**Data Cleaning in SQL**

1. **Introduction**

Data cleaning, also known as data cleansing or data scrubbing, is the process of detecting and correcting (or removing) inaccurate, incomplete, or irrelevant records from a database. In SQL, data cleaning plays a crucial role in ensuring the quality and reliability of data used for analysis and reporting. The main goal is to prepare raw data into a structured and consistent format that can be used effectively in decision-making.

1. **Importance of data cleaning**

Data cleaning is essential for:

* Improving data accuracy and consistency.
* Reducing duplication and redundancy.
* Enhancing the performance of queries and analytics.
* Ensuring better decision-making based on reliable information.
* Meeting data governance and compliance requirements.

Without proper cleaning, data analysis can produce misleading results and lead to poor business decisions.

1. **Common Data Quality Issues**

When working with large datasets, the following issues often occur:

* **Missing values:** Some fields may contain NULL or blank entries.
* **Duplicate records:** The same data may appear multiple times.
* **Inconsistent formats:** Different date or text formats (e.g., 01/02/2025 vs. 2025-02-01).
* **Invalid values:** Out-of-range or incorrect entries (e.g., negative age).
* **Trailing spaces or special characters:** Extra spaces or symbols in text fields.
* **Incorrect data types:** Numbers stored as text or vice versa.

1. **Examine dataset before cleaning**

At first, we will go through the original dataset, we will examine some main characteristics:

