

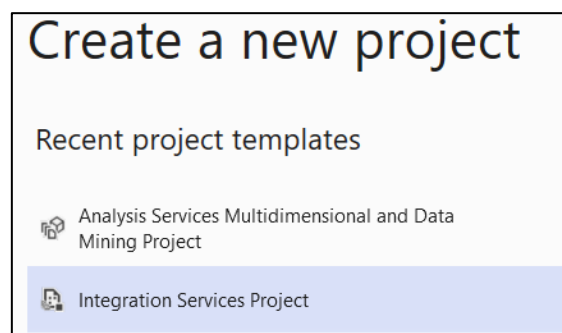
Lab Week 7

A. Sort Transformation in SSIS

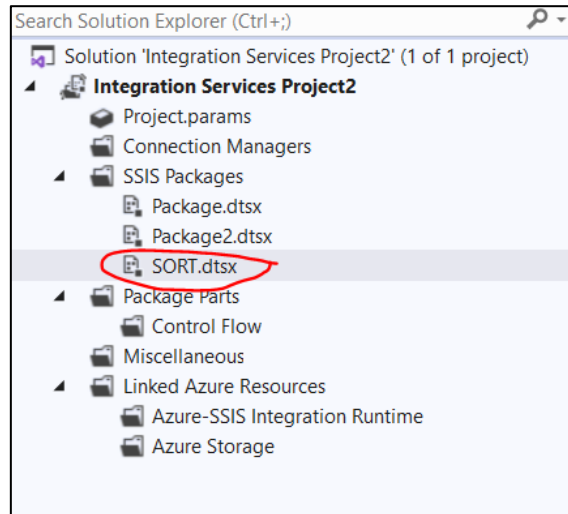
The Sort Transformation in SSIS is used to sort the source data in either Ascending or Descending order, which is similar to the SQL command ORDER BY statement.

Some transformations like Merge Transformation and Merge Join Transformation needs data to sort before using them. In these situations, we use SSIS Sort Transformation to sort the data.

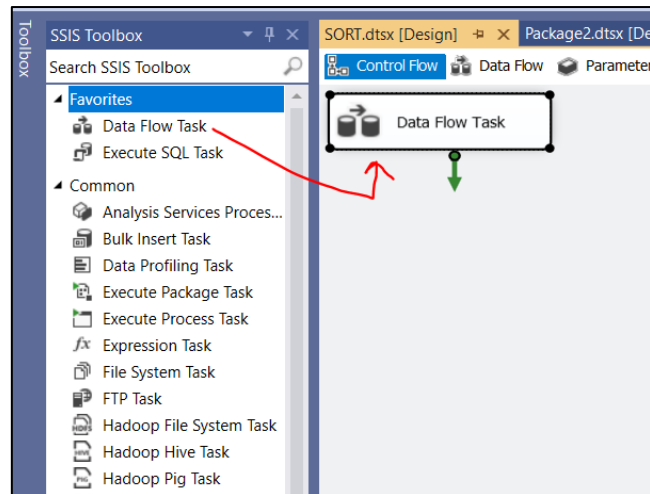
STEP 1: start SSIS and create a new SSIS project



Rename the SSIS package as SORT.dtsx

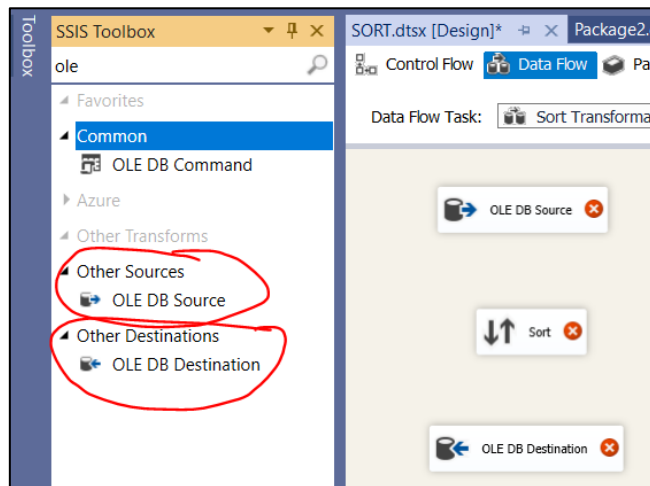


Drag and drop the data flow task from the toolbox to control flow and rename it as Sort Transformation.

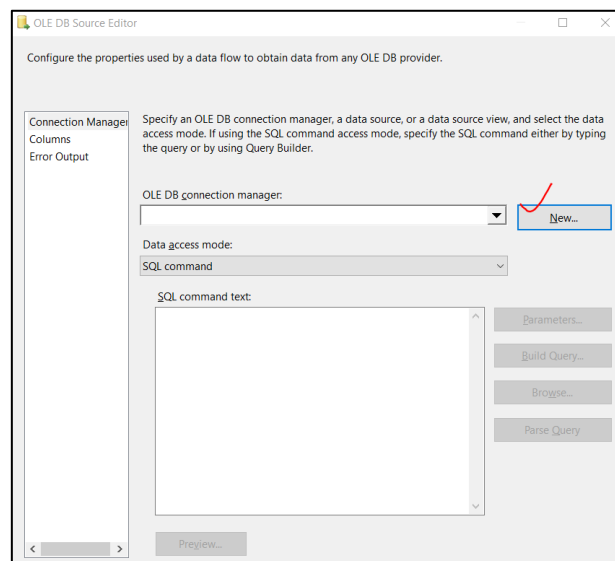


Double click on it, and it will open the data flow tab.

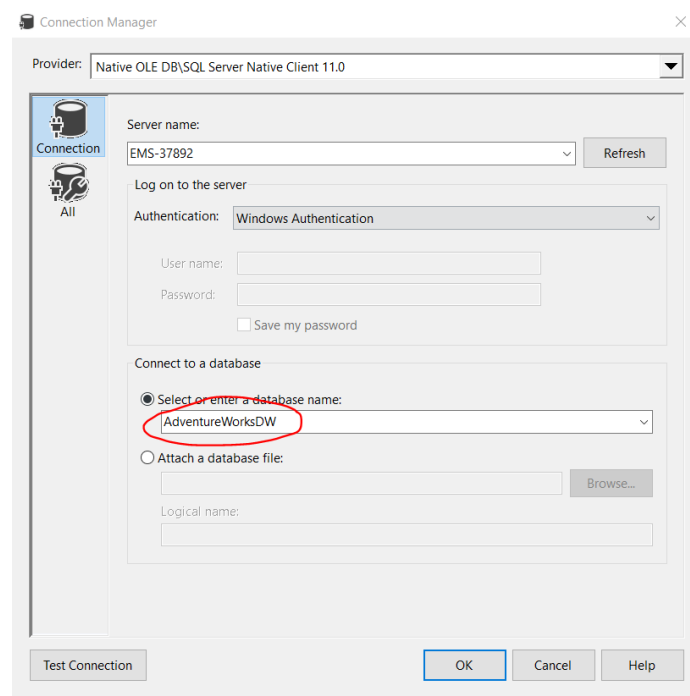
STEP 2: Drag and drop OLE DB Source, Sort Transformation, and OLE DB Destination from the toolbox to the data flow region.



STEP 3: Double click on OLE DB source in the data flow region will open the connection manager settings and provides space to write our SQL statement.



Click **New...** , **New...** and fill the server information. Choose AdventureWorksDW as DB name.



