1. Step 1:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

SS 1: FashionKey operational database with the required tables

2. Step 2:

A screenshot of a computer

Description automatically generated

SS2: ER model for FashionKey

After the operational database was successfully created, first of all, SQL query to authorize for database diagrams was executed. Then under the FashionKey, right clicked on the database diagrams and selected the new database diagrams. Then selected all the tables and Ok.

The 4-step process for designing a dimensional model for FashionKey Inc is as follows:

**- Business Process:** the business process that will be the subject of the analysis is the first step in creating a dimensional model. The sales process, in which FashionKey Inc. sells clothing items to customers through its online and physical stores, could be considered the business process in this case.

**- Grain**: Grain is the lowest level of detail. The granularity, for instance, may be at the individual transaction level in the case of FashionKey Inc., where each row in the fact table represents a distinct sales transaction and the dimensions give extra information.

**- Dimensions:** Selecting the dimensions that will be utilized to set the context for the analysis is the third stage. Date, client, product, shop, vendor, and area are a few examples of conceivable aspects in the context of FashionKey Inc.

**- Fact:** The last stage is to define the fact table, which will include the measurements or metrics that will be examined. The fact table for FashionKey Inc. might include statistics like sales revenue, amount sold, and profit margin.

1. Step 3:

A screenshot of a computer

Description automatically generated

SS 3: Created FashionKeyDW Database

Executed t-sql to create a FashionKeyDW Database

1. Step 4:

A screenshot of a computer

Description automatically generated

SS 4: Created the dimension tables

Like in FashionKey, database diagram authorization was successfully executed on FashionKeyDW. Then looking at the ER Model, dimensions (who, what, where, when) were selected from Customer, Vendor, Product, Store and date. Dimension tables were created for all of them while assigning a Surrogate key to each of the dimension tables and the included columns with their respective datatypes.

1. Step 5:

A computer screen shot of a computer screen

Description automatically generated

SS 5: Star schema with the fact table in the middle

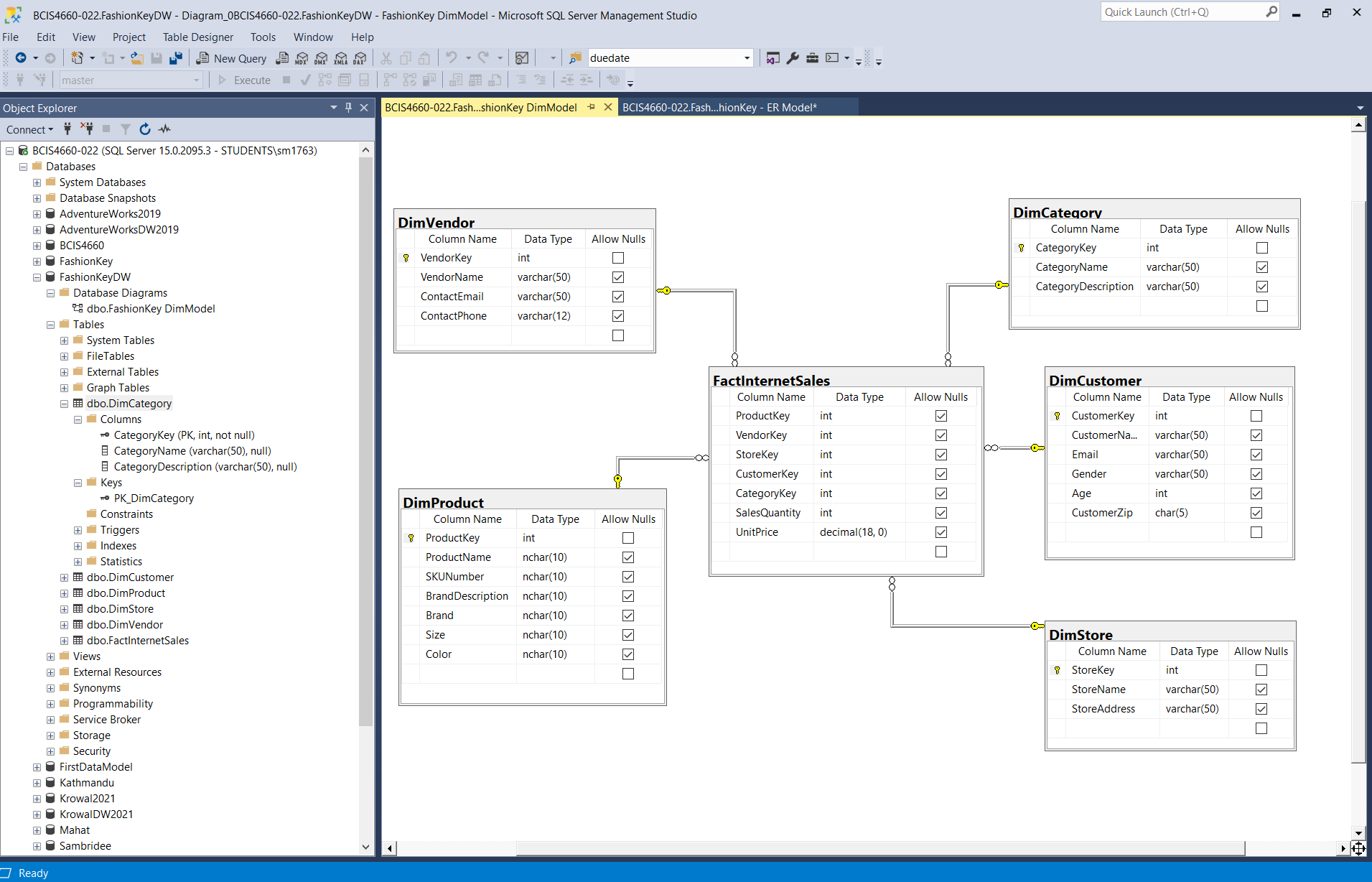
The fact table is what is in the middle of the screenshot, that is connected to all other dimension tables. To create a fact table, just right click on the database diagram page and select create new table and include all the surrogate key from each of the dimension tables and the measurement metrics like the quantity and the price.

a. A fact table contains numerical measurements like sales quantity and price while dimension tables contain descriptive information about the attributes like customer information, geographics, etc

b. Fact tables are usually larger than dimension tables as fact tables contain dynamic information while dimension tables contain static information like description.

