# Project Documentation: ETL Process with SSIS Package

## Overview

This project involves creating an SSIS (SQL Server Integration Services) package using Visual Studio 2019 to perform ETL (Extract, Transform, Load) processes. The package collects daily reports data from a CSV file, performs transformations including adding a column for "End of Day," and loads the transformed data into a SQL Server database. This database serves as a data source for Power BI reports, providing up-to-date insights. Additionally, a File System Task is configured to move the processed file to an archive folder after loading it into the database.

## Step-by-Step Process

### Setting up the SSIS Package

1. Open Visual Studio 2019 and create a new Integration Services Project.
2. Add a new SSIS package to the project.

### Data Flow Configuration

1. Drag and drop a "Data Flow Task" onto the control flow design surface.
2. Double-click the Data Flow Task to switch to the Data Flow tab.
3. Add a "Flat File Source" component to the data flow.
4. Configure the Flat File Source to read data from the CSV file containing daily reports.
5. Add a "Data Conversion" transformation to the data flow.
6. Configure the Data Conversion transformation to convert data types of columns as necessary to match the destination.
7. Add a "Derived Column" transformation to the data flow.
8. Configure the Derived Column transformation to add the "End of Day" column.
9. Add an "OLE DB Destination" component to the data flow.
10. Configure the OLE DB Destination to load data into the "DailyReports" table in the "NorthAmerica" database.

### File System Task Configuration

1. Switch back to the Control Flow tab.
2. Drag and drop a "File System Task" onto the control flow design surface.
3. Configure the File System Task to move the processed CSV file to an archive folder.
4. Set up expressions or variables to dynamically handle file paths and names.

### Creating the SQL Server Database

**Database Design**

The database "NorthAmerica" is designed to store daily reports data and serve as a reliable source for Power BI reports. The key design aspects include:

* Normalization: Ensuring the data is organized efficiently to avoid redundancy.
* Indexing: Creating indexes on frequently queried columns to improve query performance.
* Data Integrity: Implementing constraints to maintain data accuracy and consistency.

**SQL Server Setup**

1. Open SQL Server Management Studio (SSMS) and connect to the SQL Server instance.
2. Create a new database named "NorthAmerica":

CREATE DATABASE NorthAmerica;

1. Use the newly created database:

USE NorthAmerica;

1. Create the "DailyReports" table using the provided SQL code:

CREATE TABLE [DailyReports](

[End of Day] DATE,

[Ticket #] VARCHAR(50),

[Company Name] VARCHAR(50),

[Site] VARCHAR(255),

[Date Opened] DATE,

[Last Update] DATE,

[Age] FLOAT,

[Machine Status] VARCHAR(50),

[Summary] VARCHAR(255),

[Status] VARCHAR(50),

[Service Type] VARCHAR(50),

[Service Sub Type] VARCHAR(100),

[Service Sub Type Item] VARCHAR(100),

[Team Name] VARCHAR(50),

[City] VARCHAR(50),

[State] VARCHAR(50),

[Country] VARCHAR(50),

[Parent] FLOAT

);

### Testing and Execution

1. Test the SSIS package by executing it from Visual Studio.
2. Verify that data is extracted from the CSV file, transformed, and loaded into the database.
3. Confirm that the File System Task moves the processed file to the archive folder.

### Deployment and Scheduling

**Best Practices for Deployment**

1. Package Protection: Use SSIS package protection levels to secure sensitive information.
2. Environment Configuration: Use environment variables and configuration files to manage settings.
3. Logging: Implement logging to monitor package execution and troubleshoot issues.
4. Error Handling: Design robust error handling and retry mechanisms.

**Deployment Steps**

1. Once testing is successful, deploy the SSIS package to the SQL Server Integration Services Catalog.
2. Create an environment in SSISDB to manage configurations.
3. Set up a SQL Server Agent job to schedule the execution of the SSIS package at the desired frequency (e.g., daily).

### Power BI Integration

* Data Source Configuration: Connect Power BI to the "NorthAmerica" database to create reports.
* Data Refresh: Configure scheduled refresh in Power BI to ensure reports display the latest data.

## Conclusion

This comprehensive SSIS package automates the ETL process for daily reports data, ensuring seamless extraction, transformation, and loading into a SQL Server database. The integration of the File System Task adds functionality for archiving processed files, maintaining data integrity, and facilitating ongoing data management tasks. Incorporating the Data Conversion step ensures that data types in the source match those in the destination, enhancing the accuracy and reliability of the ETL process. The database design supports efficient reporting in Power BI, providing valuable insights for decision-making.