Now, select “Data access mode” as SQL command and write the below SQL commands into the SQL command text.

```
SELECT [Color]
,[EnglishProductName]
,[ListPrice]
,[DealerPrice]
,[EnglishDescription]
,[StartDate]
,[EndDate]
FROM [AdventureWorksDW].[dbo].[DimProduct]
WHERE [DealerPrice] IS NOT NULL
```

OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

Connection Manager
Columns
Error Output

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

OLE DB connection manager:
EMS-37892.AdventureWorksDW

Data access mode:
SQL command

SQL command text:

```
SELECT [Color]
      ,[EnglishProductName]
      ,[ListPrice]
      ,[DealerPrice]
      ,[EnglishDescription]
      ,[StartDate]
      ,[EndDate]
FROM [AdventureWorksDW].[dbo].[DimProduct]
WHERE [DealerPrice] IS NOT NULL
```

Parameters...
Build Query...
Browse...
Parse Query

Preview...

OK Cancel Help

Click Preview to select the top 200 records as below:

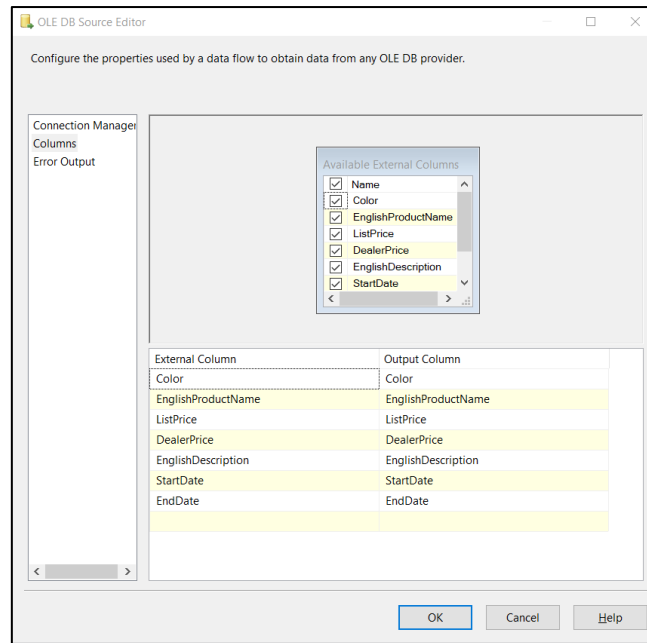
Preview Query Results

Query result (up to the first 200 rows):

Color	EnglishP...	ListPrice	DealerPr...	English...	StartDate
Red	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Red	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Red	Sport-1...	34.99	20.994	Univers...	1/07/2...
Black	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Black	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Black	Sport-1...	34.99	20.994	Univers...	1/07/2...
White	Mountai...	9.5	5.7	Combin...	1/07/2...
White	Mountai...	9.5	5.7	Combin...	1/07/2...
Blue	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Blue	Sport-1...	33.6442	20.1865	Univers...	1/07/2...
Blue	Sport-1...	34.99	20.994	Univers...	1/07/2...
Multi	AWC L...	8.6442	5.1865	Traditio...	1/07/2...
Multi	AWC L...	8.6442	5.1865	Traditio...	1/07/2...
Multi	AWC L...	8.99	5.394	Traditio...	1/07/2...
Multi	Long-Sl...	48.0673	28.8404	Unisex L...	1/07/2...

Close

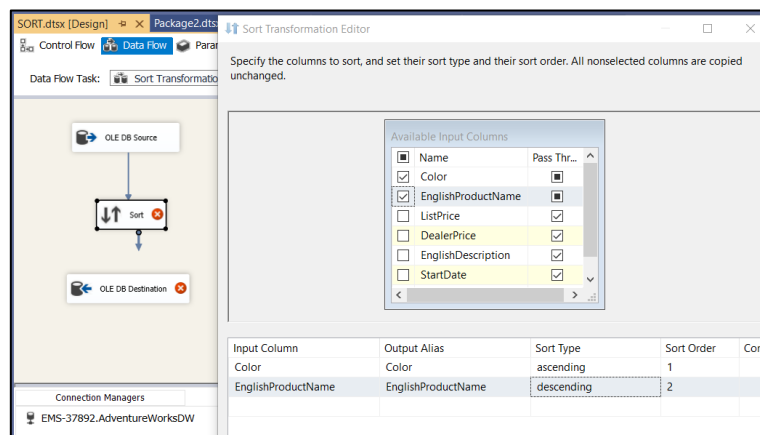
STEP 4: Click on columns tab to verify the columns. In this tab we can uncheck the unwanted columns also.



Click ok and drag the blue arrow from the OLE DB source to Sort Transformation to perform transformations (sorting) on the source data.

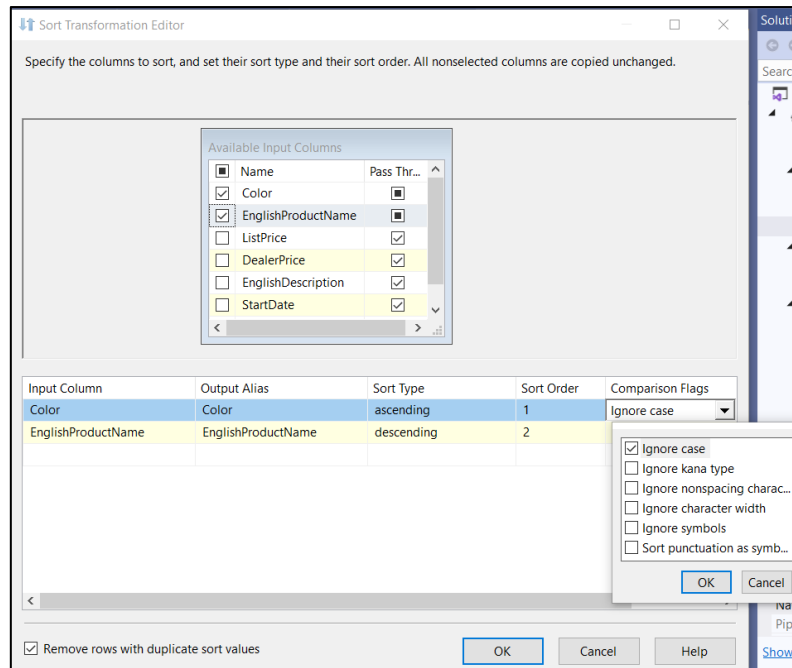
STEP 5: Double click on SSIS Sort Transformation to configure it.

Check the columns we want to sort, and don't forget to Pass Through the remaining column. If you forget to select the Pass Through then, they won't appear in output columns.



To understand the Sort Transformation in SSIS, we are sorting one column with **Ascending** order and another with **Descending** order. From the above, we are sorting the data by Color and then by [English product name] because we specified the Sort order in that way.

- First, data is sorted by Color in Ascending Order and then
- Second, data is sorted by the English product name in Descending order.



From the above screenshot, you can observe that Sort Transformation in SSIS has one more important property called Comparison Flags.

1. **Ignore case:** Specify whether you want to differentiate between uppercase and lowercase letters. If we check this option, then both XYZ is the same as xyz.
2. **Ignore Kana Type:** Specify whether you want to differentiate between the Japanese language: hiragana and katakana letters. If we check this option, then it will ignore the kana Type.
3. **Ignore nonspacing characters:** If you don't want to differentiate between the normal characters and diacritics, then check this option.
4. **Ignore Character Width:** Specify whether you want to differentiate between a single byte and a double-byte representation of the same character. If we check this option, SSIS Sort Transformation ignores the difference.
5. **Ignore Symbols:** Specify whether you want to consider the normal letters and letters with symbols (such as white spaces, currency symbols, operators, etc.) as same or not. If we check this option, both %xyz is the same as xyz.
6. **Sort punctuation as symbols:** If we check mark this option, all the punctuation symbols except the hyphen and apostrophe sorted before the actual letters. For instance, SSIS Sort Transformation will sort ? xyz before x.

Remove rows with duplicate sort values: If you checkmark this option then, Sort Transformation will remove the duplicate columns. If not, then this transformation will copy all the columns, including duplicate rows.

