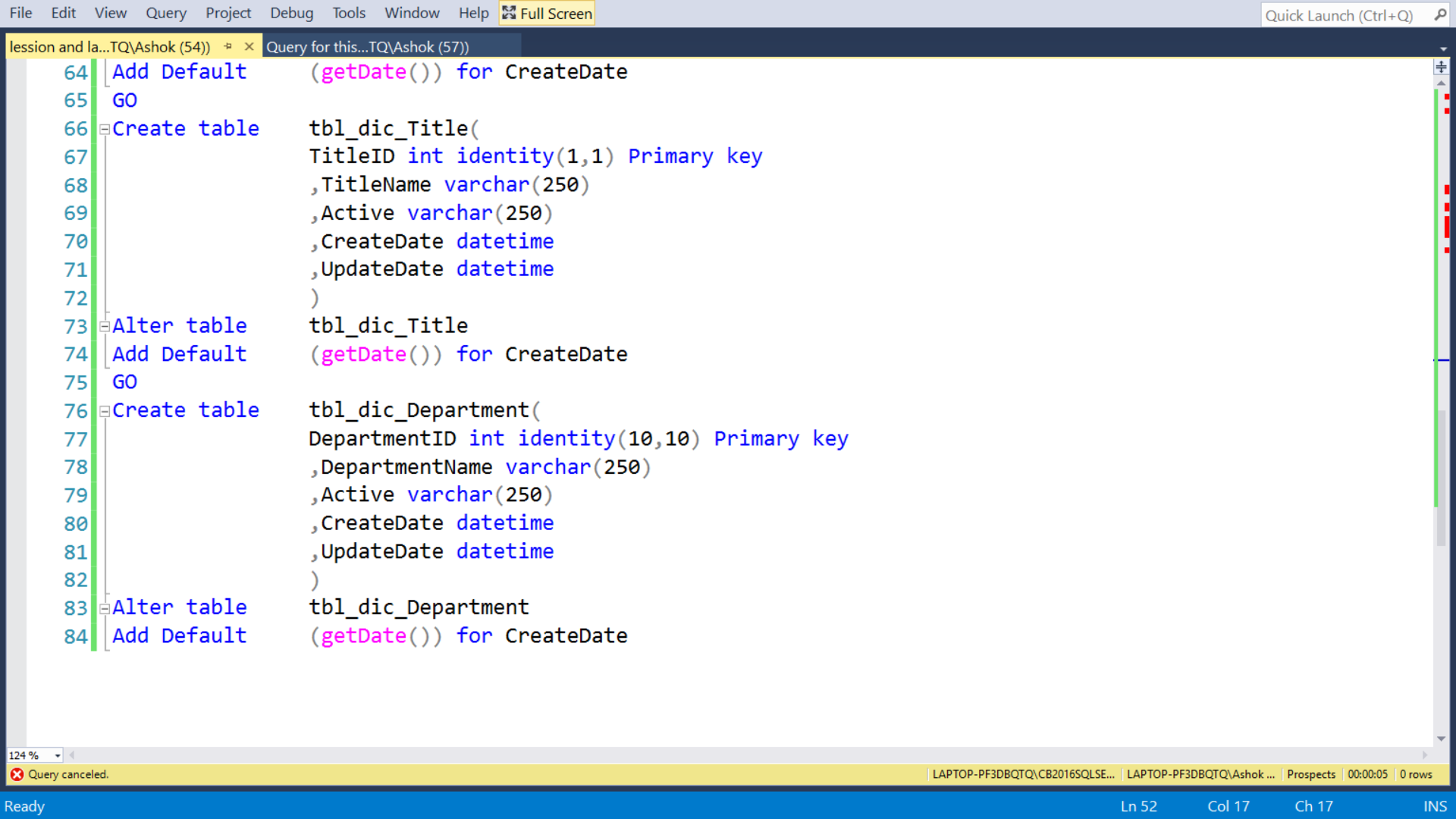
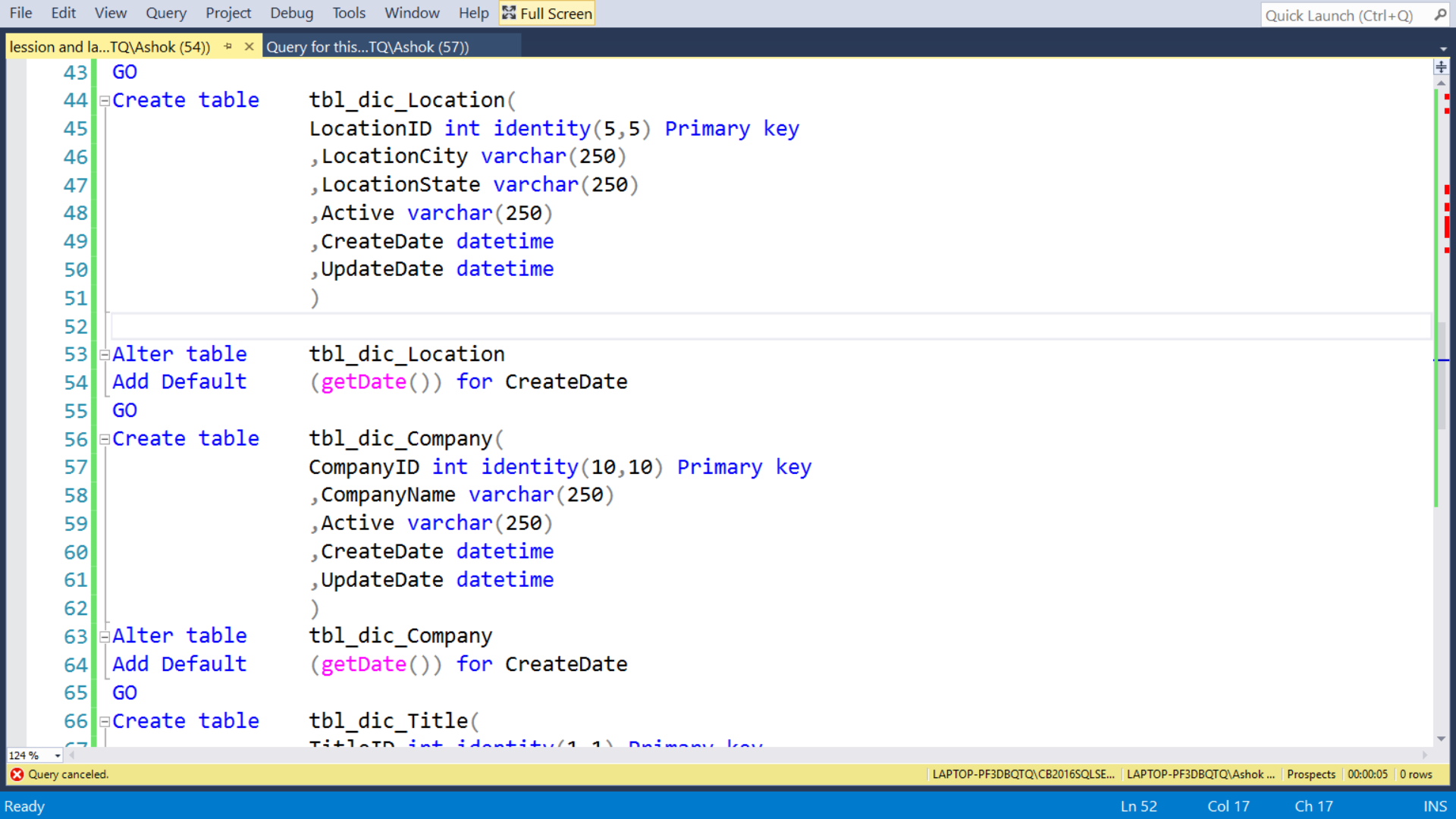
/\* Ashok

Auditing and Error Handling Lab

6.27.19\*/

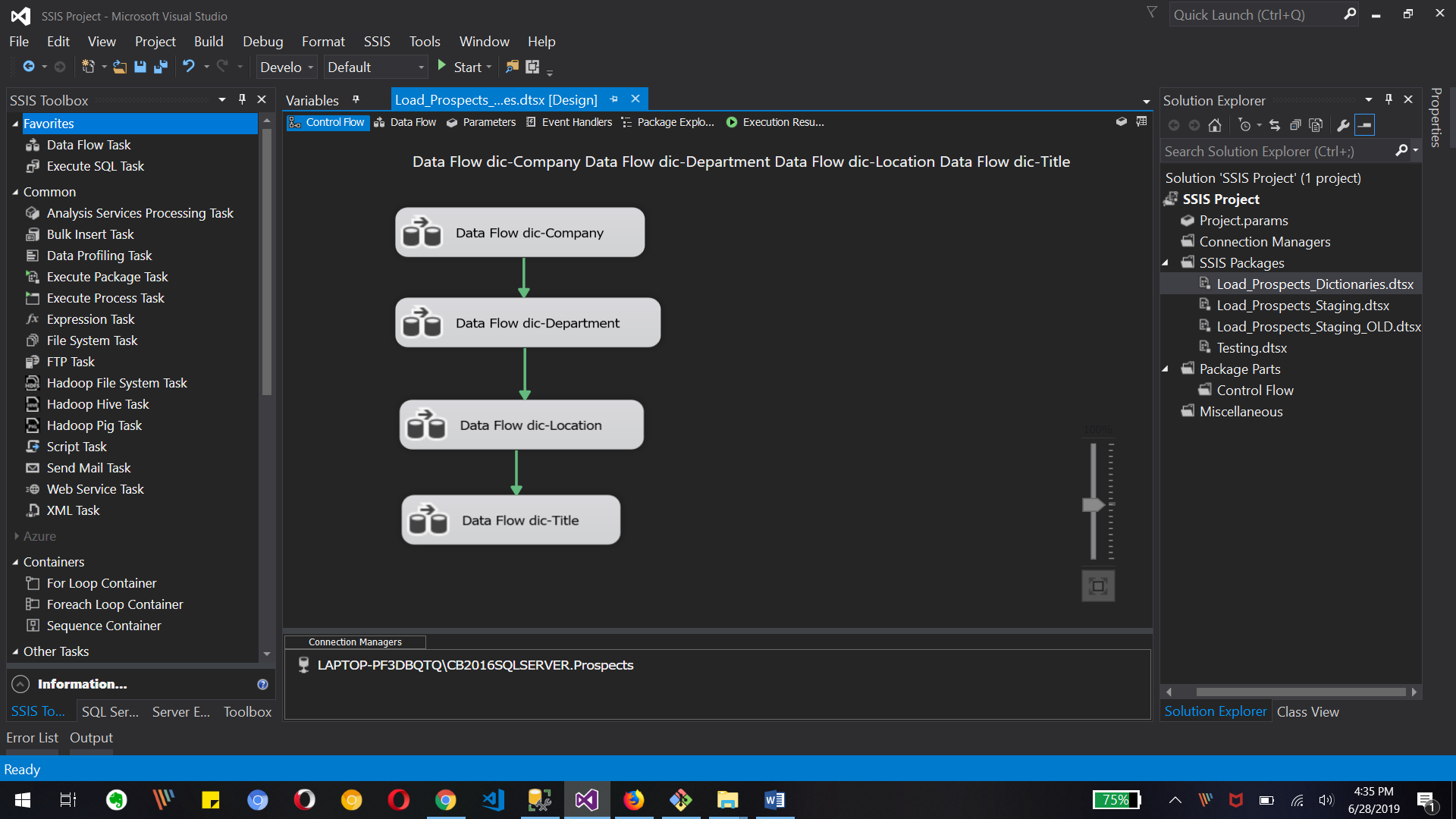
1. Create dictionary tables in Prospects database for Title, Department, Company and Location –

(Project document)

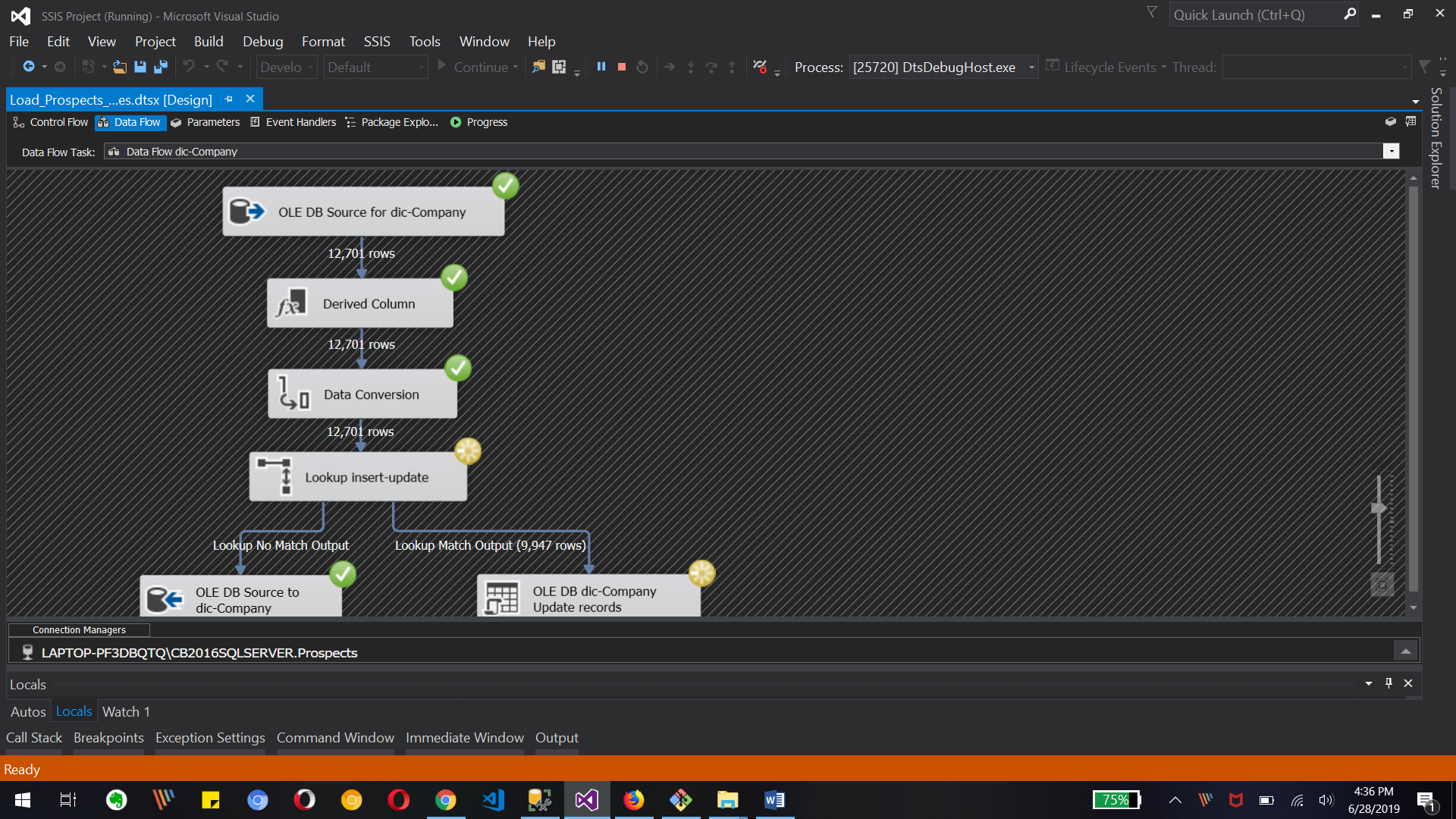


2... Create a SSIS package Load\_Prospects\_Dictionaries.dtsx to load the dictionary table (make