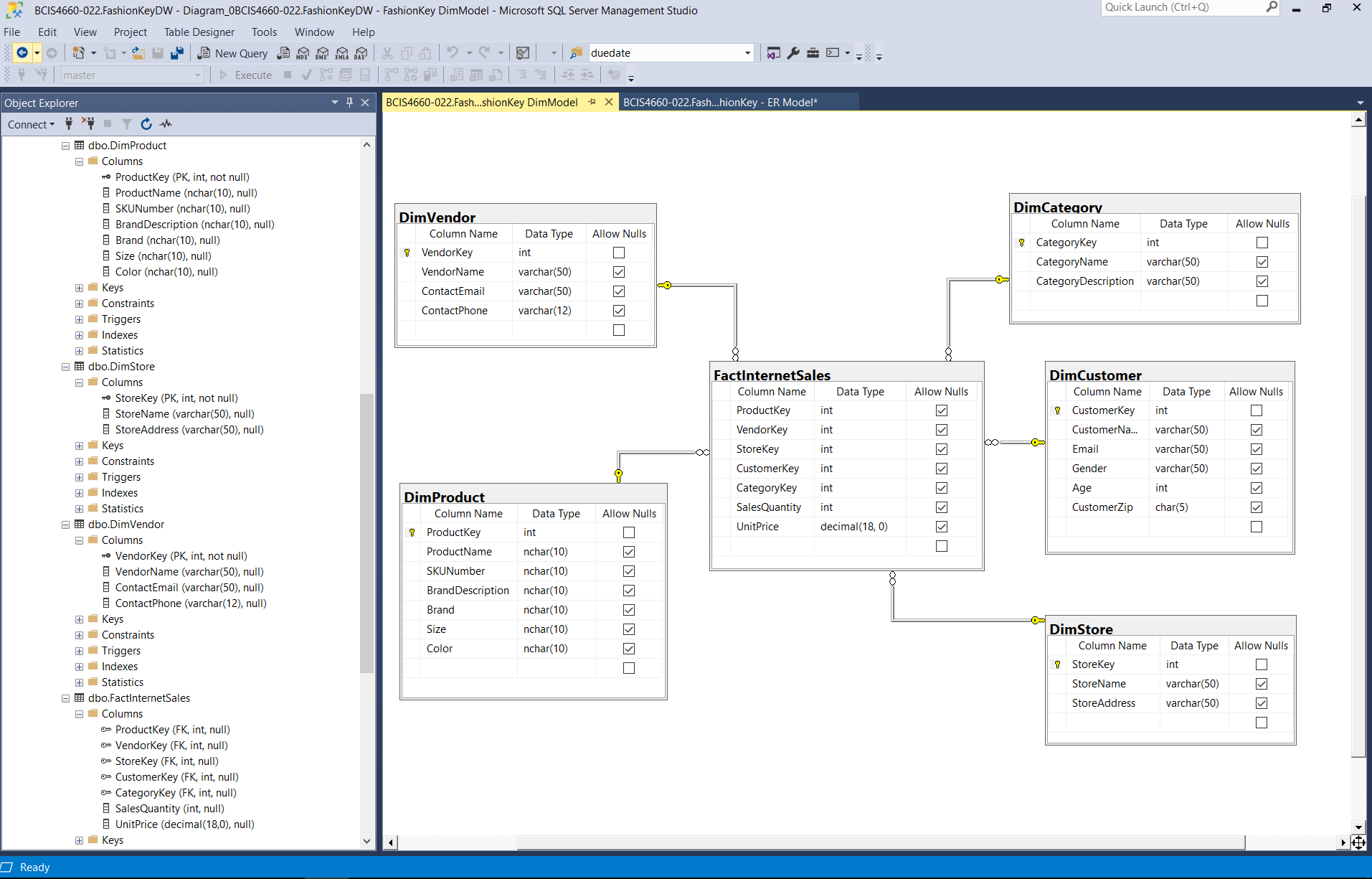
c. The fact table's primary key is made up of foreign keys from each of the dimension tables, that joins the fact tables with dimension tables.

1. Step 6:

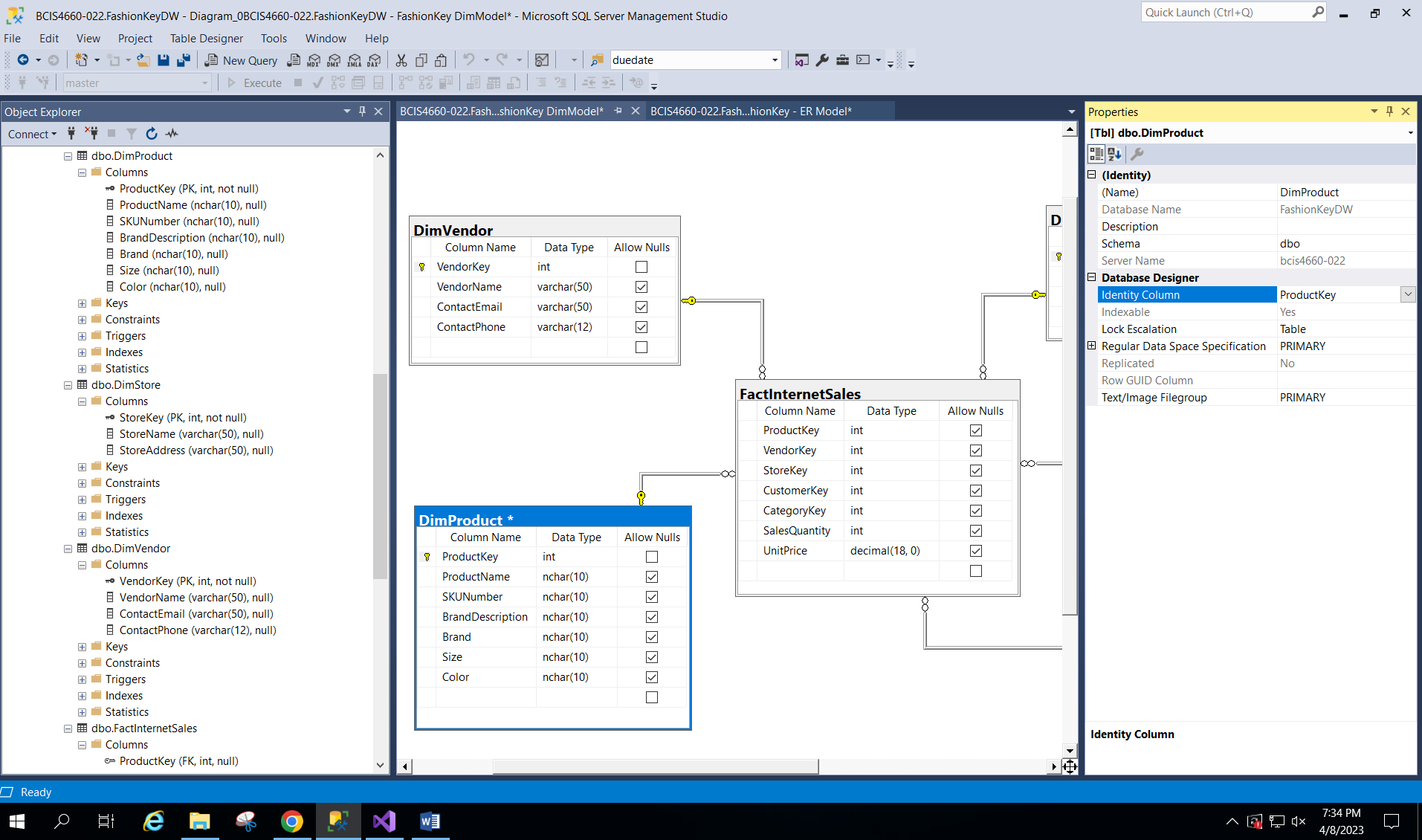


SS 1a: Dimensional Model

Database diagram showing the five dimension tables called DimVendor, DimProduct, DimCustomer, DimStore, and DimCategory and a fact table called FactInternetSales.