STEP 6: Drag the blue arrow from Sort Transformation to OLE DB Destination.

Create a new Database `AdventureWorksDWHNew` and create a table `DimProductDest` using the below commands. (Copy paste in SSMS new Query tab and execute after SQLCMDMODE on)

```

:setvar DatabaseName "AdventureWorksDWNew"

-- *****
-- Drop Database
-- *****
PRINT '';
PRINT '*** Dropping Database';
GO

IF EXISTS (SELECT [name] FROM [master].[sys].[databases] WHERE [name] =
N'$(DatabaseName)')
    DROP DATABASE $(DatabaseName);

-- If the database has any other open connections close the network connection.
IF @@ERROR = 3702
    RAISERROR('$(DatabaseName) database cannot be dropped because there are still
other open connections', 127, 127) WITH NOWAIT, LOG;
GO

-- *****
-- Create Database
-- *****
PRINT '';
PRINT '*** Creating Database';
GO

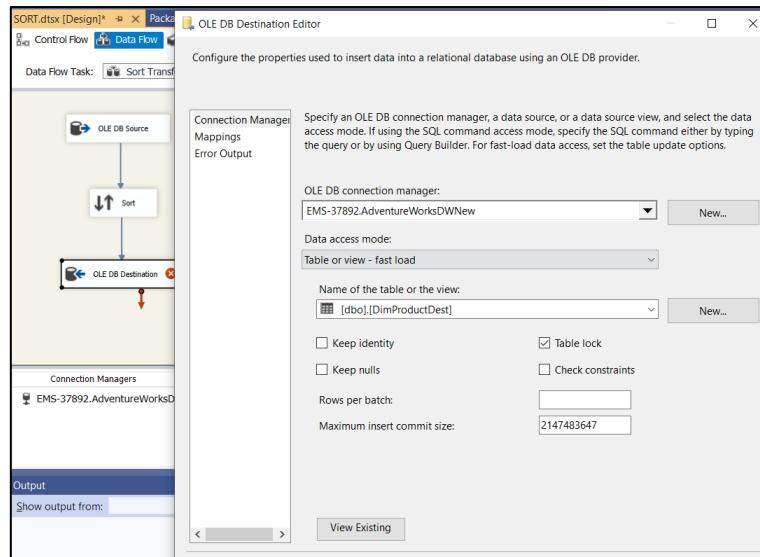
CREATE DATABASE $(DatabaseName);
GO

USE $(DatabaseName);
GO

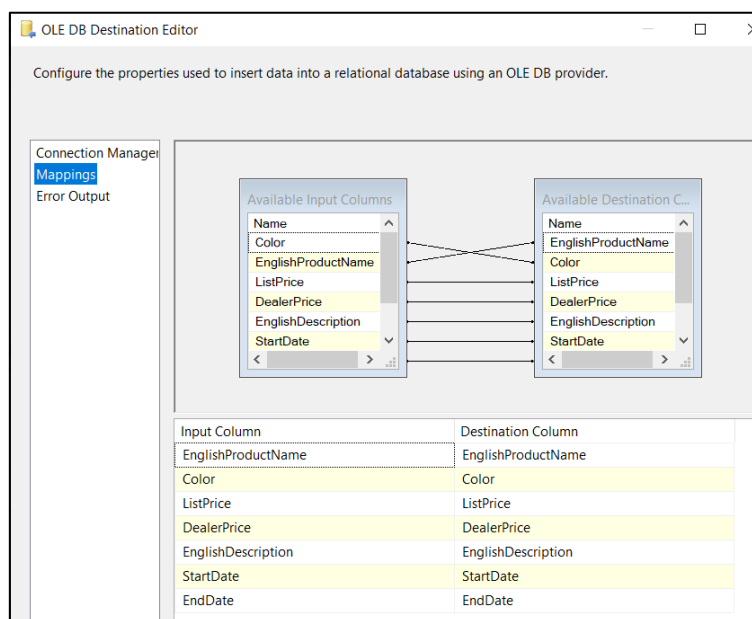
CREATE TABLE [dbo].[DimProductDest](
    [EnglishProductName] [nvarchar](50) NOT NULL,
    [Color] [nvarchar](15) NOT NULL,
    [ListPrice] [money] NULL,
    [DealerPrice] [money] NULL,
    [EnglishDescription] [nvarchar](400) NULL,
    [StartDate] [datetime] NULL,
    [EndDate] [datetime] NULL
) ON [PRIMARY];
GO

```

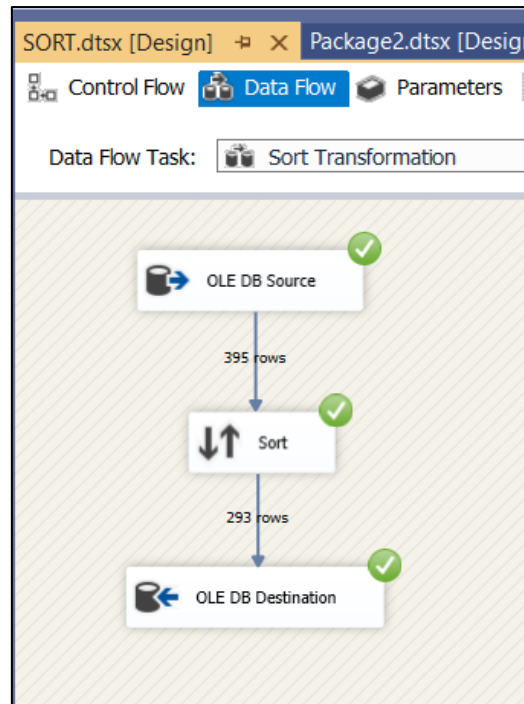
Now we have to provide the Server, database, and table details of the destination. So double-click on the OLE DB Destination and provide the required information.



STEP 7: Click on the Mappings tab to check whether the source columns exactly mapped to the destination columns.



Click ok to finish designing the SSIS Sort Transformation package. Let us run the package (Press F5, or Debug >> Start Debugging)



Let us open the SQL Server Management Studio and check the Order by results

```
SELECT TOP 1000 [Color]
, [EnglishProductName] AS ProductName
, [ListPrice]
, [DealerPrice]
, [EnglishDescription] AS Description
, [StartDate]
, [EndDate]
FROM [AdventureWorksDWNew].[dbo].[DimProductDest]
```

SQLQuery2.sql - E...MS-37892\Nur (53))*

SQLQuery1.sql - E...MS-37892\Nur (52))*

instawdbdw.sql - E...MS-37892\Nur (55))

```
SELECT TOP 1000 [Color]
, [EnglishProductName] AS ProductName
, [ListPrice]
, [DealerPrice]
, [EnglishDescription] AS Description
, [StartDate]
, [EndDate]

FROM [AdventureWorksDWNew].[dbo].[DimProductDest]
```

