# ****Data Migration Plan: Acquired Customer to Northwind Database****

## ****1. Introduction****

The goal of this migration is to extract, transform, validate, and load data from the “acquired customer” database into the Northwind database. The Northwind database will serve as the target, with data related to customers, products, orders, and sales to match the relevant data from the "acquired customer" database.

This plan covers:

* Data Extraction
* Data Transformation
* Data Validation
* Data Loading
* Data Integrity checks
* Challenges faced during migration and solutions

## ****2. Migration Scope****

The migration will focus on the following tables from the Northwind database, as they align with the data in the “acquired customer” database:

* **Customers**: From Entities-related data.
* **Products**: From Items-related data.
* **Orders**: From Batches-related data.
* **OrderDetails**: From Transactions-related data.
* **Employes**: From locations data.

## ****3. Extraction from the Source Database****

### ****Source Database: Acquired Customer Database****

* **Extract Data**: Identify each table and extract all the information to handle all the change with python and sql.
* **Tools/Method**: Use SQL queries and python
* **Data Structure**: Pyton dataframe.

The idea was to create a dynamic method where all the information from all the tables was taken using a previous structure created in JSON that combined the core necessities:

* Identify the source structure and the target structure.
* Include the necessary validations for each structure.
* Include the transformations required for each structure.

## ****4. Transformation and Normalization****

Data transformation involve several steps to make the data compatible with the Northwind database schema, ensuring consistency and data integrity.

### ****4.1 Data Normalization****

**Customer Data**:

* Standardize customer names (e.g., upper case for consistency).
* Normalize address formatting (e.g., trimming spaces, consistent postal codes).

**Product Data**:

* Check product descriptions for length consistency (e.g., no product names exceeding 100 characters).

**Order Data**:

* Convert date formats (ensure consistency with Northwind’s date format YYYY-MM-DD).
* Ensure order total values are in the correct format (e.g., currency or float).

### ****4.2 Data Type Conversion****

**Date Formats**:

Convert date fields to match the format used in Northwind (mdY).

**Example**:

"2024-11-29" in the source database should be converted to "11-29-2024 00:00:00" for the target database.

**Currency Handling**:Ensure that any monetary fields (e.g., price, total\_value) are correctly cast to the DECIMAL or MONEY type in Northwind.

### ****4.3 Mapping Data to Target Schema****

**Example of Source to Target Field Mapping**:

* Locations"."location\_id": Employees."EmployeeID",
* "Locations"."street\_addr": Employees."Address",
* "Locations"."municipality": Employees."City",
* "Locations"."region\_code": Employees."Region",
* "Locations"."postal\_code": Employees."PostalCode"

## ****5. Data Validation****

Before loading the data into the Northwind database, the step to take is validate the integrity of the data to ensure there are no discrepancies.

### ****5.1 Data Validation Steps****

**Null Value Validation**:

* Ensure that no critical fields (such as CustomerID, ProductID, OrderID,EmployeeID) are null in the extracted data.

**Range and Format Validation**:

* Check that all numerical values (e.g., quantity, price) are within valid ranges.
* Validate that dates are in the correct format and fall within reasonable ranges.
* Ensure that the length of specific fields is validated appropriately.

**Integrity Check**:

Ensure that OrderID exists in the orders table and ProductID exists in the products table before inserting data into order\_details.

## ****6. Loading Data into Northwind****

I can't complete the loading process because I was struggling with the Docker image of MSSQL. I didn't focus on this problem at the beginning because it was the final stage of the entire process. By the time I realized there were other types of solutions, it was too late to take action and meet the established timeline.

## ****7. Data Integrity and Verification****

This was the next stage where, for the same reason, I can't complete without making the loading process.

## ****Challenges Faced and Resolutions****

### ****Challenge 1****: Docker virtualization for MSSQL its not Anymore avaible for Arm64 (Macbook M1,...) architecture.

* **Issue**:I spent too much time investigating why the SMSQL image was not working, I cant acomplish the loadind data process.
* **Resolution**: I try to take advantege with the other things that I can acomplish.

### ****Challenge 2****: Different Date Formats

* **Issue**: The source database used a different date format (e.g., YYYY-MM-DD HH:MM:SS), while the Northwind database expected YYYY-MM-DD.
* **Resolution**: Used STR\_TO\_DATE function to convert the source date format into the target format during transformation.

### ****Challenge 3****: Currency Conversion

* **Issue**: The source data had product prices stored as strings with currency symbols.
* **Resolution**: Used SQL REPLACE() function to strip currency symbols and then converted to the appropriate FLOAT type.