DATA WAREHOUSING FOR SMALL SCALE ENTERPRISE

BY

EBUNOLUWA, OLOWO ADEMOLA

17010301042

DEPARTMENT OF COMPUTER SCIENCE AND MATHEMATICS, COLLEGE OF BASIC AND APPLIED SCIENCES, MOUNTAIN TOP UNIVERSITY, MFM PRAYER CITY

September, 2021

DECLARATION

I hereby declare that this project has been written by me and is a record of my own research work. It has not been presented in any previous application for higher degree of this or any other University. All citation s and sources of information are clearly acknowledged by means of reference.

…….

OLOWO, EBUNOLUWA

…….

Date

CERTIFICATION

This project titled, **DATA WAREHOUSING FOR SMALL SCALE ENTERPRISE,** prepared and submitted by **OLOWO EBUNOLUWA ADEMOLA** in partial fulfillment of the requirements for the degree of **BACHELOR OF SCIENCE** (**Computer Science)**, is hereby accepted.

…… (Signature and Date)

Dr F.A. Kasali

Supervisor

…… (Signature and Date)

Dr M.O Adewole

Coordinator of Department

DEDICATION

I dedicate this project work to God almighty.

ACKNOWLEDEMENT

I want to show my appreciation to my parents and also to every member of staff that helped me to work on this project research. Special appreciation to my supervisor Dr. F.A Kasali and the head of department of computer science and mathematics department.

ABSTRACT

This project work contains five chapters and references. The chapter one includes introduction to the data warehousing and the basic concepts to why and how data warehouses can be useful to small scale businesses, the chapter two contains literature review of the project summing up all the nitty-gritties of the data warehousing solution in view ranging from the history to facts tables and variety of dimension tables, the chapter three contains the methodology to design and develop the data warehouse as well as the necessary tools used in the development process and all of the requirements for the solution to be implemented by a user, the chapter four contains implementation details of the data warehouse highlighting the four major modules of this data warehousing solution: data source module, data integration module, data warehouse and business intelligence which may be known as reporting, the chapter five contains the summary being the result of my research and conclusion listing the recommendations and round-up.

TABLE OF CONTENTS

DECLARATION i
CERTIFICATION ii
DEDICATION iii
ACKNOWLEDEMENT iv
ABSTRACT v
TABLE OF CONTENTS vi
LIST OF FIGURES viii
LIST OF TABLES x
CHAPTER ONE: INTRODUCTION 1
1.1 Background to The Study 1
1.2 Statement of The Problem 2
1.3 Aim and Objectives 2
1.4 Scope and Limitations of Study 2
1.5 Significance of Study 2
1.6 Definition of Terms 3
CHAPTER TWO: LITERATURE REVIEW 4
2.0 LITERATURE REVIEW 4
2.1.1 Definition of data warehouse 4
2.1.2 History of data warehouse 5
2.2 Impact of Data Warehouses 5
2.3 Objectives of Proposed Data Warehouse 6
2.4 Inmon vs Kimball Data Warehouse: 7
2.5 Data Warehousing and Data Warehouse. 9
2.6 Fact and Dimension Tables 9
2.6.1 Fact tables 9
2.6.2 Dimension tables 9
2.7 Schema Types in Data Warehouse Design 11
2.7.1 The star schema 11
2.7.2 The snowflake schema 11
2.7.3 The constellation schema 12
2.8 Data warehouse types. 12
2.9 Information Systems. 13
2.10 Review of Related Works. 13
CHAPTER THREE: METHODOLOGY 16
3.1 Requirement Gathering 16
3.2 Architecture of The Designed Data Warehouse System 16
3.2.1 System of records (SOR) 16
3.2.2 System of integration (SOI) 16
3.2.3 System of analysis (SOA) 16
3.2.4 Business intelligence (BI) 18
3.3 INSTALLATION OF SOFTWARE 18

CHAPTER FOUR: IMPLEMENTATION OF DESGNED SYSTEM 19
4.1 Implementation Environment 19
4.3 Description of Developed System for Data Warehousing 20
4.3.1 Analysis of system of records 20
4.3.2 Implementation of system of integration 23
4.2.3.1 Creation of tables in database: 23
4.2.3.2 Data population from SSIS: 34
4.3.3 Implementation of business intelligence 53
CHAPTER FIVE: SUMMARY AND CONCLUSION 55
5.1 Summary 55
5.1 Limitations 55
5.2 Recommendation 55
REFERENCES 56
APPENDIX 59

LIST OF FIGURES

Figure 4.1: sales_datasample CSV data source (Example of System of Record) 22
Figure 4.2: Creation of Staging Table..... 24
Figure 4.4 Creation of Date Dimension Table..... 27
Figure 4.6: Creation of Product Dimension Table..... 29
Figure 4.7: Creation of Contact Dimension Table..... 30
Figure 4.8 Display of Dimension Tables..... 31
Figure 4.9: Creation of Fact Table and Relationship..... 32
Figure 4.11: Data Flow Task to Populate Staging Table..... 35
Figure 4.12 Module-to-module Sequence in Data Flow..... 36
Figure 4.13: Flat File Configuration and Column Structure..... 37
Figure 4.14: Data Conversion for Column Datatypes 38
Figure 4.15: Column Mapping From Data Source to Destination..... 39
Figure 4.16: Successful Execution of Data Flow Task From CSV to SQL Staging Table. 40
Figure 4.17: Selection of Populated Staging Table 41
Figure 4.18: Various Data Flow Tasks for Populating Dimension Tables..... 42
Figure 4.19: SQL Command Accessing for Product Dimension Table..... 43
Figure 4.20: Column Mappings for Columns From Staging Tables into Product Dimension Table. 44
Figure 4.21: Successful Execution of Data Flow Tasks From Staging Table to Product Dimension Table..... 45
Figure 4.22: Successful Execution of Data Flow Tasks From Staging Table to Contact Dimension Table..... 46
Figure 4.23: Successful Execution of Data Flow Tasks From Staging Table to StateDimension Table..... 47
Figure 4.24: Successful Execution of Data Flow Tasks From Staging Table to Date Dimension Table..... 48
Figure 4.25: Selection of Data From Dimension Table..... 49
Figure 4.26: SQL Command to Populate Data From Staging Table into Fact Table..... 50
Figure 4.27: Mapping Columns From Staging Table to Fact Table..... 51
Figure 4.28: Successful Execution of Data Flow Tasks From Staging Table to Facts Table..... 52
Figure 4.29: Data visualisation of data warehouse data with Microsoft Power BI 54

LIST OF TABLES

TABLE 2.1: INMON VS KIMBALL DWH ARCHITECTURE..... 8

Table 2.1 Slowly Changing Dimensions (Type 1)- Before Change. 10

Table 2.3: Slowly Changing Dimensions (Type 1)- After Change. 10

CHAPTER ONE
INTRODUCTION

1.1 Background to The Study

Small scale enterprises contribute highly to the economic buoyancy of any country in the world. A lot of focus is now lately placed on small scale enterprises following the success of SMEs in the economic growth of Asian countries Ojo, (2003), Small scale enterprises are concerned majorly with the satisfaction of customers in their environment but they are often growth-oriented. Although, as stated in Masurel and Montfort, (2006), not every business is growth-oriented and for certain enterprises, growth is voluntary. Data management is a common problem for small scale businesses that may face a lack of skilled staff. If data management is done properly the growth of a business is partially assured. According to a business intelligence website, bi-survey.com, 18% of companies are using 20 or more data sources for decision-making, businesses use data for decision making and that is a great deal.

Data Warehouse (DWH) is a Decision Support System (DSS) that integrates data from multiple heterogeneous sources and integrates them under a unified schema into a single site (Inmon, 2002). The modern Data Warehouse depends on the problem and it has no reference architecture or standard terminology except if there are some product-oriented architectures from the vendor i.e., the customer (Santoso & Yulia, 2017). This project examines how product-oriented modern warehouses can be implemented for small scale enterprises to ensure cost optimization and improved data management. This data warehouse would combine data from operational and transactional systems (ERP, CRM, finance apps, IoT devices, online and mobile systems) to allow small business owners to access data from a variety of sources and present it in a structured format, ready for analysis and further exploitation.

1.2 Statement of The Problem

Small scale enterprise owners are faced with a lot of record-keeping challenges which is an aftermath of the traditional record-keeping method (Pen and Paper). Errors could be easily overturned without many difficulties if record keeping is enabled digitally.

1.3 Aim and Objectives

This project aims to design and implement a data warehouse for small scale business enterprises and the specific objectives of this study are:

i. Identify a data collection model suitable for a large percentage of small-scale businesses.
ii. Design a data warehouse system to suit business processes of small-scale businesses generically.
iii. Implementation of data warehouse system.

1.4 Scope and Limitations of Study

This work will research different techniques, various database applications that can be used for developing a successful data integration system. This project will be focused on the design of data warehouse systems for small scale businesses. This study is restricted to only Lagos State, Nigeria.

1.5 Significance of Study

This research work is done to reduce the impact of poor management and other grossing problems affecting their growth as to the benefit of immediate users. The benefits of data warehousing include its ability to enhance business intelligence, improvement of data quality and consistency, increased time efficiency and enhanced performance. On the other

hand, Researchers will benefit from more anecdotes as regard documentations that guide to build and manage a data warehouse.

1.6 Definition of Terms

DSS- Decision support systems: A decision support system (DSS) is an information system that aids in the making of corporate or organizational decisions.

DWH- Data Warehouse: This is a repository of heterogeneous sources accumulated under a unified schema at a single site to facilitate decision making.

SSE- Small Scale Enterprise: This is a term used to describe a small, privately owned business with a small number of employees and low sales volume. This is the typical outcome of a successful small business's slow and steady expansion.

MSE- Medium Scale Enterprise: This is the typical outcome of a successful small business's slow and steady expansion.

CHAPTER TWO

LITERATURE REVIEW

This chapter sums up all the nitty-gritties of the data warehousing solution in view ranging from the history to facts tables and variety of dimension tables.

2.1 Definition of data warehouse

A data warehouse is a tool that compiles data from multiple sources so that it can be juxtaposed and analyzed for business intelligence. A data warehouse must be subject-oriented, time-variant, integrated and non-volatile (Almeida, 2017)

Subject oriented: Data in warehouses is structured around various topics like transactions, clients, and goods. This is in contrast to transactional structures, which arrange data according to business processes like order entry, inventory management, and accounts receivable. Although service-oriented architectures SOA, will modularize the services and simultaneously provide the possibility of integrating them to execute a business process (Rojas, Arias, & Renteria, 2020)

Integrated: Data is obtained from several sources and grouped around topics. Customer data, for example, maybe derived from internal (and external) databases and combined around a customer identifier to provide a holistic view of the customer.

Time variant. A warehouse maintains historical data (i.e., it includes time as a variable). Unlike transactional systems, where only recent data, such as for the last day, week, or month, are maintained, a warehouse may store years of data. Historical data is needed to detect deviations, trends, and long-term relationships.

Nonvolatile. A warehouse is nonvolatile – users cannot replace or update the data. Non-volatility ensures that all users are working with the same data. The warehouse is updated, but through automated controlled load processes rather than by users. As stated in (Watson, 2002) Because of the first characteristic (Subject Oriented), Warehousing can be

designed to suit different scales of businesses of which this project will be focusing on a small scale.

2.1.1 History of data warehouse

Data warehouses provide availability of data that may originate from different sources. Although, the idea of the data warehouse is not novel it dates as far back as the 1980s when Paul Murphy and Barry Devlin worked on something they called “business data warehouse”. The aim of data warehouses has always been to ensure better decision support systems.

The history of the data warehouse is protracted. In the 1960s General Mills at Dartmouth College started joint research to develop the terms dimensions and facts. As time went on, in the 1970s A.C. Nielsen and IRI developed multi-dimensional data marts for retail sales and Bill Inmon started to define the term “data warehouse”. In 1988, Barry Devlin and Paul Murphy published an article called "business data warehouse". In 1990, Ralph Kimball introduced Red Brick Warehouse software specifically for data warehouse, in 1991 Bill Inmon, introduced Prism Warehouse Manager, a software for developing a data warehouse. Ever since data warehouse development has been a subtle competitive market.

2.2 Impact of Data Warehouses

A lot of businesses benefit from data warehouses in numerous ways which include improvement of decision making, growth management and proper organization of historical data. Asides from the aforementioned benefits data warehouses seem to be behind the chain of events that leads to data analysis and data security in most enterprises.

According to (Golfarelli & Rizzi, 2009), Data warehousing has proven to be very useful in the following areas:

Trade: Data warehouses help to organize sales and claim analysis, customer relationship and shipment & inventory control.

Craftmanship: Data warehouses manage production cost control along with demand and supply chain.

Transportation industry: Vehicle scheduling and management.

Health care services: Patient admission and discharge analysis, as well as bookkeeping in accounts departments, are all part of the health care services.

Financial services: Risk analysis and fraud detection in financial services

Telecommunication services: Data warehouses track client profiles and do call flow analysis.

2.3 Objectives of Proposed Data Warehouse

The proposed enterprise data warehouse aims to integrate data from multiple heterogeneous sources of data from businesses, this might contain information from a POS machine, data from billing software, and unprocessed data from excel sheets for businesses.

However, for the effective functioning of the enterprise data warehouse, the data warehouse must have the following objectives:

Outline of Data Collection Model using MS Data Collector: Specification of the data collection model is necessary to help guide a continuous data control flow. The Data Collector is a SQL Server component that gathers several types of data. Data gathering can be done on an ongoing basis or according to a user-defined schedule. The data collector stores the collected data in a relational database known as the management data warehouse. (Microsoft, 2021)

Integration of Data from Different Sources with SSIS: SSIS is a component of the Microsoft SQL server that is used to perform data integration and transformation tasks. This will create a control flow diagram to outline the data integration process. (Microsoft, 2021)

Creation of Fact and Dimension Tables: Fact and dimension tables are created for the relational data. A couple of Structured query language SQL queries will be used to secure this. (Edward M. Leonard, 2011)

Training of Data Warehouse end-users: Most end-users have to familiarize themselves with the data warehouse.

Continuous Review of the Data Warehouse System: Training of the data warehouse users and administrators is very cogent because the data warehouse artefact is created in a way that it interacts with users through complex queries. Therefore, the DWH has to be updated so that more queries can be executed on it.

2.4 Inmon vs Kimball Data Warehouse:

Bill Inmon and Ralph Kimball have both contributed to data warehouse methods and evaluation of both data warehouse types is necessary. Bill Inmon's approach to the data warehouse is called the enterprise data warehouse and Ralph Kimball's approach to the data warehouse is called the dimensional data warehouse.

TABLE 2.1: INMON VS KIMBALL DWH ARCHITECTURE

Architecture	Kimball	Inmon
Model	Used dimensional model e.g., star schema, snowflake schema to organize data in the data warehouse. The dimensional model is used for all of the data	Used entity-relation ER model for data warehouse and dimensional model for data marts
Data Separation	In Kimball's Architecture separation of the dimensional data warehouse from data mart is not needed	Used data marts as separation from the EDW
Analytic Systems	In Kimball's Architecture analytic systems access data directly	In Inmon's architecture, analytic systems can only access data via data marts

2.5 Data Warehousing and Data Warehouse.

Data warehousing includes a mechanism for end-users to gain access to data collected and stored which may include analysis and a data warehouse is simply datasets stored in form of fact tables, aggregated fact tables and lookup fact tables and an accumulation of data that facilitate the process of decision making of the management. (Qureshi & Jamal, 2012)

2.6 Fact and Dimension Tables

Every data warehouse has fact and dimension tables that describe the data warehouse.

2.6.1 Fact tables

A fact table usually has two types of columns: the foreign key which would be the primary key of the dimension tables and the measures that contain numeric values most of the time. A fact table typically contains data in an aggregated or detailed level. A fact-value can be additive, semi-additive or nonadditive. For example, in a fact table with sales items the profit can be said to be additive, the percentage gain or loss is semi-additive and the product name is nonadditive.

2.6.2 Dimension tables

The dimension table is usually structured with one or more hierarchies that classify data. In cases where the dimension has no hierarchies and levels, it is called a flat dimension or a list. The primary key of dimension tables is part of the foreign keys of fact tables. Dimension tables are smaller in size than the fact tables. There are different types of dimensions but the most determinant type is the slowly changing dimension.

Slowly Changing Dimensions (SCD): Dimensions that vary extremely slowly over time rather than according to the regular timetable.

Below are some popular ways of SCDs:

Type 0: This approach is passive. There are no changes to records that have been added.

Type 1: Without recording previous data, the old record is overwritten with fresh values.

Table 2.1 Slowly Changing Dimensions (Type 1)- Before Change.

Cust_ID	Cust_Name	City	State
1234	John James	Parkot	Myannar

Table 2.3: Slowly Changing Dimensions (Type 1)- After Change.

Cust_ID	Cust_Name	City	State
1234	John James	Parkot	Delhi

Type 2: This method uses a version number or the start date of a column to track historical data. The downside is that the table expands in height. Also, more space is required.

		Customer			
Cust_ID	Cust_Name	City	State	Country	
1234	ABC	Albany	NY	US	
		Customer_History			
Cust_ID	Cust_Name	City	State	Country	CreateDate
1233	ABC	Rochester	NY	US	22nd Dec 2015
1234	ABC	Albany	NY	US	7th July 2013

Figure 2.1 Slowly Changing Dimensions (Type 2). (Nikhilesh, 2015)

Type 3: This approach is also used to track historical data by adding a new value to the column. It retains a historical limit.

Cust_ID	Cust_Name	Current_City	State	Country	EffectiveDate	Old_City
1234	ABC	Albany	NY	US	22nd Dec 2015	Rochester

Figure 2.2 Slowly Changing Dimensions (Type 3). (Nikhilesh, 2015)

Type 4: This technique utilizes a separate table to store the previous data and holds the current data. Type 4: And they will both point to the same substitute key.

Customer

Cust_ID	Cust_Name	City	State	Country
1234	ABC	Albany	NY	US

Customer_History

Cust_ID	Cust_Name	City	State	Country	CreateDate
1233	ABC	Rochester	NY	US	22nd Dec 2015
1234	ABC	Albany	NY	US	7th July 2013

Figure 2.3 Slowly Changing Dimensions (Type 4). (Nikhilesh, 2015)

2.7 Schema Types in Data Warehouse Design

There are basically, three schema types used in data warehouse design. Star schema, Snowflake schema and constellation schema. But before that, the principle of fact and dimension tables must be clarified.

2.7.1 The star schema

In a star schema, all dimension tables are connected directly with a fact table. Hence the Dimension tables are highly denormalized in a star schema. In a star schema, only one join is required to get data from fact and dimension tables. The main advantage of star schema is that it saves space and it processes queries faster because of fewer joins.

2.7.2 The snowflake schema

Some dimension tables in a snowflake schema are not directly related to the fact table; instead, they are connected to another dimension table, which is connected to either a fact or another dimension table. Snowflake schema means that there would be more joins. The major tradeoffs of the snowflake schema are that the DWH will take slightly more space and query processing might be slower because of the many joins.

2.7.3 The constellation schema

It is feasible to build a factual constellation scheme for any star scheme (for example by splitting the original star schema into more star schemas, each of them describes facts on another level of dimension hierarchies). There are numerous fact tables in the constellation architecture which share many dimensional tables.

The major weakness of the scheme is a more difficult design as numerous versions must be examined and selected for certain types of aggregation. In addition, tables of dimension remain big.