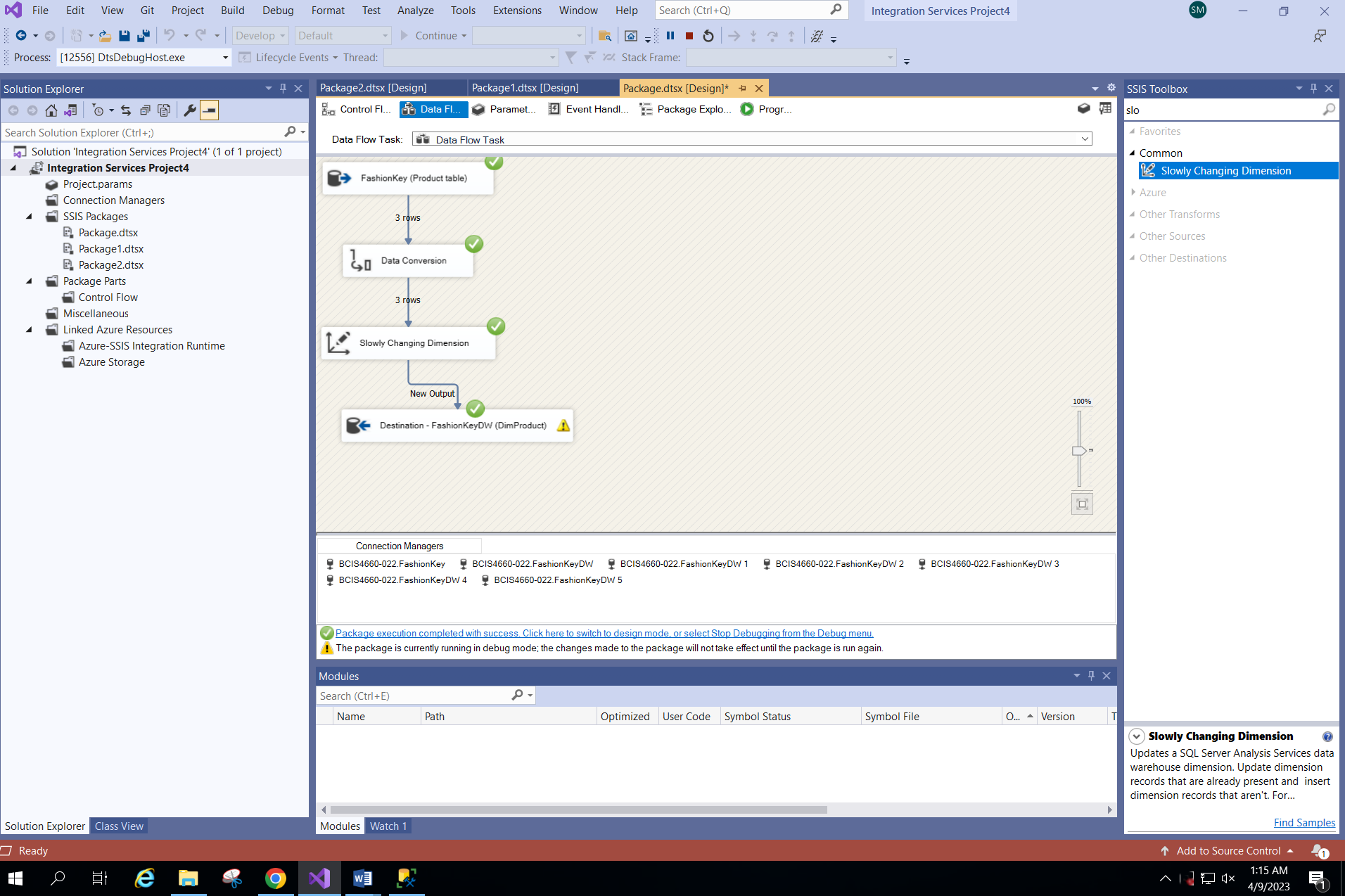


SS 1b. Object explorer showing the tables created



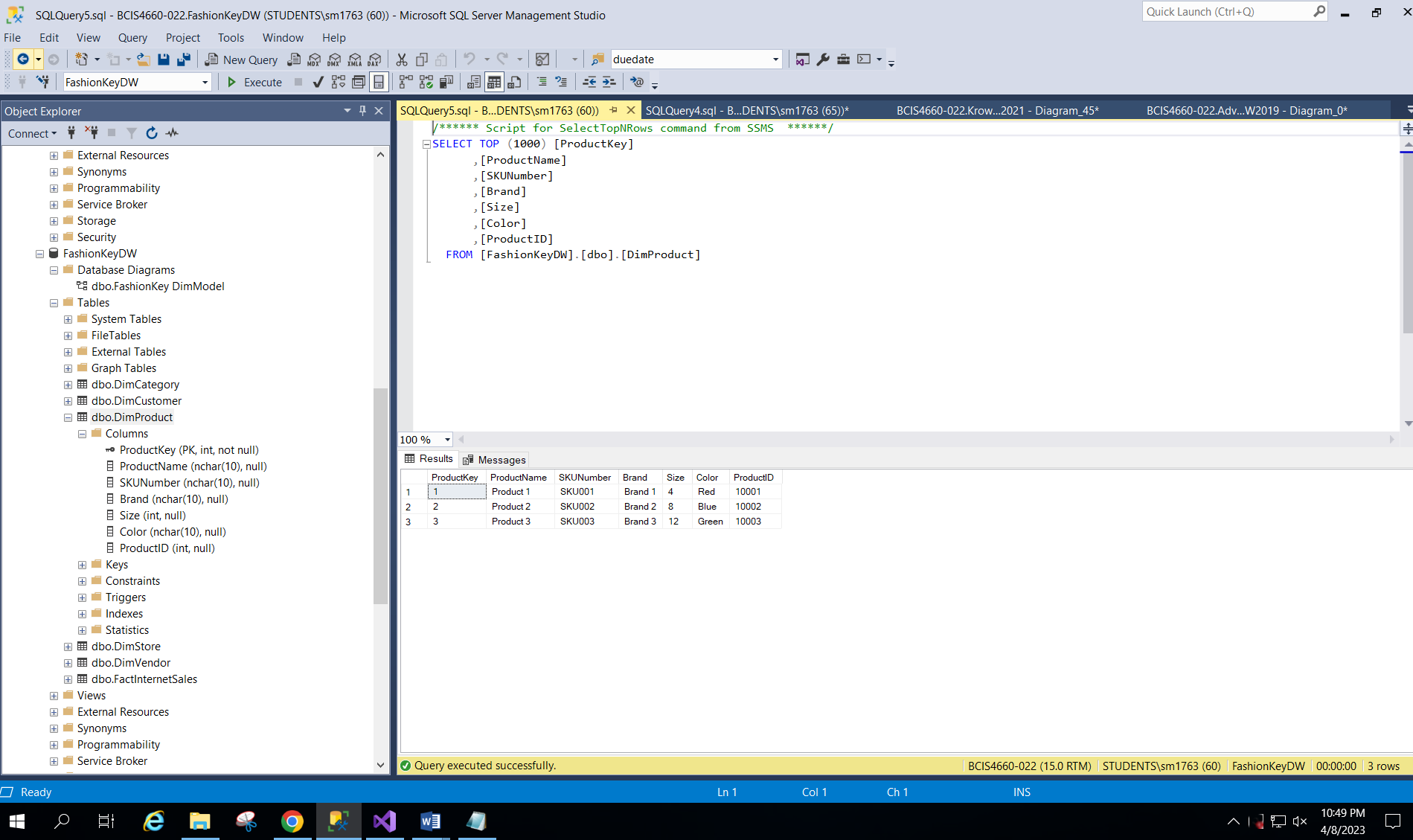
SS 1C : SSMS showing the Identity key property of the DimProduct table