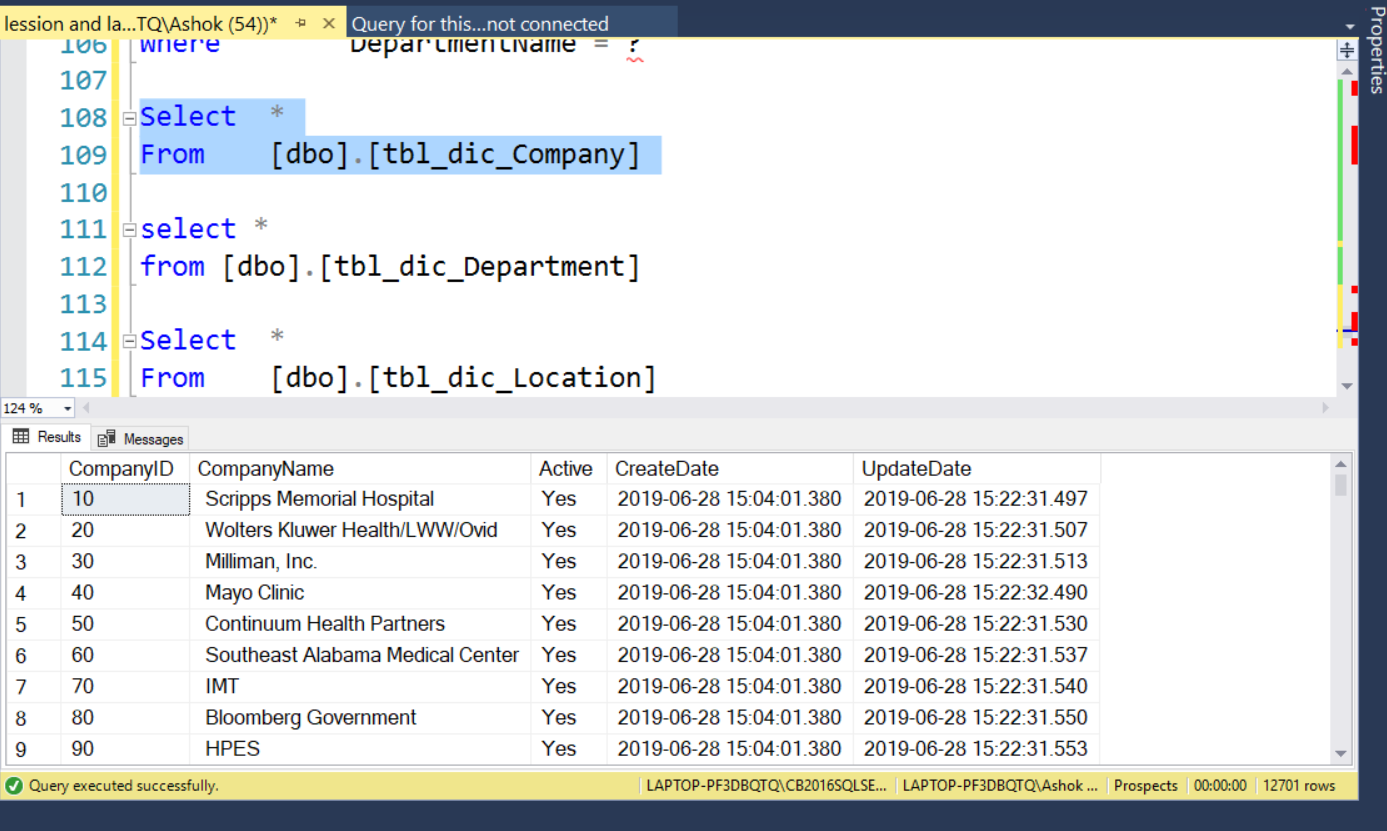
sure to lookup for records and insert/update accordingly the dictionary table)



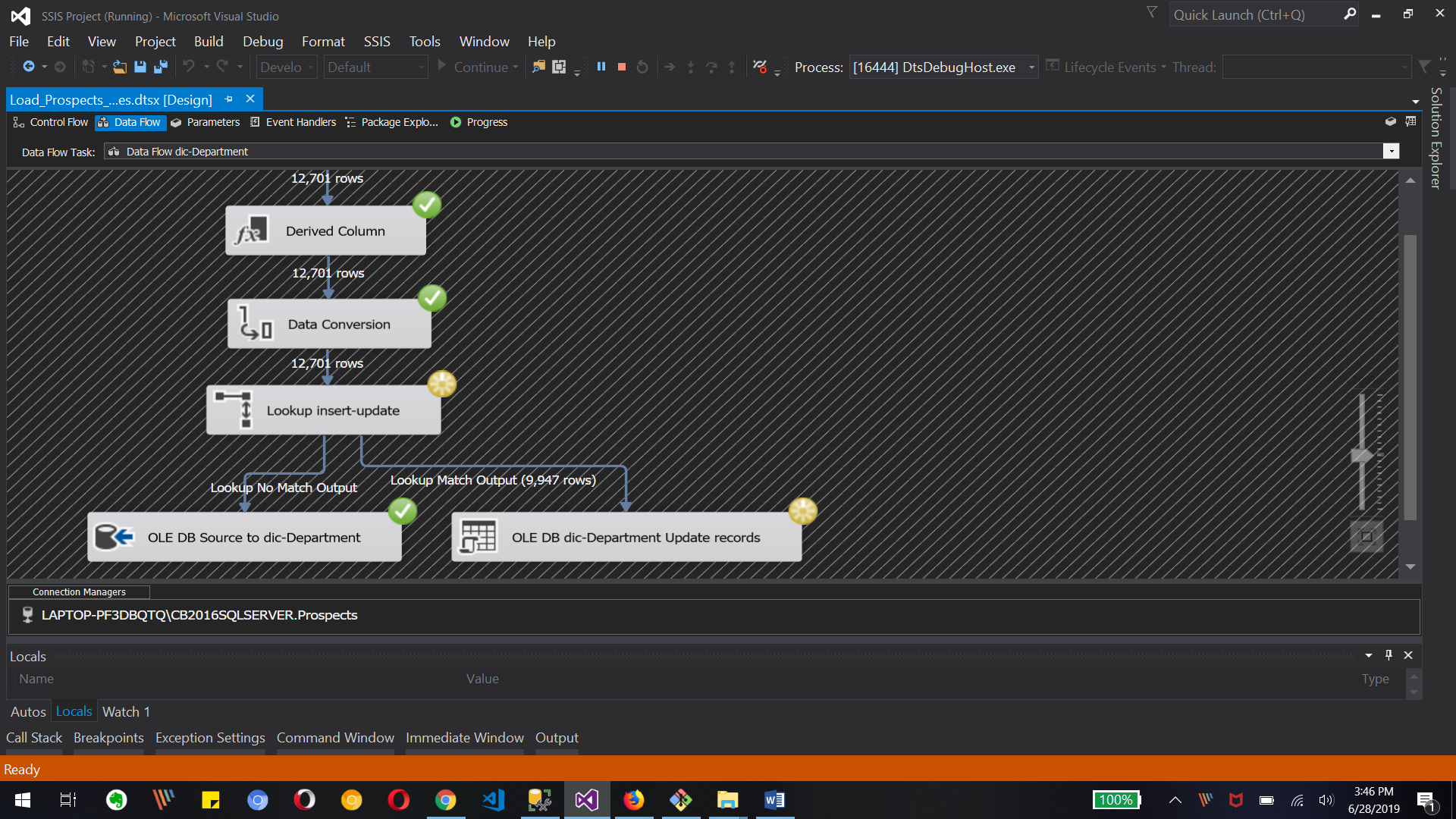
Data Flow dictionary-Company, dictionary -Department, dictionary -Location and dictionary-Title in Control flow tab



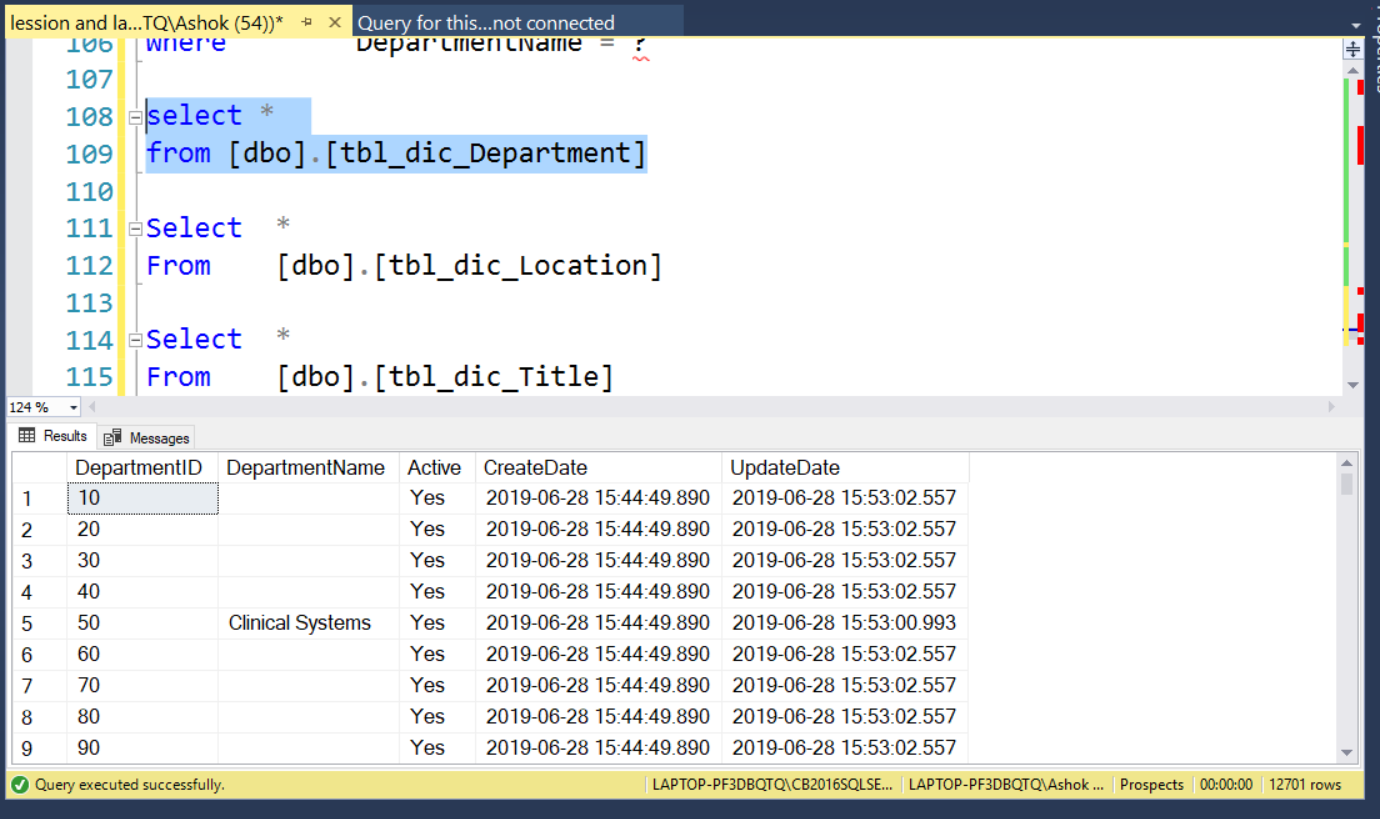
Debugging screen for dictionary-Company Data Flow



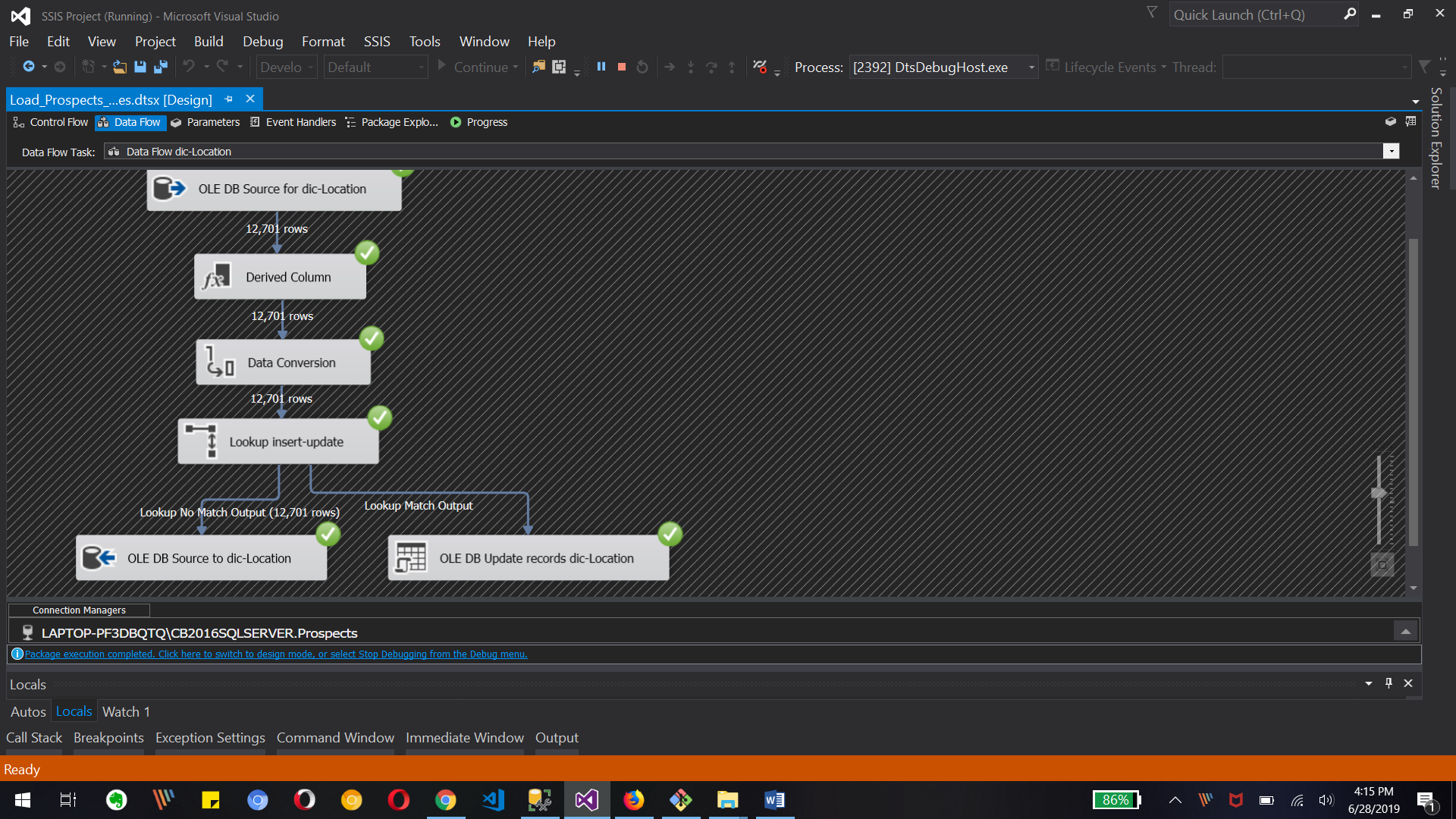
Updated records of Company table in SSMS