2.8 Data warehouse types.

Data marts: collect the data from the data warehouse and send it to the OLAP layer. Now data is ready to be used by end-users as an accessible tool. A data mart is a subcategory of the data warehouse. It is specially designed for a particular line of the bus.

Operational Data Store: The OPD, which is also called the ODS, is nothing more than but data store when reporting needs are not supported by the Data warehouse or by OLTP systems. Data storage in ODS is updated in real-time. For regular operations such as keeping employee records, it is therefore often favoured.

Enterprise Data Warehouse (EDW): is a centralized warehouse. It delivers decision support services across the organization. It gives the organization and representation of data a unified approach. It also offers the ability to categorize and grant access to data according to the subject.

2.9 Information Systems.

Information systems are formal, socio-technical, organizational information systems (IS) aimed at collecting, processing, storing and distributing information. From a socio-technical point of view, information systems consist of four components: task, persons, structure, and technology. Information systems may be described as the integration of data components that offer information, store and process data, contribute to knowledge and digital products that aid decision-making. Information systems can also include data collecting. Information systems are classified into several categories, including transaction processing systems, decision support systems, knowledge management systems, learning management systems, database management systems, and office information systems. Most information systems rely on information technologies, which have traditionally been designed to enable humans to perform tasks for which the human brain is not well suited, such as handling large amounts of data, performing complex calculations, and controlling multiple processes at once.

2.10 Review of Related Works.

(Santoso & Yulia, 2017), Data Warehouse with Big Data Technology for Higher Education was a research work made to tackle the lack of efficiency of big data technology to handle the load and analytic process of data. With the use-case of a university system where vast amounts of data are collected and is growing significantly. Hadoop was proposed as a big data analytics tool for data staging and ingestion. The research came to a conclusion outlining future directions relating to the development and implementation of data warehouse technology on Big Data.

In 2011 Edward Leonard of Marquette university presented a thesis called ‘design and implementation of an enterprise data warehouse’ for higher institutions in which he

implemented an enterprise data warehouse that showed how sets of different databases go through the process of Extract, transform and load (ETL) and loaded into a star schema. The limitation of this system is that it does not offer a chance to share reports against an existing system and the DWH was not structured in a way that other institutions can use.

(Qureshi & Jamal, 2012), Taxonomy based Data Marts depicted an approach that added a new layer for categorization of queries using data warehouse technology and quickly responds to multidimensional analytic queries. The algorithm in this journal article proved to be more accurate and time-saving which was possible because of the query classification technique.

(Almeida, 2017), Concepts and Fundaments of Data Warehousing and OLAP, discussed the fundamental concepts of data warehouses and explores the concepts associated with data

warehousing and analytical information analysis using OLAP. The fundamental concepts of data warehouses and explores the concepts associated with data warehousing and analytical information analysis using OLAP.

In 2016, Hughes, R. made research for Agile Data Warehousing for the Small Enterprise. The work contained the building of different types of data warehouses using different agile methods. It was difficult to find limitations in this work because of the versatile approaches employed. Hughes concluded in his work that: The combination of XP and SCRUM could be used in the development of a data warehouse; RUP could be used in place of XP and SCRUM.

In a research work performed by Abhisek Gupta and Arun Sahayadhas (2020) a few punchy attributes were explained including Slowly Changing Dimension types, Schemas, ETL and

ELT architectures and MD5 (Multi-Dimensional) function. The survey established that there are still many open future aspects of research in the same area.

CHAPTER THREE
METHODOLOGY

This chapter explains the methodology for the design and development of a data warehouse, the tools essential to build a user-developed solution and all needs

3.1 Requirement Gathering

The data warehouse is subject oriented meaning it must be modelled after the business process. Matter of fact this entire process is called IDENTIFICATION OF BUSINESS PROCESS in data warehousing. Identification of business process was done by an interview with organization stakeholders and by first-hand observation of the data integrated from the existing data sources. The data warehouse should provide a generic structure for SSEs and MSEs by applying the business process in sales information gathering.

3.2 Architecture of The Designed Data Warehouse System

3.2.1 System of records (SOR)

The system of record is an information storage and retrieval mechanism in a system with numerous sources of the same element, the authoritative source for a certain data element. In the case of this proposed system, it most likely contains spreadsheets, data from point-of-sale machines, and unprocessed data. The System of Integration receives each data source choice.

3.2.2 System of integration (SOI)

The system of integration consolidates the datasets from different sources into a unified schema in other words the imperative part of data warehousing is done in the system of integration.

3.2.3 System of analysis (SOA)

The availability of analysis options is done by the system of analytics. The whole process is conducted by the OLAP Server with the help of the data marts so the necessary queries can be executed on the data warehouse.

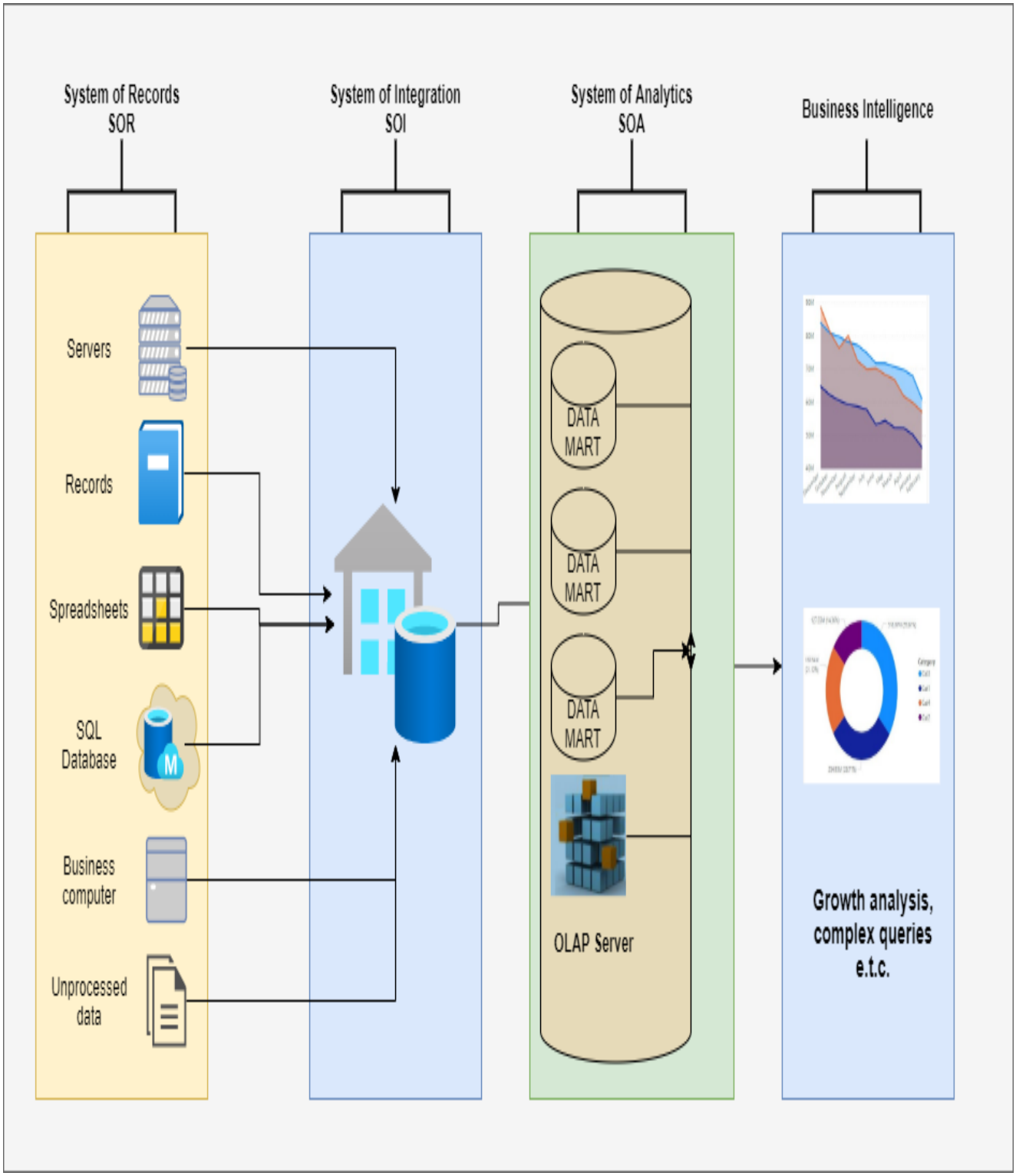

Figure 3.1: Data Warehousing System Architecture

3.2.4 Business intelligence (BI)

Business intelligence is the preferred front-end tool for decision support. It directly interacts with the OLAP server in the System of Analysis.

3.3 INSTALLATION OF SOFTWARE

Microsoft SQL Server 2019 was used for writing queries to design schemas and create joins, Microsoft Excel 2019 was used to create random values for the data warehouse which were later exported to the SQL database, Microsoft Visual Studio Community 2019 hosted the SSIS package where integration projects were performed, Microsoft Power BI 2019 was used for reporting data that was on the data warehouse and Microsoft SQL Server Integration Services Package was used to integrate the data warehouse. The development of this work was carried out on a DELL LATITUDE E6440 with the following hardware specification;

- 500GB Hard Disk
- 8 Gigabytes RAM (Random Access Memory)
- 2.6 Gigahertz Processor Speed (Intel Core i5)
- 64-bit Operating System (Windows 10)

The Installation process was done step by step.

CHAPTER FOUR

IMPLEMENTATION OF DESIGNED SYSTEM

This chapter shows the implementation of the designed system. For implementation, multiple data sources were reviewed.

4.1 Implementation Environment

The implementation for the data warehousing was done using five software: Microsoft SQL Server 2019, Microsoft Excel 2019, Microsoft Visual Studio Community 2019, Microsoft Power BI 2019 and Microsoft SQL Server Integration Services Package.

Microsoft SQL Server 2019 is a Microsoft-designed relational database management system. As a database server, it is a software product that mainly has the role of storing and retrieving data on the same or other computer via a network, as required by other software programs.

Microsoft Excel 2019 is a spreadsheet developed by Microsoft. It includes computing, graphic design tools, pivot tables and a Visual Basic for applications macro programming language. For those systems, in particular since version 5 in 1993, it has been an extremely often used spreadsheet, and Lotus 1-2-3 has been supplanted as an industry standard. Excel is part of the software package of Microsoft Office.

Microsoft Visual Studio 2019 is a Microsoft-based built-in development environment. It is used for the development of computer software, websites, web applications, and mobile applications. Visual Studio is using the Windows API, Windows Forms, Windows Presentations Foundation, Windows Store and Microsoft Silverlight software development platforms. It is capable of producing both native and managed code.

Microsoft Power BI 2019 is a Microsoft business analytics service. It seeks to offer a simple interface to allow interactive visualizations and business intelligence capabilities to construct their reports and dashboards for end-users

Microsoft SQL Server Integration Service is a component of the Microsoft SQL Server database program that may be utilized to carry out the various data migration activities is the SQL Server Integration Services. SSIS is a data integration and workflow platform. It has a data storage tool used to extract, process and load data. The program may be used also to automate SQL Server database maintenance and changes to multidimensional cube data.

4.3 Description of Developed System for Data Warehousing

The system for the data warehousing developed for small scale businesses is built after the architecture in FIGURE 3.1 Taking the prospective systems one after the other the implementation will be explained in the following subsections

4.3.1 Analysis of system of records

The implementation of the system of records was done with the Microsoft excel 2019 spreadsheet software where a certain comma separated file (csv) sales_datasample.csv which was sourced from Kaggle.io was chosen because the dataset has high similarity with small scale business operations and the dataset was analyzed. Some certain formulas were used to generate the values e.g.,

```
=RANDBETWEEN($A1, 50)

=RAND(()*(33-45)*15)
```

Other formulas were just used to manipulate the data sets and to make them structured e.g.,

```
=VLOOKUP(A13,SalesR2,2,FALSE)

=YEAR(G8)=$C$2

=SUM(M3:M503)

=IF(AND([@[Total
Sales]]>B6,[@Account]>=B7),[@[Total
```

The sales operations were analyzed to find out the measures so as to implement fact and dimension tables in the data mart. The CSV file is a data source that will provide the records and fields in the system of integration.

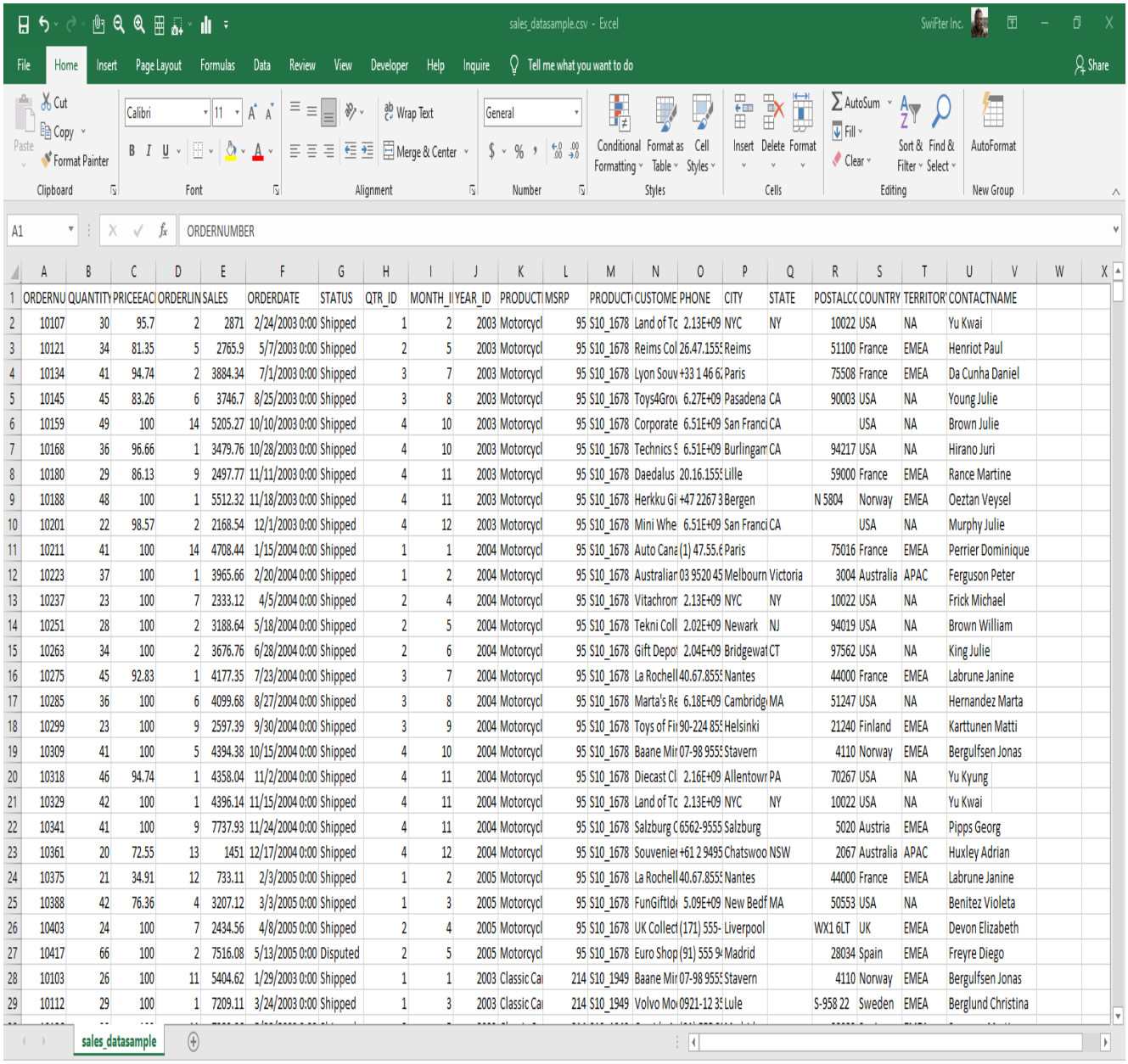

Figure 4.1: sales_datasample CSV data source (Example of System of Record)

4.3.2 Implementation of system of integration

The integration of the data warehouse was done with Microsoft SQL Server Management studios. A couple of SQL queries were used to create joins and relationships. In this module, the dimension and fact tables were created to match the aforementioned sales_datasample.csv records and fields a staging table to hold initials from the csv file was also created. The idea of a staging table is to hold data verbatim from the csv flat file. In the integration the data will be passed on from the csv file into the SQL server via SQL server integration service package.

4.2.3.1 Creation of tables in database:

A staging table was created with a series of complex queries, afterwards, the dimension tables were created with respect to the hierarchy of the schema. For instance, in a snowflake schema that has two or more dimensioin tables that are related to each other the dimension table that is farthest from the fact table will be created first. The hierarchy of table creation ensures consistency of the data warehouse. The creation and display of the table is outlined in FIGUREs 4.2 & 4.3