100 %

Results

Messages

	Color	ProductName	ListPrice	DealerPrice	Description	StartDate	EndDate
1	Black	Women's Tights, S	74.99	44.994	Warm spandex tights for winter riding; seamless ch...	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
2	Black	Women's Tights, M	74.99	44.994	Warm spandex tights for winter riding; seamless ch...	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
3	Black	Women's Tights, L	74.99	44.994	Warm spandex tights for winter riding; seamless ch...	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
4	Black	Women's Mountain Shorts, S	69.99	41.994	Heavy duty, abrasion-resistant shorts feature seaml...	2013-07-01 00:00:00.000	NULL
5	Black	Women's Mountain Shorts, M	69.99	41.994	Heavy duty, abrasion-resistant shorts feature seaml...	2013-07-01 00:00:00.000	NULL
6	Black	Women's Mountain Shorts, L	69.99	41.994	Heavy duty, abrasion-resistant shorts feature seaml...	2013-07-01 00:00:00.000	NULL
7	Black	Touring Rear Wheel	245.01	147.006	Excellent aerodynamic rims guarantee a smooth ri...	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
8	Black	Touring Front Wheel	218.01	130.806	Aerodynamic rims for smooth riding.	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
9	Black	Sport-100 Helmet, Black	34.99	20.994	Universal fit, well-vented, lightweight, snap-on visor.	2013-07-01 00:00:00.000	NULL
10	Black	Road-750 Black, 58	539.99	323.994	Entry level adult bike; offers a comfortable ride cros...	2013-07-01 00:00:00.000	NULL
11	Black	Road-750 Black, 52	539.99	323.994	Entry level adult bike; offers a comfortable ride cros...	2013-07-01 00:00:00.000	NULL
12	Black	Road-750 Black, 48	539.99	323.994	Entry level adult bike; offers a comfortable ride cros...	2013-07-01 00:00:00.000	NULL
13	Black	Road-750 Black, 44	539.99	323.994	Entry level adult bike; offers a comfortable ride cros...	2013-07-01 00:00:00.000	NULL
14	Black	Road-650 Black, 62	782.99	469.794	Value-priced bike with many features of our top-of-t...	2012-07-01 00:00:00.000	2008-12-27 00:00:00.000
15	Black	Road-650 Black, 60	699.0982	419.4589	Value-priced bike with many features of our top-of-t...	2011-07-01 00:00:00.000	2007-12-28 00:00:00.000
16	Black	Road-650 Black, 58	699.0982	419.4589	Value-priced bike with many features of our top-of-t...	2011-07-01 00:00:00.000	2007-12-28 00:00:00.000
17	Black	Road-650 Black, 52	699.0982	419.4589	Value-priced bike with many features of our top-of-t...	2011-07-01 00:00:00.000	2007-12-28 00:00:00.000

B. Data Profiling in SSIS

The Data Profiling Task in SSIS (SQL Server Integration Service) is used to compute various profiles that help us to become familiar with the data source and to identify the problems in the data (if any) that have to be fixed. Today's lab will demonstrate how to profile the source data using the Data Profiling Task in SSIS with an example.

Among the various applications of SQL Server Integration Services (SSIS), one of the more common is loading a data warehouse or data mart. SSIS provides the extract, transform, and load (ETL) features and functionality to efficiently handle many of the tasks required when dealing with transactional source data that will be extracted and loaded into a data mart, a centralized data warehouse, or even a master data management repository, including the capabilities to process data from the relational data warehouse into SQL Server Analysis Services (SSAS) cubes. SSIS provides all the essential elements

Note: The Data Profiling Task in SSIS is associated with the data available in SQL Server. The SSIS Data Profiling Task doesn't support the data present in the file system.

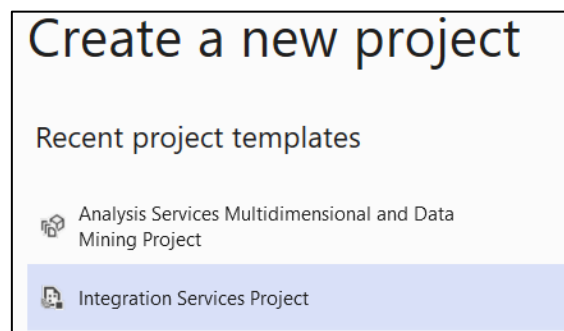
Why Data Profiling is important in DW designing?

Let's assume that we have a table called **customers** which has information on my customers containing the email address, zip code, country, date of birth, customer ID etc. If one wants to answer the general questions such as, 'How many people live in a particular zip code?', 'What is the min and max length of emails?' All this information can easily be extracted by just profiling the data.

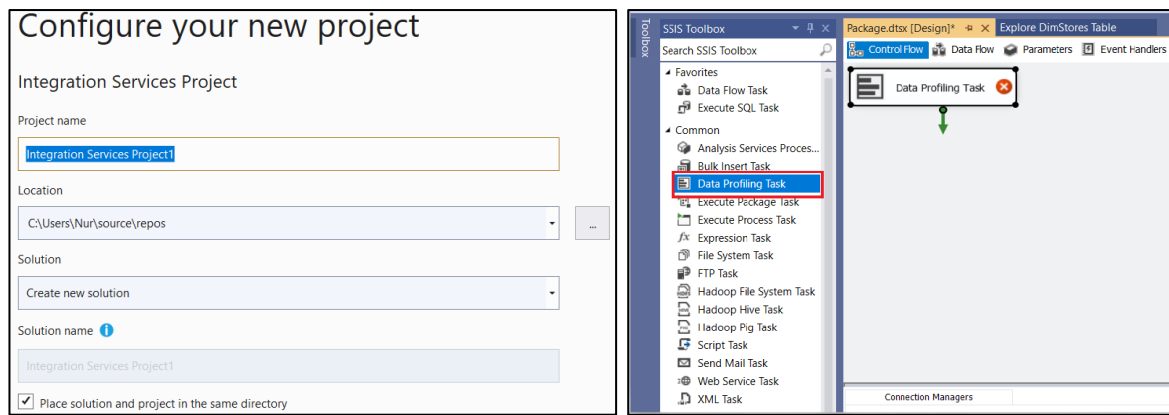
Example: if you have a requirement that when you pull the customers from a particular system, more than 90% must have their email address filled, data profiling in the stage can give you the count of nulls and you use its percentage to decide if the data is good to move to the data warehouse or not. During the designing stage of a data warehouse, you can also profile different tables to determine what will be the appropriate table structure including data types and the lengths to use in the final table.