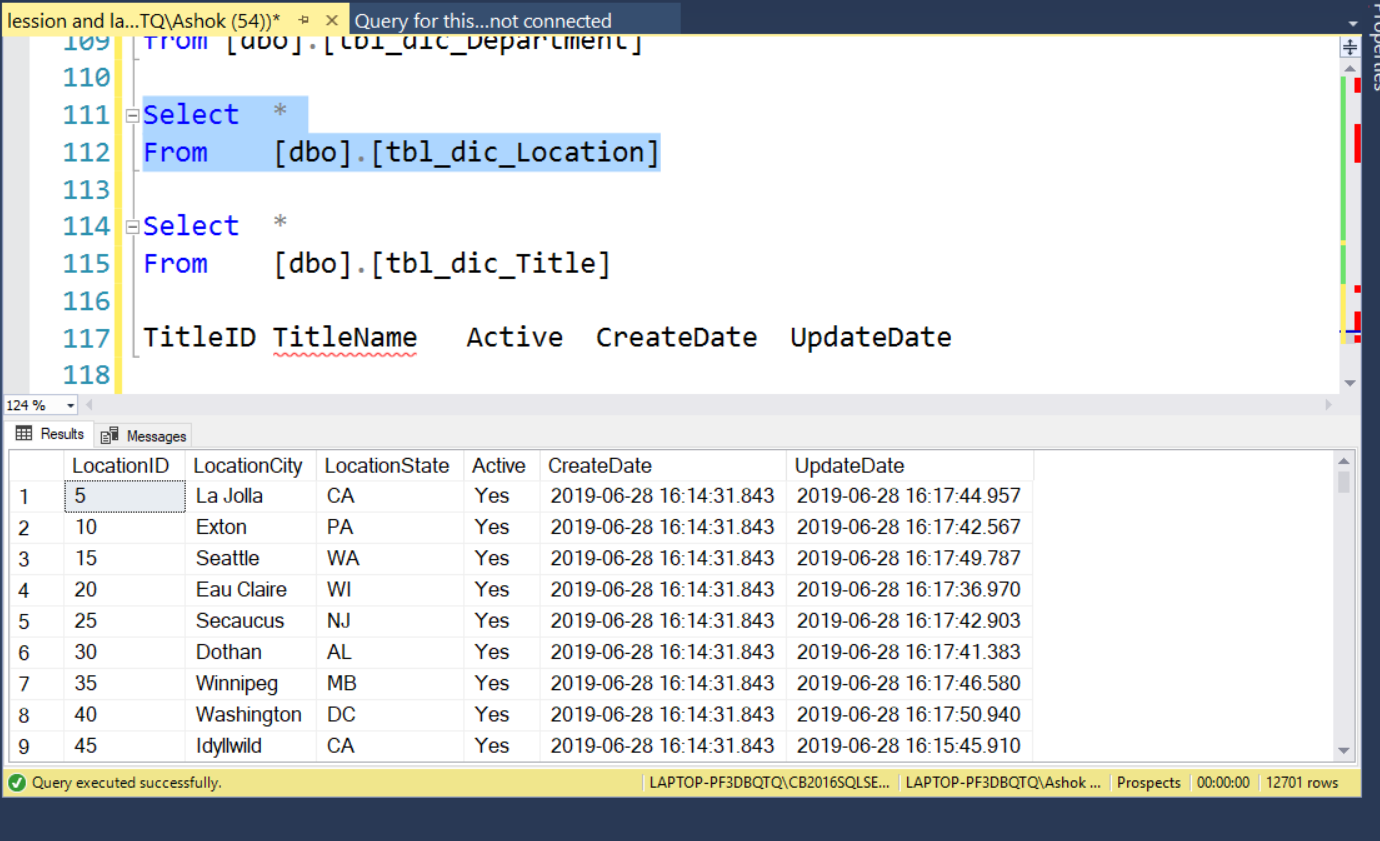
Debugging screen of dictionary-Deparment Data flow tasks



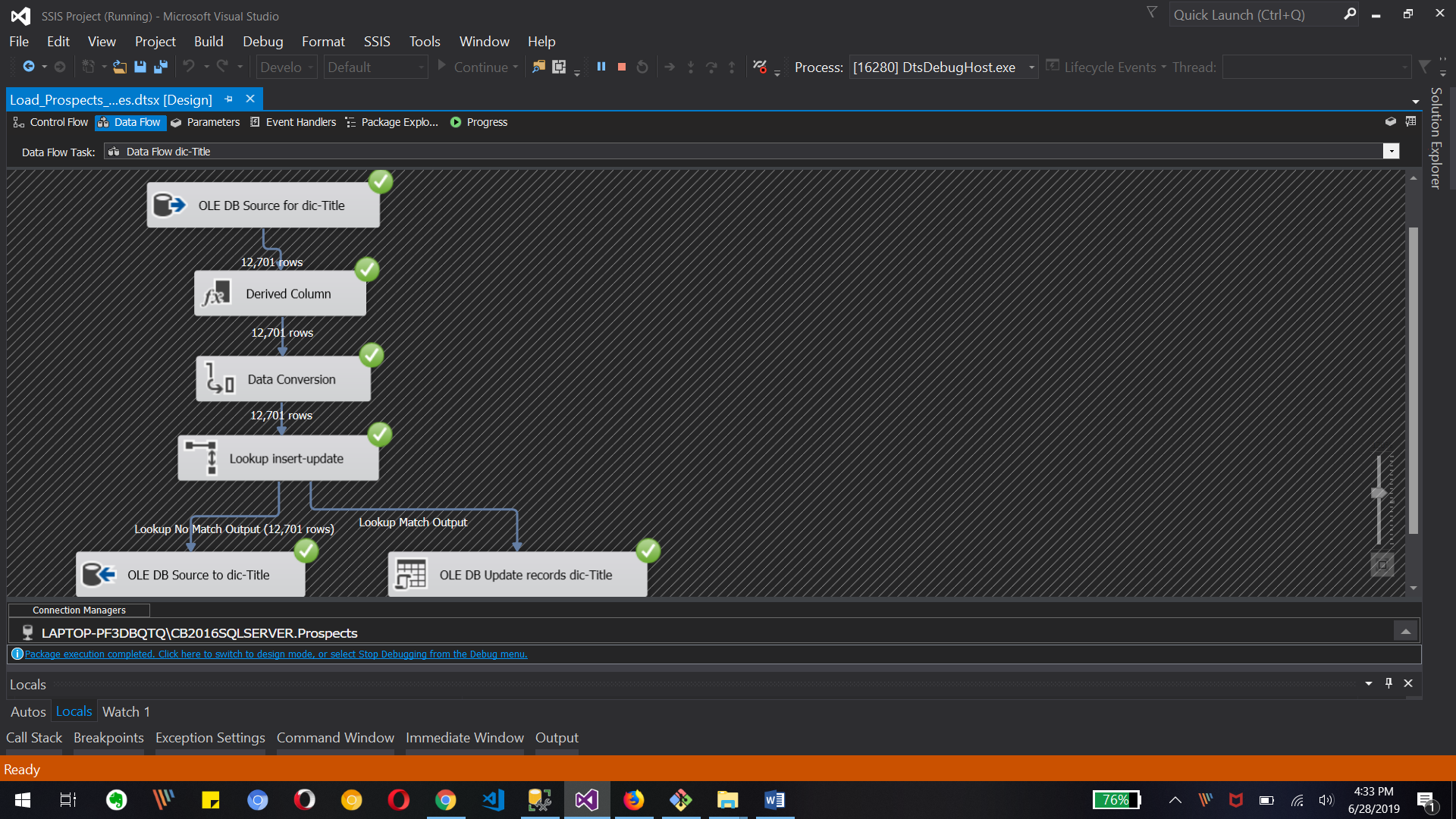
Updated records of Department table in SSMS



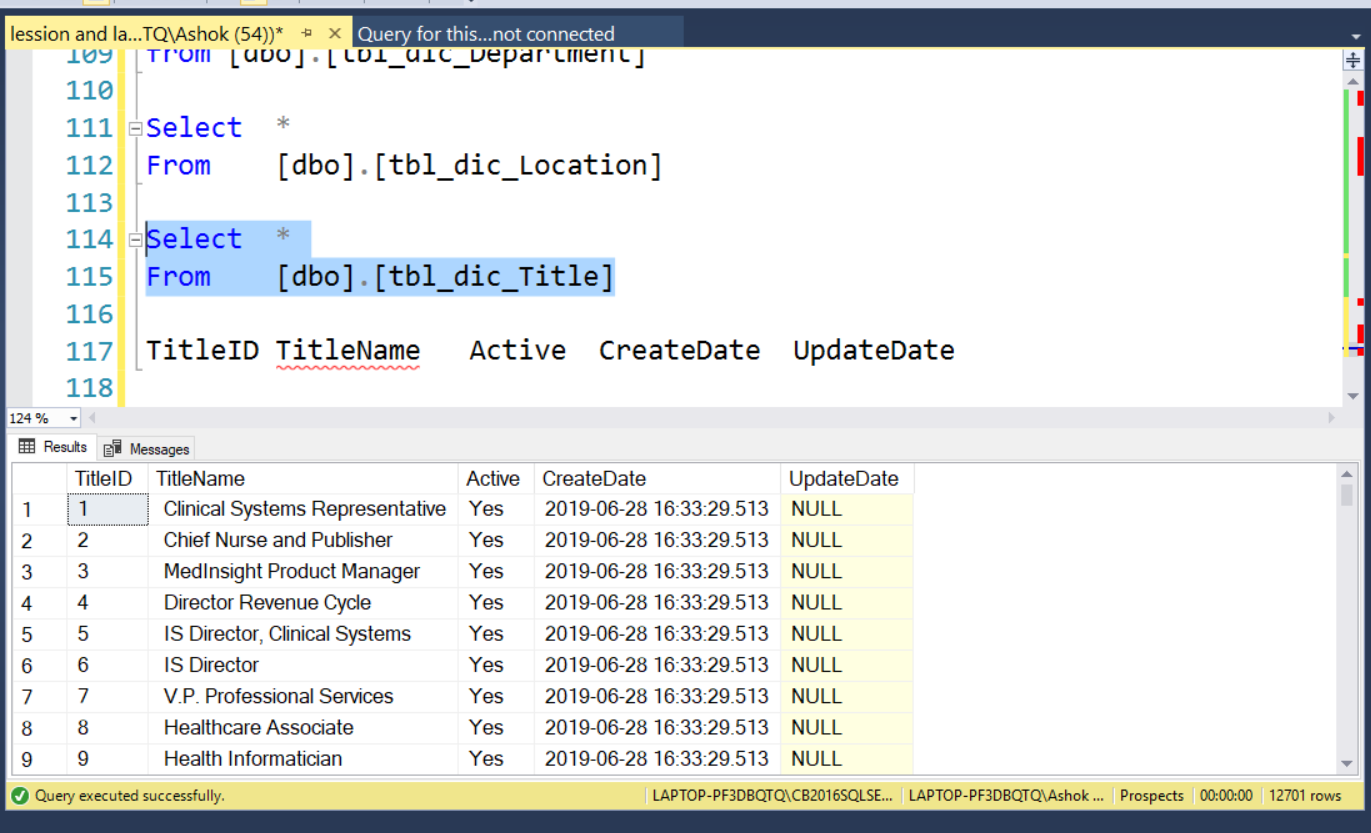
Debugging screen of dictionary-Location data flow tasks