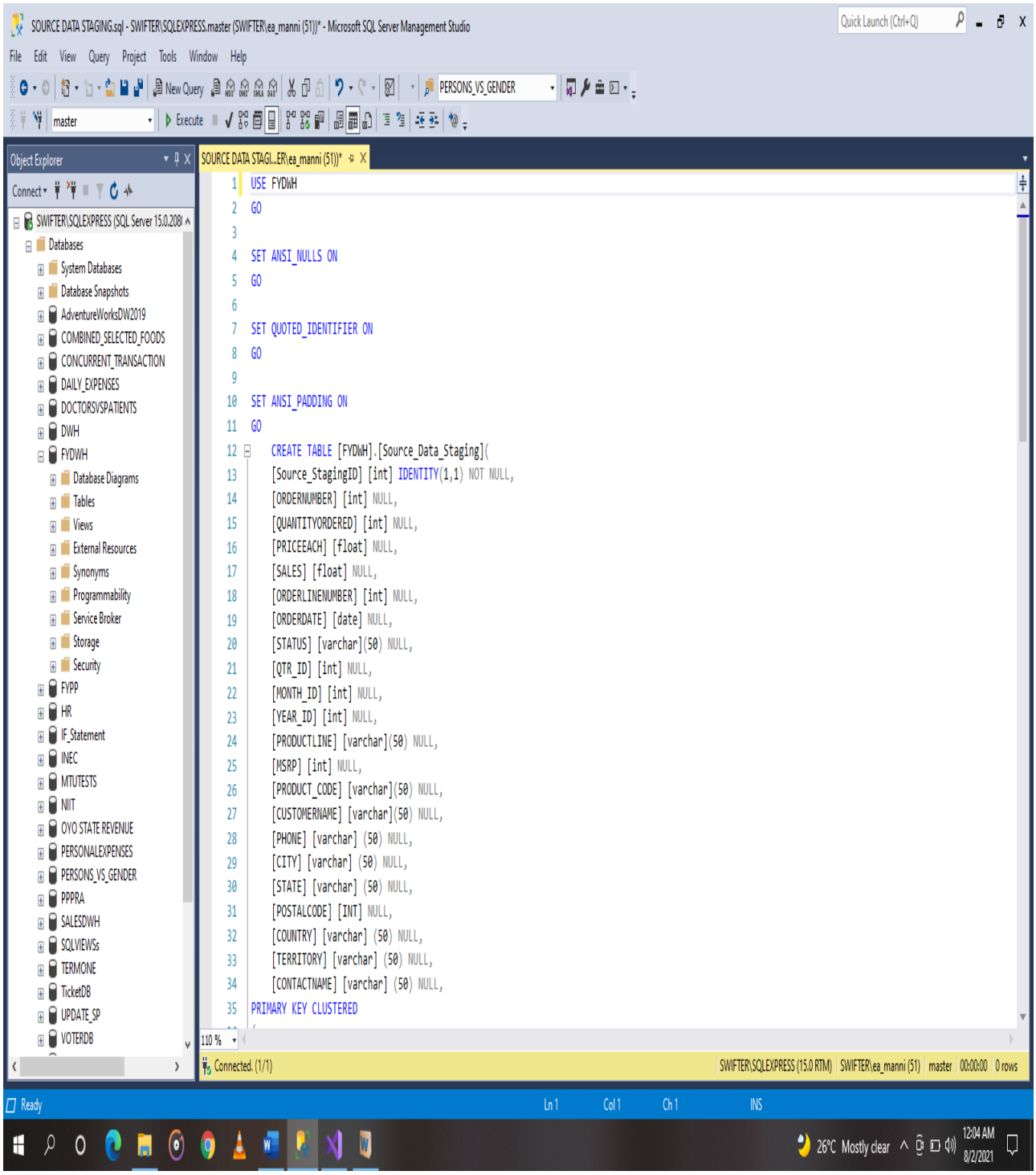

Figure 4.2: Creation of Staging Table

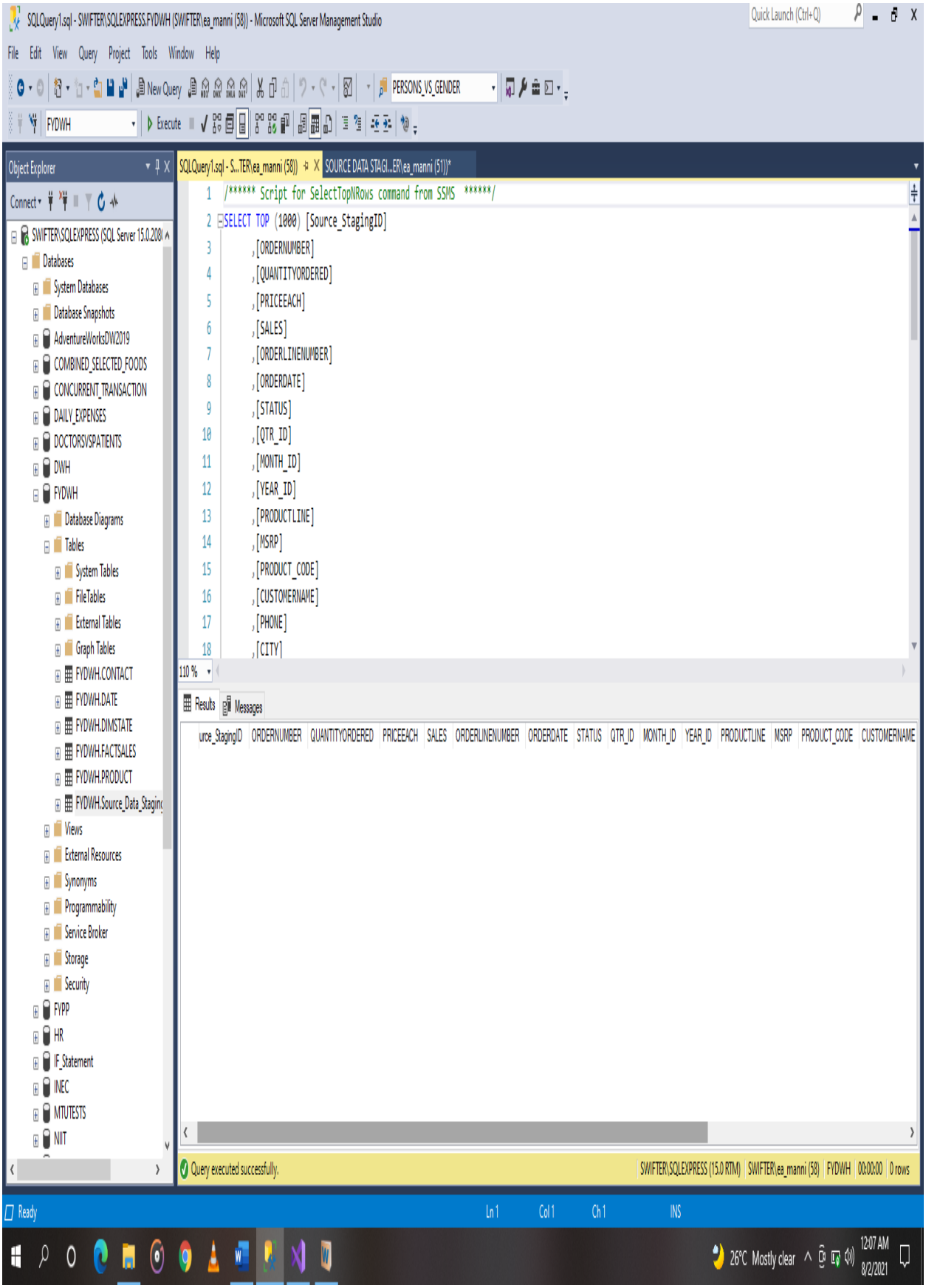

Figure 4.3: Viewing Staging Table

The records are absent from the staging table because the Source_Data_Staging table is yet to be populated.

A dimension table is created for date, product, address and contact details distinctively. The date table has the following columns: ORDERDATE, MONTH_ID, QTR_ID and YEAR_ID. The product table has the following columns: PRODUCTLINE, MSRP, PRODUCT_CODE. The state dimension table contains the following columns: CITY, STATE, POSTALCODE. The contact dimension contains CUSTOMERNAME, PHONE and CONTACTNAME columns. The creation of the tables were displayed in Figure 4.4,Figure 4.5, Figure 4.6, Figure 4.7.

A fact table is then created containing relationships with the other tables. The reason for the hierarchy of creation becomes clearer at this point because if you try to create a relationship with a table that does not exist yet a lot of error will follow the execution simply because of the current inexistence of those dimension tables. The fact table will have the primary keys of the dimension tables as the foreign key of the fact table. Therefore, a clean and clear relationship.

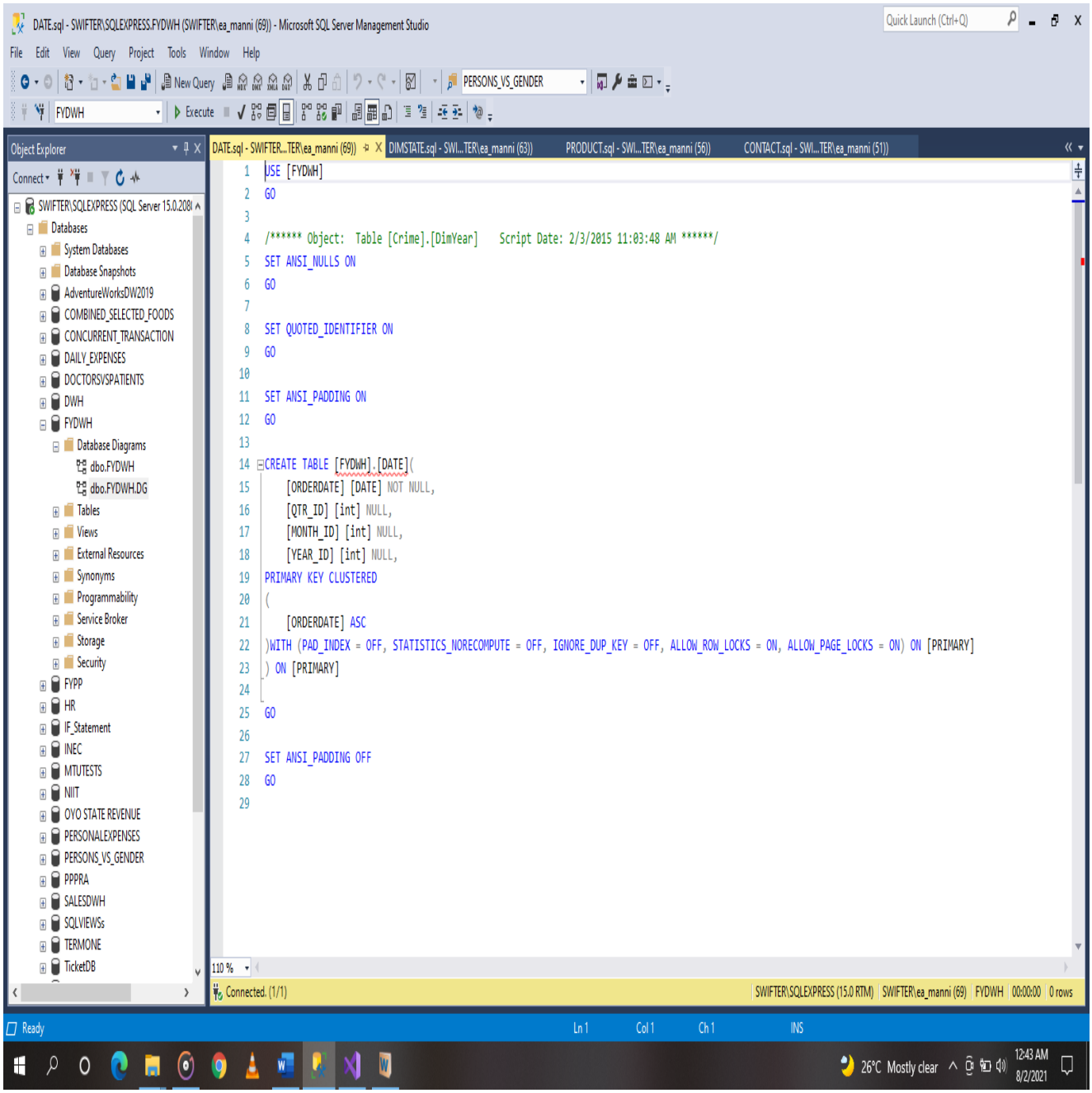

Figure 4.4 Creation of Date Dimension Table.