Steps for Data Profiling in SSIS

Step 1: start SSIS and create a new SSIS project

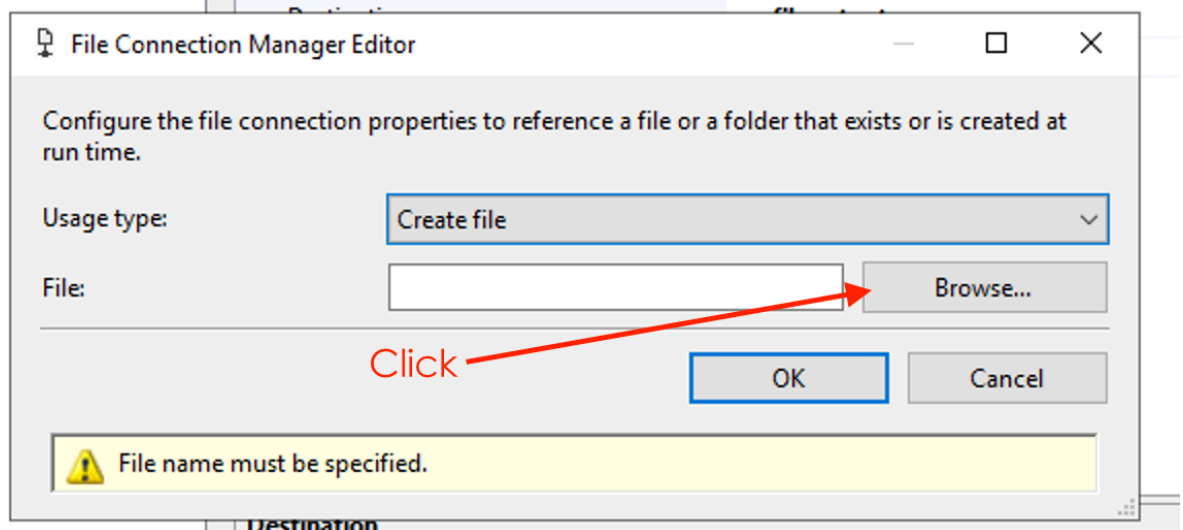


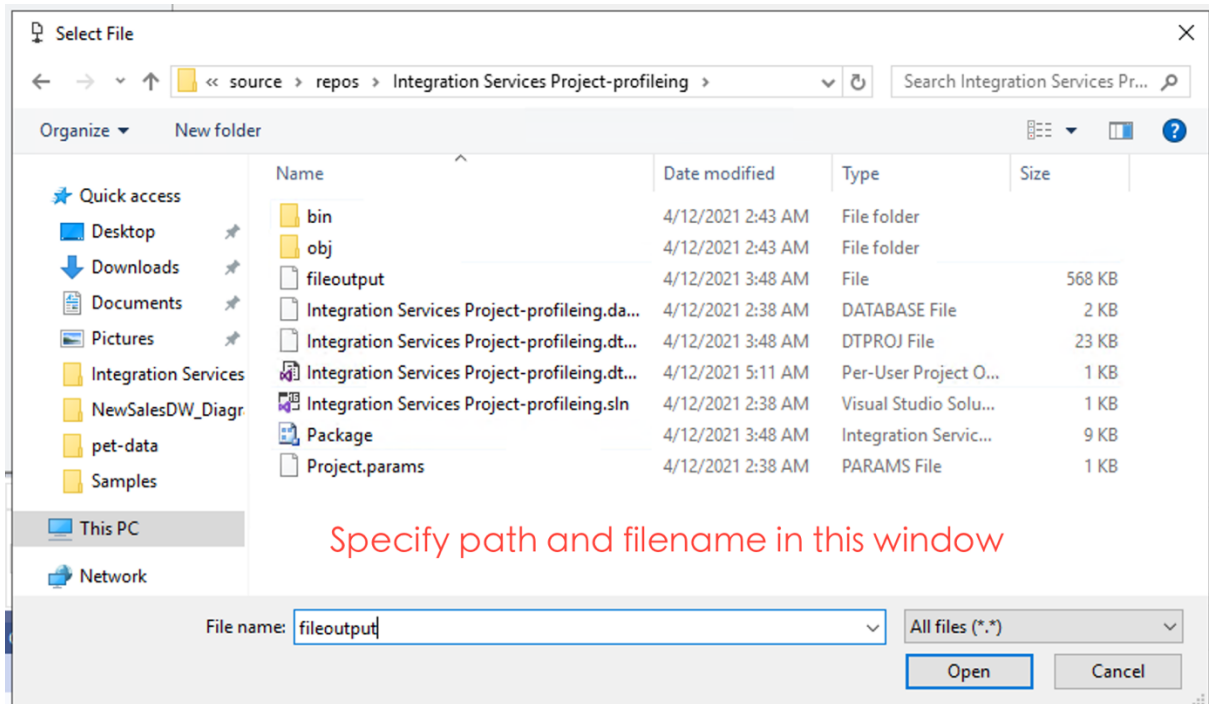
By default a new package will be created and in this package add a new data profiling task from the common section in SSIS toolbox. You should have similar screens as below.



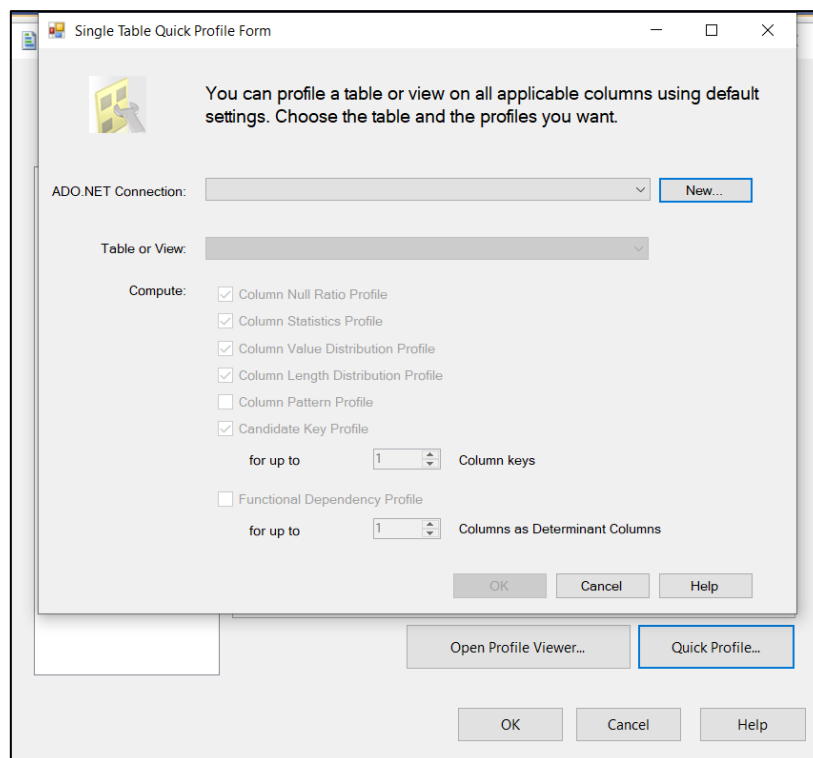
Step 2: Next click on the Data Profiling Task (in the Design Pane) see the screen shot below to config next.

- Choose FileConnection for the DestinationType
- For Destination, click and select new file connection then chose create new file and specify output path and filename by clicking on 'Browse..'
- Specify the output file path and file name. Any path or file name is fine. But please DO remember this path and file name.



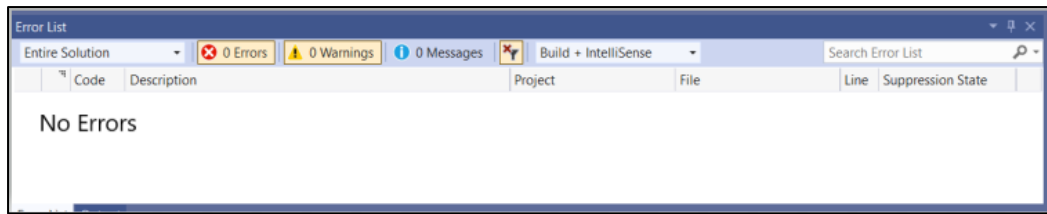


Step 3: Click the ok button to come back to the screen shot above. Next click on the **quick profile** button to go to connect to a server and perform the profiling.



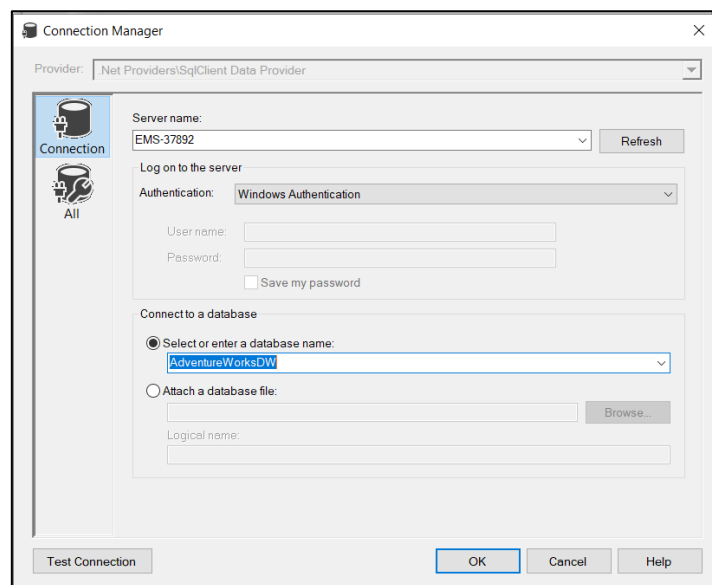
Note: SSIS data profiling allows only ado.net connection and it works only in sql server. You can only profile the data in sql server using SSIS data profiler so wherever your data is example flat file, spreadsheet, to use the SSIS data profiler, you must store it into sql server.