1. Step 7:



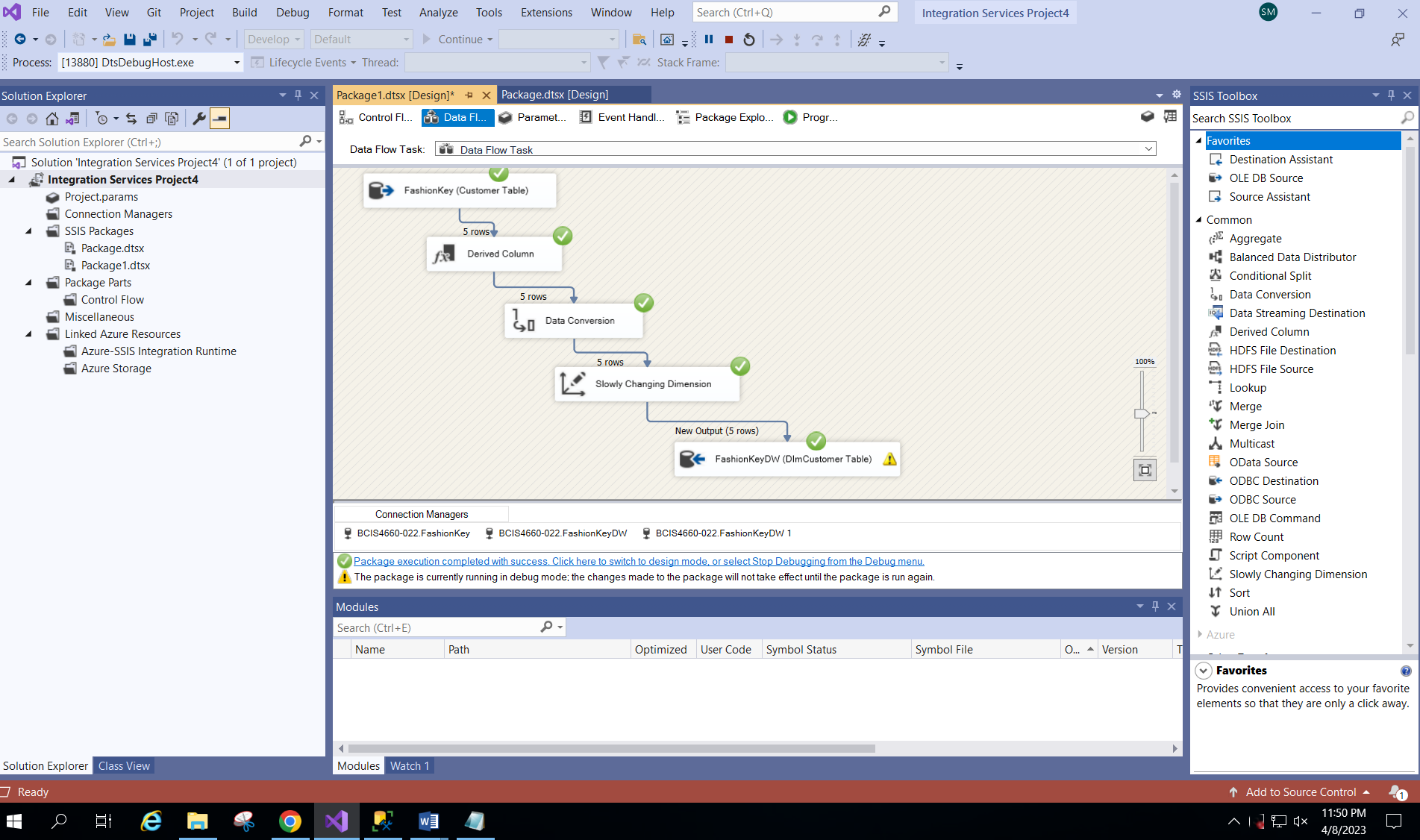
SS 2A: ETL package to extract data from Product table to DimProduct table

The data from the operational database FashionKey table in the Product table was extracted using a ETL package which included tasks like data flow task. The source was connected, the data was converted to appropriate data types, slowly changing dimension was added and the destination was set to FashionKeyDW, DimProduct table.



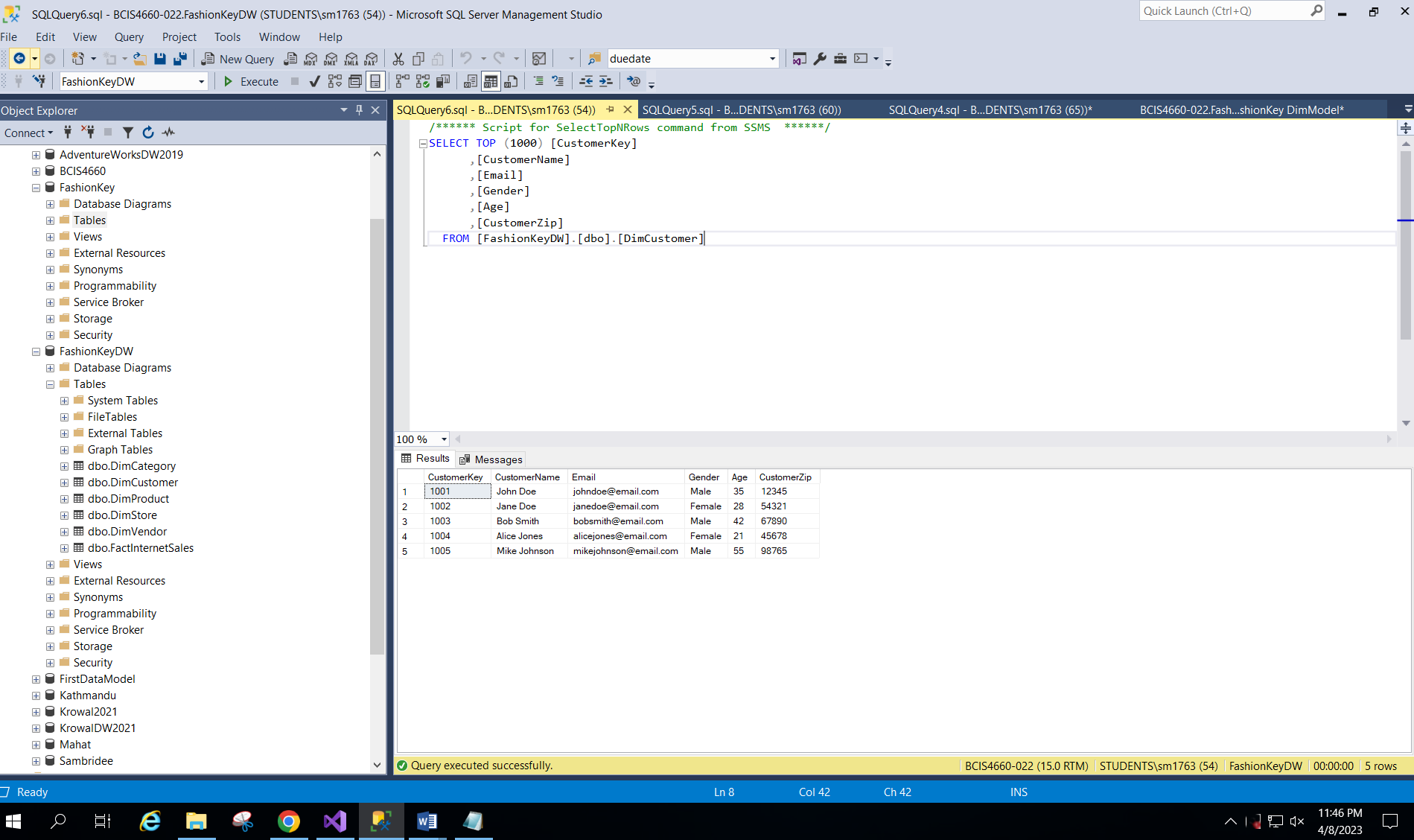
SS 2B: SSMS showing the data loaded into the destination table