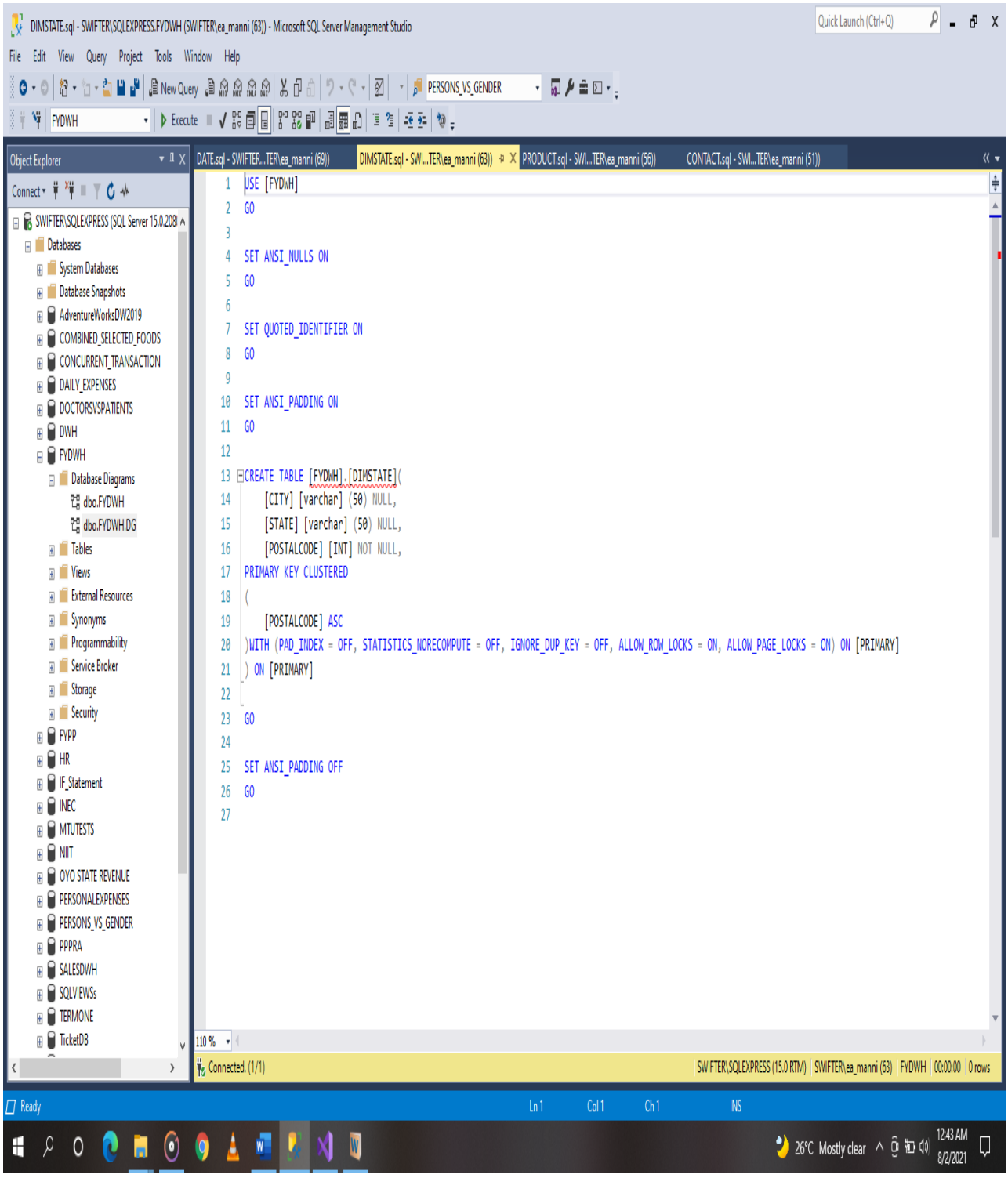

Figure 4.5: Creation of State Dimension Table

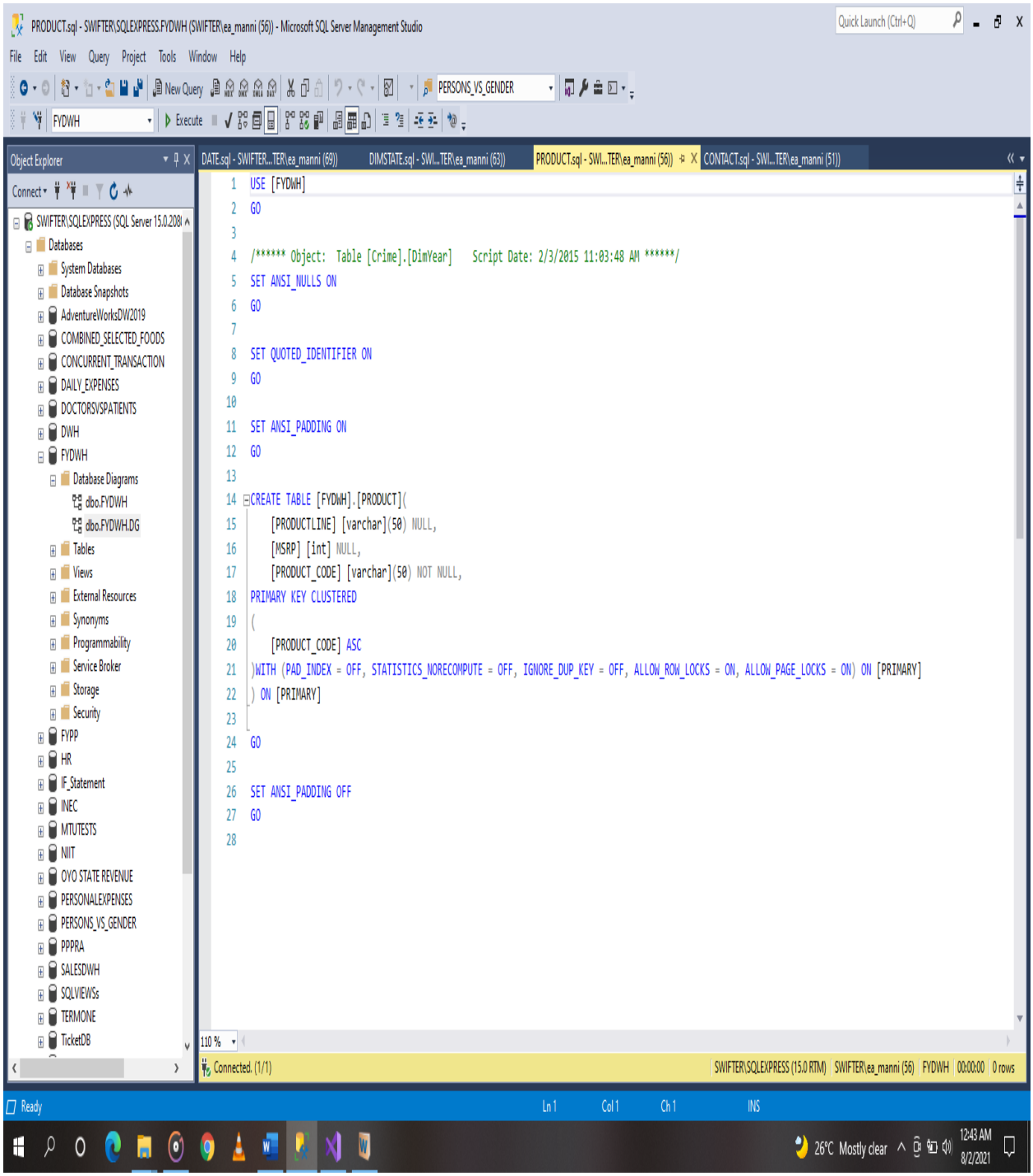

Figure 4.6: Creation of Product Dimension Table

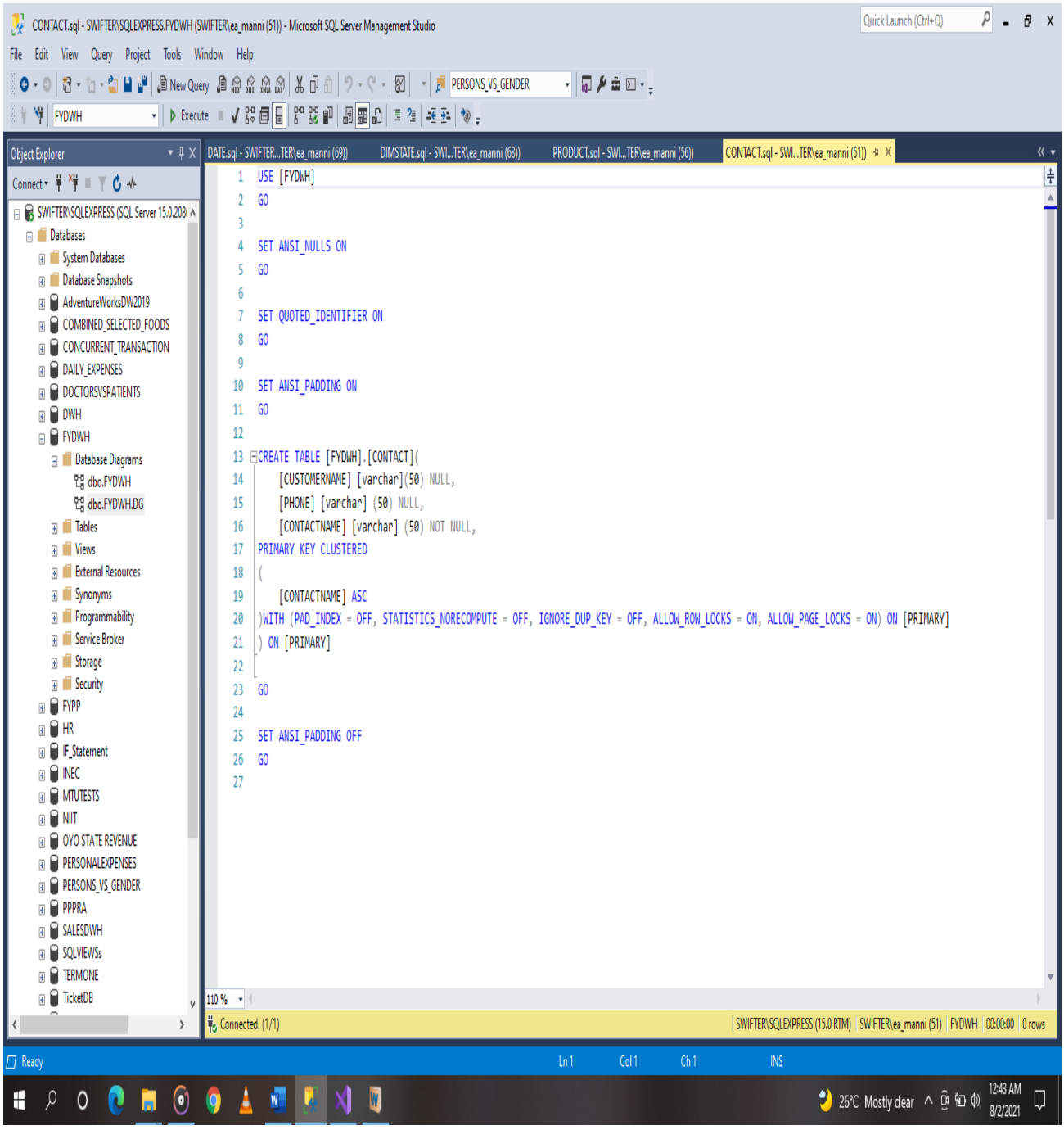

Figure 4.7: Creation of Contact Dimension Table

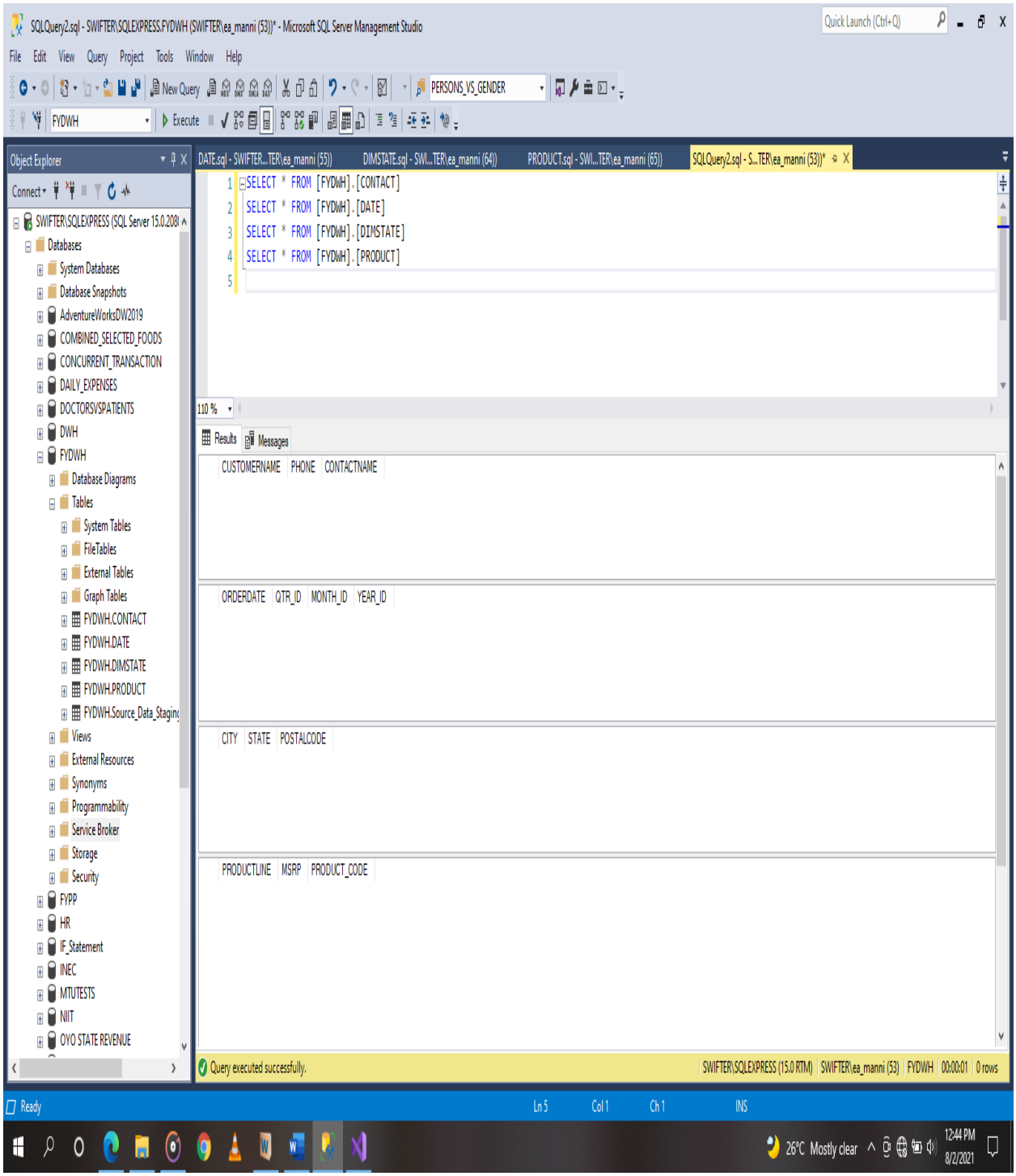

Figure 4.8 Display of Dimension Tables

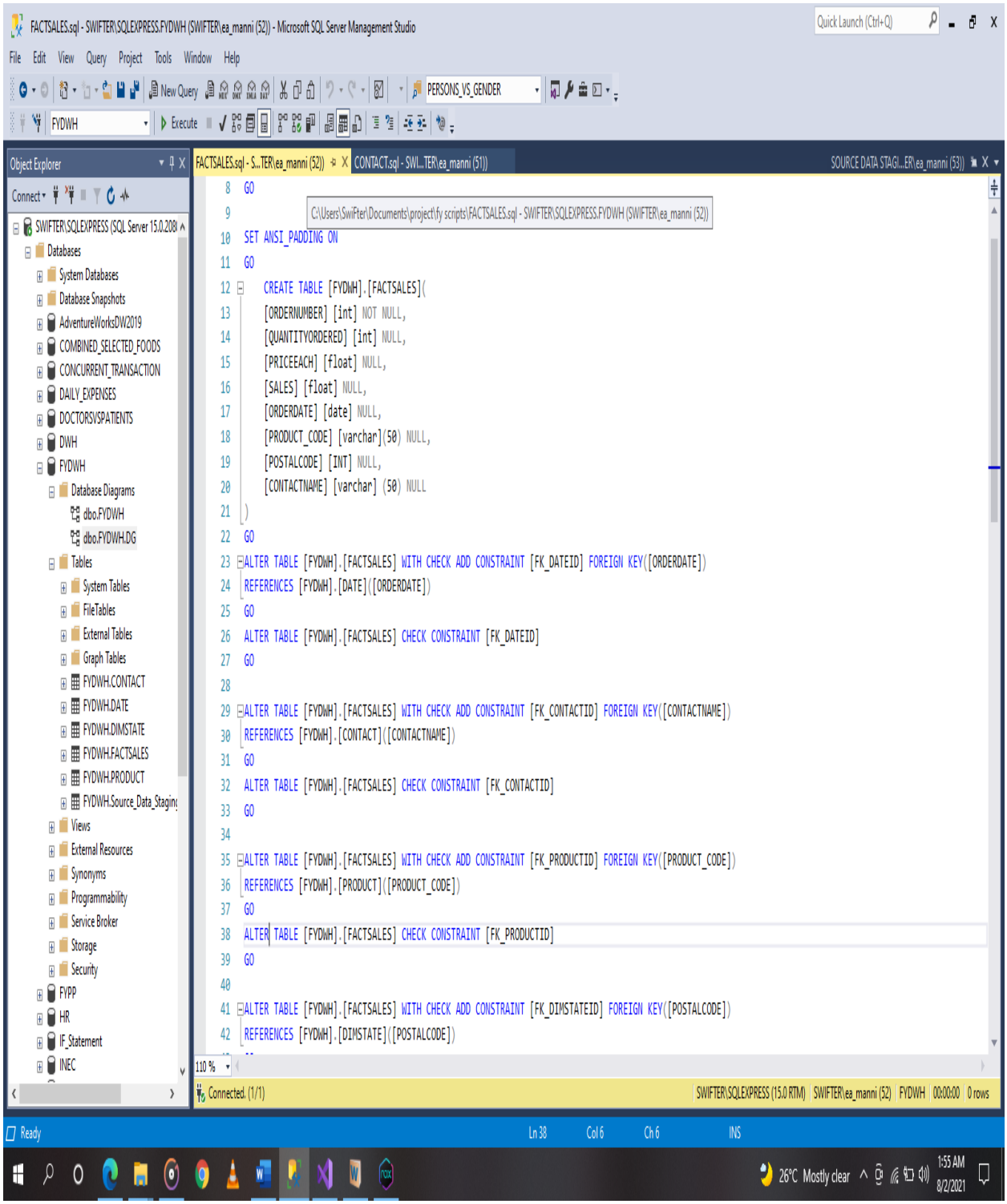

Figure 4.9: Creation of Fact Table and Relationship.

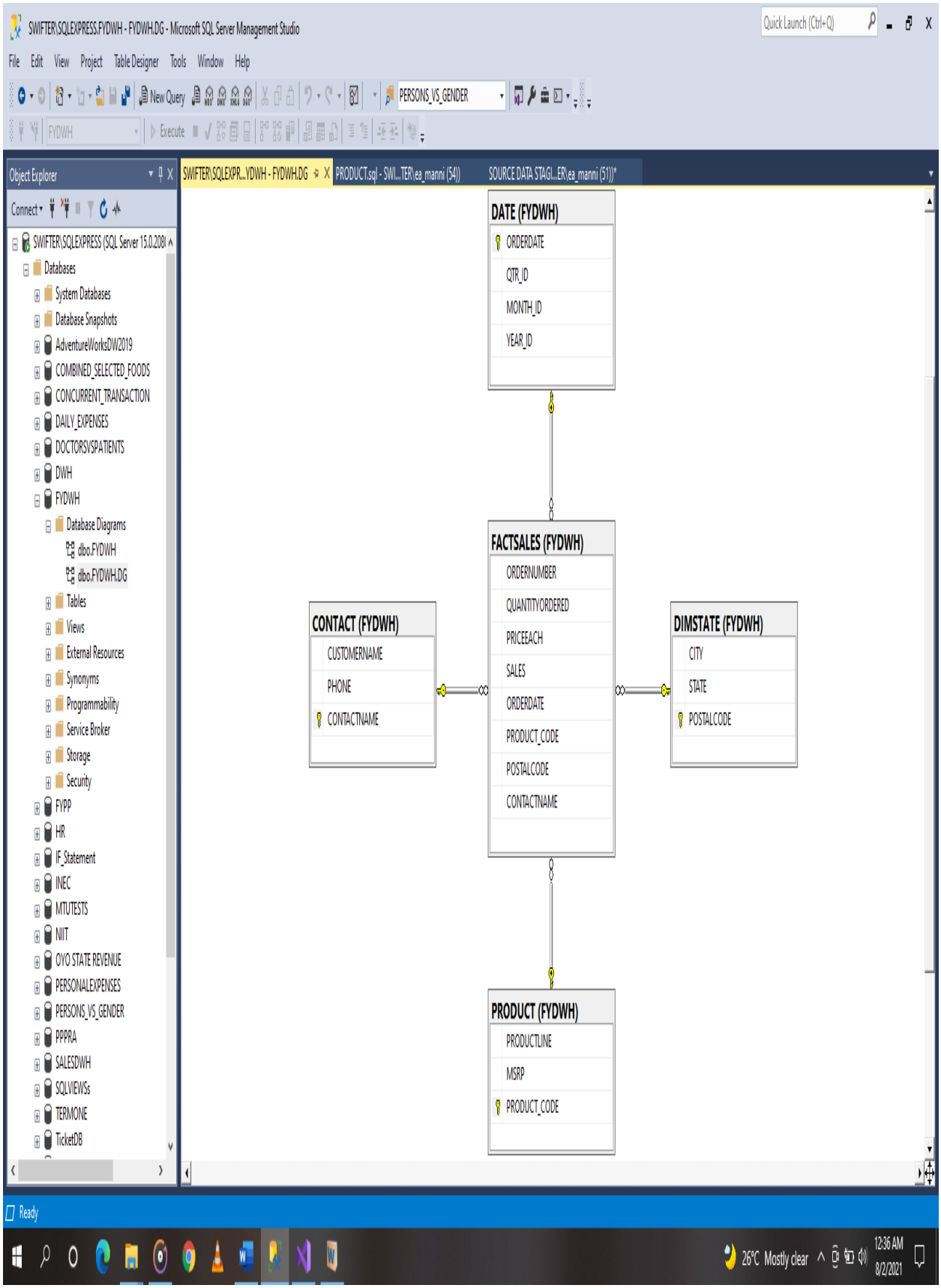

Figure 4.10: Diagram of Star Schema Implemented From Sales Dataset

4.2.3.2 Data population from SSIS:

Because each source table or file should have its own staging table in a warehouse, the staging table is loaded from the csv data source. The raw data is supplied by operational databases and external sources, as depicted in the diagram. Special logic monitors changes in data sources and generates periodic extracts of new data. Staging tables record the history of data entered from periodic extracts. This is accomplished through the use of a data flow job, which consists of data transitioning from a flat file source through a data conversion module, and finally into the SQL table destination. Following that, the Dimension tables are filled.