You also can see no error in the error tab

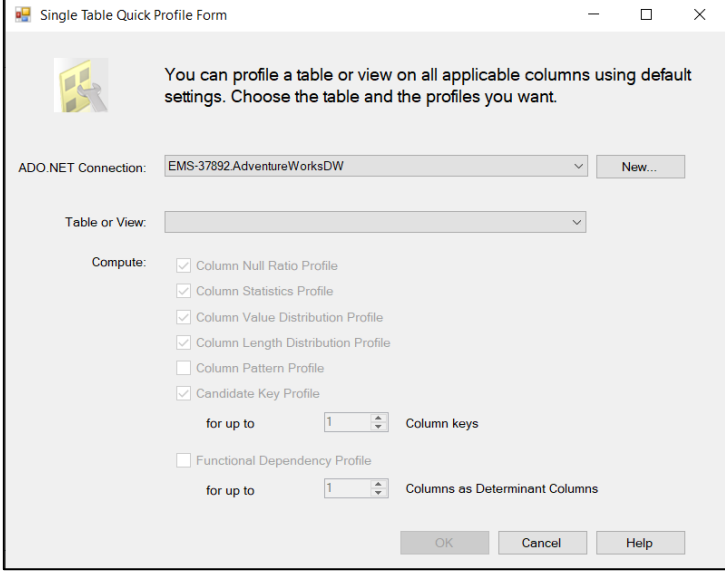


Step 4: To connect to the sql server where the data is, click on the new button in the snapshot above at Step 2.c, and enter the sql server to connect and select the database to connect and click ok. In the below example, the server name is EMS-37892 (you should have your own DB engine) and the selected database is **AdventureWorksDW**. See the snapshot below.

Note: You should have completed week-3 lab to get **AdventureWorksDW** database in your machine. And, don't forget to check "Test Connection" with the message "*Test connection succeeded.*"



Step 5: Once you are done setting the connection to the sql server and the database, click ok to go back to the profiler form configuration. You should see below screen after clicking ok.



Single Table Quick Profile Form

You can profile a table or view on all applicable columns using default settings. Choose the table and the profiles you want.

ADO.NET Connection: EMS-37892.AdventureWorksDW [New...]

Table or View: [Empty dropdown]

Compute:

- ☒ Column Null Ratio Profile
- ☒ Column Statistics Profile
- ☒ Column Value Distribution Profile
- ☒ Column Length Distribution Profile
- ☐ Column Pattern Profile
- ☒ Candidate Key Profile

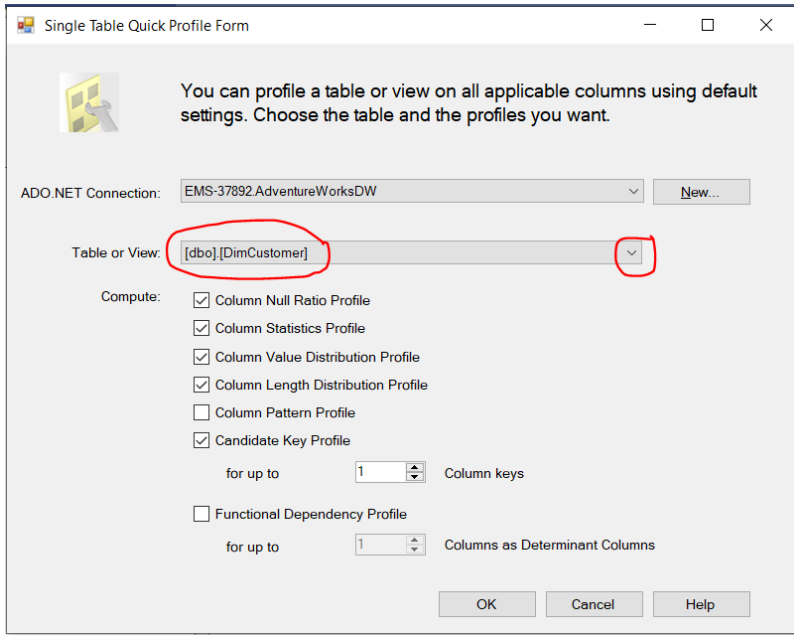
for up to 1 Column keys

☐ Functional Dependency Profile

for up to 1 Columns as Determinant Columns

[OK] [Cancel] [Help]

Step 6: Connect to the table or the view you want to profile. In this example, we will go to profile DimCustomer table in AdventureWorksDW.



Single Table Quick Profile Form

You can profile a table or view on all applicable columns using default settings. Choose the table and the profiles you want.

ADO.NET Connection: EMS-37892.AdventureWorksDW [New...]

Table or View: [dbo].[DimCustomer] [Dropdown arrow]

Compute:

- ☒ Column Null Ratio Profile
- ☒ Column Statistics Profile
- ☒ Column Value Distribution Profile
- ☒ Column Length Distribution Profile
- ☐ Column Pattern Profile
- ☒ Candidate Key Profile

for up to 1 Column keys

☐ Functional Dependency Profile

for up to 1 Columns as Determinant Columns

[OK] [Cancel] [Help]

Each of the options available in the previous figure is described below:

- i. Column Null Ratio Profile: This will go over all the columns in the table and give you **count of nulls in each of the columns** and the corresponding percentage of the null count in that column.
- ii. Column Statistics Profile: This is mostly for integers and dates profiling. For an integer it calculates the **minimum, maximum, mean and standard deviation**.
- iii. Column Value Distribution Profile: This is calculating the **uniqueness or repetition** of values in a column.

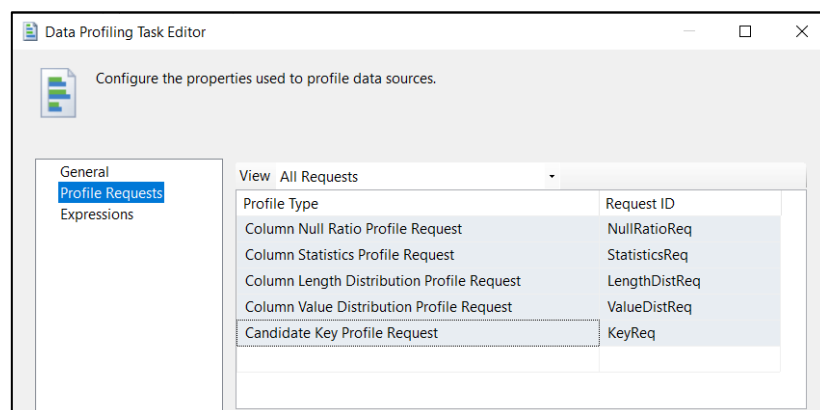
iv. Column Length Distribution Profile: the **length** of **characters** in the values in a column. It can tell you like 40% of customers have email of length 25, 10% has email of length 30 etc.

v. Column Pattern Profile: It helps you identify **problems** in your data, such as invalid strings, and can suggest **regular expressions** that can be used in the future to validate new values.

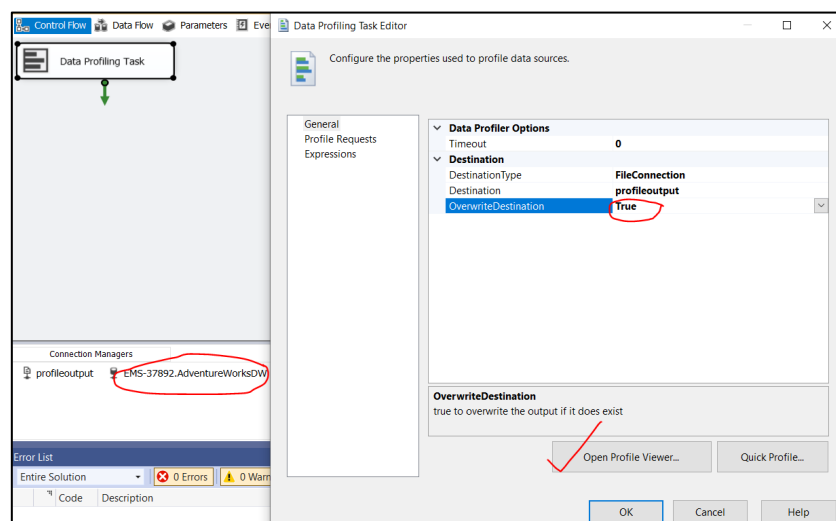
vi. Candidate Key Profile: This profiles all columns in the data and report which columns or combination of columns can be used as **primary keys**.

vii. Function Dependency Profile: This profile tells the extent to which a value in a column depends on a value in another column or set of columns.