**4.1 Total rows and columns**

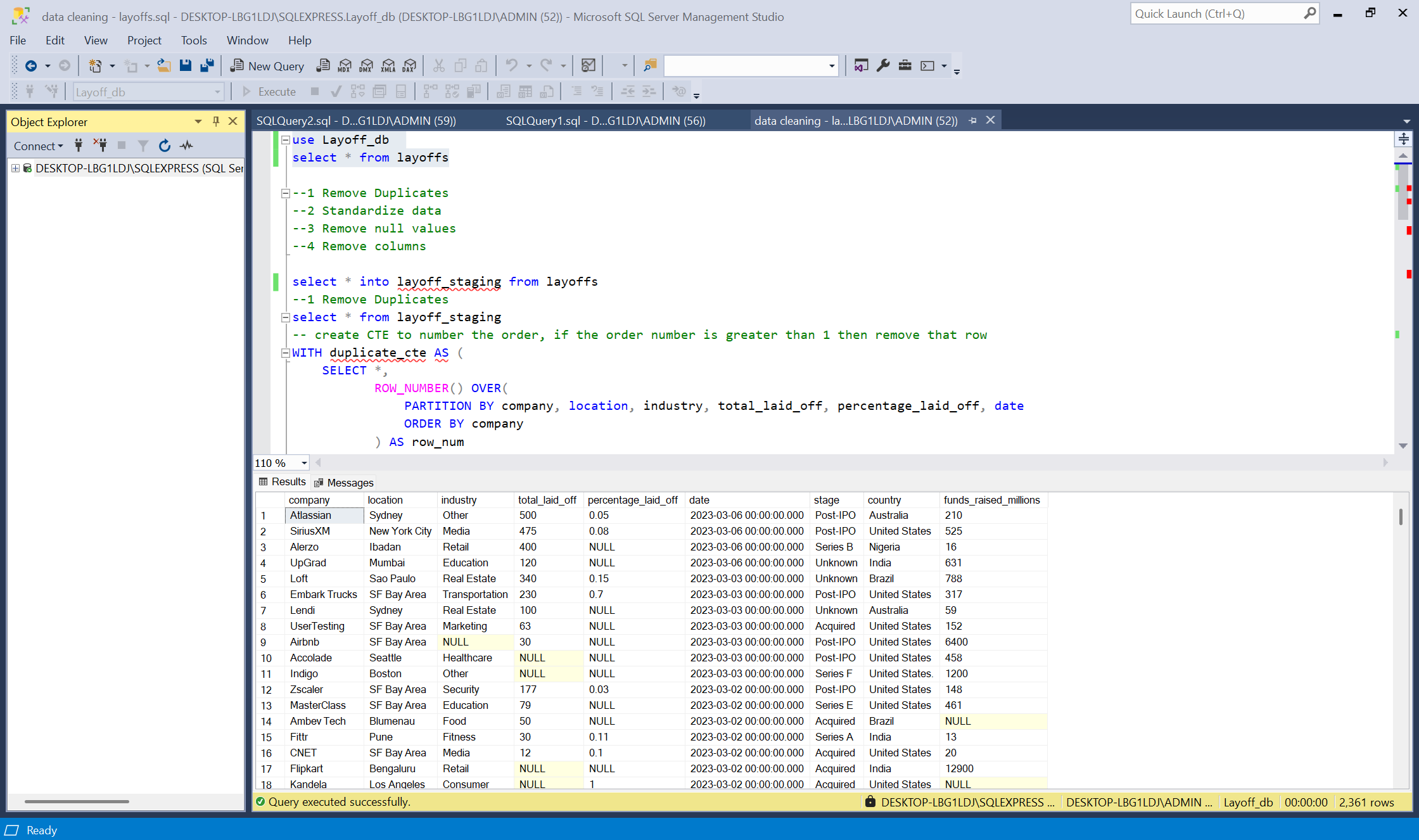


Figure 1: Total rows and columns

The original dataset has **2361 rows** and **9 columns.** We will compare these numbers after applying cleaning techniques

**4.2 Check number of null values in each column**

We will check missing values of each column in “layoff” table, this is the original table.

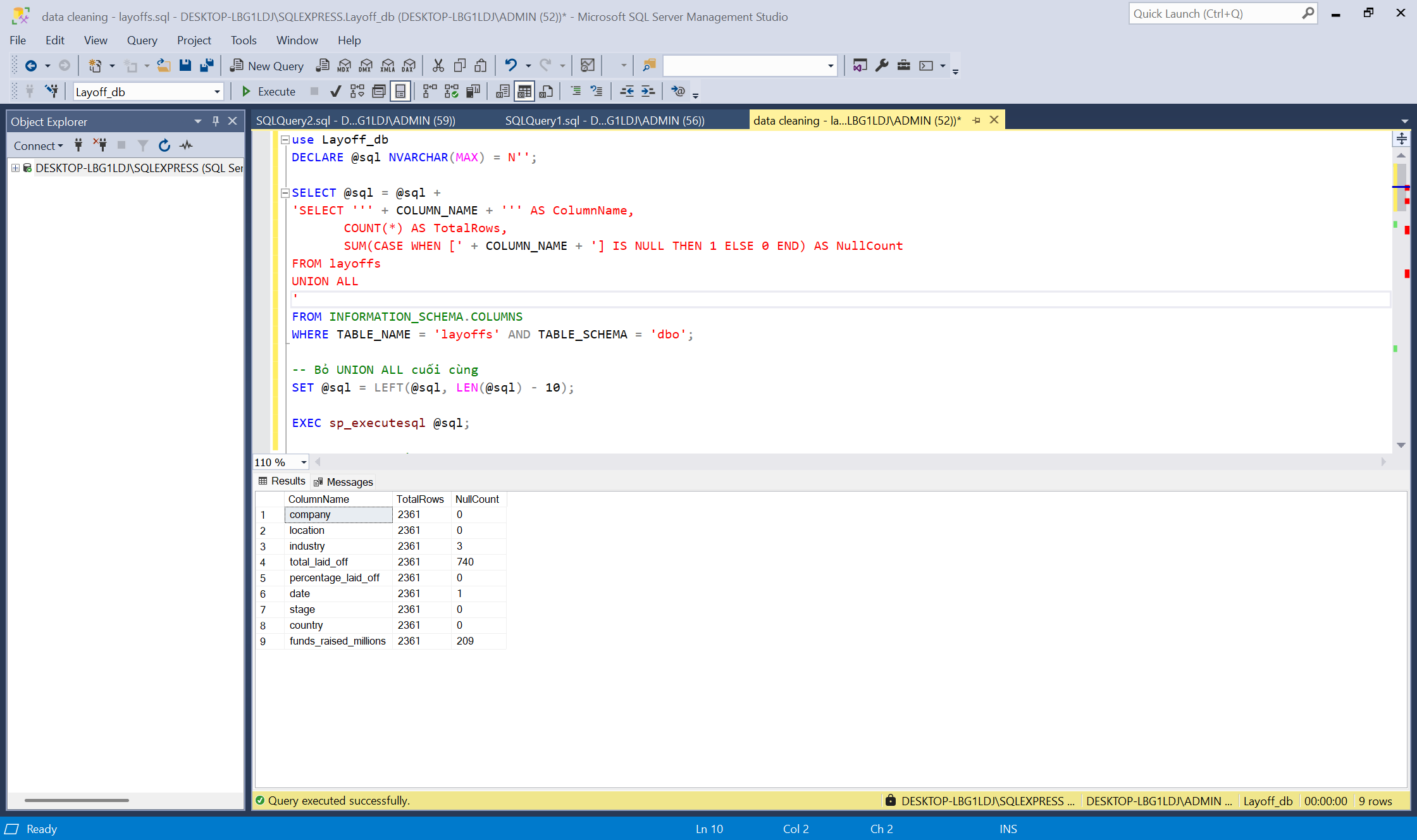


Figure 2: Checking missing values

As we can see, the column **“industry”** has 3 missing values, **“total\_laid\_off”** has 740 missing values, **“date”** has 1 missing value and **“fund\_raised\_millions”** has 209 missing values. Next we will find if there are any invalid values in the table.