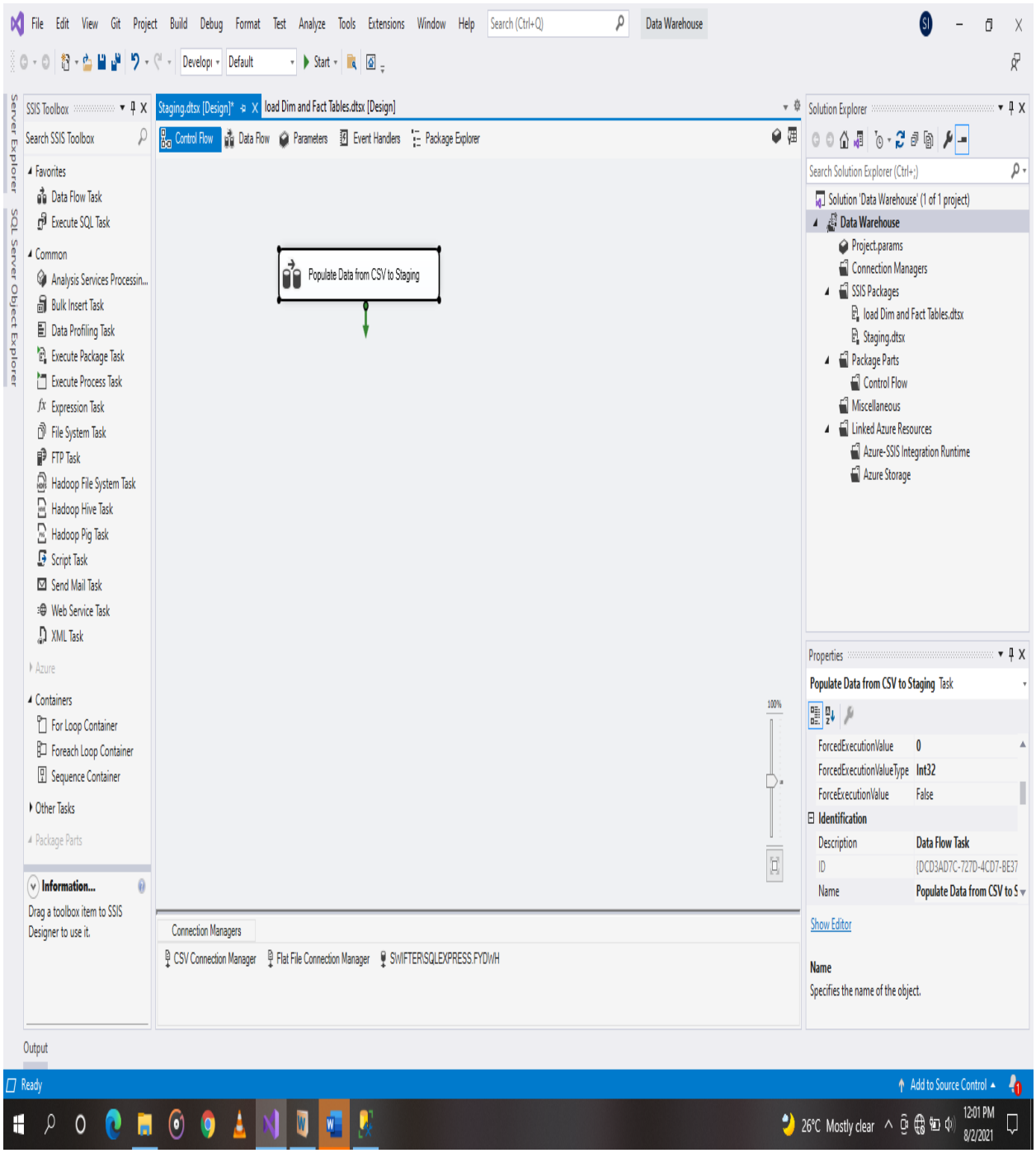

Figure 4.11: Data Flow Task to Populate Staging Table

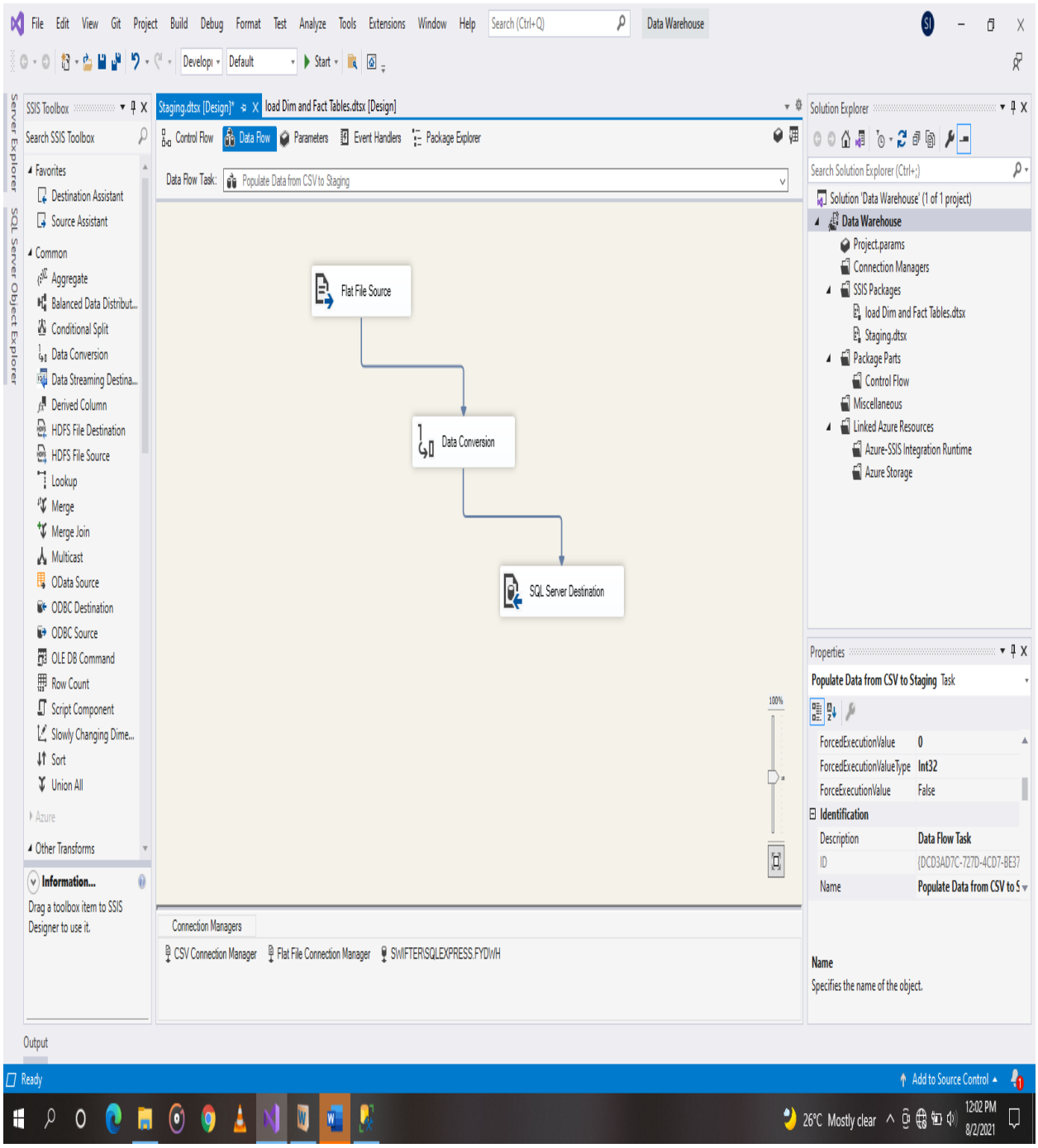

Figure 4.12 Module-to-module Sequence in Data Flow

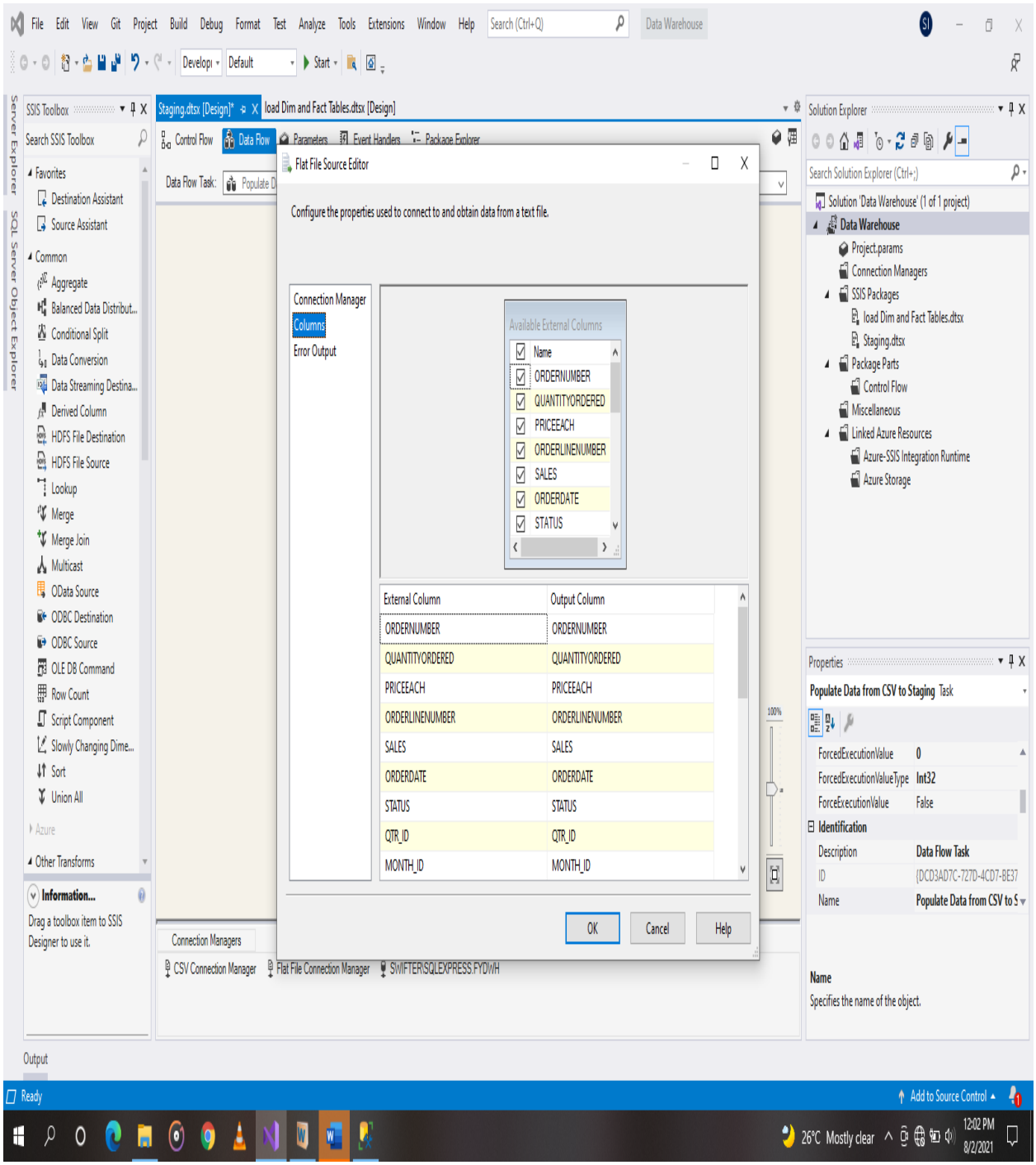

Figure 4.13: Flat File Configuration and Column Structure

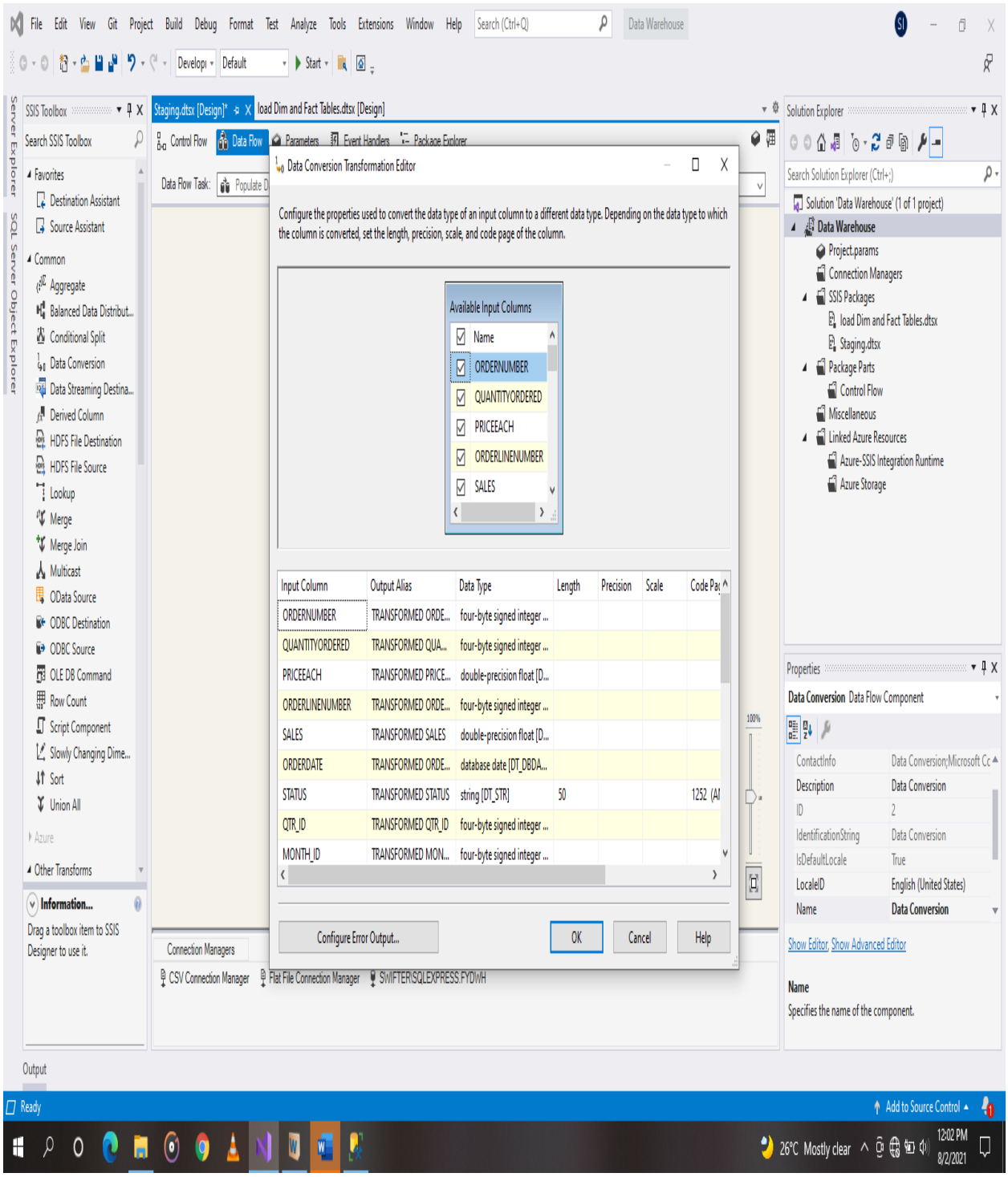

Figure 4.14: Data Conversion for Column Datatypes

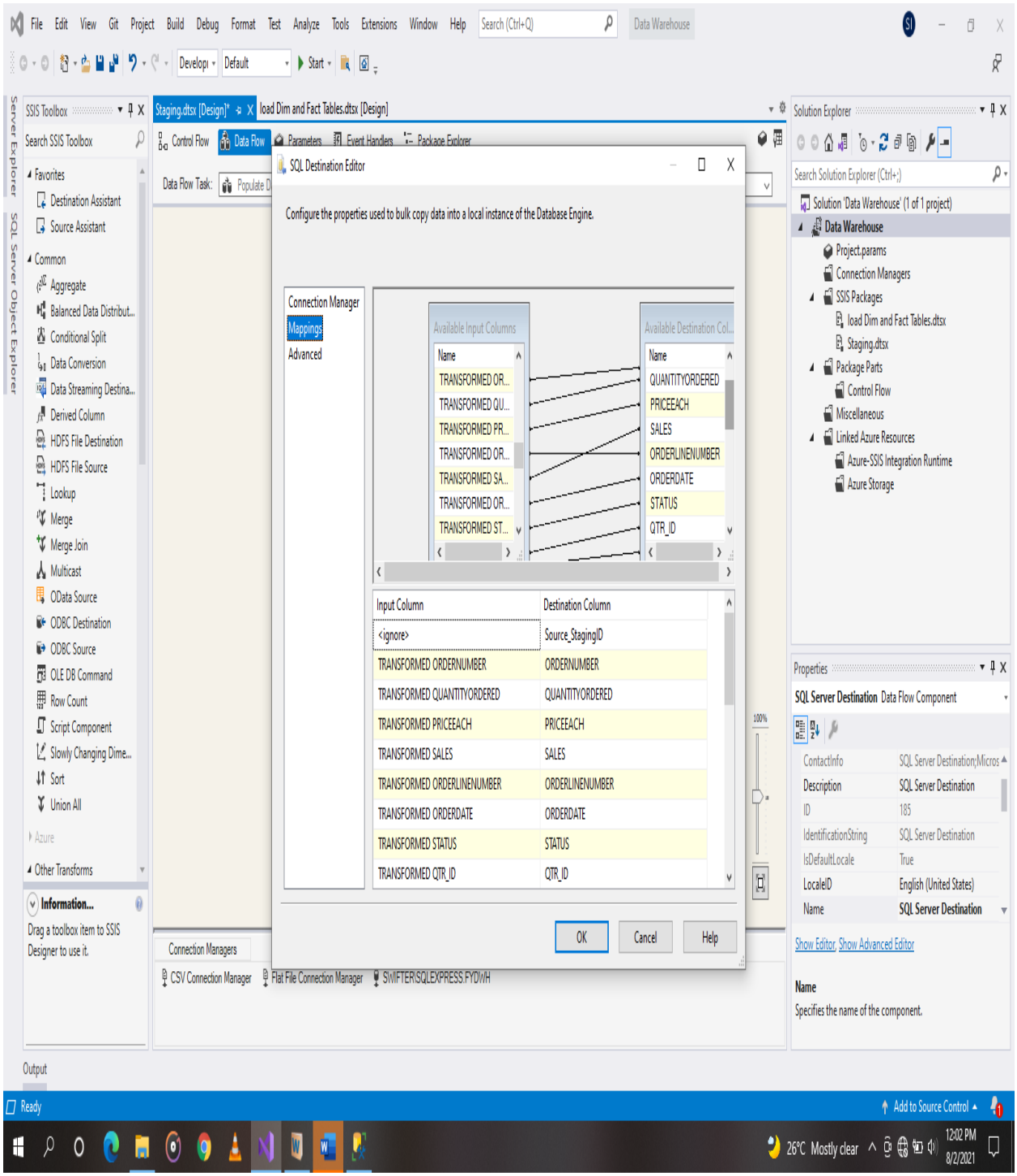

Figure 4.15: Column Mapping From Data Source to Destination.

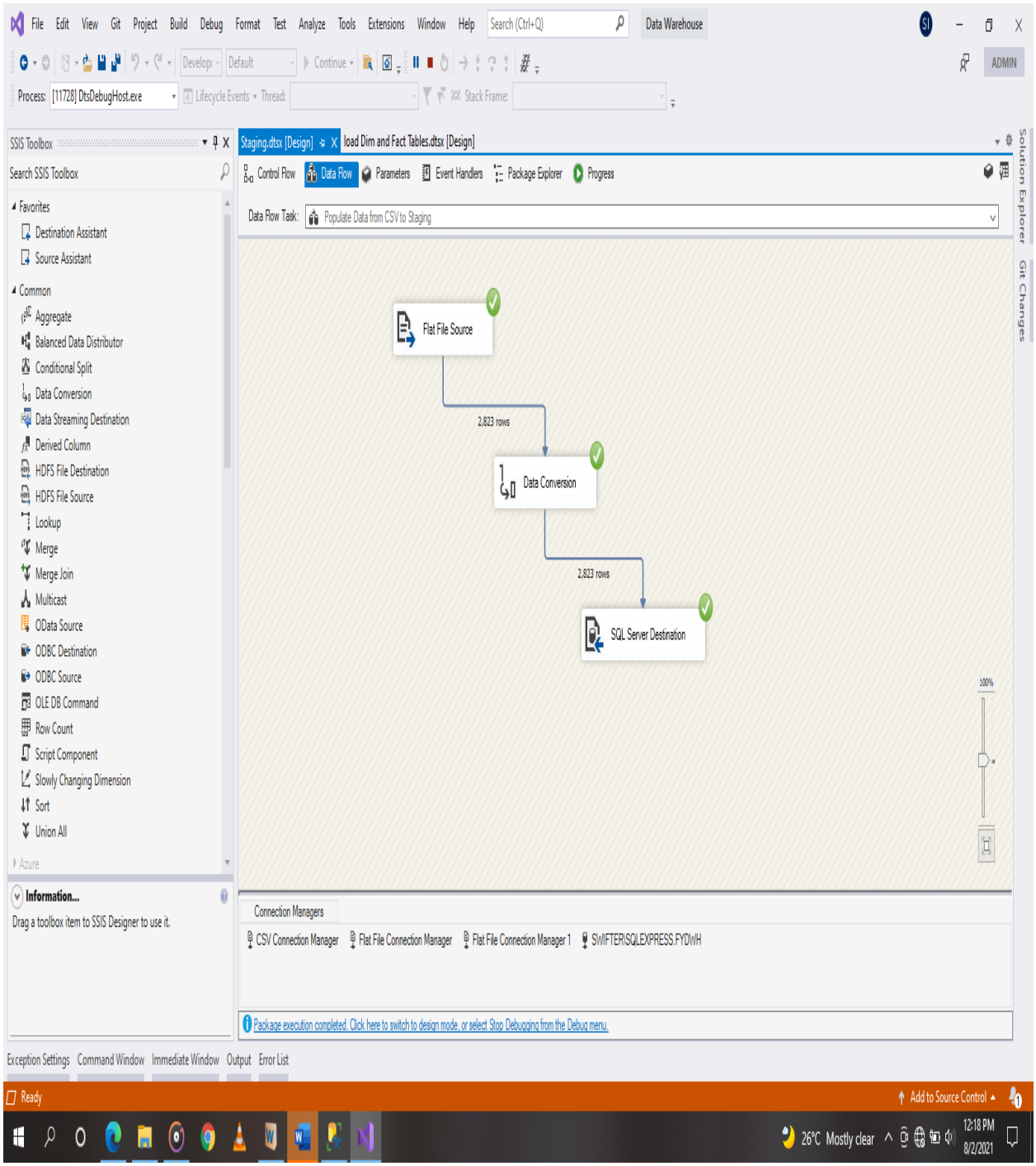

Figure 4.16: Successful Execution of Data Flow Task From CSV to SQL Staging Table.