Updated records of Location table in SSMS

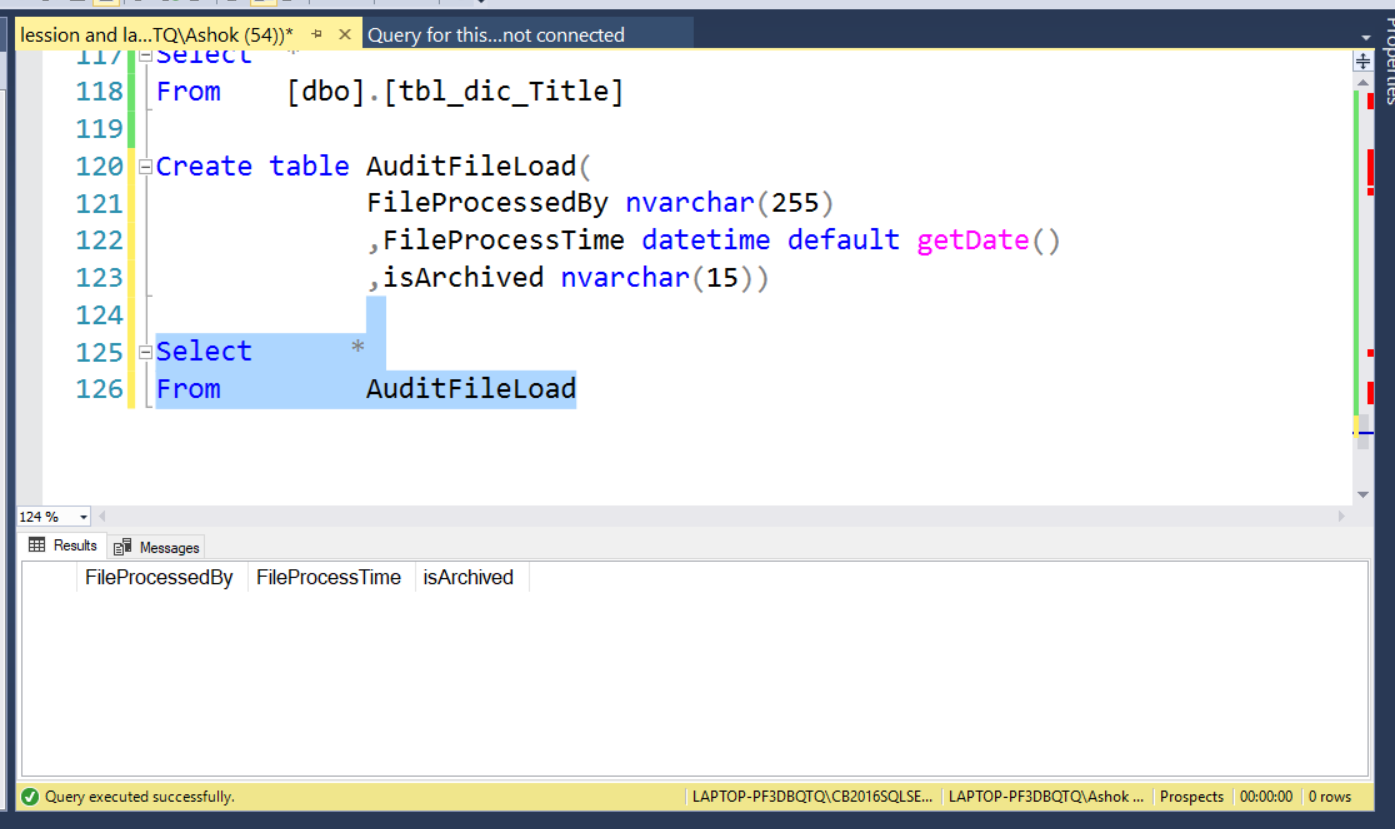


Debugging screen of dictionary-Title data flow tasks

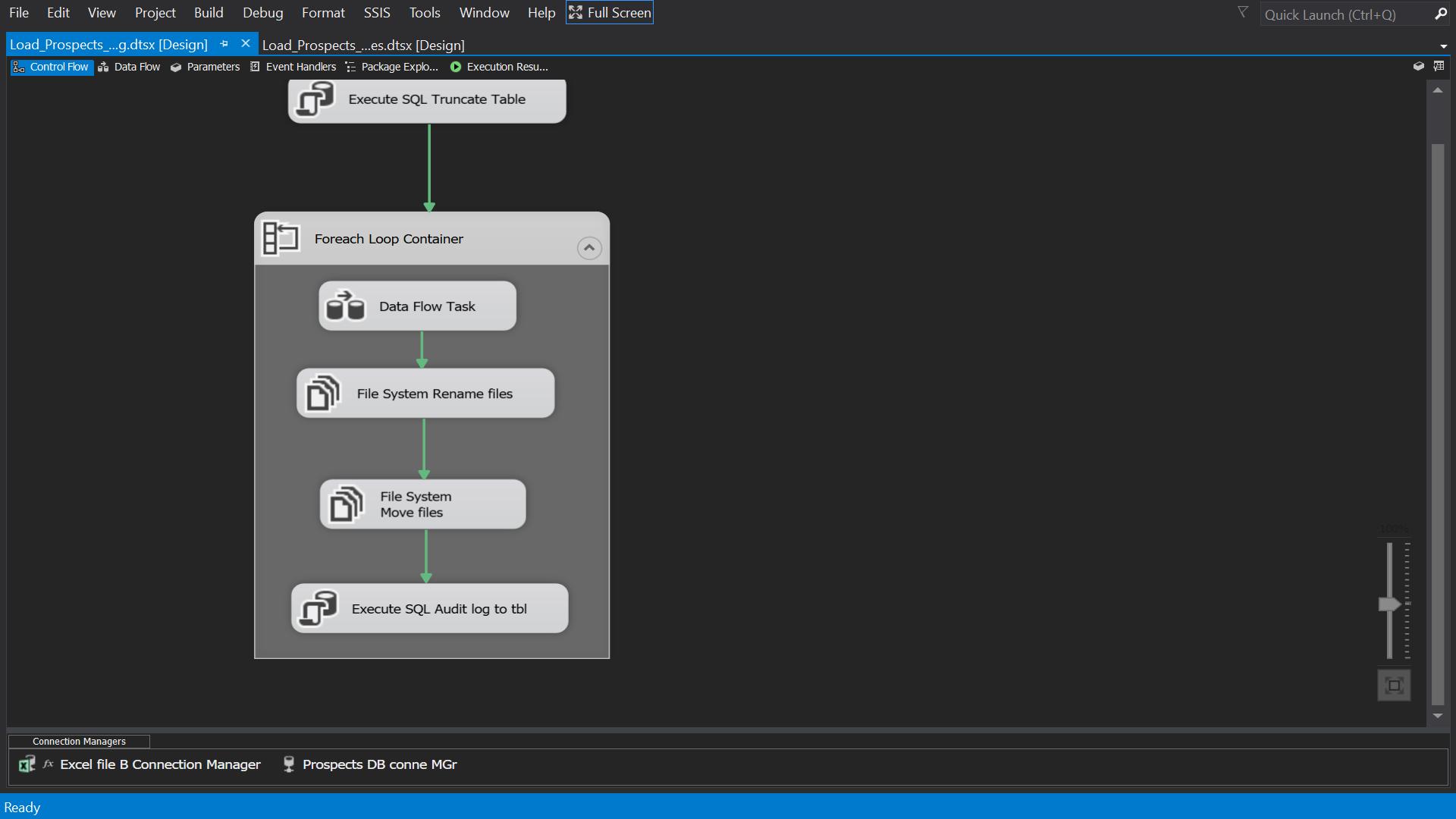


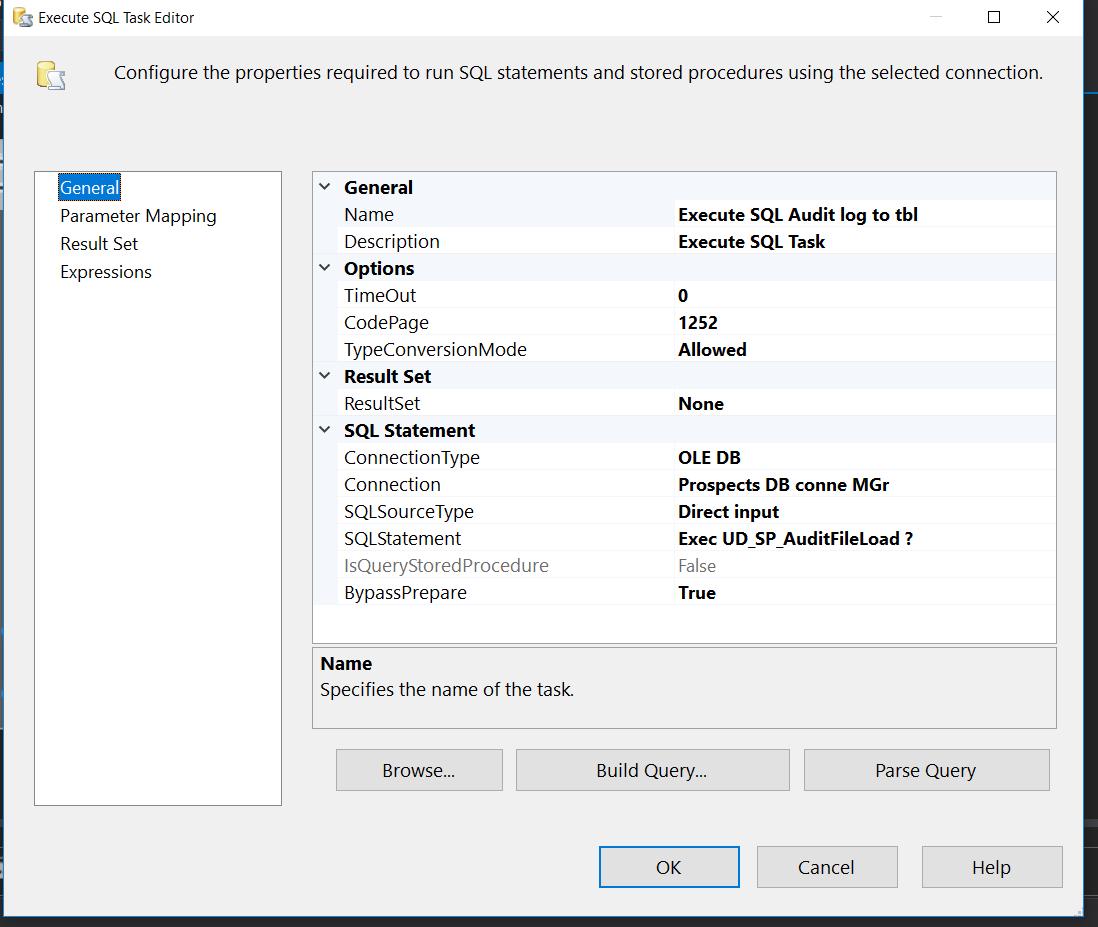
Updated records of Title table in SSMS

3. Create the audit table to log file load process – (Project document)

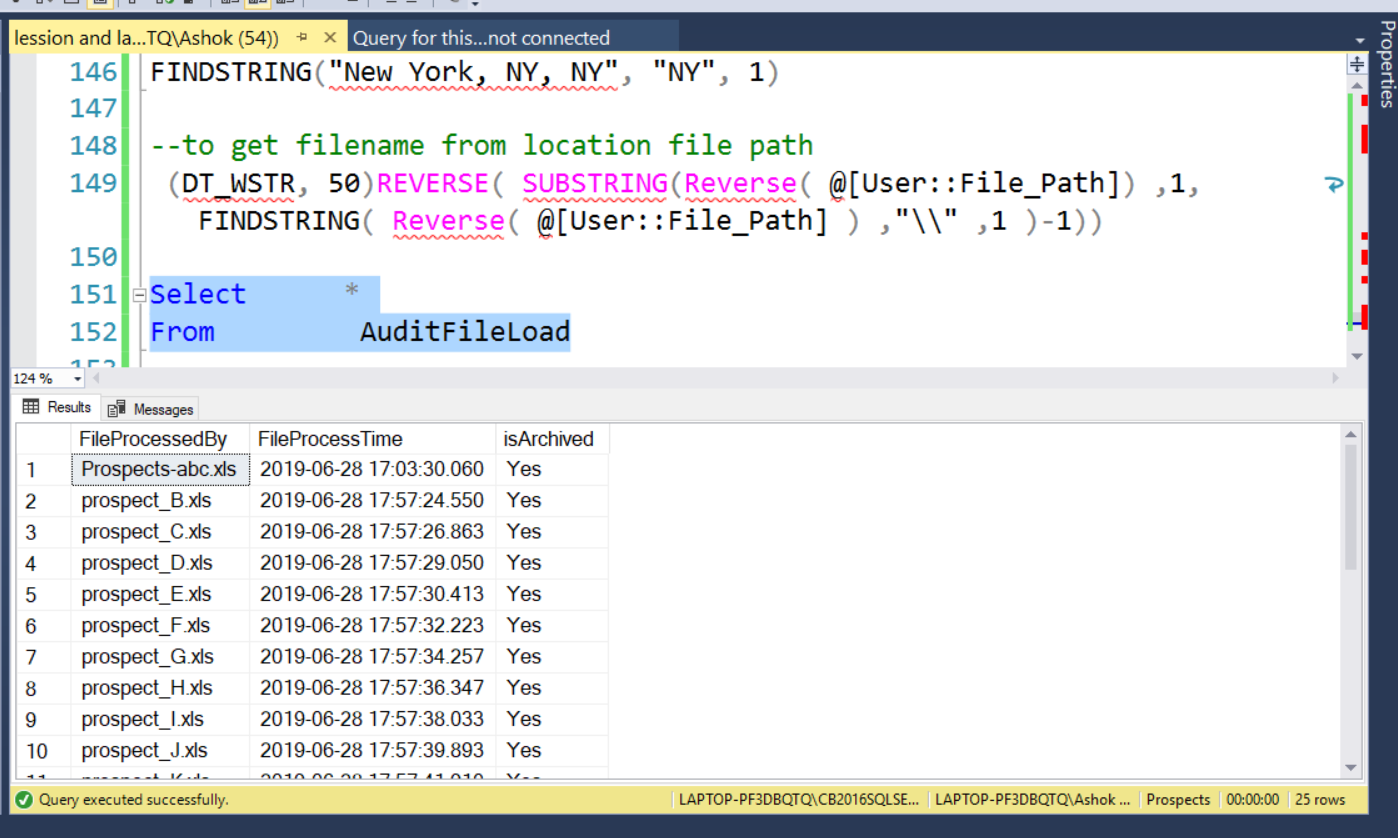


Creating AuditFileLoad table in Prospect database





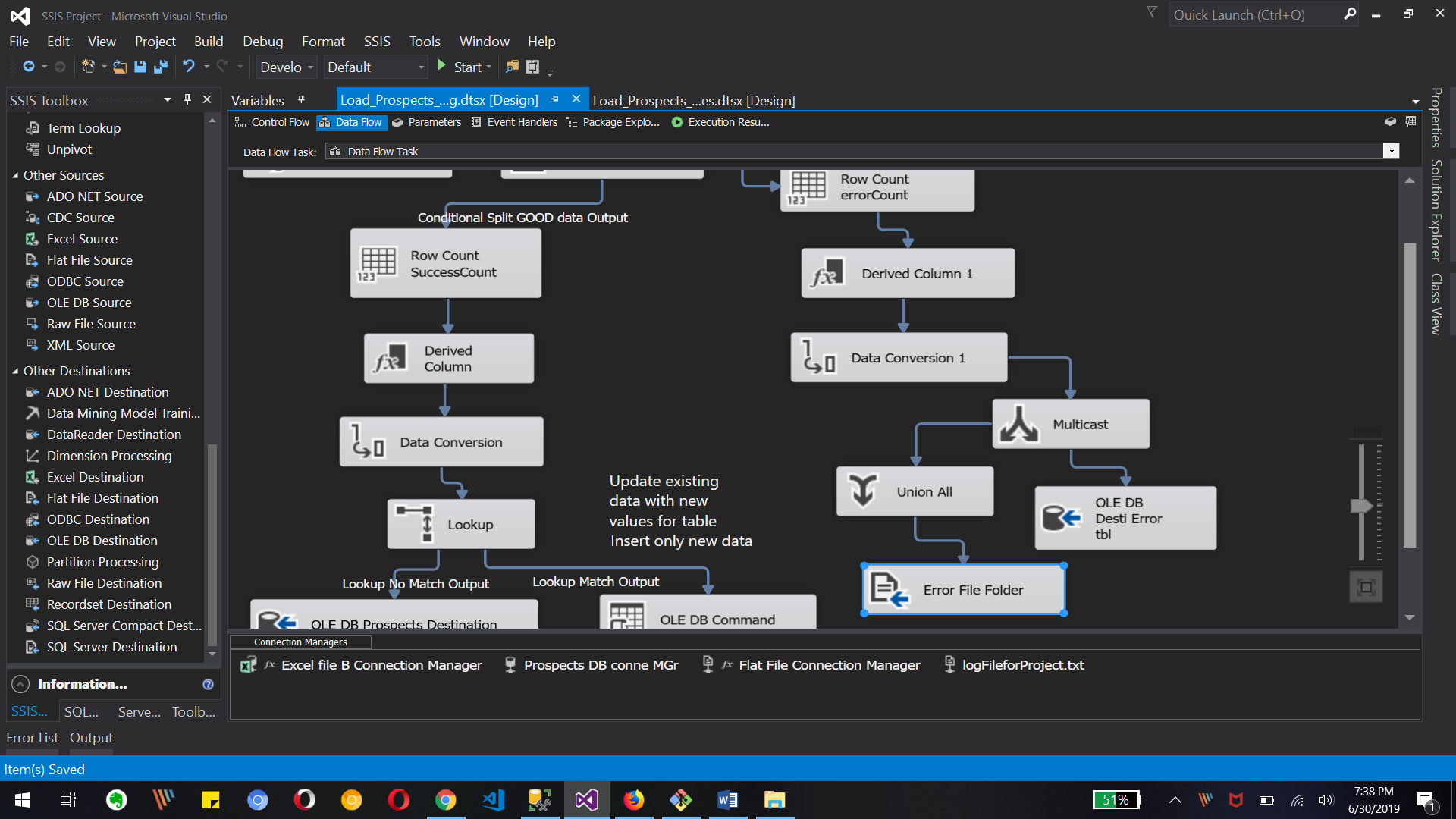
Creating Execute sql task for filling AudiFileLoad table

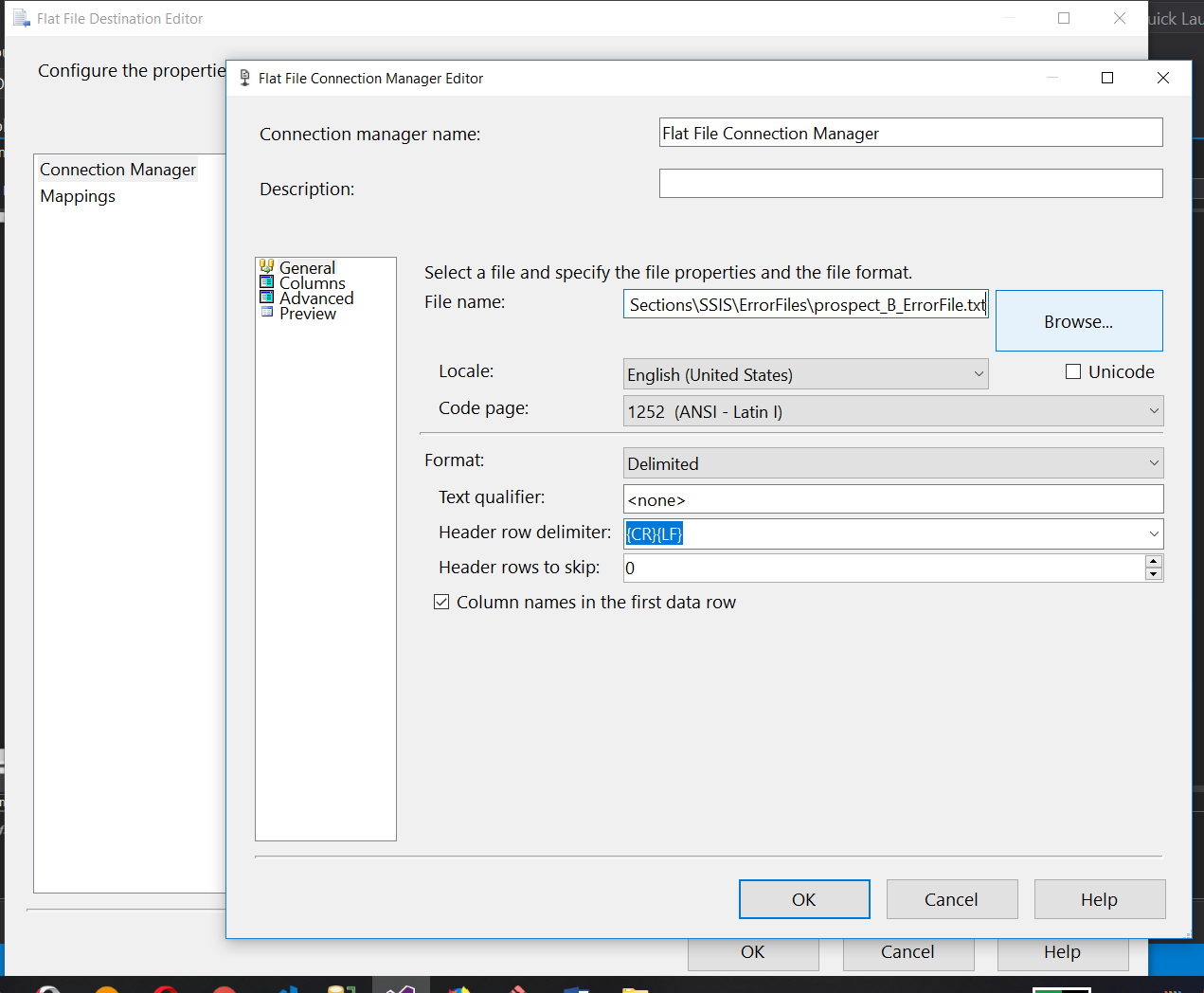