1. Step 8:



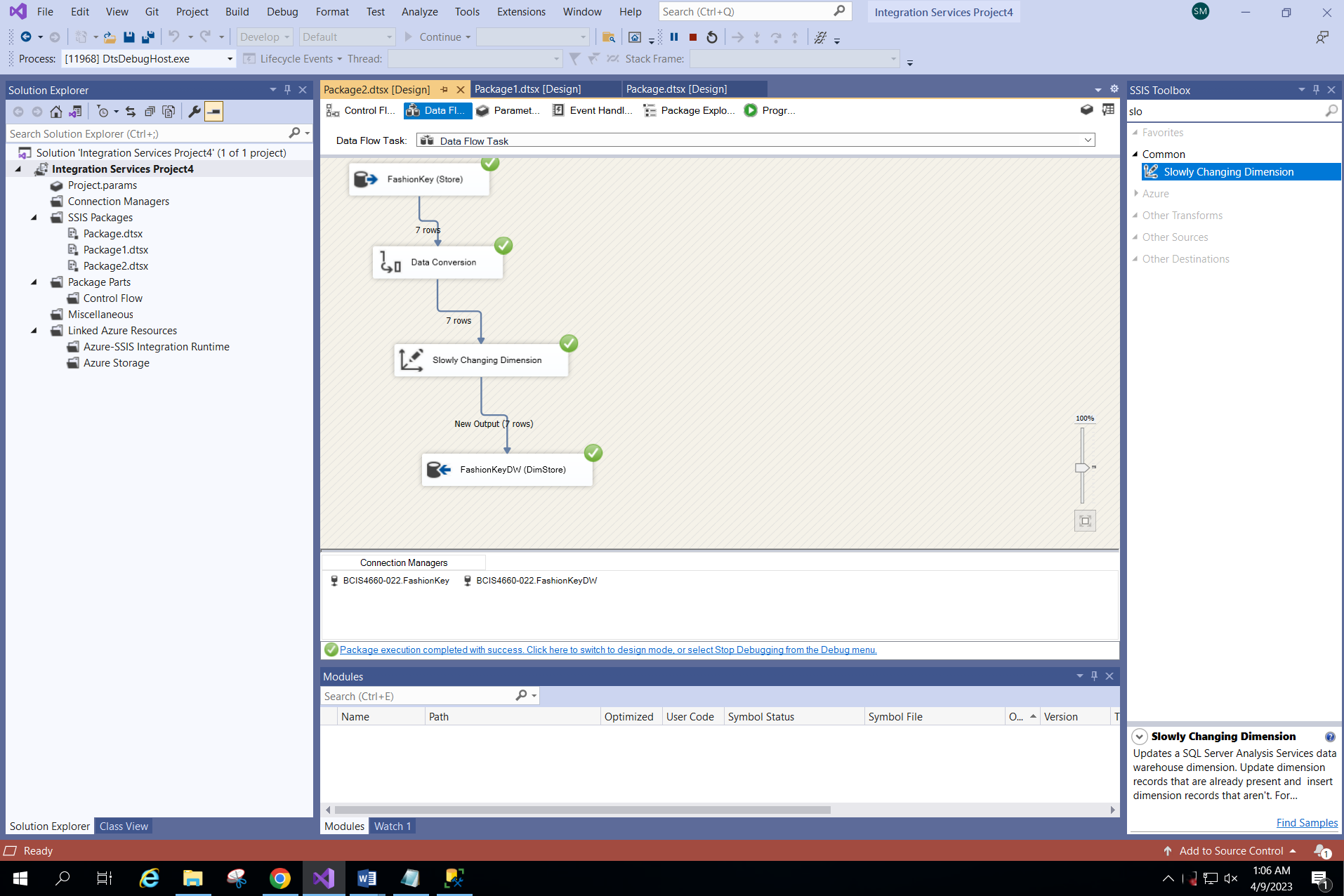
**SS 3A:** ETL package to extract data from Customer table to DimCustomer table

The data from the operational database FashionKey table in the Customer table was extracted using a ETL package which included tasks like data flow task. The source was connected, data was transformed (FirstName and LastName columns were merged into CustomerName column, the data was converted to appropriate data types, slowly changing dimension was added and the destination was set to FashionKeyDW, DiCustomer table.



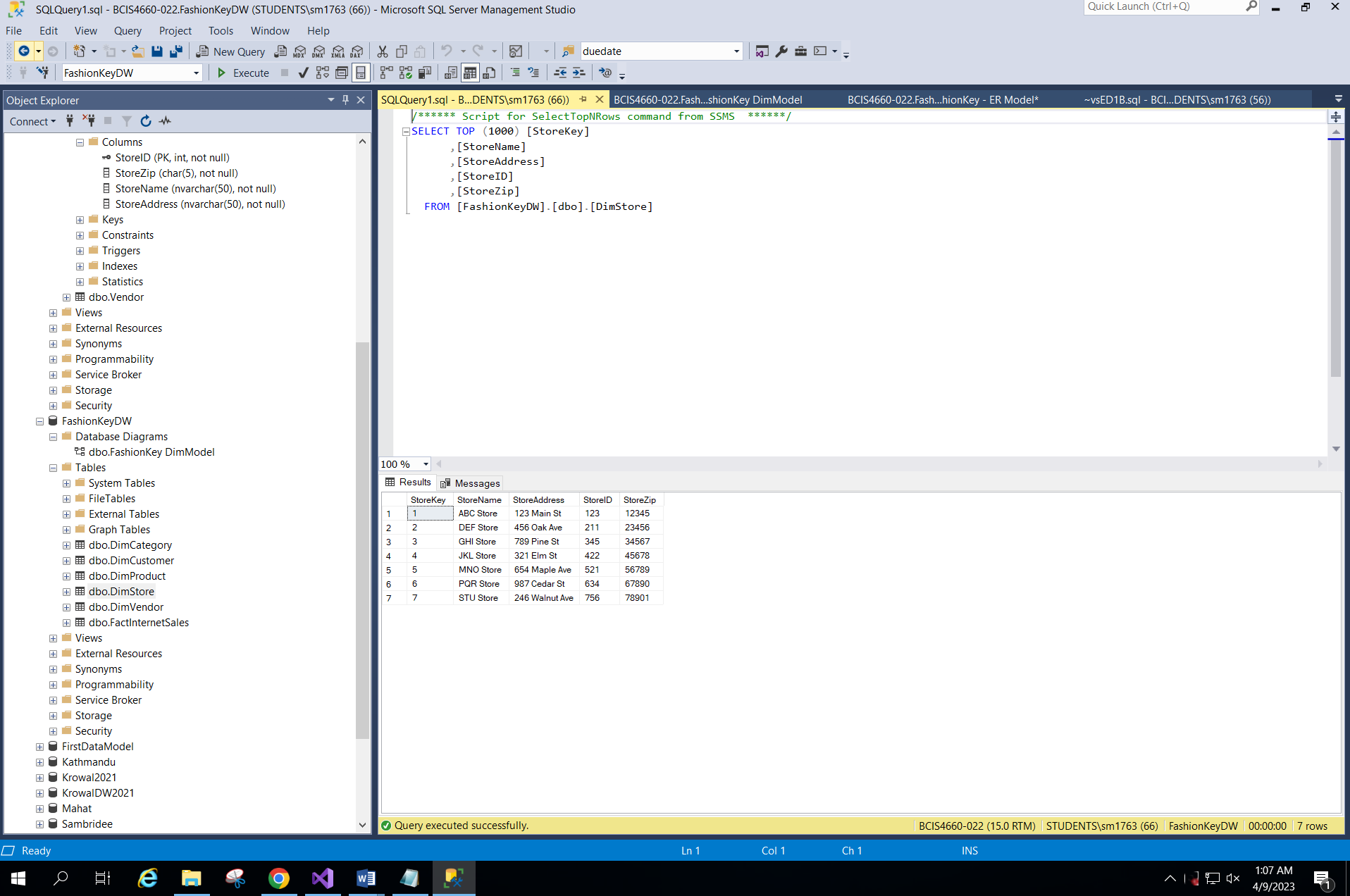
**SS 3B:** SSMS showing the data loaded from Customer table into the destination table (DimCustomer)