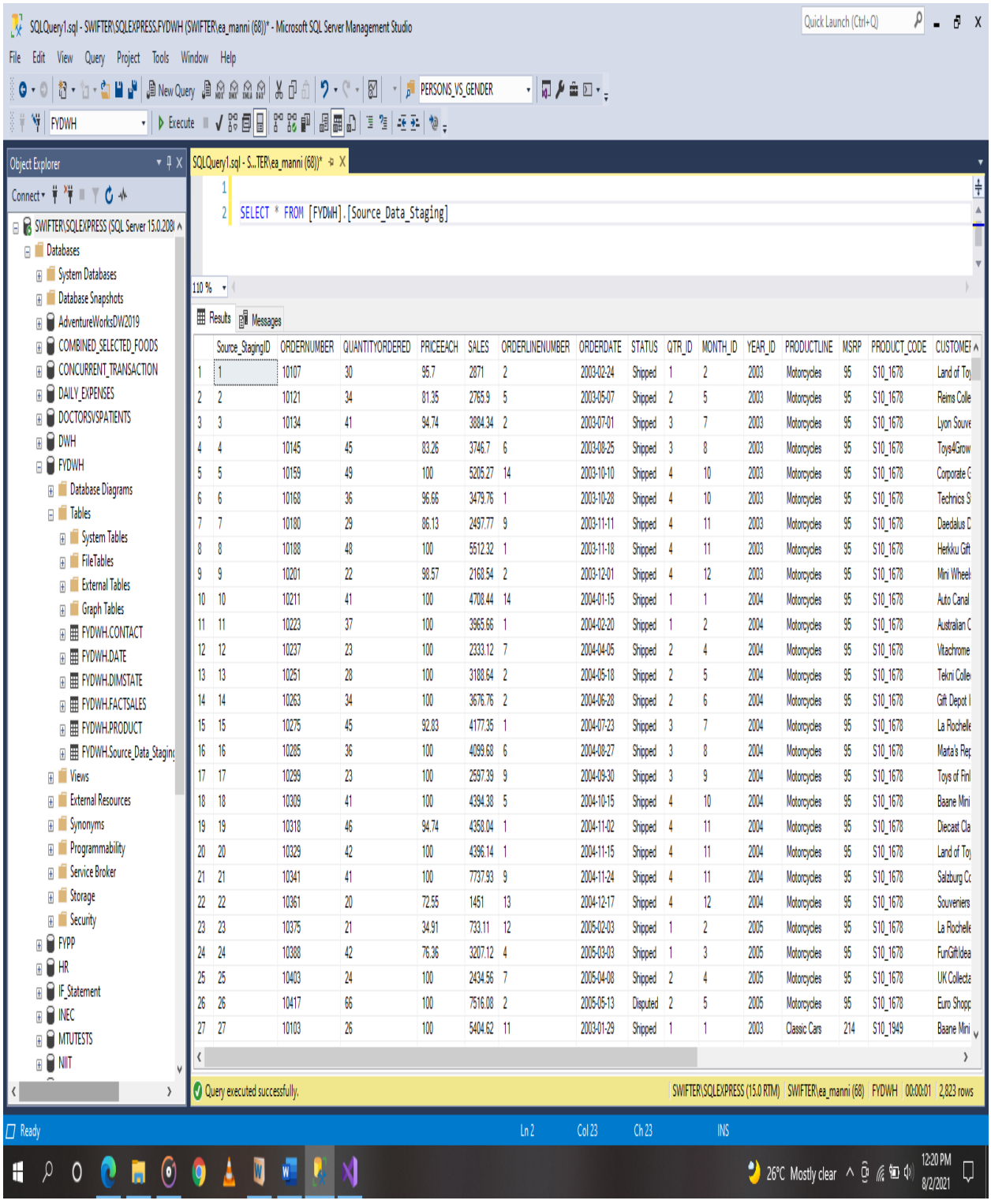

Figure 4.17: Selection of Populated Staging Table

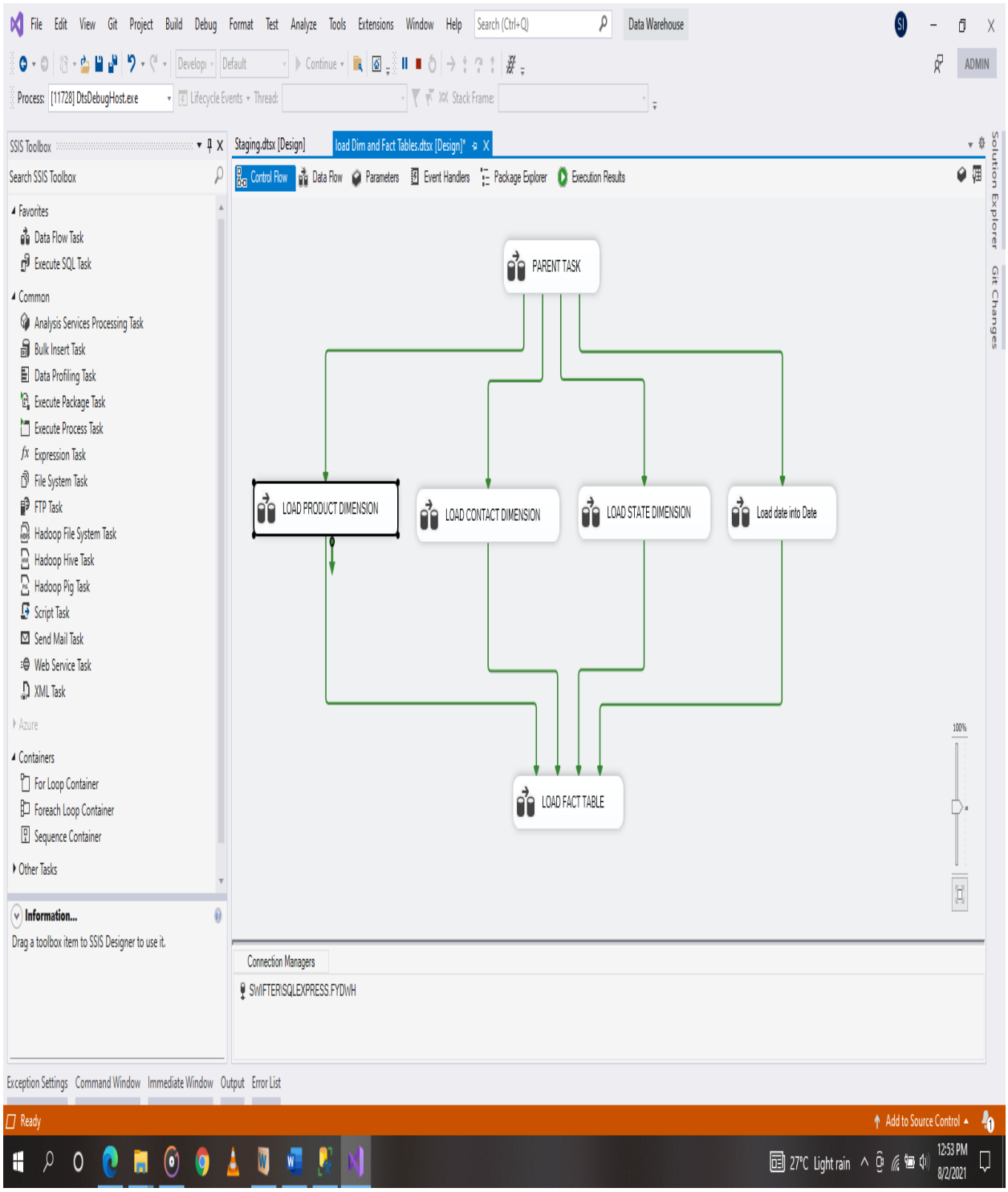

Figure 4.18: Various Data Flow Tasks for Populating Dimension Tables.

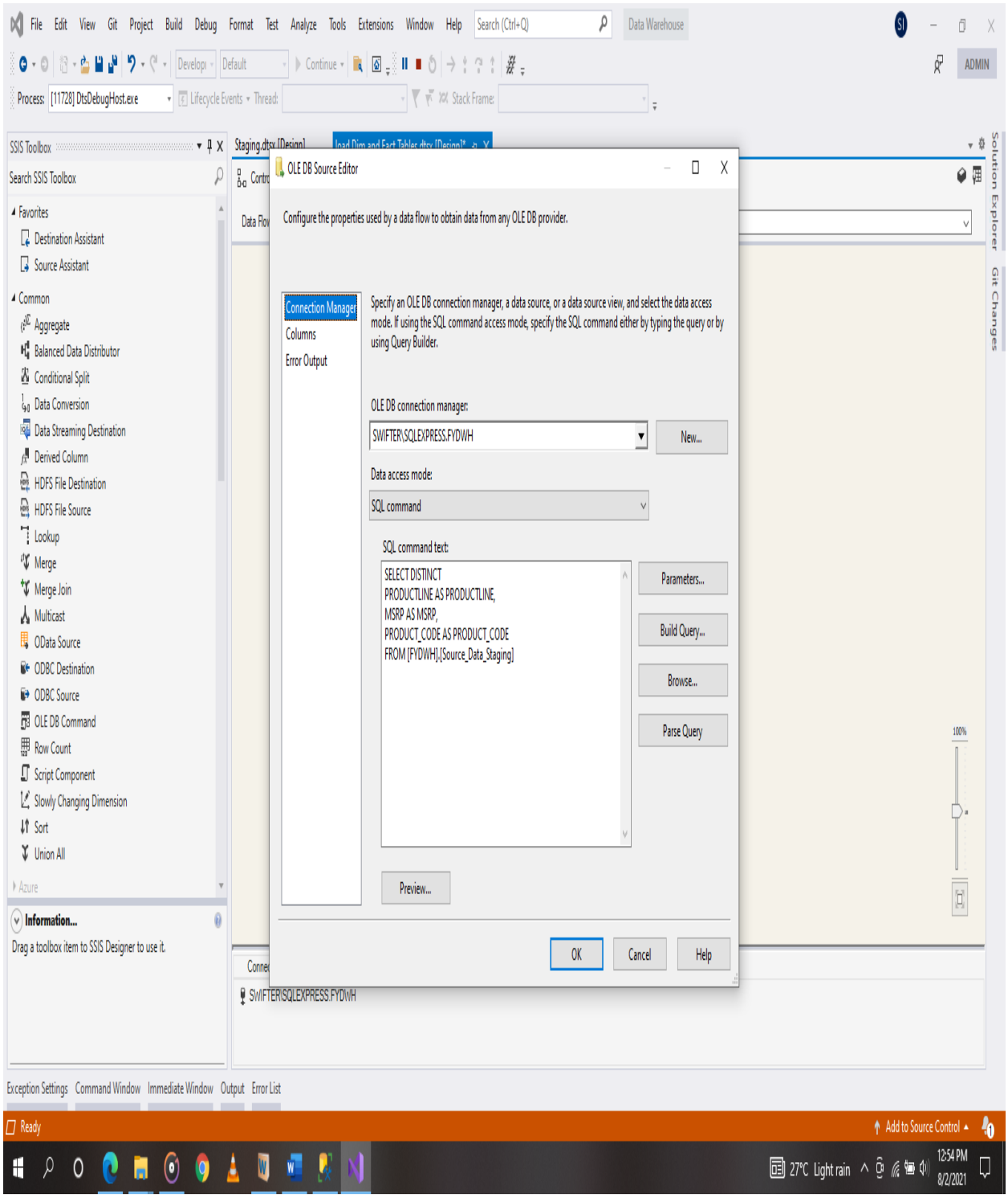

Figure 4.19: SQL Command Accessing for Product Dimension Table.

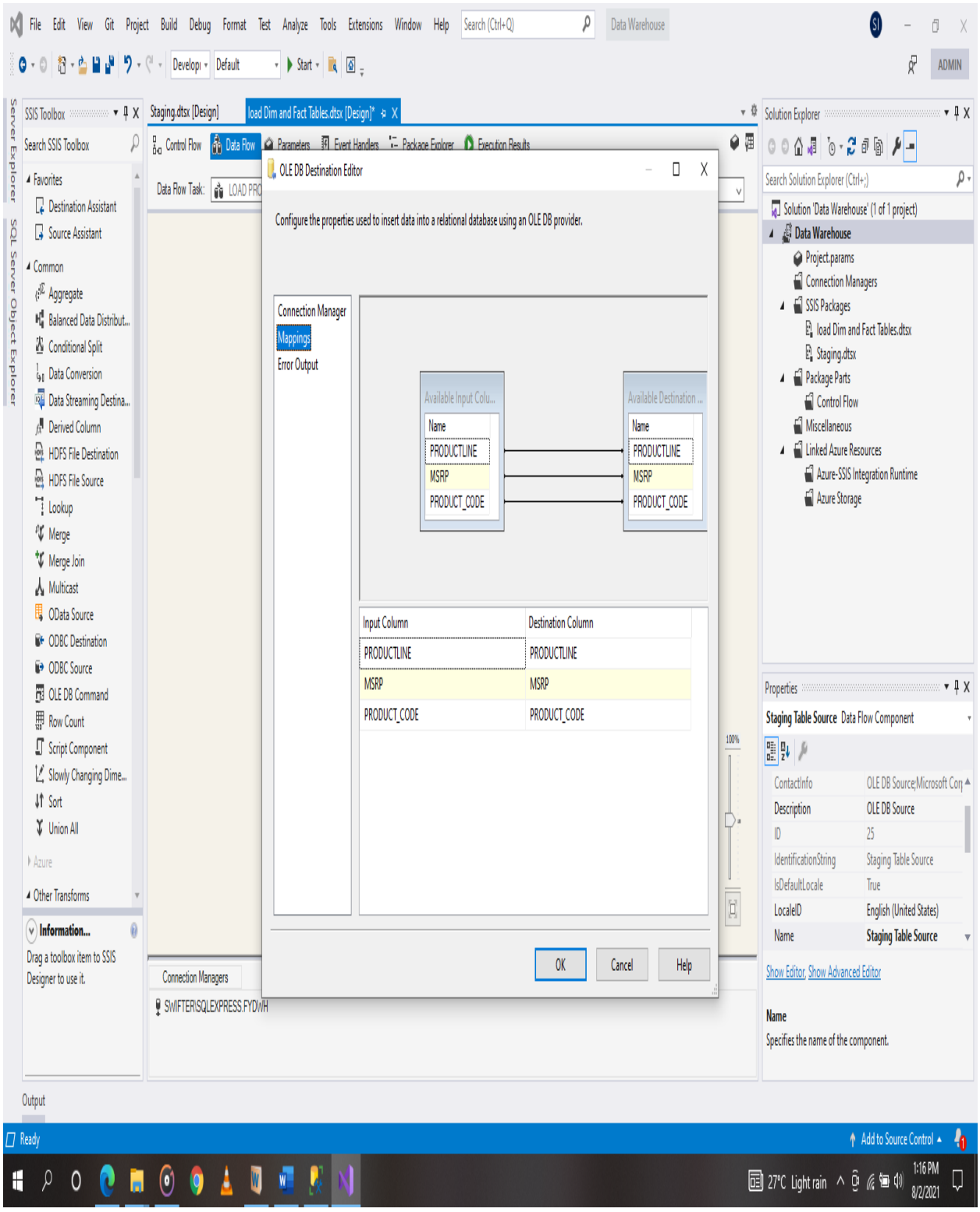

Figure 4.20: Column Mappings for Columns From Staging Tables into Product Dimension Table.

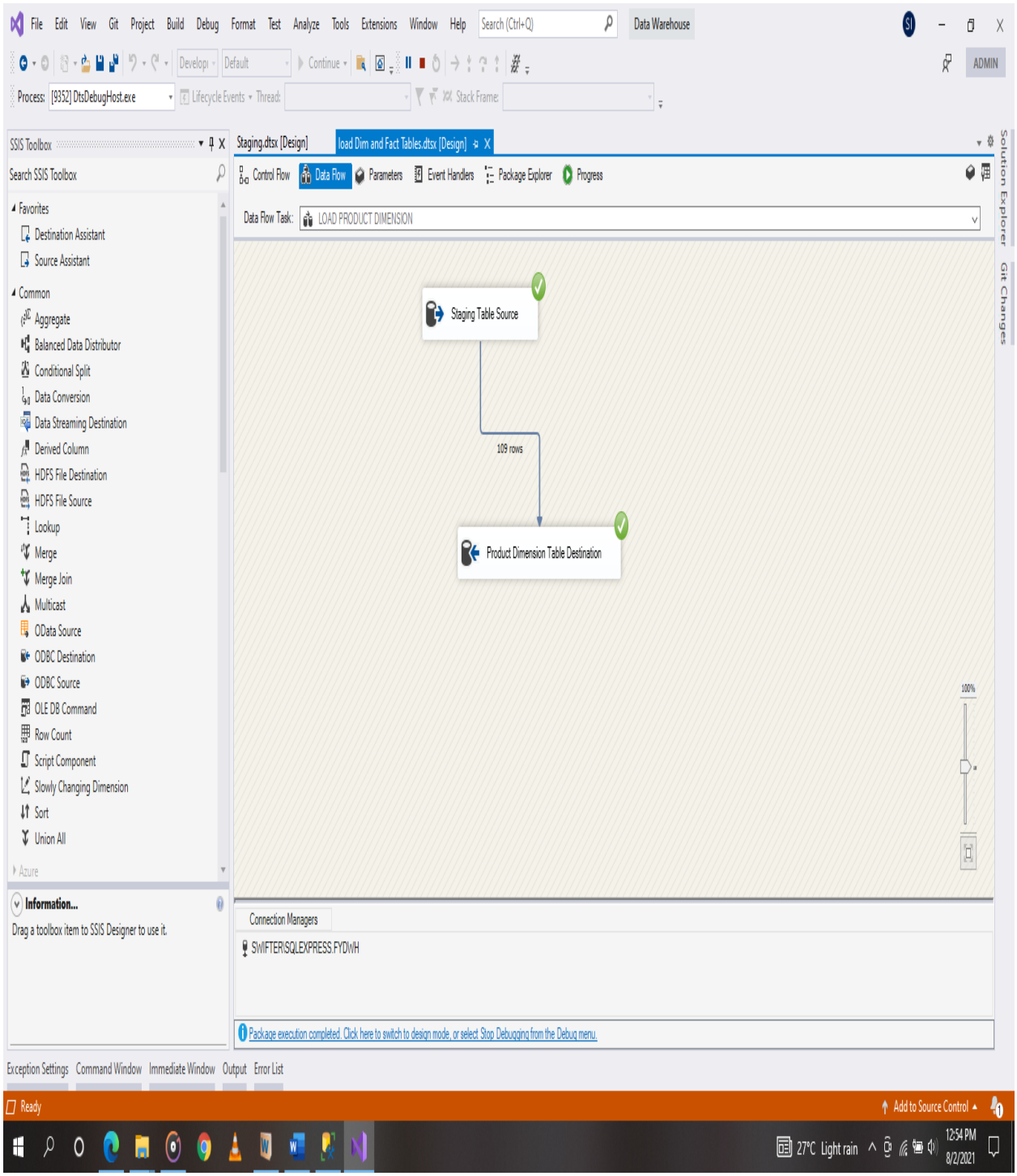

Figure 4.21: Successful Execution of Data Flow Tasks From Staging Table to Product Dimension Table

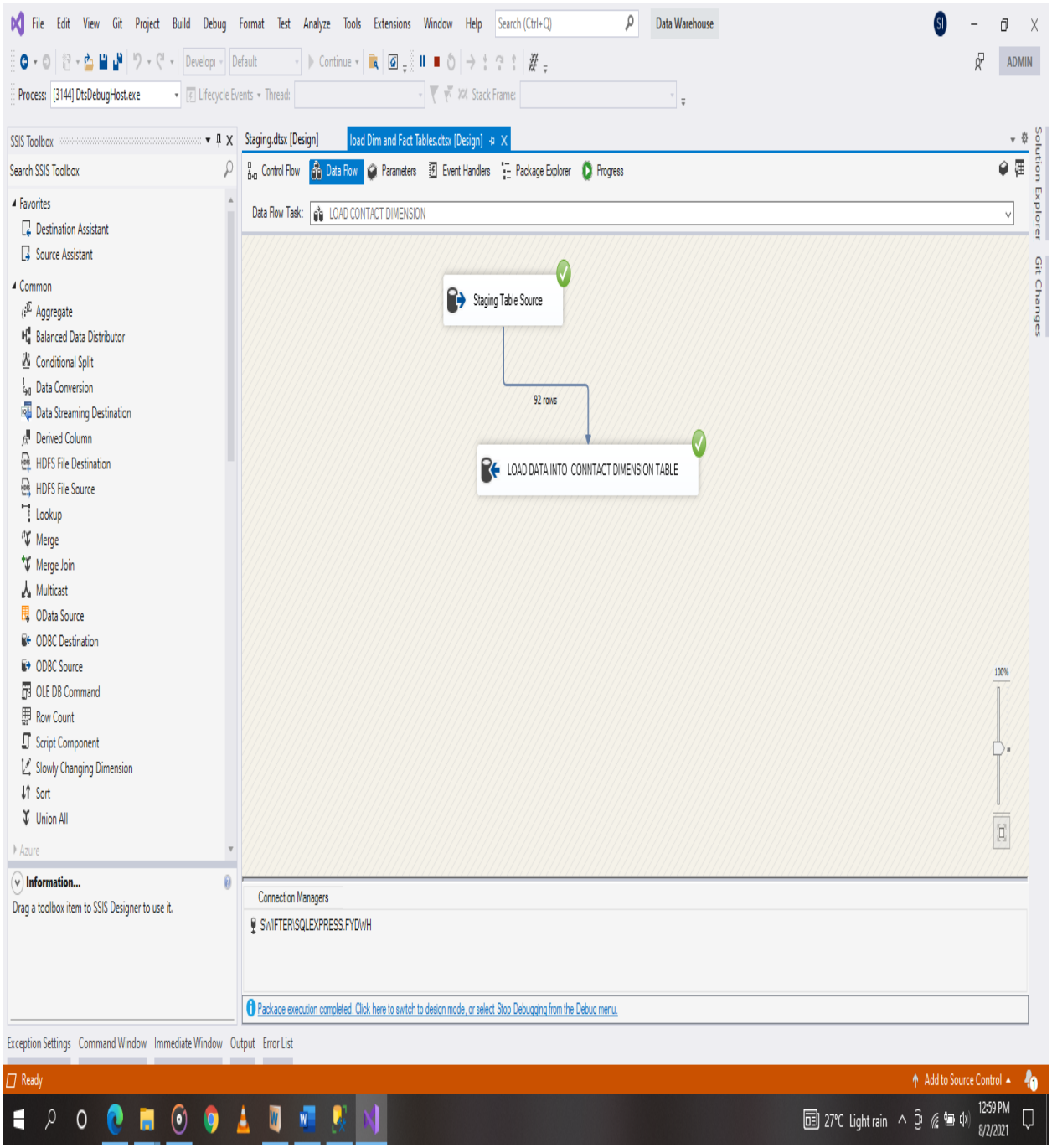

Figure 4.22: Successful Execution of Data Flow Tasks From Staging Table to Contact Dimension Table