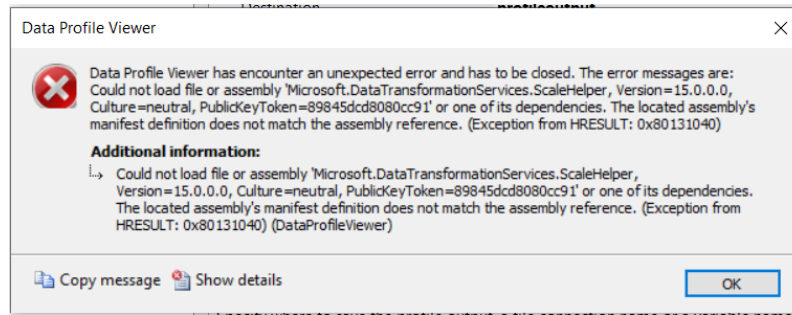
Step 7: Click ok to complete the configuration. A new page will show all the selected profiling to be performed. From here, you can go to previous steps to edit configuration.



Step 8: Now run the package and see it complete then click on the data profiler task to open it again. Select “True” against OverwriteDestination.



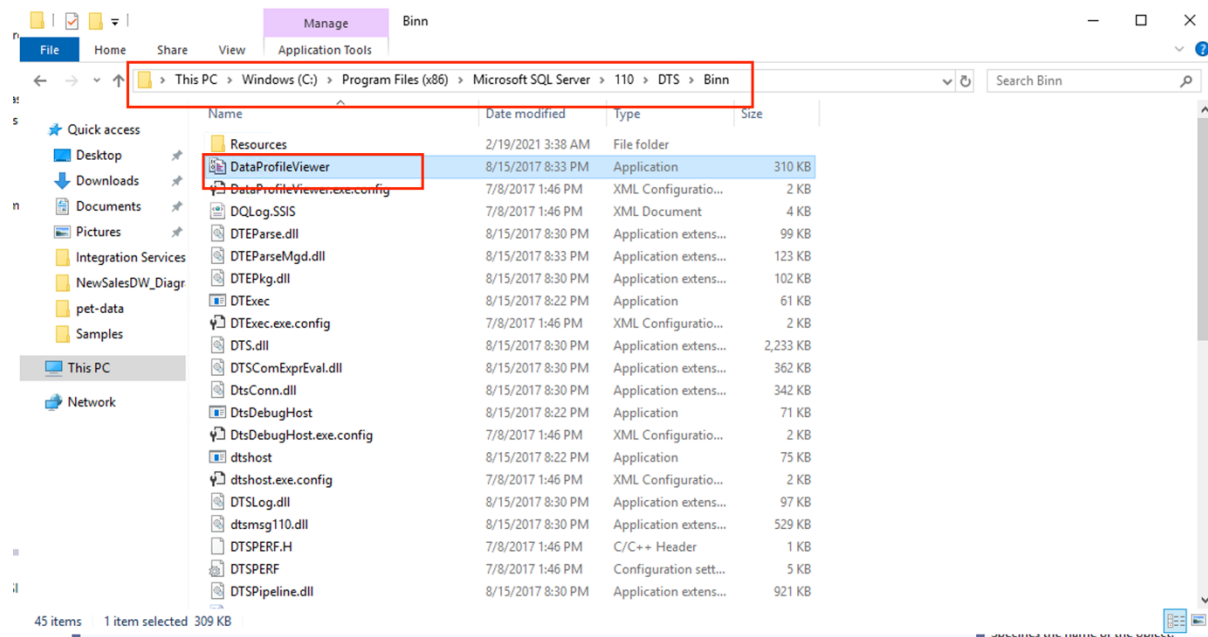
Open Profile Viewer which will open the latest profiling done as shown below. If you get some error like:



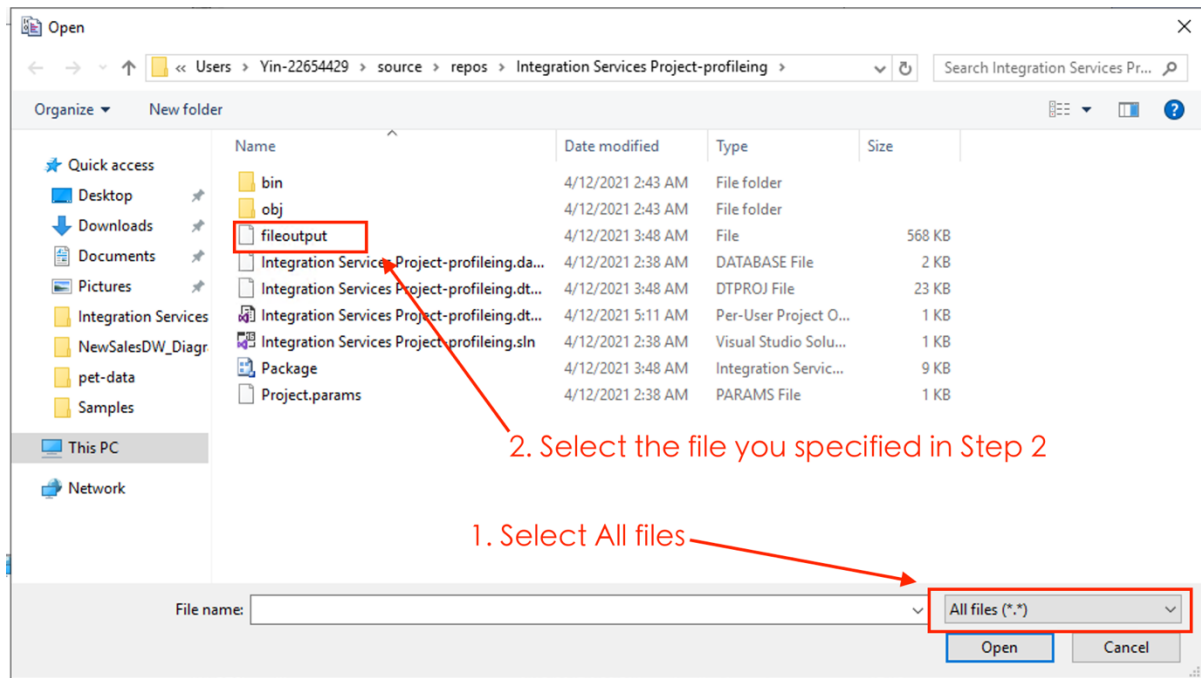
In your File Explorer, go to

C:\Program Files (x86)\Microsoft SQL Server\110\DTs\Binn

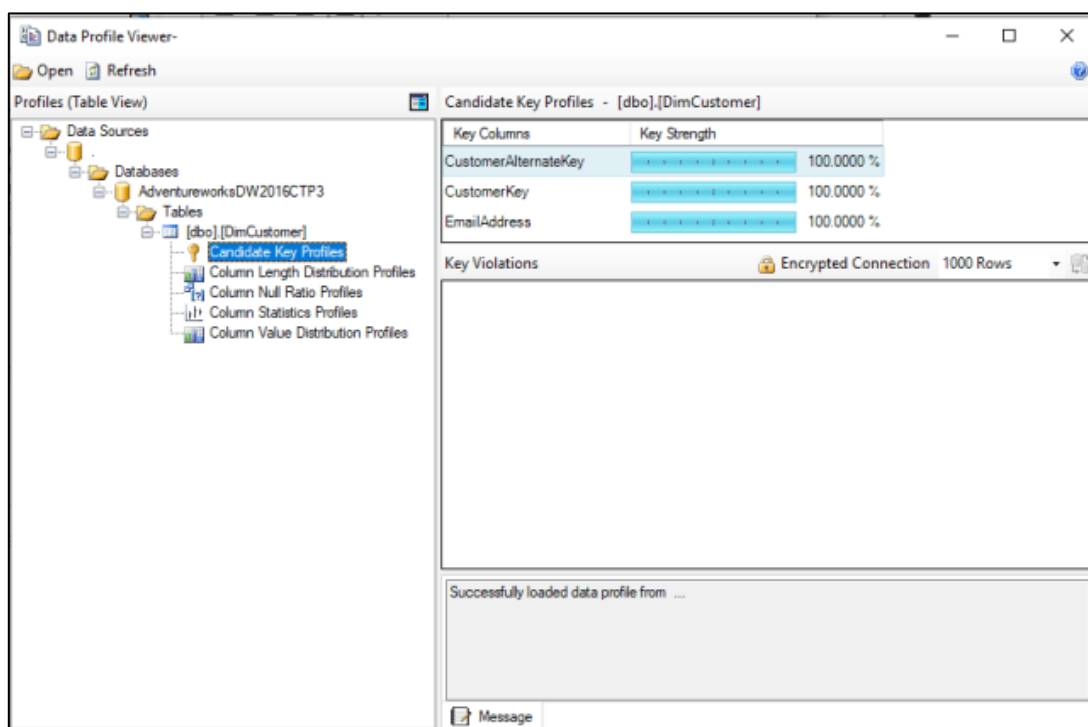
And double click on DataProfileViewer, to manually run it.



In your DataProfileViewer, click on open then select the file you specified in Step 2.

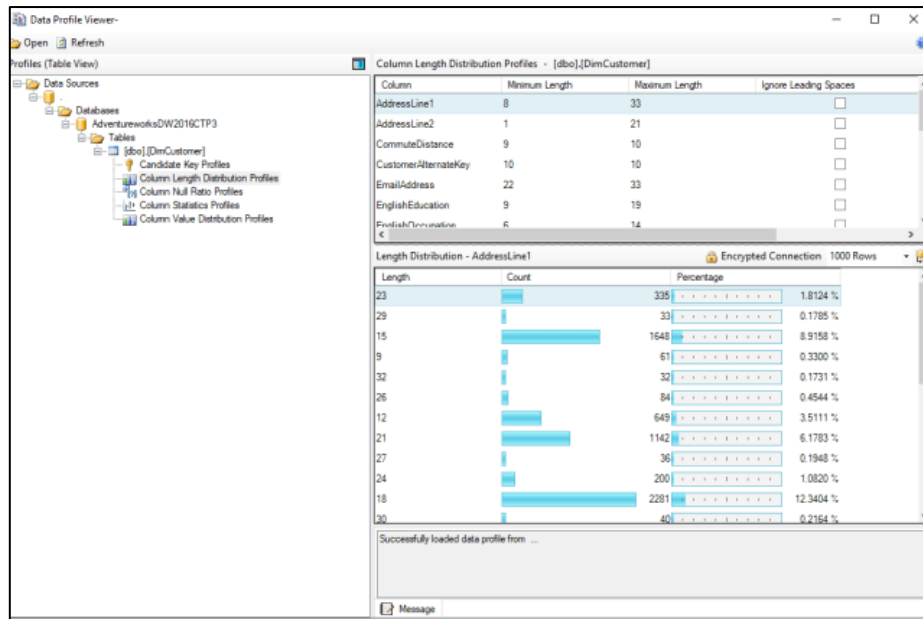


Step 9: click on each of the calculated profiles to see the details, for example the candidate key profile looks like below



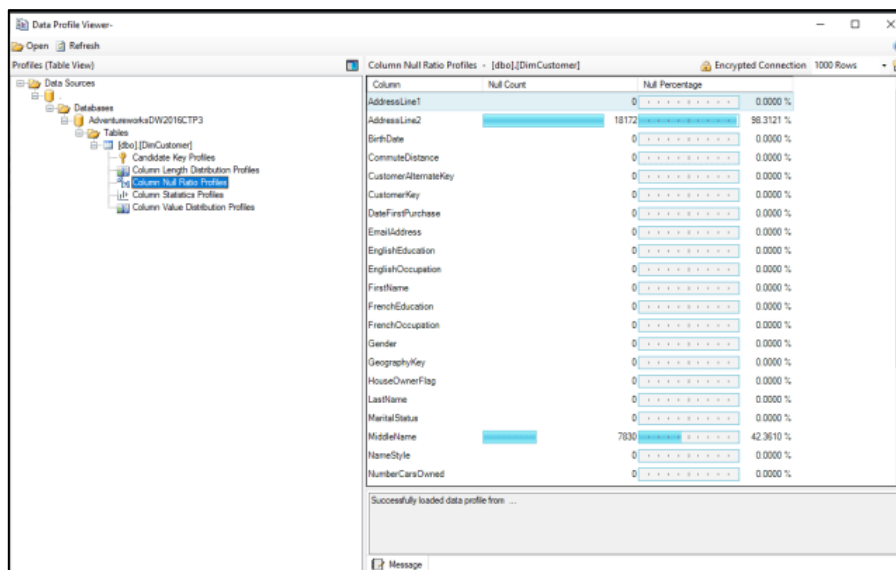
It means the CustomerAlternateKey, CustomerKey, EmailAddress all has 100% key strength to be used as primary key for the DimCustomer table.

Step 10: Look at the Length Distribution Profiles as shown below.



Addressline1 shows in the column length distribution profiles shown the minimum length is 8 and the maximum length is 33. In the length distribution for that column there are 335 values have length of 23 forming 1.8124% of values in that column.

Step 11: Check Null Ratio Profiles. There are 98.3121% of values in addressline2 with nulls and addressline1 has no null.



Step 12: Click column statistics profile. This can help you identify how the table data looks.

Column Statistics Profiles - [dbo].[DimCustomer]

Column	Minimum	Maximum	Mean	Standard Deviation
BirthDate	2/10/1916 12:0...	6/25/1986 12:0...		
CustomerKey	11000	29483	20241.5	5335.87118004174
DateFirstPurchase	12/29/2010 12:...	1/28/2014 12:0...		
GeographyKey	2	654	257.95628651807	196.5257461411
NumberCarsOwned	0	4	1.50270504219...	1.13836294670722
NumberChildrenRtHo...	0	5	1.00405756329...	1.52261846927002
TotalChildren	0	5	1.8443518718892	1.61236431793752
YearlyIncome	10000.0000	170000.0000	57305.7779701...	32284.9683495135

Successfully loaded data profile from ...

Message

Step 13: Click column distribution profile. Addressline1 had 12797 distinct values and no repeated values. Let's look at addressline2 to see how it looks below. 166 of the values are distinct but the value Verkaufsabteilung is repeated 34 times forming 0.1839% in the addressline2 column.

Column Value Distribution Profiles - [dbo].[DimCustomer]

Column	Number Of Distinct Values
AddressLine1	12797
AddressLine2	166
BirthDate	6139
CommuteDistance	5
CustomerAlternateKey	18484

Frequent Value Distribution (0.1000 %) - AddressLine2

Value	Count	Percentage
Verkaufsabteilung	34	0.1839 %

Successfully loaded data profile from ...

Message

With all these information, one can decide how to build the final schema or design an ETL to have some intelligence when the staging data did not meet the requirement to be merged into the final data warehouse.