1. Step 9:



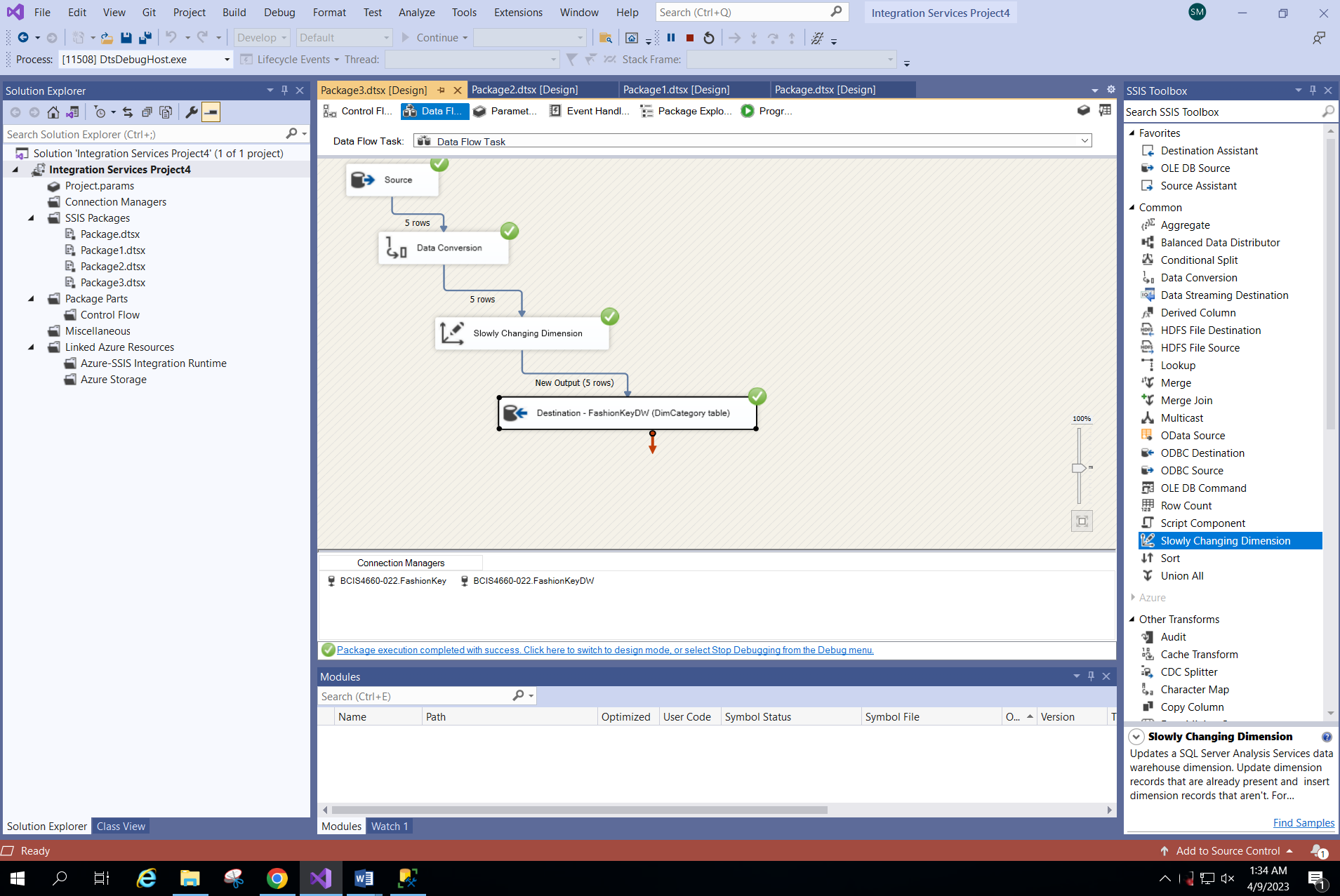
**SS 4A: ETL package to extract data from Store table to DimStore table**

The data from the operational database FashionKey table in the Store table was extracted using a ETL package which included tasks like data flow task. The source was connected, the data was converted to appropriate data types, slowly changing dimension was added and the destination was set to FashionKeyDW, DimStore table.