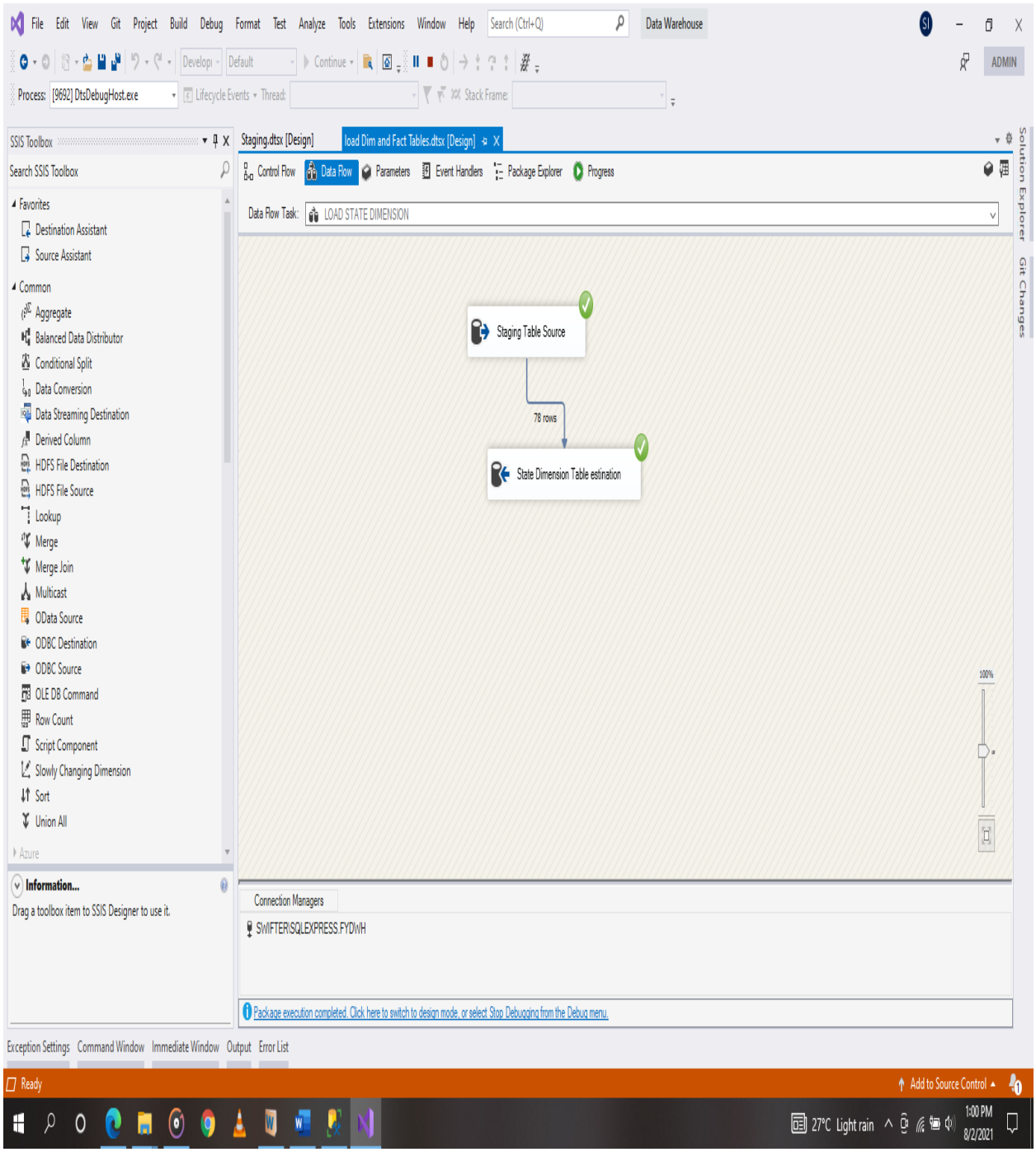

Figure 4.23: Successful Execution of Data Flow Tasks From Staging Table to StateDimension Table

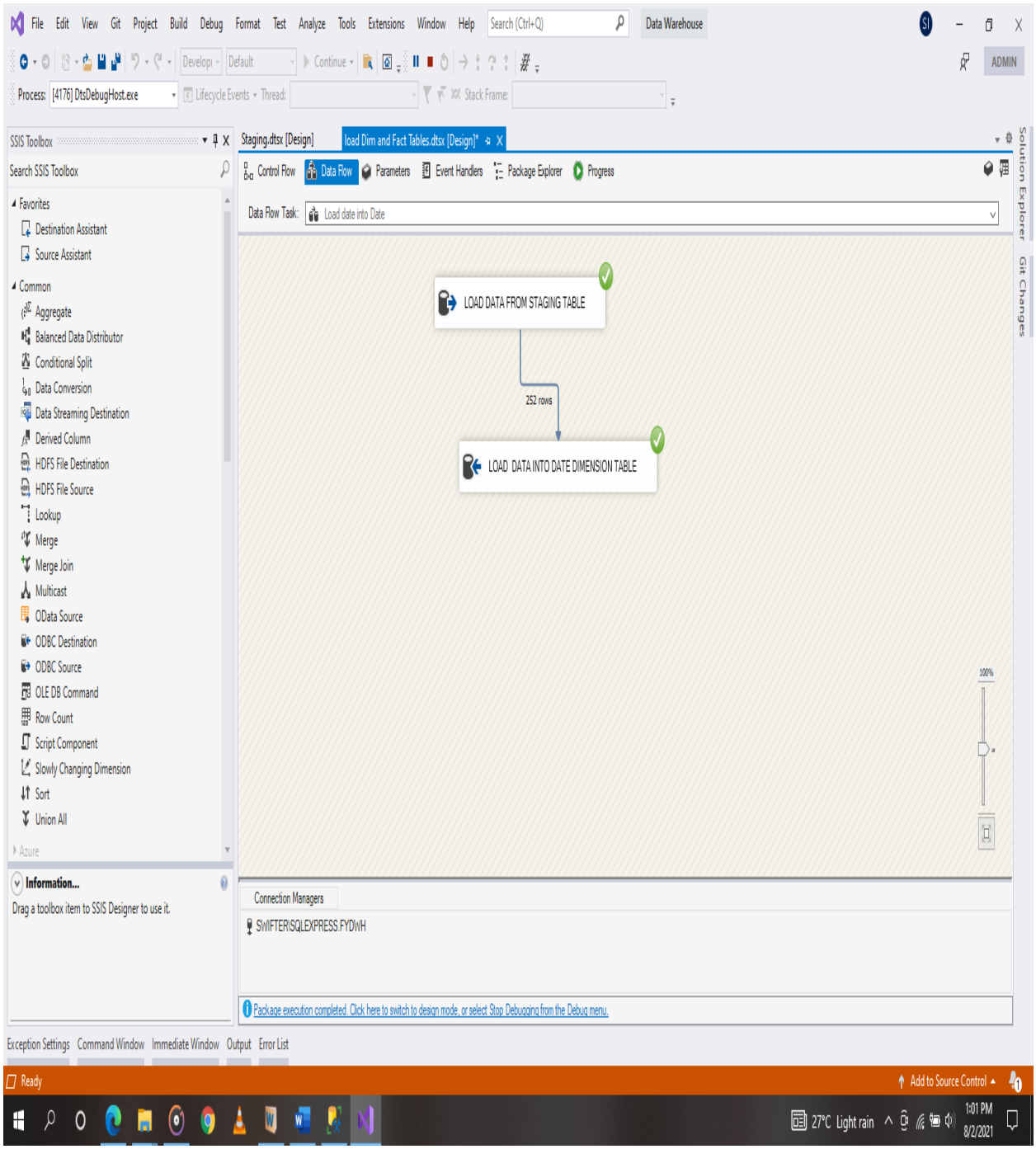

Figure 4.24: Successful Execution of Data Flow Tasks From Staging Table to Date Dimension Table.

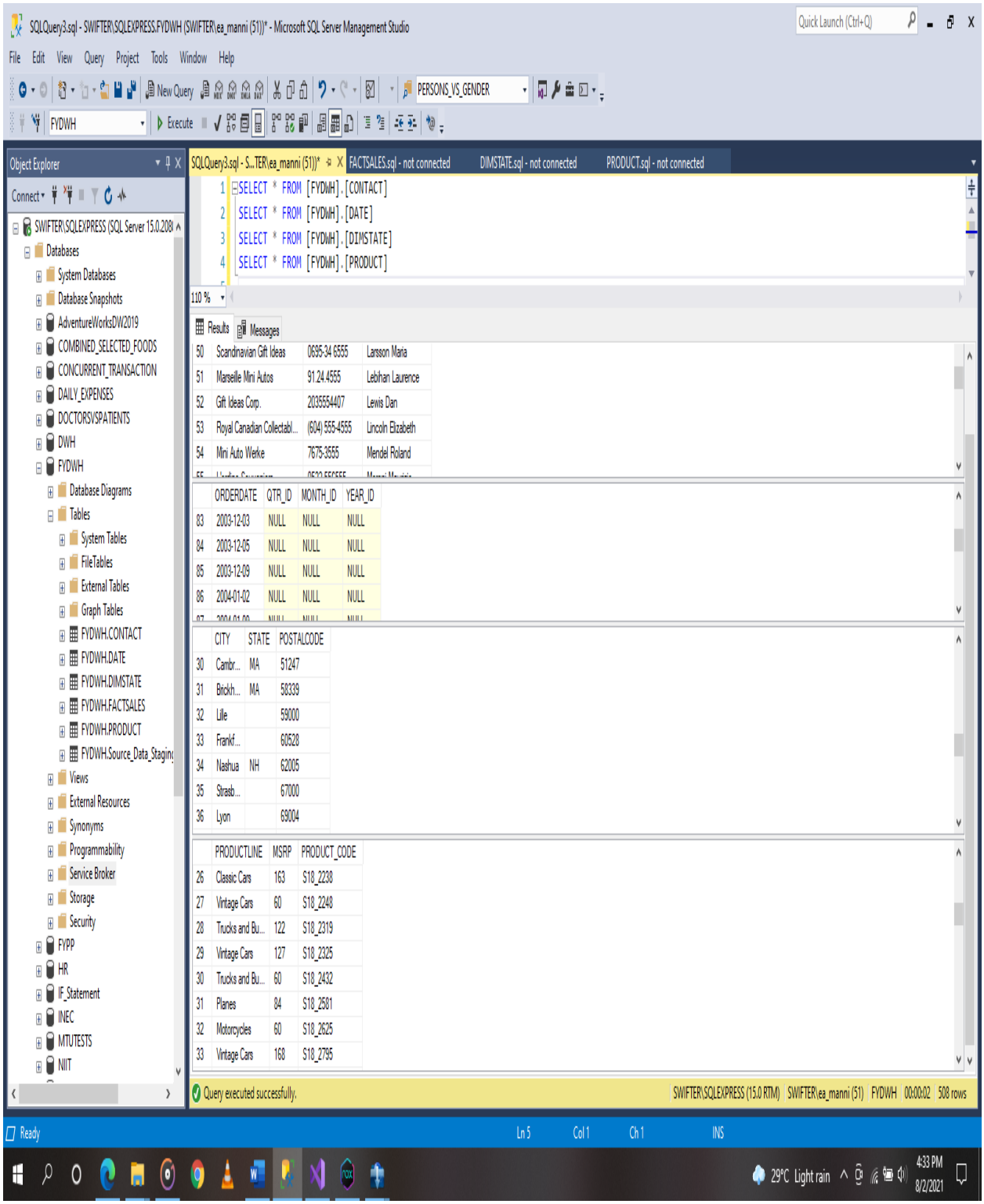

Figure 4.25: Selection of Data From Dimension Table.

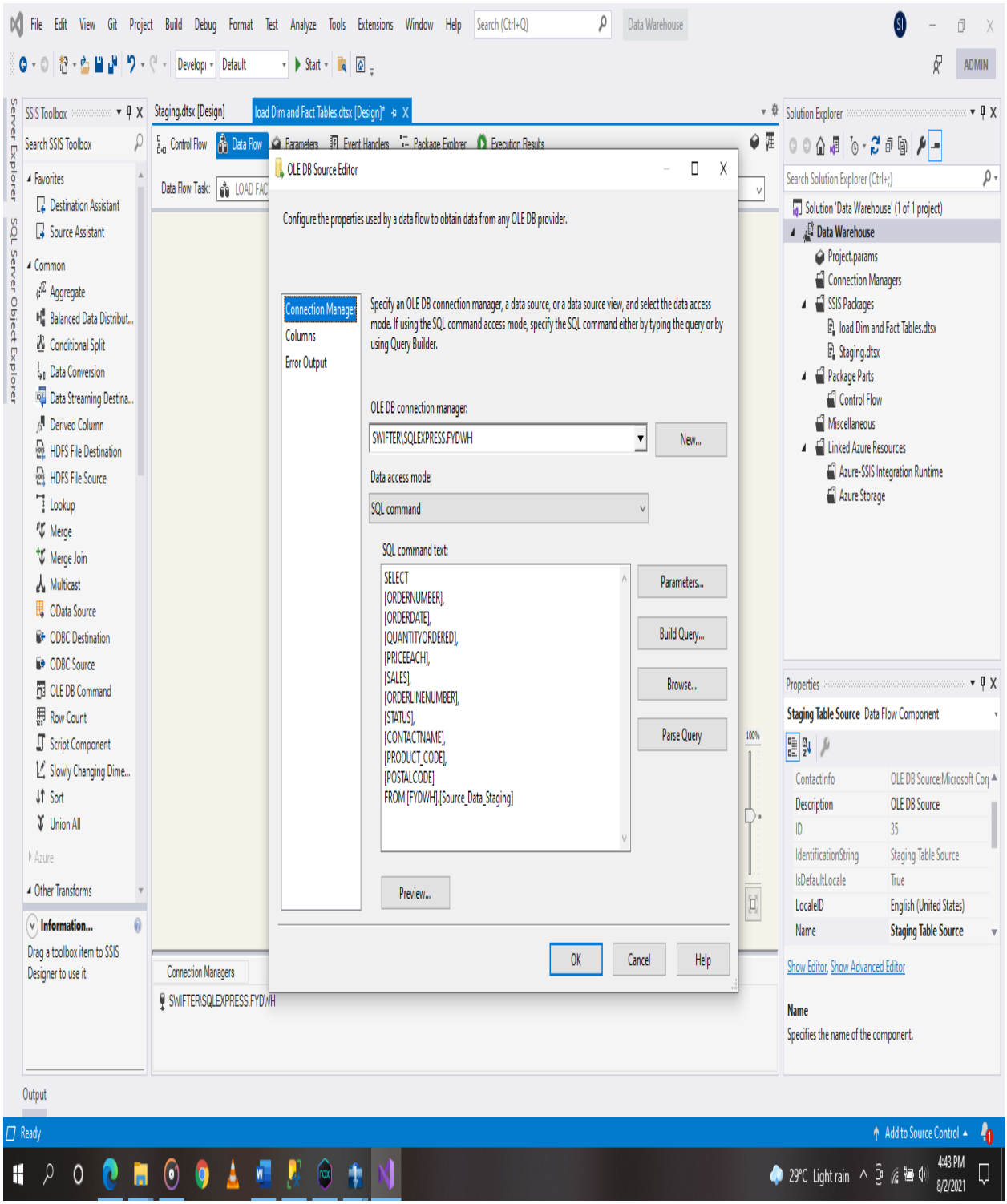

Figure 4.26: SQL Command to Populate Data From Staging Table into Fact Table.

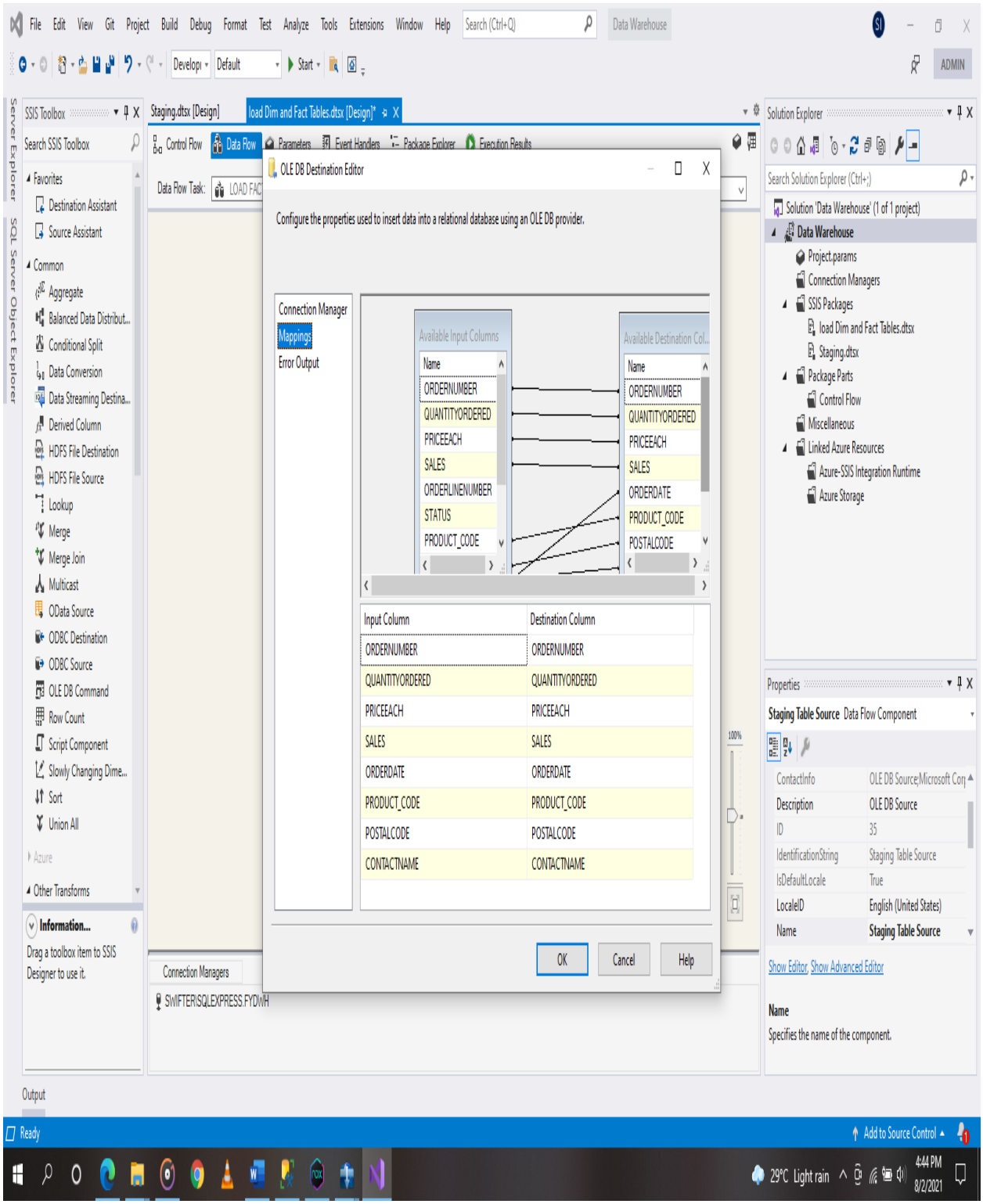

Figure 4.27: Mapping Columns From Staging Table to Fact Table.

