Luna McBride

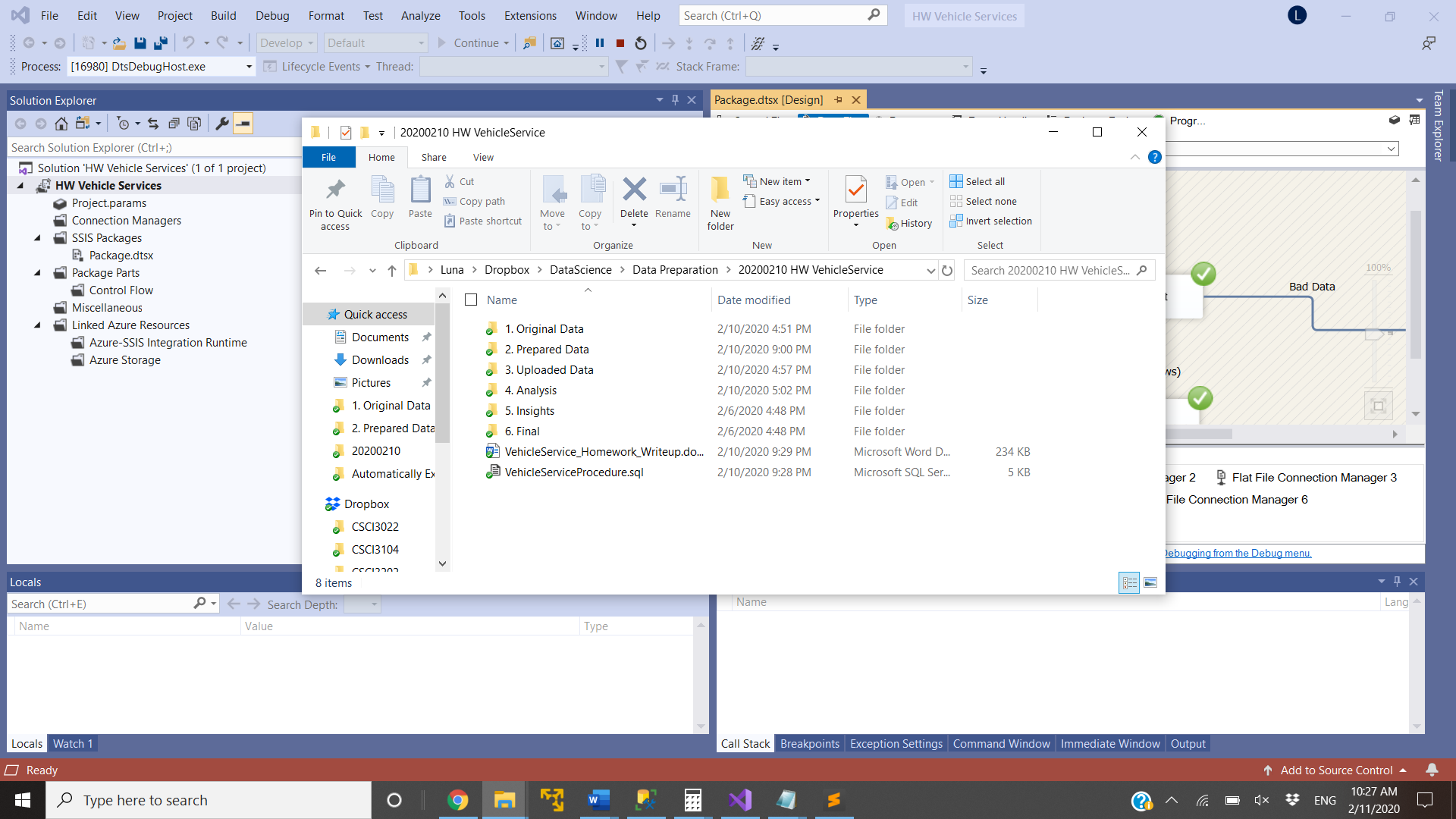
Udemy: Data Science A-Z

**Homework: ETL #2**

**Opening**

This is a write-up for the second ETL assignment, Vehicle Services, of the Data Science A-Z class found on Udemy. This one covers the major steps of ETL, including data formatting in SQL and addressing errors. The dataset given is 1.05 million rows long. Unused folders will likely not show in Github, so trust the directory setup to show it exists. Please note that Github has an upload size limit of 25mb, so folders 1,2, and 3 are being zipped to accommodate that limitation before being uploaded to Github. This is not meant to be a substitute for doing the class, but rather a tool for myself to show the skills used.