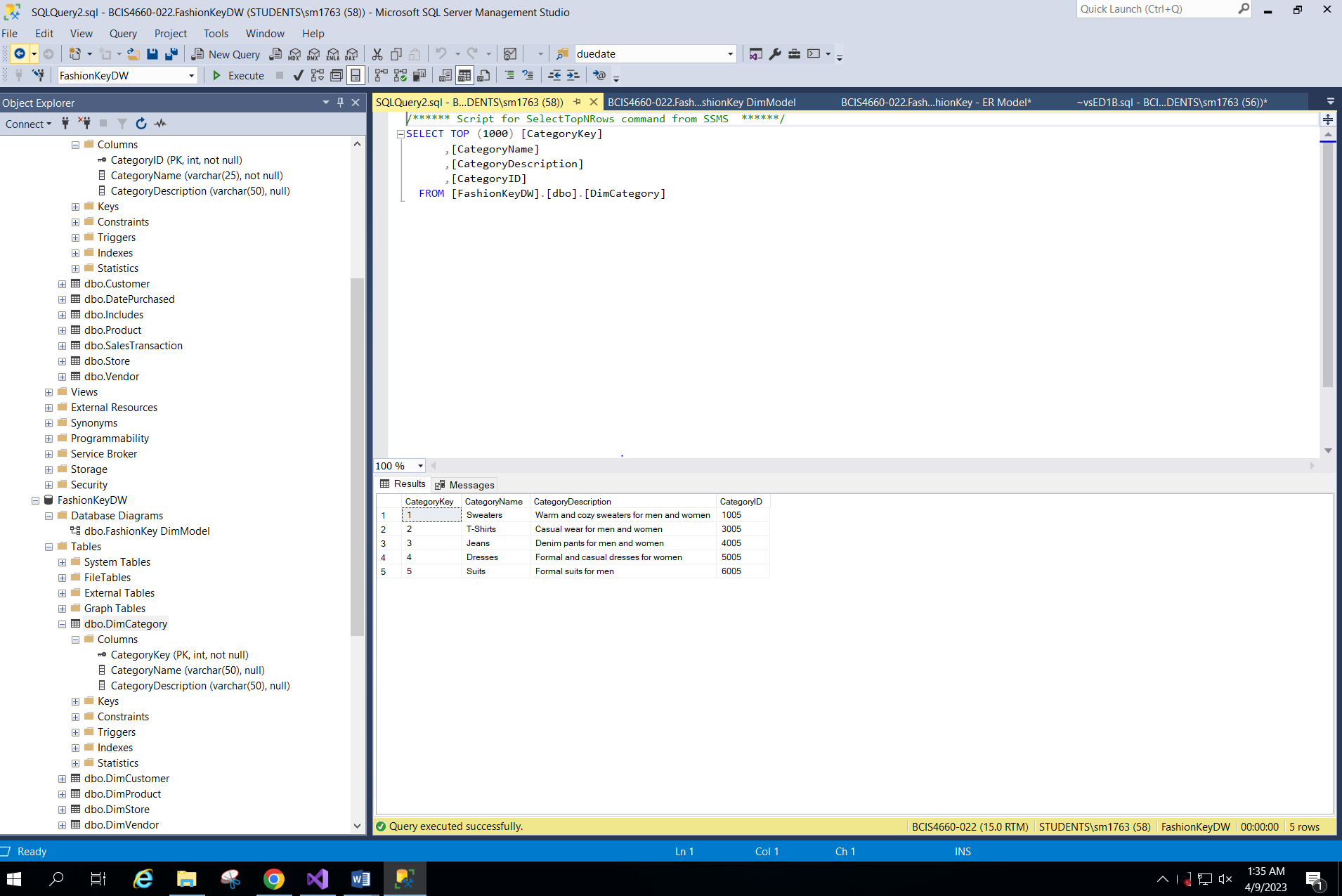
**SS 4B: SSMS showing the data loaded from Store table into the destination table (DimStore)**

1. Step 10:



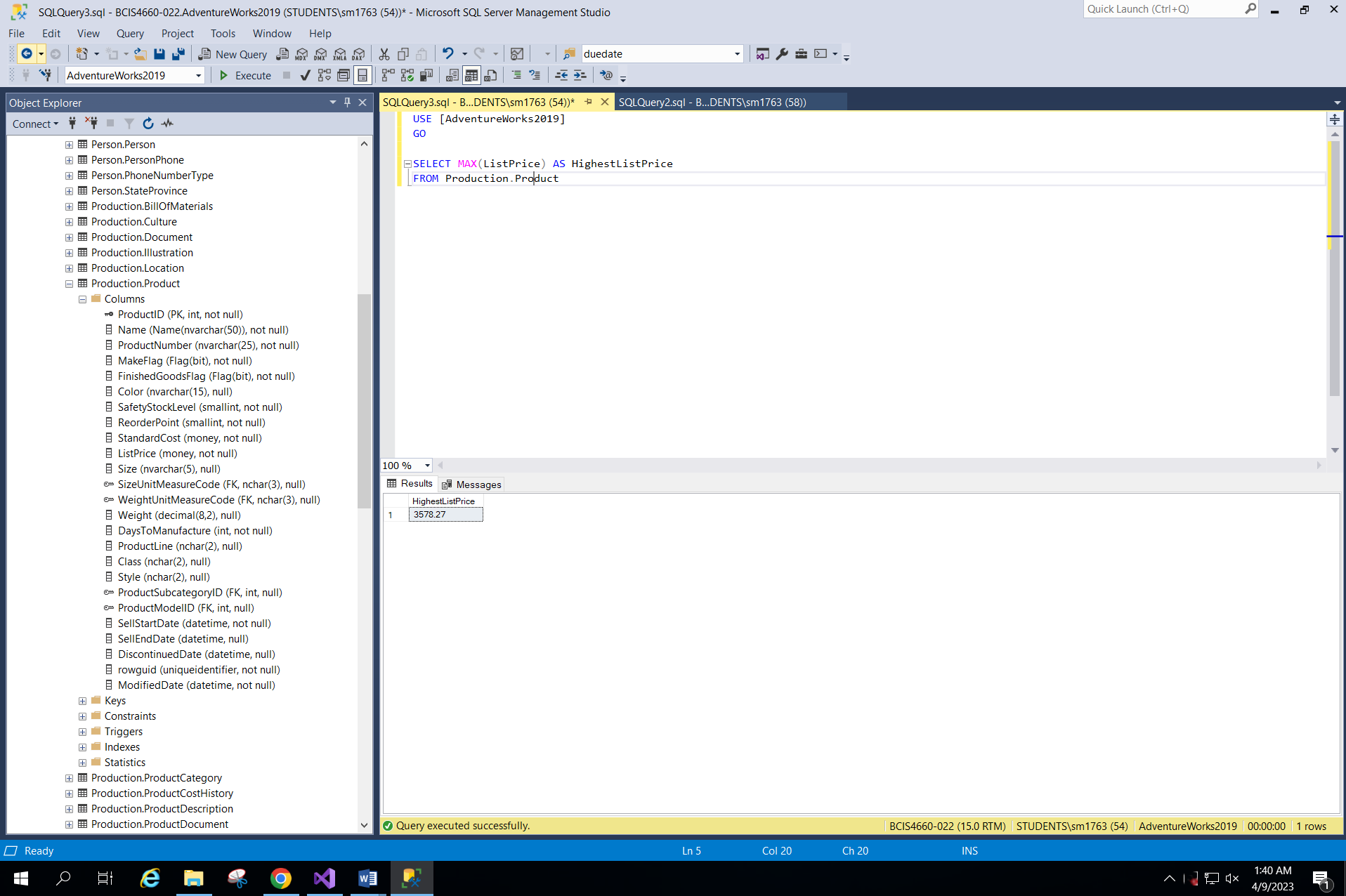
**SS 5A:** **ETL package to extract data from Category table to DimCategory table**

The data from the operational database FashionKey table in the Category table was extracted using a ETL package which included tasks like data flow task. The source was connected, the data was converted to appropriate data types, slowly changing dimension was added and the destination was set to FashionKeyDW, DimCategory table.