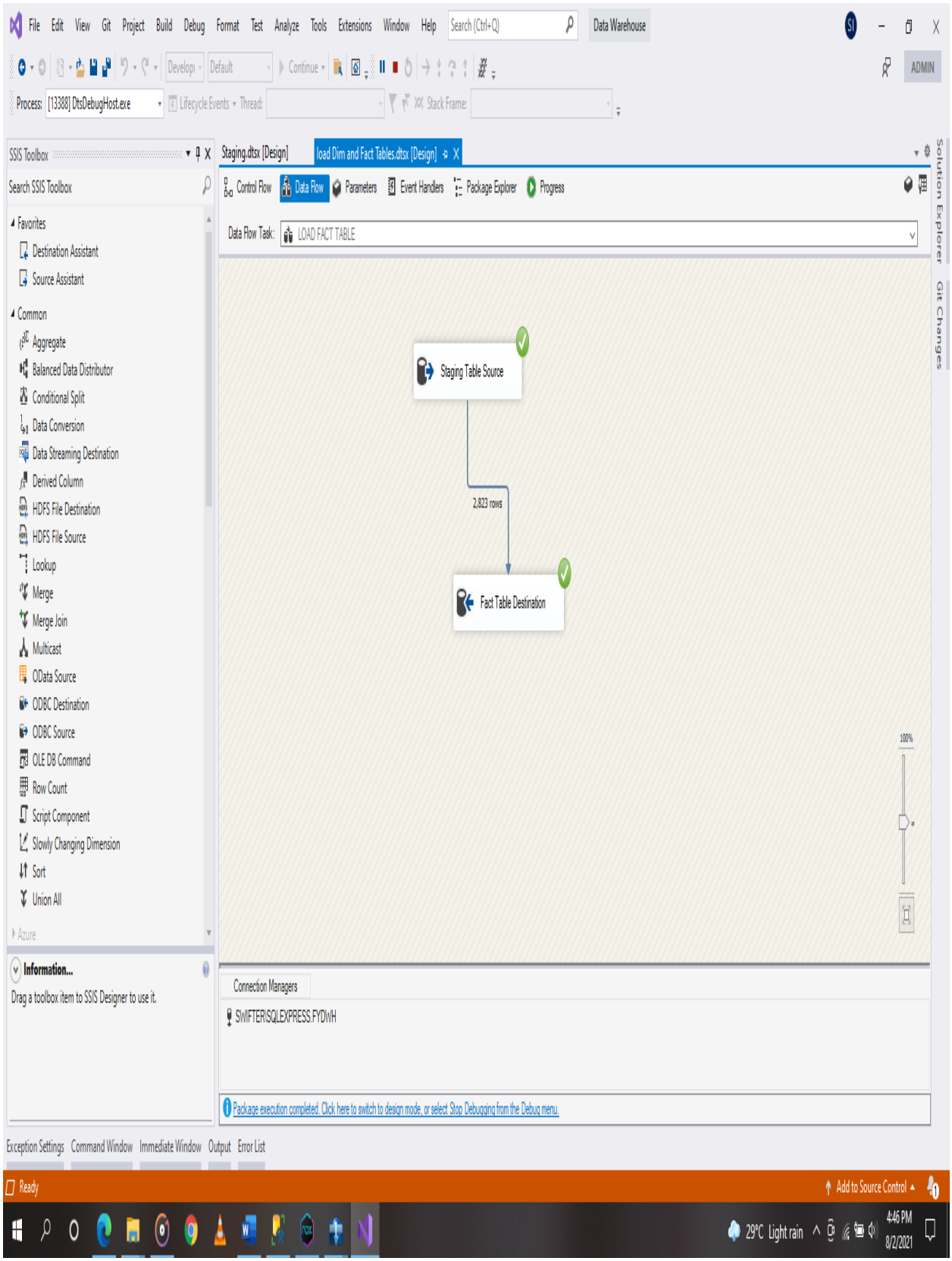

Figure 4.28: Successful Execution of Data Flow Tasks From Staging Table to Facts Table.

At this point implementation of the data warehouse is completed but the business intelligence module avails mentioning.

4.3.3 Implementation of business intelligence

The database holds a data warehouse but it is needed to create a reporting systems for the data present in the data warehouse, Visualizations are the best ways to report the data coming from any data source for human digestion. Afterall, the entire purpose of a data warehouse is to enhance better decision support.

After the data is imported measures are created. It is important to understand that measures in power BI are mathematical calculations for some certain facts to be expressed e.g., Imagine a column of the cost price of which we are to find out the total cost of items, a measure would be

```
Cost Price plus Cost Price =
SUM('''Fact    Table$'''[Cost    Price])    +    SUM('''Fact
Table$'''[Cost Price])
```

created, to sum up, all of the cost prices across all the “cost price” columns in the table.

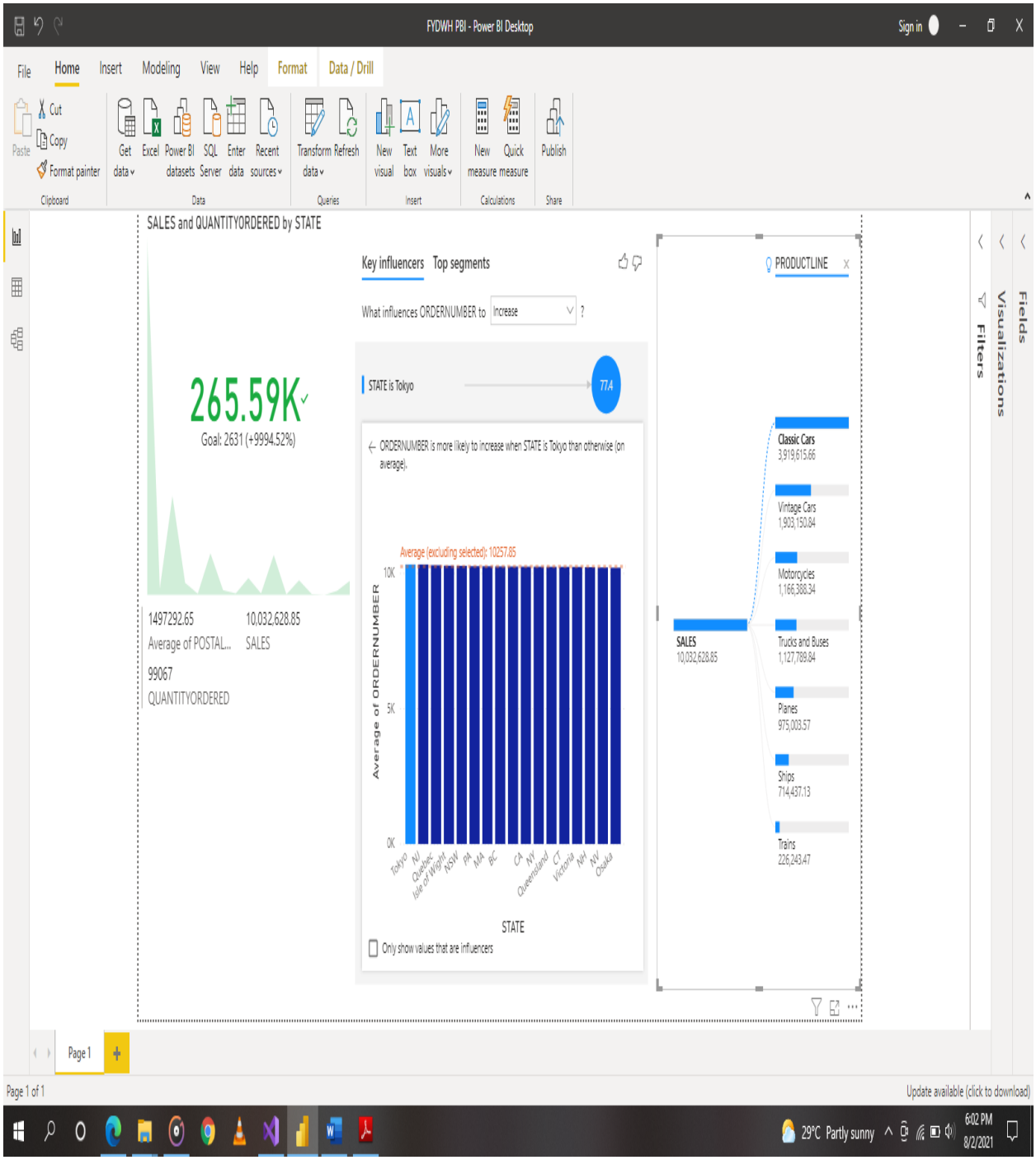

Figure 4.29: Data visualisation of data warehouse data with Microsoft Power BI

CHAPTER FIVE

SUMMARY AND CONCLUSION

This chapter contains the result summary of the project

5.1 Summary

The data warehouse facilitates reporting by implementing OLAP features, however data sources must be accommodated when conversion and extractions are performed. Although, the efficiency of a data warehouse will never be at it's peak because of some certain constraints, the data warehouse can be exploited to give certain results.

This paper presents a system for data warehousing and reporting services that will help small scale businesses thrive. With the application of different data flow tasks we are able to extend ordinary datasets into standard data warehouses.

5.1 Limitations

i. Time constraints.
ii. Previous works are scarce.
iii. Datasets are scarce.

5.2 Recommendation

It is recommended that other researchers should engage implement a datawarehouse with other progamming languages e.g., python, R e.t.c., or solutions such as postgresql database server to find out which of the following approaches yields better performance of a data warehouse.

REFERENCES

Almeida, F. (2017). *Concepts and Fundaments of Data Warehousing and OLAP*. Porto: Instituto Superior Politécnico Gaya.

Arya, K. (2019, June 4). *Latest Database Management Trends To Look Out For In 2021*. Retrieved from TechJockey.com: https://www.techjockey.com/blog/latest-database-management-trends-look-2019

Babajide, A. (2012). Effects of Microfinance on Micro and Small Enterprises (MSEs) Growth in Nigeria. *Asian Economic and Financial Review, 2*, 463-477.

Business Application Research Center. (2017, September 2). *Using internal and external data for decision-making*. Retrieved from bi-survey.com/: https://bi-survey.com/data-sources

Edward M. Leonard, B. (2011). *ABSTRACT DESIGN AND IMPLEMTATION OF AN ENTERPRISE DATA WAREHOUSE*. Marquette University.

Golfarelli, & Rizzi. (2009). *Data Warehouse Design: Modern Principles and Methodologies.*

Gupta, A. (2019). A Complete Reference for Informatica Power Center ETL Tool. *International Journal of Trend in Scientific Research and Development*, 1063-1070. doi:10.31142/ijtsrd19045

Gupta, A., & Sahayadhas, A. (2020, 7 11). A Comprehensive Survey to Design Efficient Data Warehouse for Betterment of Decision Support Systems for Management and Business Corporates. *International Journal of Management*, 463-471. Retrieved from http://www.iaeme.com/IJM/issues.asp?

Hughes, R. (2016). *Agile Data Warehousing for the Small Enterprise*. Miami: Elsevier Inc.

Inmon, W. H. (2002). *Building the Data Warehouse*. NEW YORK; CHICHESTER; WEINHEIM; BRISBANE; SINGAPORE; TORONTO: John Wiley & Sons, Inc.

Linstedt, D., & Olschimke, M. (2016). Introduction to Data Warehousing. *Building a Scalable Data Warehouse with Data Vault 2.0*.

Microsoft. (2021, 5 3). *SQL Server documentation.* Retrieved from SQL Server Technical Documentation: https://aka.ms/sqldocs

Mohania, M., Karlapalem, K., & Bellatreche, L. (2002). *Some Issues in Design of Data Warehousing Systems* .

Nielsen, M. A. (2015). *Neural Networks and Deep Learning.* Determination Press.

Nikhilesh. (2015, December 31). *Types of Dimensions in Data Warehouse*. Retrieved from helicaltech.com.

Ponniah, P. (2002). *Data Warehousing Fundamentals.* New York: John Wiley.

Qureshi, A. A., & Jamal, S. M. (2012, December). Taxonomy based Data Marts. *International Journal of Computer Applications (0975 – 8887), 60*, 6-13.

Rojas, H., Arias, K. A., & Renteria, R. (2020). Service-oriented architecture design for small and medium enterprises with infrastructure and cost optimization. *International Conference on Computer Science and Computational Intelligence*, 488-497.

Santoso, L. W., & Yulia. (2017). Data Warehouse with Big Data Technology for Higher Education. *Information Systems International Conference*, 93-99.

Schutz, C., Neumayr, B., Schrefl, M., & Neubock, T. (2016). Reference Modeling for Data Analysis: The BIRD Approach. *International Journal of Cooperative Information Systems*, 25. doi:10.1142/S0218843016500064.

Sreenivasaiah, P. K., & Kim, D. H. (2010). Current Trends and New Challenges of Databases and Web Applications for. *Frontiers in Physiology · December 2010*, 3-7. doi:10.3389/fphys.2010.00147

Watson, H. J. (2002). Recent Developments in Data Warehousing. *Communications of the Association for Information Systems*, 1-25. doi:10.17705/1CAIS.00801

APPENDIX

```
	CREATE TABLE [FYDWH].[Source_Data_Staging](
	[Source_StagingID] [int] IDENTITY(1,1) NOT NULL,
	[ORDERNUMBER] [int] NULL,
	[QUANTITYORDERED] [int] NULL,
	[PRICEEACH] [float] NULL,
	[SALES] [float] NULL,
	[ORDERLINENUMBER] [int] NULL,
	[ORDERDATE] [date] NULL,
	[STATUS] [varchar](50) NULL,
	[QTR_ID] [int] NULL,
	[MONTH_ID] [int] NULL,
	[YEAR_ID] [int] NULL,
	[PRODUCTLINE] [varchar](50) NULL,
	[MSRP] [int] NULL,
	[PRODUCT_CODE] [varchar](50) NULL,
	[CUSTOMERNAME] [varchar](50) NULL,
	[PHONE] [varchar] (50) NULL,
	[CITY] [varchar] (50) NULL,
	[STATE] [varchar] (50) NULL,
	[POSTALCODE] [INT] NULL,
	[COUNTRY] [varchar] (50) NULL,
	[TERRITORY] [varchar] (50) NULL,
	[CONTACTNAME] [varchar] (50) NULL,
PRIMARY KEY CLUSTERED
(
	[Source_StagingID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
```

Sql Queries to Create Staging Table

Sql Queries to Create Dimension Tables

```
	CREATE TABLE [FYDWH].[PRODUCT](
	[PRODUCTLINE] [varchar](50) NULL,
	[MSRP] [int] NULL,
	[PRODUCT_CODE] [varchar](50) NOT NULL,
PRIMARY KEY CLUSTERED
(
	[PRODUCT_CODE] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
CREATE TABLE [FYDWH].[DIMSTATE](
	[CITY] [varchar] (50) NULL,
	[STATE] [varchar] (50) NULL,
	[POSTALCODE] [INT] NOT NULL,
PRIMARY KEY CLUSTERED
(
	[POSTALCODE] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
CREATE TABLE [FYDWH].[CONTACT](
	[CUSTOMERNAME] [varchar](50) NULL,
	[PHONE] [varchar] (50) NULL,
	[CONTACTNAME] [varchar] (50) NOT NULL,
PRIMARY KEY CLUSTERED
(
	[CONTACTNAME] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
CREATE TABLE [FYDWH].[PRODUCT](
	[PRODUCTLINE] [varchar](50) NULL,
	[MSRP] [int] NULL,
	[PRODUCT_CODE] [varchar](50) NOT NULL,
PRIMARY KEY CLUSTERED
([PRODUCT_CODE] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
```

Sql Queries to Create Fact Tables:

```
GO
       CREATE TABLE [FYDWH].[FACTSALES](
       [ORDERNUMBER] [int] NOT NULL,
       [QUANTITYORDERED] [int] NULL,
       [PRICEEACH] [float] NULL,
       [SALES] [float] NULL,
       [ORDERDATE] [date] NULL,
       [PRODUCT_CODE] [varchar](50) NULL,
       [POSTALCODE] [INT] NULL,
       [CONTACTNAME] [varchar] (50) NULL
)
GO
ALTER TABLE [FYDWH].[FACTSALES] WITH CHECK ADD CONSTRAINT [FK_DATEID] FOREIGN
KEY([ORDERDATE])
REFERENCES [FYDWH].[DATE]([ORDERDATE])
GO
ALTER TABLE [FYDWH].[FACTSALES] CHECK CONSTRAINT [FK_DATEID]
GO

ALTER TABLE [FYDWH].[FACTSALES] WITH CHECK ADD CONSTRAINT [FK_CONTACTID]
FOREIGN KEY([CONTACTNAME])
REFERENCES [FYDWH].[CONTACT]([CONTACTNAME])
GO
ALTER TABLE [FYDWH].[FACTSALES] CHECK CONSTRAINT [FK_CONTACTID]
GO

ALTER TABLE [FYDWH].[FACTSALES] WITH CHECK ADD CONSTRAINT [FK_PRODUCTID]
FOREIGN KEY([PRODUCT_CODE])
REFERENCES [FYDWH].[PRODUCT]([PRODUCT_CODE])
GO
ALTER TABLE [FYDWH].[FACTSALES] CHECK CONSTRAINT [FK_PRODUCTID]
GO

ALTER TABLE [FYDWH].[FACTSALES] WITH CHECK ADD CONSTRAINT [FK_DIMSTATEID]
FOREIGN KEY([POSTALCODE])
REFERENCES [FYDWH].[DIMSTATE]([POSTALCODE])
GO
ALTER TABLE [FYDWH].[FACTSALES] CHECK CONSTRAINT [FK_DIMSTATEID]


GO
```