**Directory Setup**



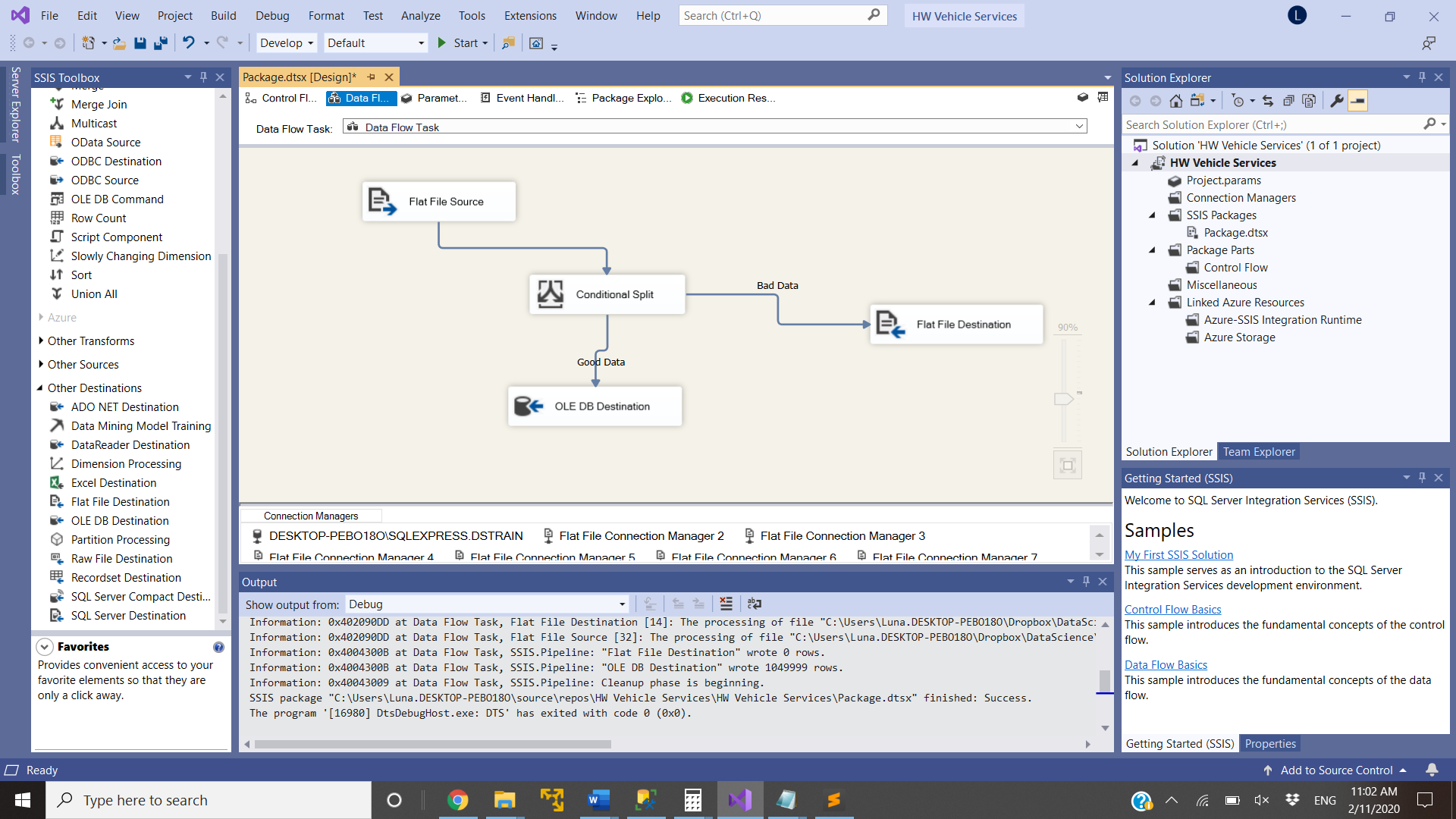
The Vehicle Services directory has eight components, all set up to keep things organized. The file system is very similar to the FakeNamesUK homework, with files 5 and 6 also empty, but there for organization purposes. The rest contains as follows:

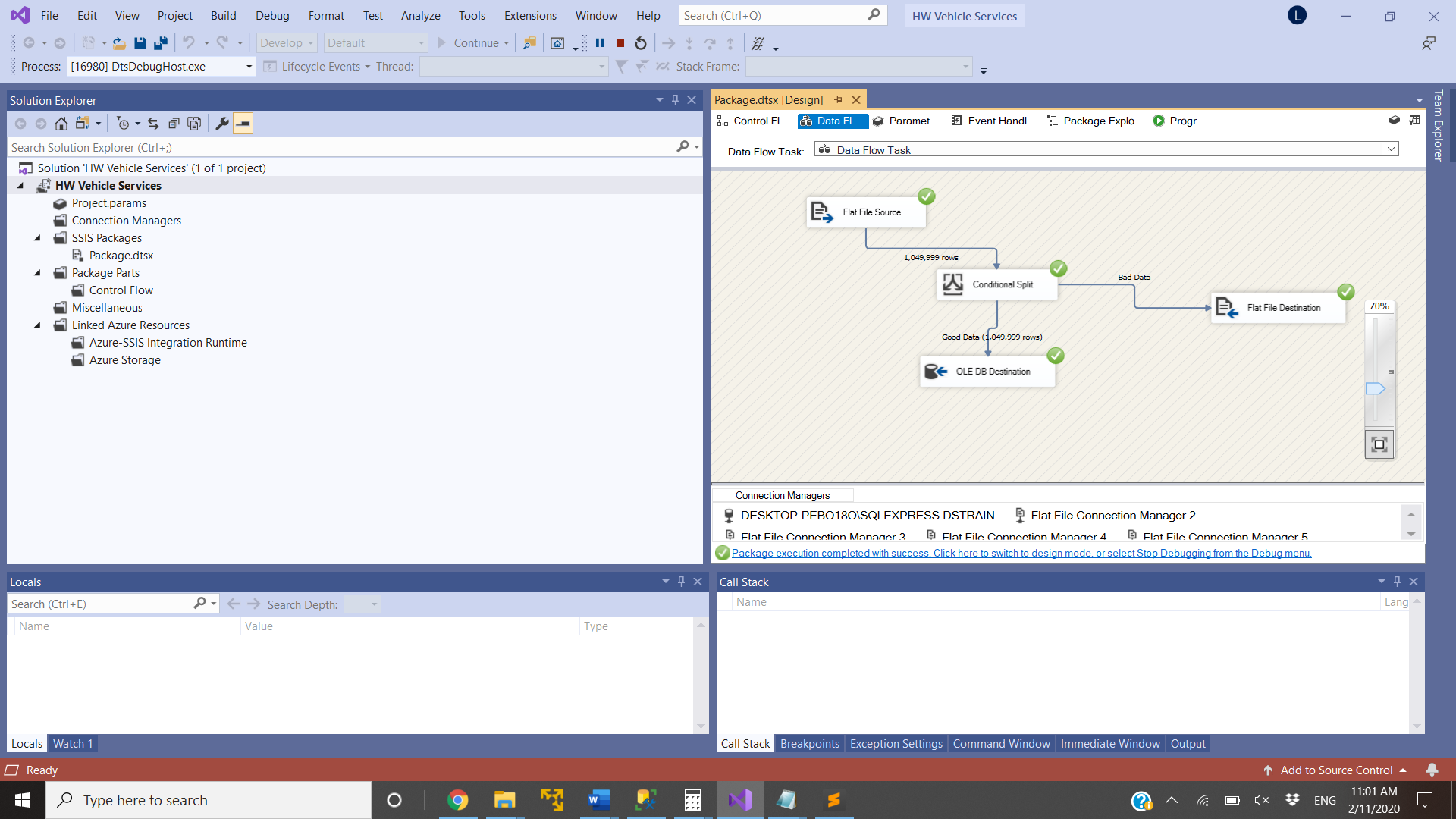
1. This folder contains 2 key files:
   1. VehicleServices.csv: The original file, unopened. This is kept untouched to prevent changes to the data that can occur from opening it in programs such as Excel. This time around, Excel would cut off many data points due to the dataset being larger than Excel can manage, so Excel will not be used at all. All usage of this file is through copy-pasting it elsewhere.
   2. P12-VehicleService-The-Challenge.docx: This is a word file with stipulations given during the assigning of the homework. It is very important to keep in mind when looking at the assignment further on.
2. This folder holds a file also called VehicleServices.csv, which is the original data set, just copied here with one row removed (line 1000001, which cased a truncation error) and one row fixed.
3. This folder contains two folders, one as a template for the date and one with the actual date of usage. This actual folder holds a copy of the prepared data, segmented for uploading purposes.
4. This is the analysis folder. This homework does not have actual analysis, as that is not the point of this section. It does, however, contain files of data manually and automatically removed, each segmented in different folders, but held together in another Data Errors folder. The automatic removals file is actually blank, as there were none that triggered Virtual Studio trigger. There is one manual removal though, which was the line 1000001 with a truncation error stated above.

