.
- **Transformed Sales Data (SQL Server Database):** Will store processed sales data with additional fields for reporting purposes, such as TotalAmount and StoreRegion.

## Tasks

1. **Flat File to Flat File:**

   - Read sales data from the source flat file.
   - Filter records for a specific date range and calculate the total amount for each transaction.
   - Write the transformed data to a new flat file for archival purposes.

2. **Database to Flat File:**

   - Query the SQL Server database to retrieve customer data.

   - Format the data into a CSV-friendly structure.

   - Write the formatted data to a flat file for use in a third-party mailing system.

3. **Flat File to Database:**

   - Read daily sales data from the source flat file.

   - Calculate the total sales amount and map StoreID to StoreRegion.

   - Insert the transformed data into the SQL Server database.

4. **Database to Database:**

   - Retrieve relevant customer data from one SQL Server database.

   - Cleanse and standardize customer contact information.

   - Insert the cleansed data into another SQL Server database for use by the marketing team.

## Expected Outcome

- A new flat file containing filtered and transformed sales data, ready for archival or further analysis.

- A flat file with formatted customer data, ready for import into a third-party system.

- Transformed sales data successfully loaded into the SQL Server database, with calculated fields such as TotalAmount and mapped StoreRegion.

- Cleaned and standardized customer data loaded into the target SQL Server database, ensuring data consistency and accuracy for marketing campaigns.