Final result of AudiFileLoad table after running SSIS package

4. Modify Load\_Prospects\_Staging.dtsx to include the following

a. Move the rejected files to the error folder

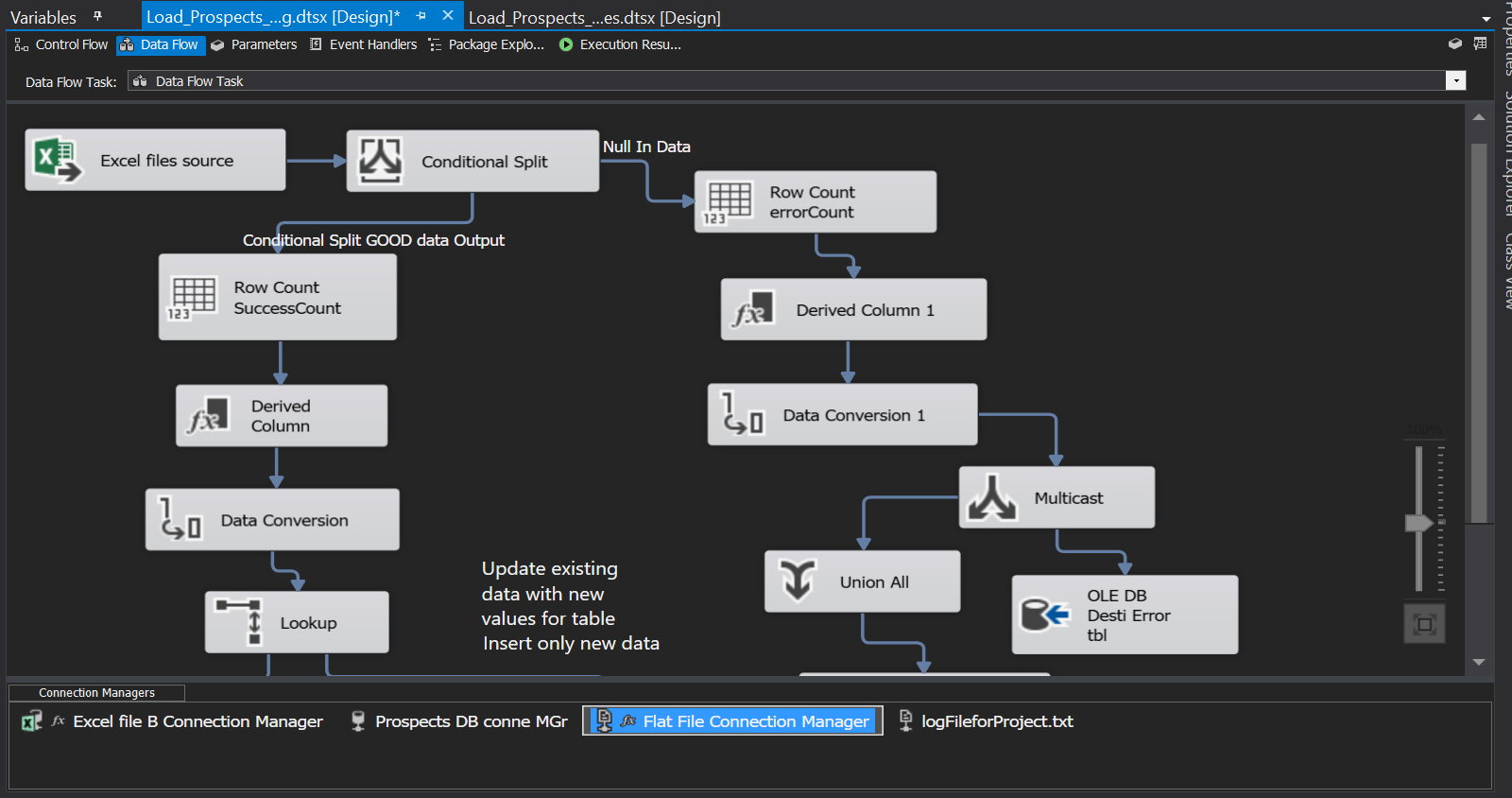




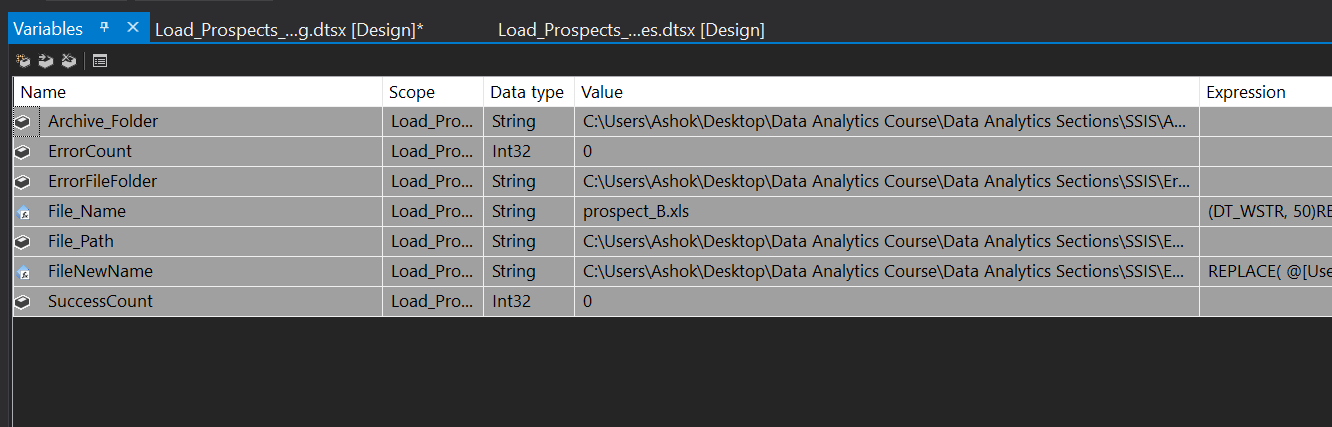
We created Flat file destination to move null data to text files with same name as xls file

Mapping for destination file.

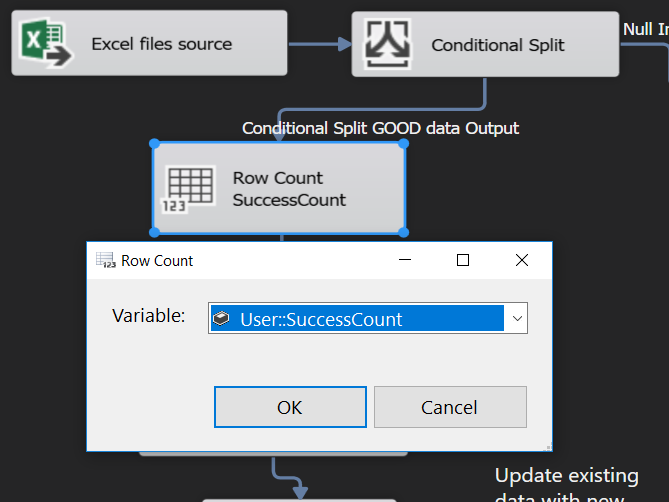
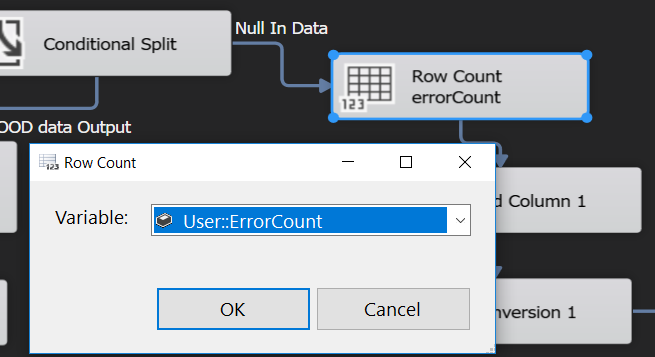
b. Get count of rows that were successfully inserted /updated and those that failed validation.