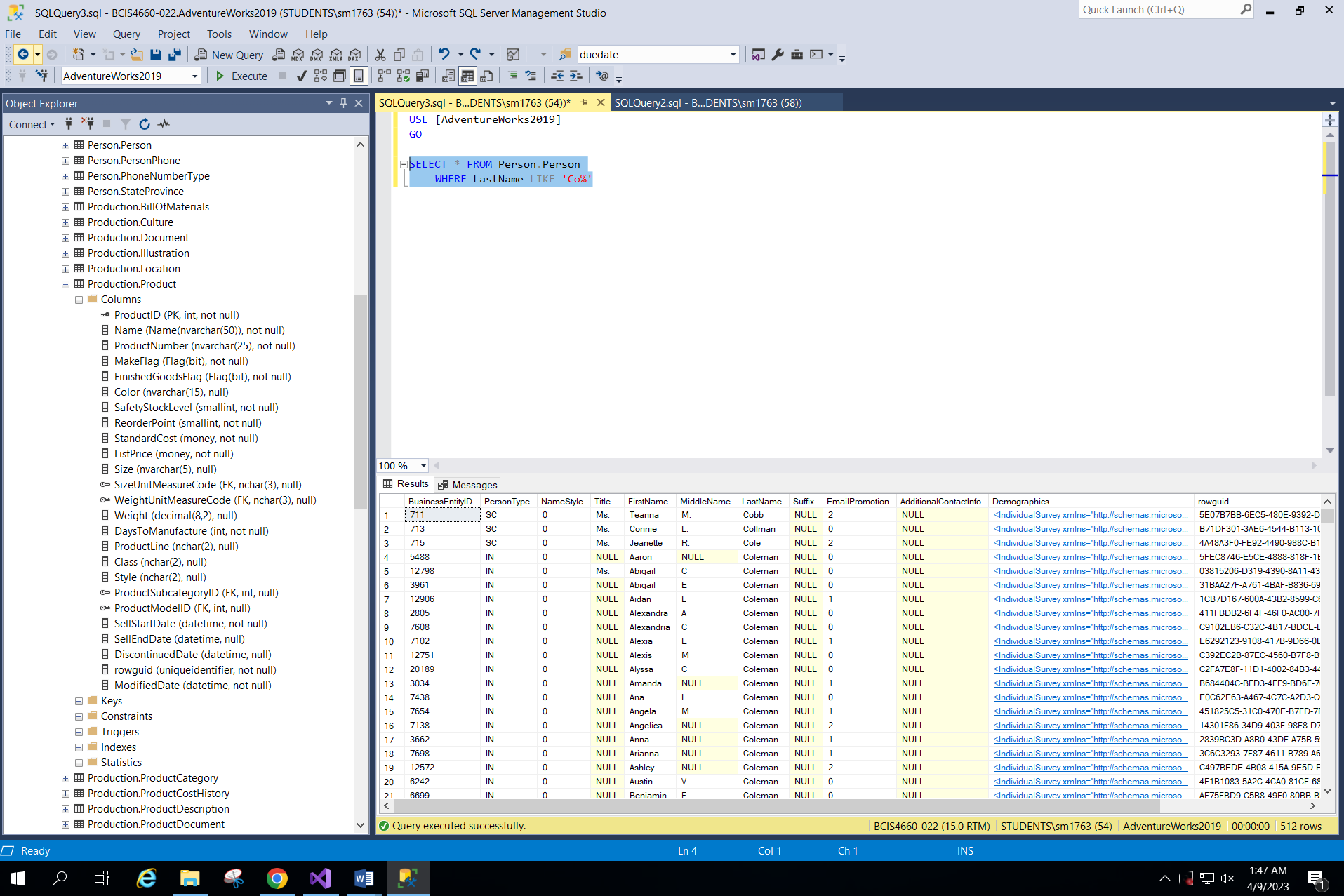
**SS 5B: SSMS showing the data loaded from Category table into the destination table (DimCategory)**

Exercise T-SQL

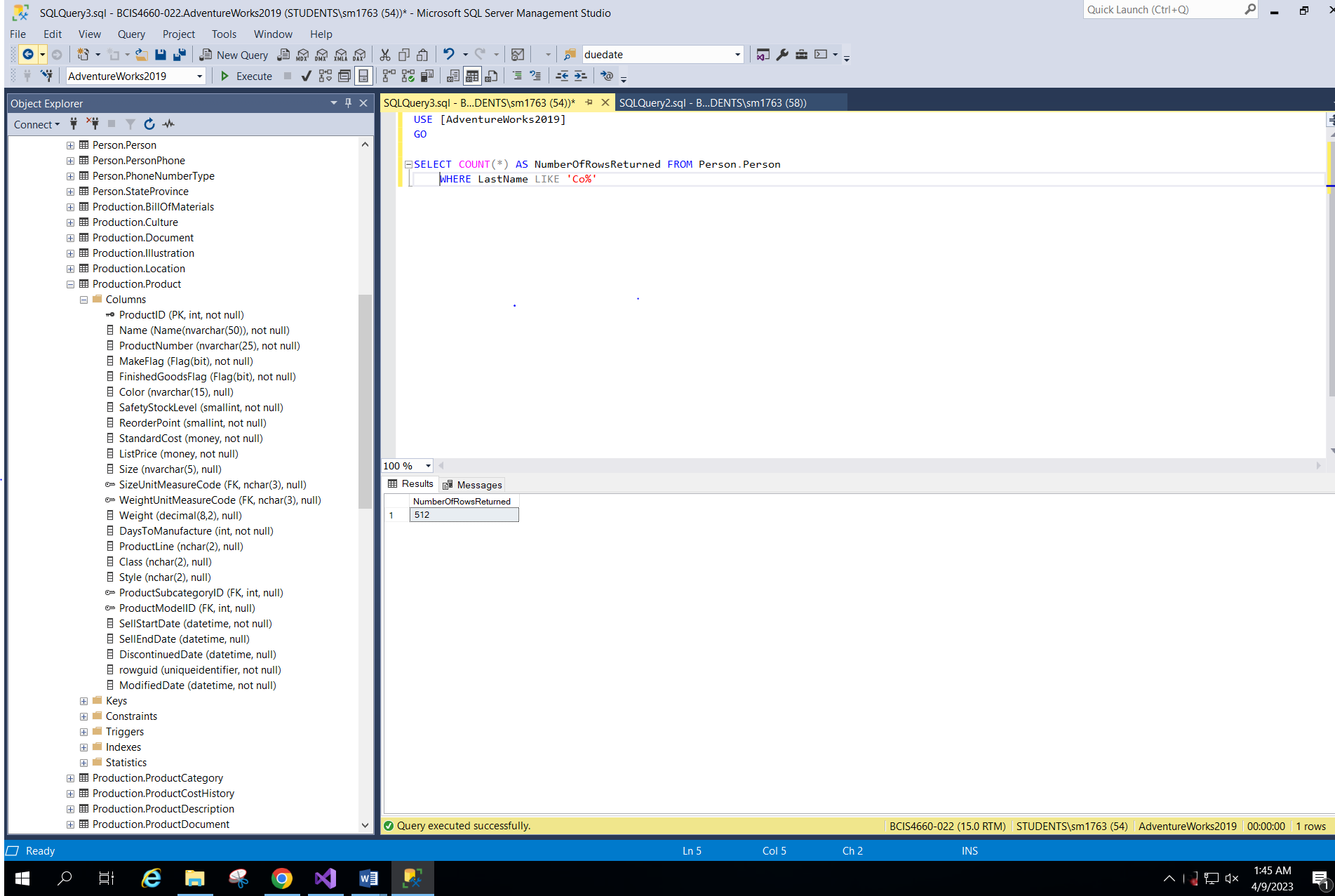


SS 7: Highest ListPrice in AdventureWorks2019

Executed in SSMS using t-sql



**SS 8: LastNames in Person.Person table that start with the letters ‘Co’**



**SS 9: Showing the total number of rows displayed with the above SS 8 SQL**

Number of rows returned = 512