(5 and 6 are skipped here due to being empty. 7 and 8 are the files in the image below 5 and 6, shown in order)

1. This file is the write up, which is the file being viewed now. This explains the homework in order to show knowledge and skills gained from it.
2. This file is the SQL procedure made to do all the post-Virtual Studio adjustments to the data. To use it, update the database information to fit your database and change the ‘alter’ keyword to ‘create’. Press execute without highlighting anything to save the procedure. Highlight any code that you would like to use. Be careful when doing so, as highlighting things incorrectly could lead to issues down the road, such as not truncating before inserting, which will cause multiple occurrences of the same values.

**Virtual Studio Setup**





Virtual Studio here has 4 main components:

1. Flat File Source: This is the component that takes in the VehicleSerices.csv file for use further down the line.
2. Conditional Split: This component checks for overflow and underflow of rows based on if the last row is empty or if another row has formed out of bounds. The data was not put into Excel and thus not changed to create another row, and thus an error further on was not caught here. That was caught in SQL and fixed, however.
3. Flat File Destination: This is the file where errors would be placed if there were any caught.
4. OLE DB Destination: This one takes the good data in and places it into the determined database.

**Errors**

The assignment specified to find the errors and describe them (what they are and how to fix them).

1. This whole time in the class Excel was used to make sure formatting on dates and other parts are correct. In this case, when Excel was tried, it limited the number of rows allowed and, thus, cut off a large chunk of data. This was fixed by not using Excel, as the data just so happened to be in a good format.
2. A manual exclusion was made to line 1000001, which had a field way bigger than it should be and, thus, caused truncation errors in Visual Studio. The long field did not show in any of the text editors I use, so I fixed it by just typing the values in as an insert statement in SQL. All data still existed, so it was something that could be fixed.
3. Various lines caused an error from the ‘2014’ column, as those values were left as blanks for customers who were not customers in 2014. Blank values did not allow conversions from varchar to floats. These values can then be changed to ‘0’, which is a number that can be converted to a float while not effecting any calculations while getting a sum for revenue.
4. Row 676803 had an error in the ‘2015’ column. The stated value was ‘781$37’, which means somebody just put a dollar sign instead of a decimal point. Simply changing that value to ‘781.37’ works just fine and fixes conversion errors in the 2015 column.
5. Row 203974 had an error in which the 2014 column had a semicolon instead of a decimal point, thus extending the value into the 2015 column. This manifested in the 2016E column, which had the 2015 value, a semicolon, and the 2016E value all in one. If put into Excel, this would have been changed into an extra column situation that would have been caught by Virtual Studio. The best way to fix this was to change the semicolon in 2014 to a decimal point in the text file, truncating the table, and reuploading the table. This could also be fixed in SQL using regular expressions, but I thought fixing it in the file would be safer in this scenario.

Corrections after looking at solution video:

The assignment was not looking at Excel nor the 2014 empty spaces as errors. The truncation error also did not occur for the instructor, meaning it might have been something caused by not splitting the file. The expected errors were:

1. Two customers with the same CustomerID.
2. Customer Since year of 1899, which was likely supposed to be 1999
3. Max value in 2014 absurdly high in comparison to max values in 2015 and 2016E.

Though I might also consider these as errors in the way that they might possibly be incorrect, talk with the company about them might be smarter than trying to fix the value, as they have no direct pointers to what they should be.

Due to how the homework was assigned, I thought the items I gave were fair as errors. I, personally, would not consider the correction errors in the same way, thus making this a miscommunication in the assignment. I do, however, see the importance in these checks.

**Result**

The end result is a working table in SQL with appropriate data types and errors displayed and fixed. After all is fixed, the sum of the 2016E column ends at $419,896,187.87, which is the value specified in the instructions. This means it is all prepared to gain insights from for the company behind this, be it real or hypothetical. Of course, I could have split the file into two and went from there, but the file already being well-formatted really worked in my favor, as it was easier to upload in one bulk than to do two.

Even so, the ambiguity within what was considered errors and thus the difference between my answers and the instructors is something to be noted. The checks given as answers are of course important, but the framing of errors in previous examples and the emphasis on how to fix them created a disconnect between definitions of errors. The mistypes given cannot be inherently fixed without input from the company; yet with no company to ask in the setting of a pre-recorded online course, the directions given become unclear. If the makers of the course ever see this, I would recommend changing up the assignment based on these ideas. Despite this, the database was made, the files were set up, and the value came out to the same value in the end, so I consider this to be a success.