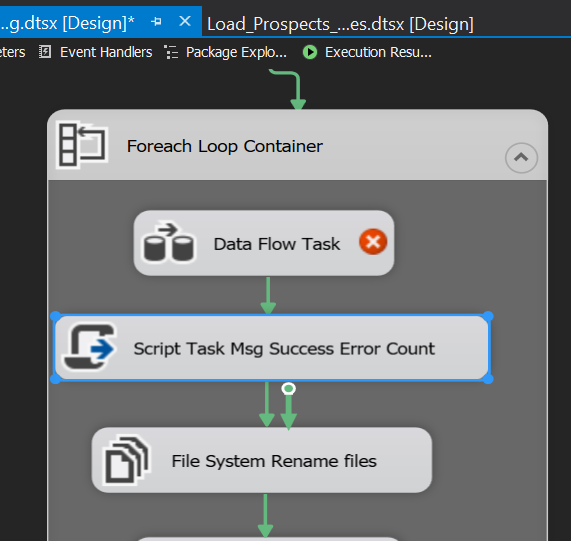
Creating Row Count with Row count transformation.



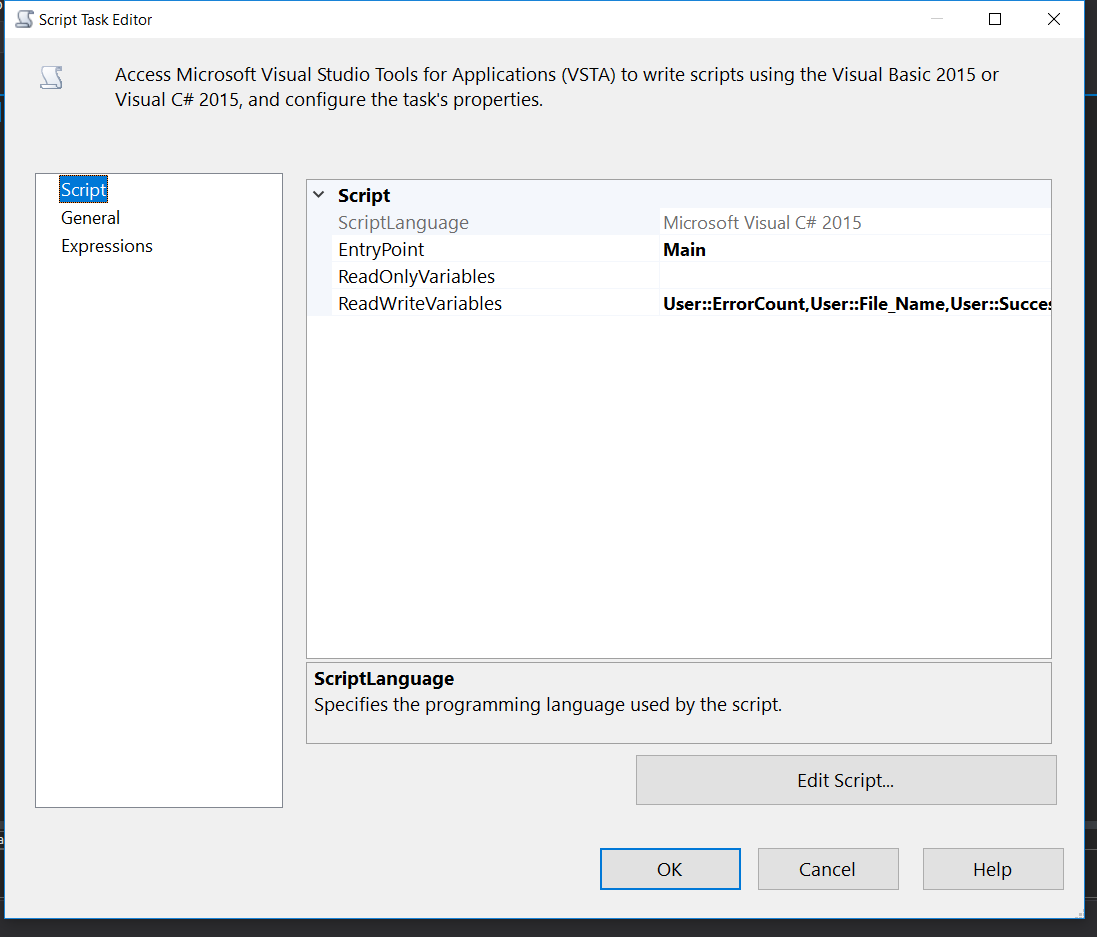
Creating Variable for row count 1 is ErrorCount and 2nd is SuccessCount of rows

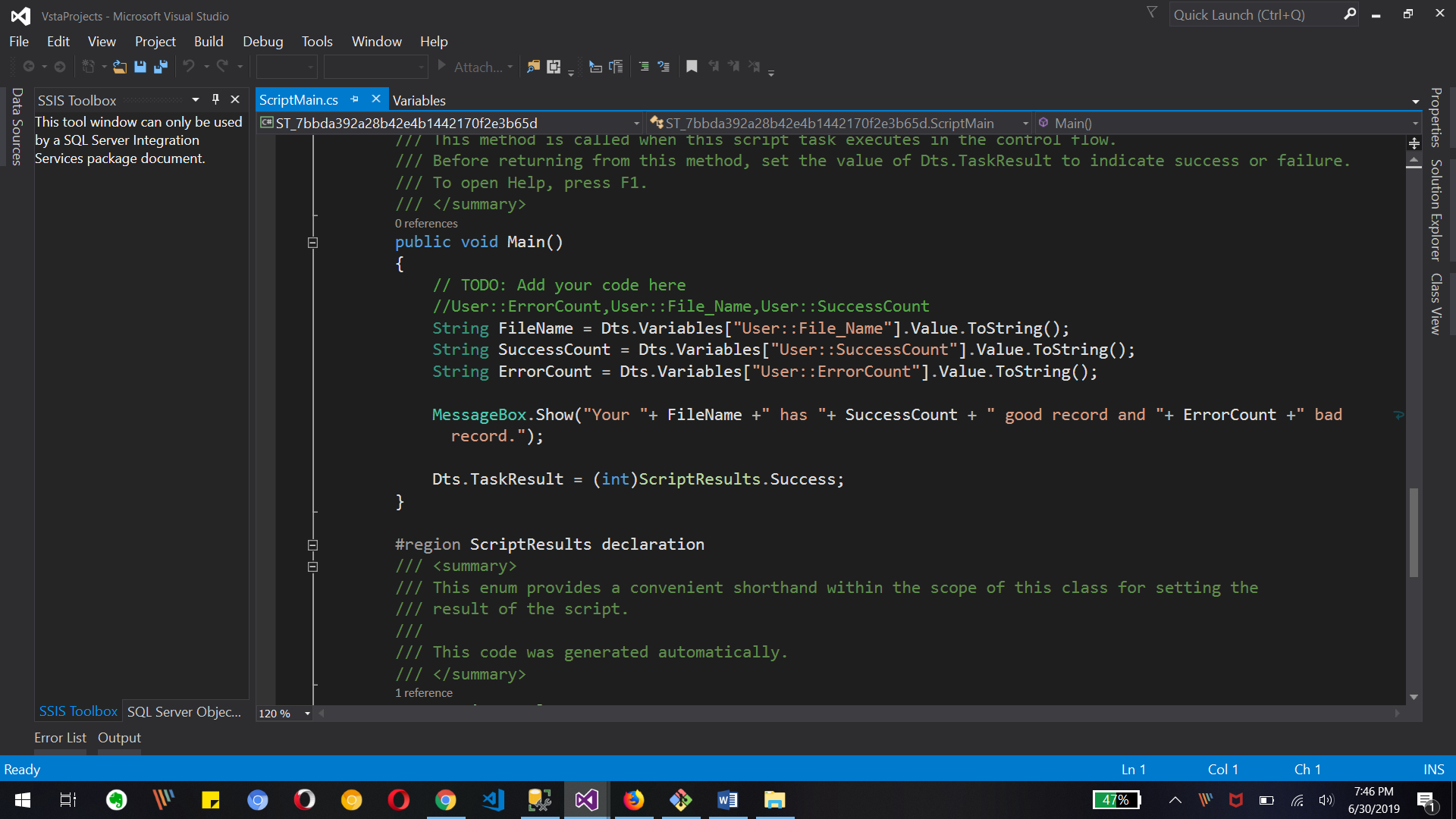
Using variable for different row counts transformation to store count



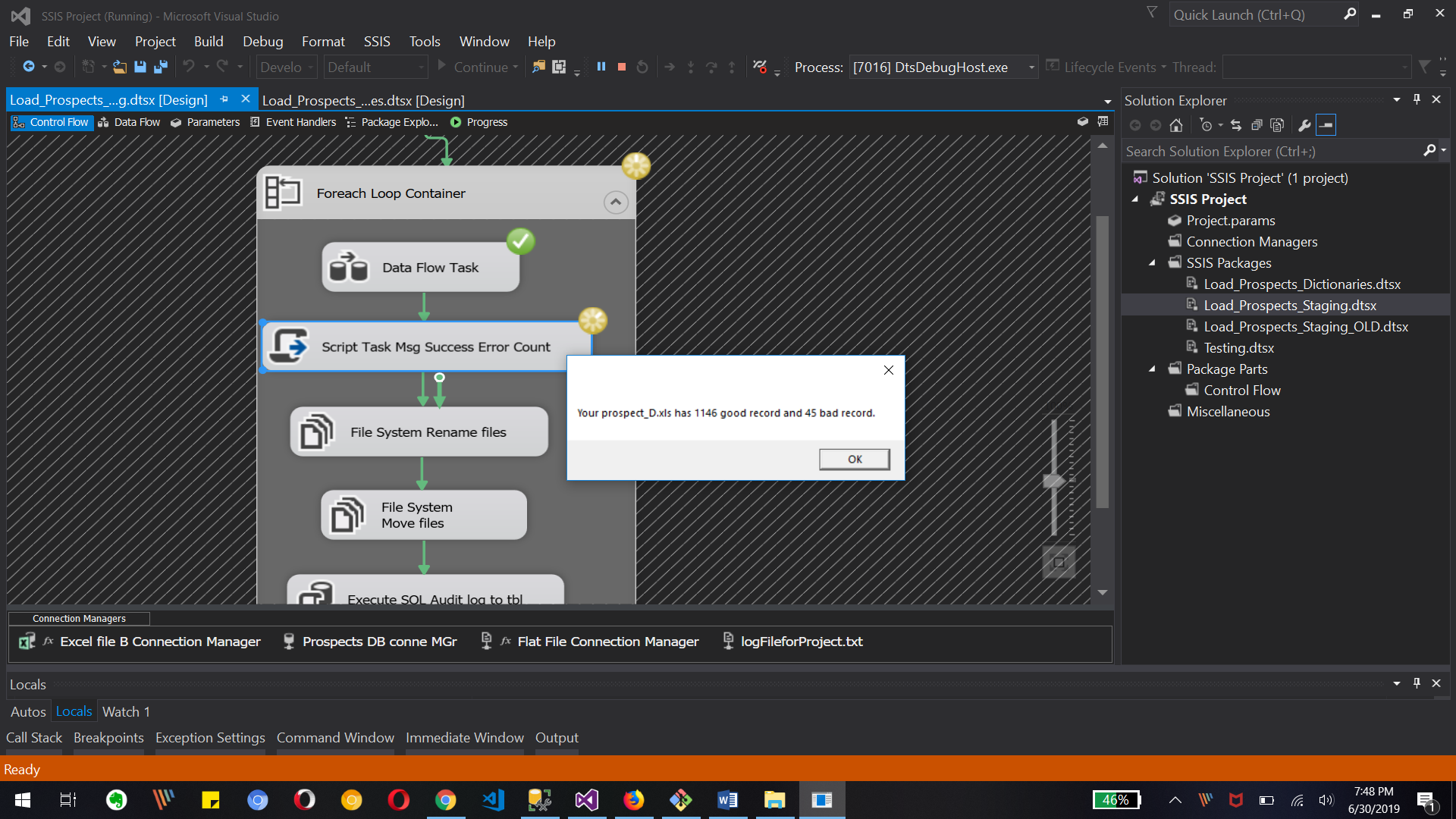
Now we will use Script Task to show row count as message box with C# script



Script task Editor for selecting variables to update and use. Also we can use Edit Script button for new Script for showing messagebox for row count

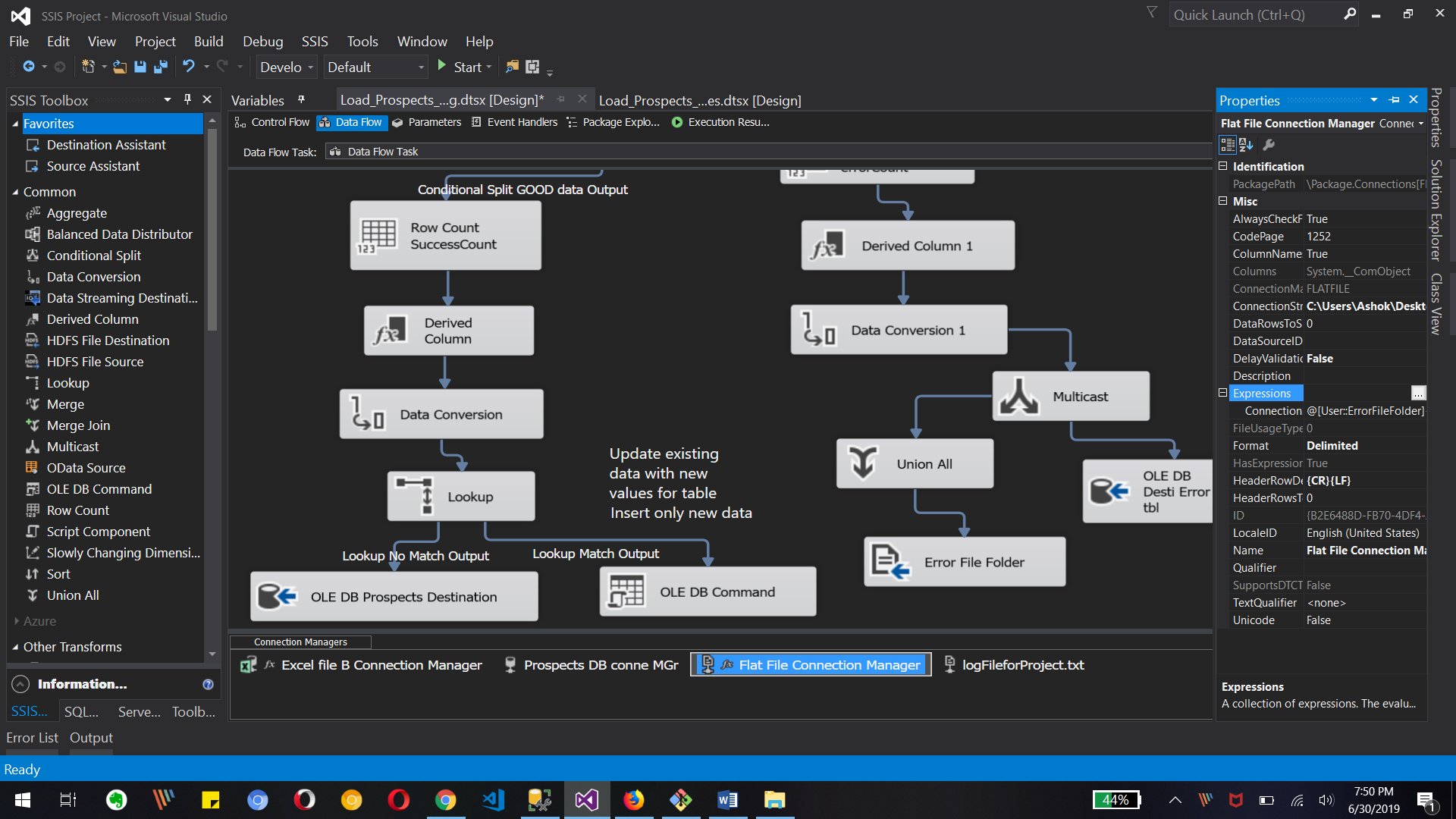


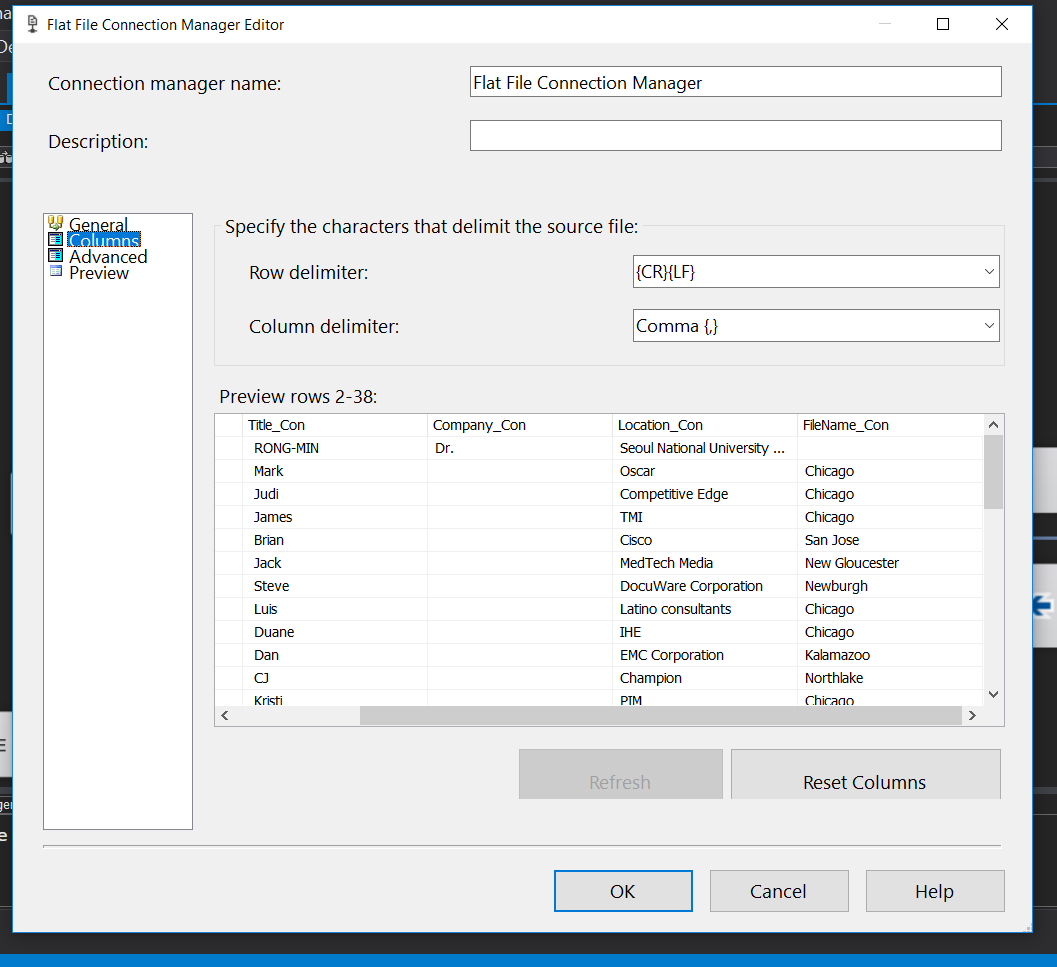
The above code will show us message box for detail row counts.



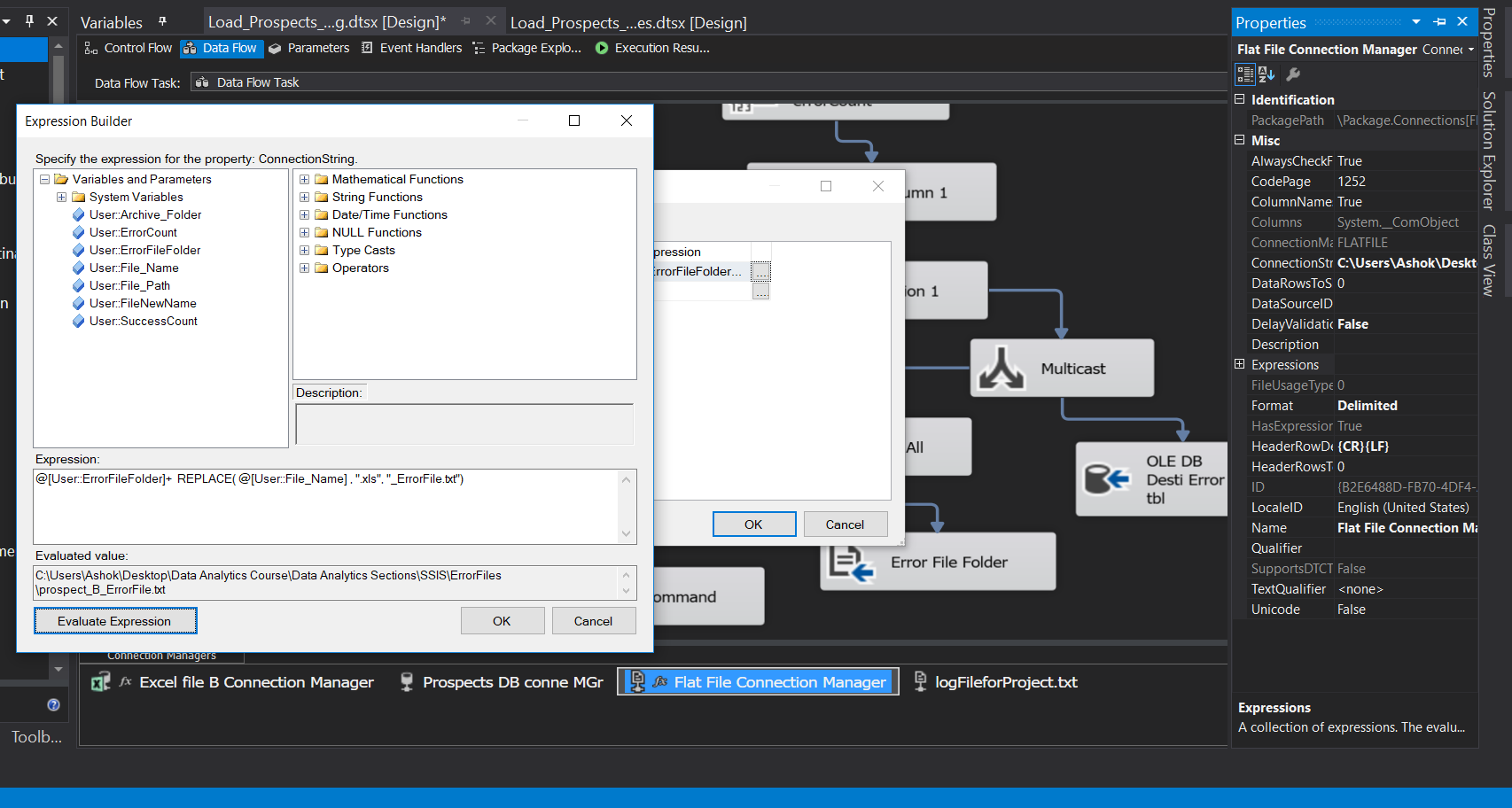
The above image shows the rowcount detail for success and error data for specific file.

c. Create separate error file for each file loaded containing the records that failed the NULL validation that you implemented in Lab – 3.





Above images are for creating separate errorFiles for null validation for each Excel files.



We need to create File name with ErroFiles as attached text to original name so we have to use connection manager’s properties expression to generate name for file in txt as shown above.



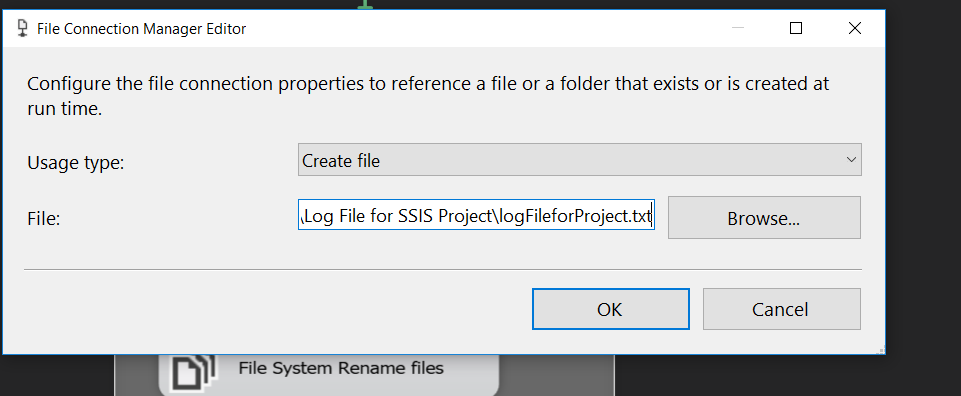
The above text files are error file with data with error as null validation.

d. Implement logging functionality in the package.

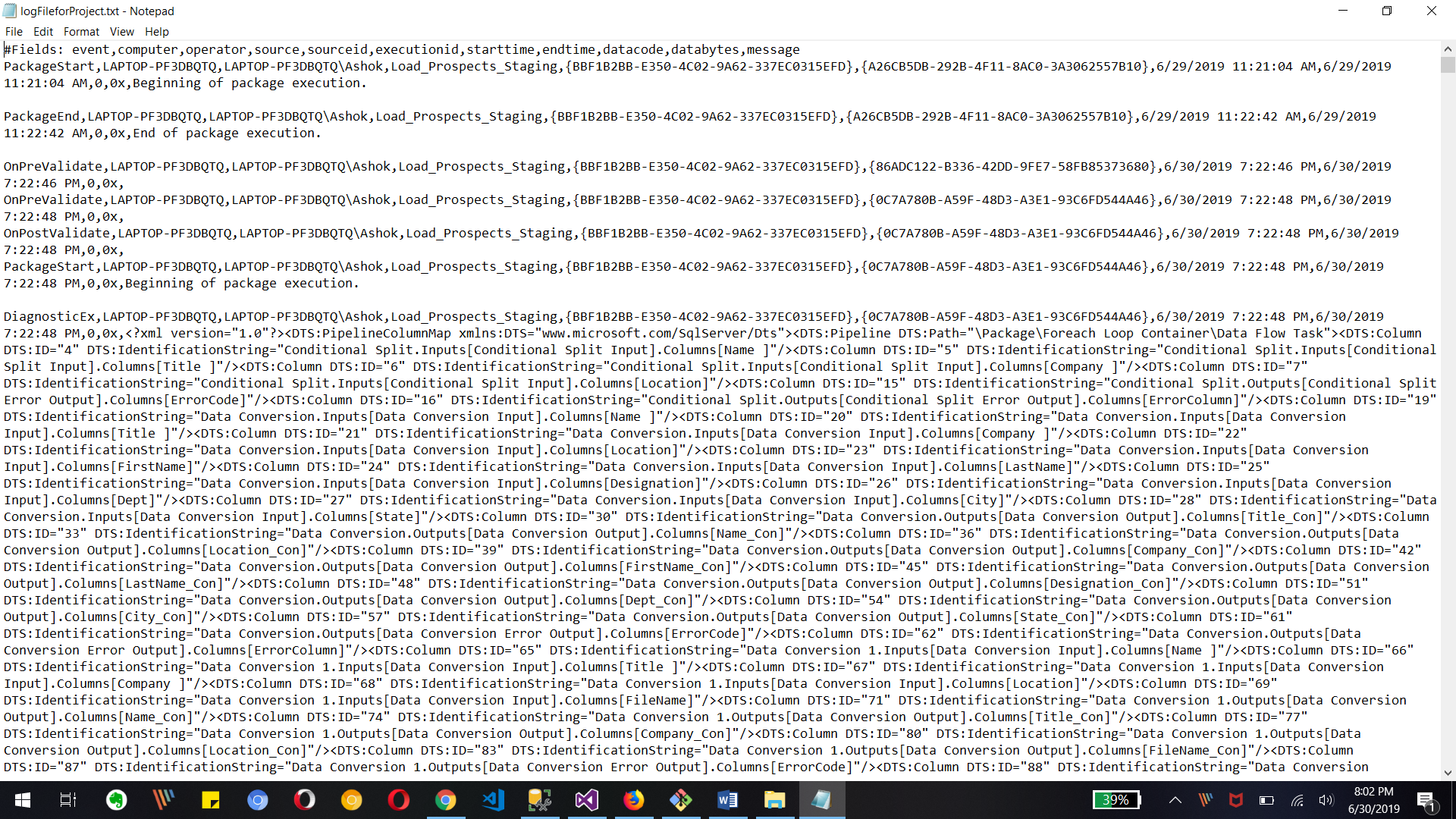


We can create log file for whole project with right click on control flow section and selecting logging

Above image is showing the selecting text file as Log file and configuring it for new text file connection to create file and sore logging detail

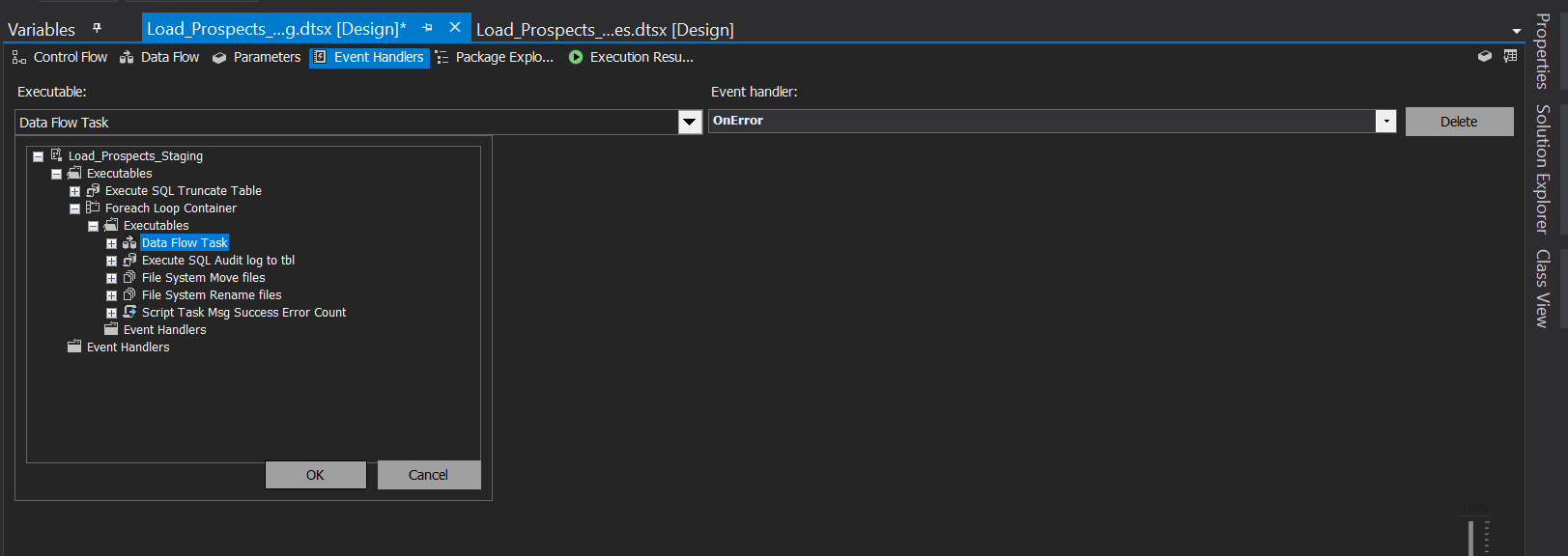


The connection manager for logfile in text.



The above image is showing log file detail for project start, any error etc.

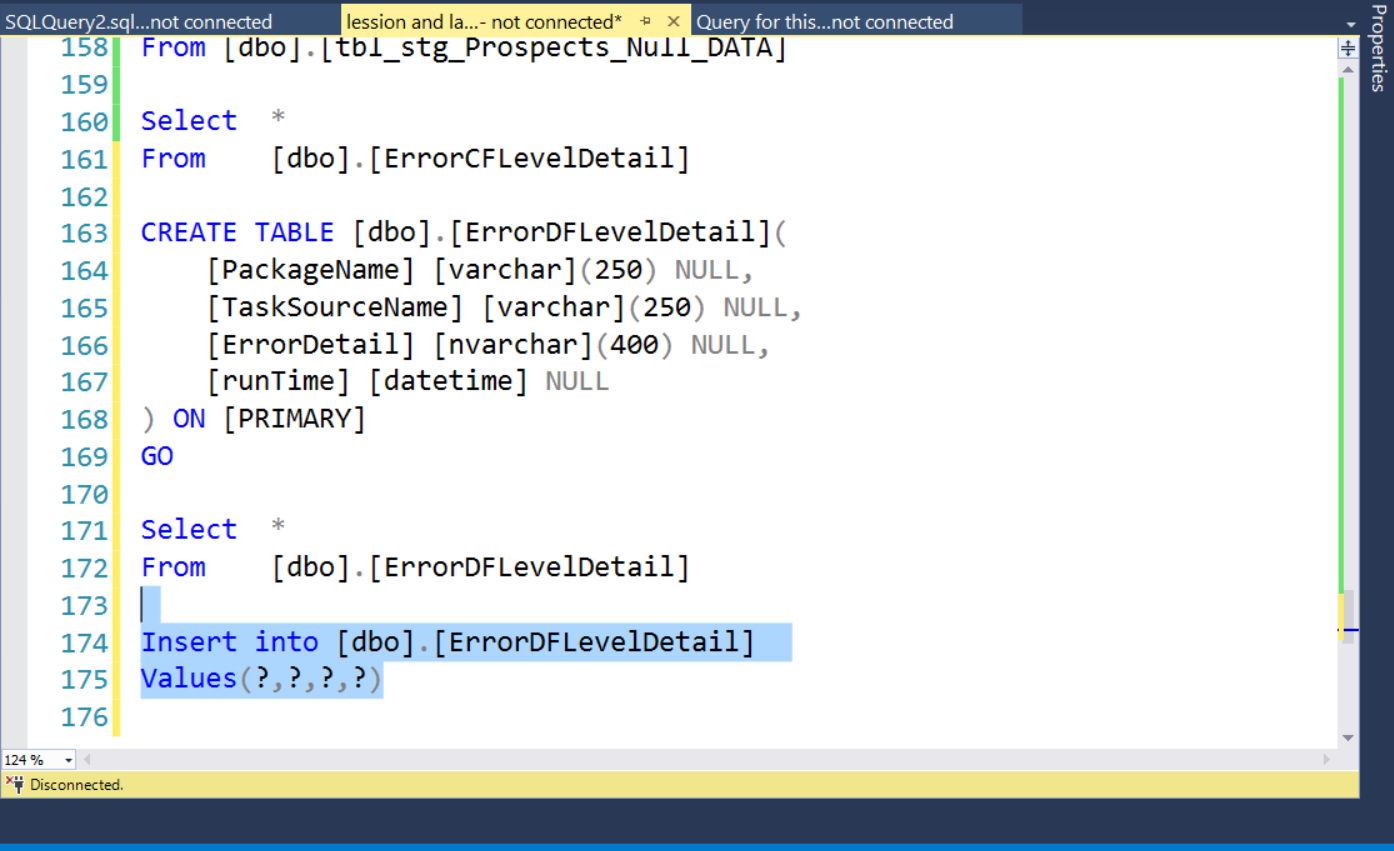
e. Implement error handling on the data flow task to avoid package failure in the event of any unforeseen errors encountered.



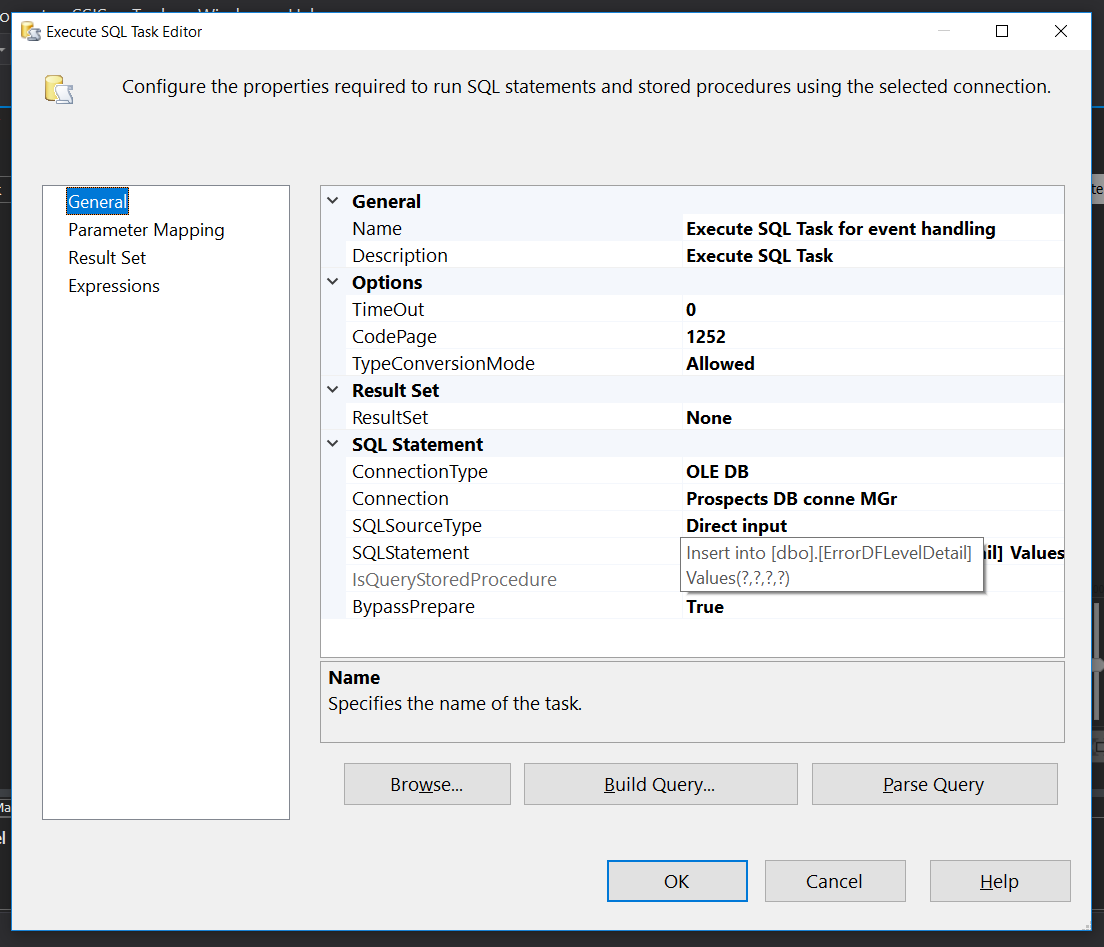
On the even handler tab of project, we can add data flow task event handler as shown in image.



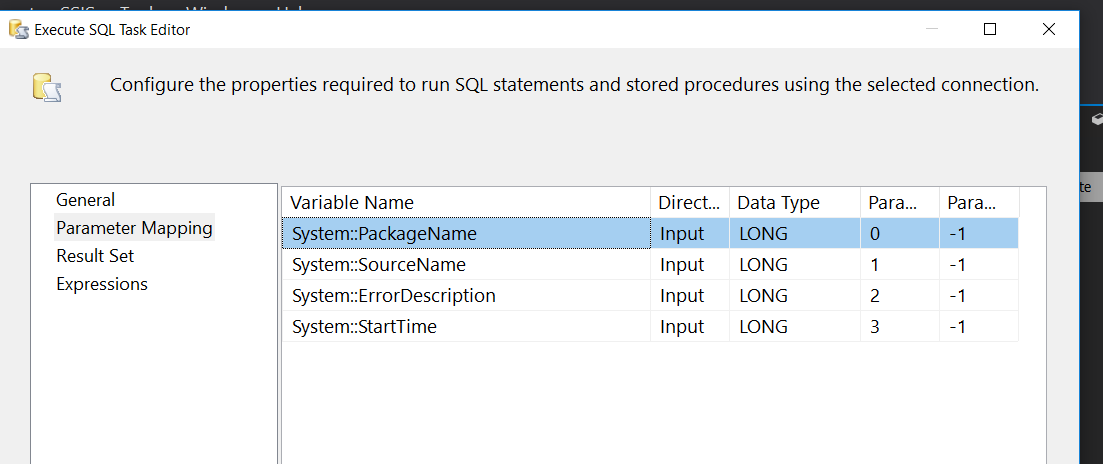
We need to add Execute Sql task for inserting error data to table as above image



Let’s create table in Sql server for error information at data flow level as shown above image. Table name is ErrorDFLevelDetail.



The above image is for Execute SQl task editor for ole db sql table ErrorDFLevelDetail insert into query for different error detail with using parameters.



The shown image is for parameters which we need to fill table with different error message.

We use different system variable for getting information for package error.

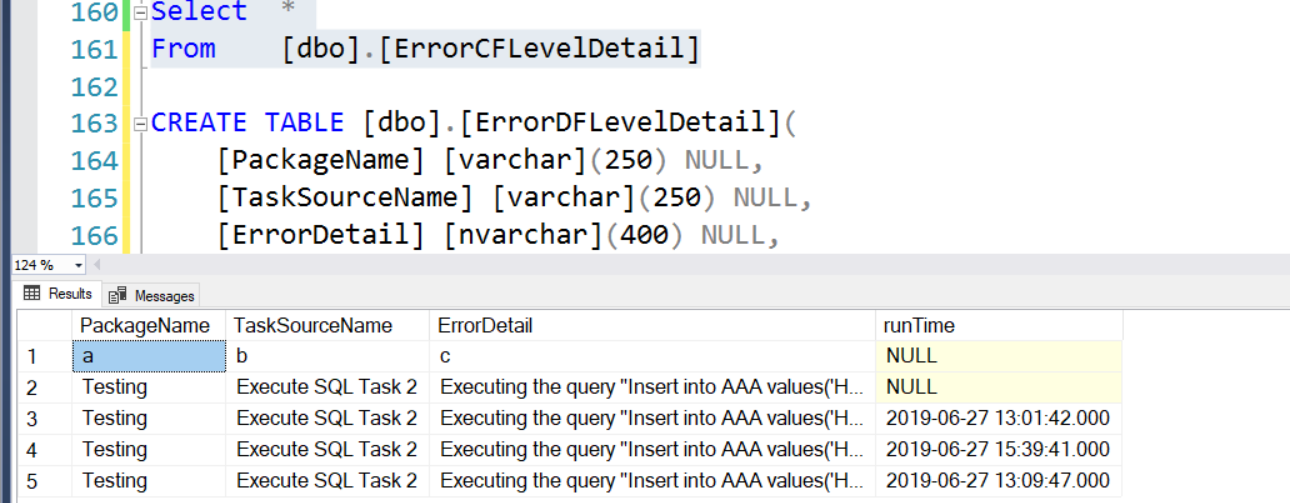


Table data for any error occurred in execution.