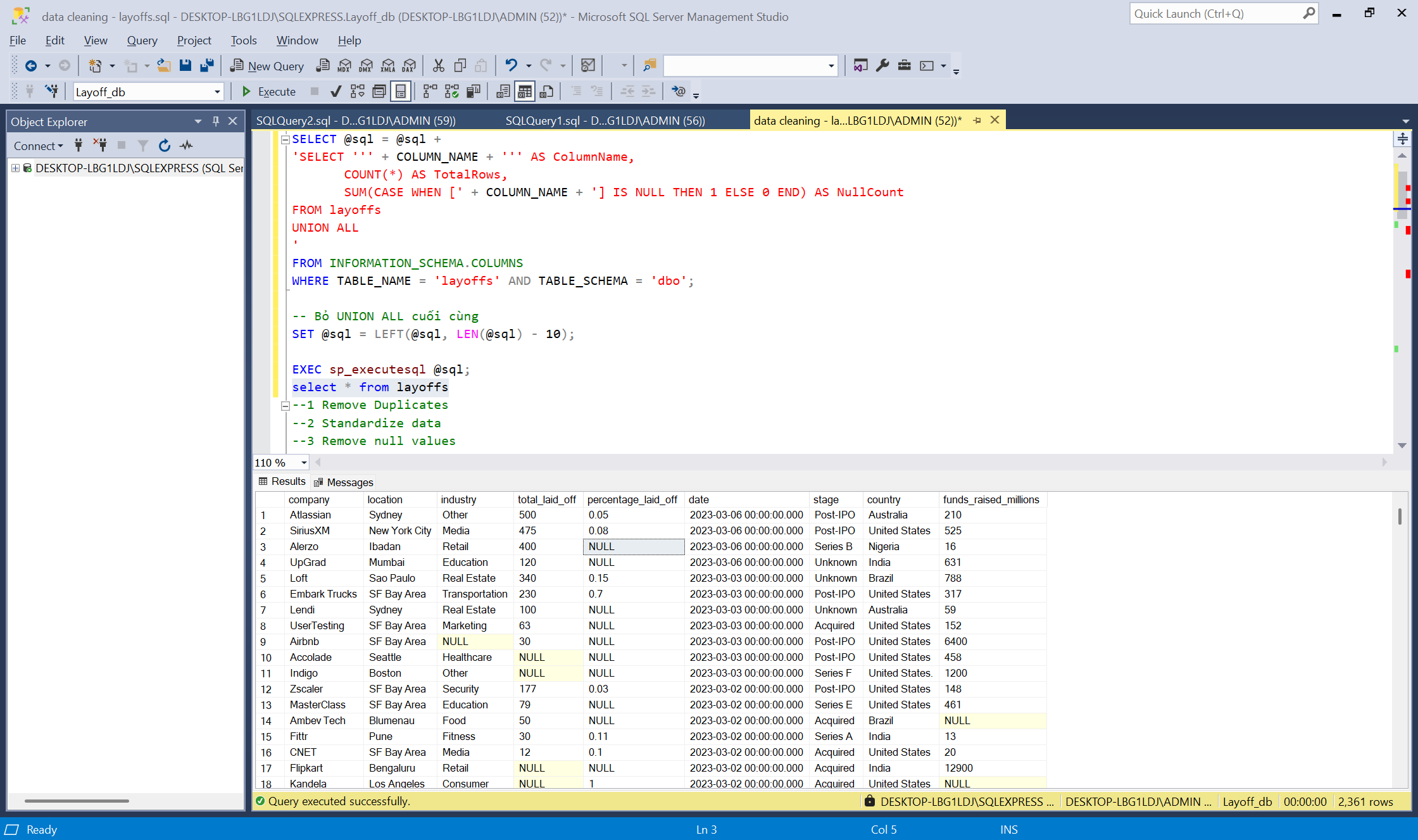


Figure 3: Checkign invalid values

In “percentage\_laid\_off” column, there is no missing value but there are “NULL” words in the column, we will replace them later. Now we have a summary table to list the characteristics of the columns we need to clean and transform.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Missing values** | **How many missing values** | **Invalid values** | **How many invalid values** |
| industry | ☑ | 3 |  |  |
| total\_laid\_off | ☑ | 740 |  |  |
| date | ☑ | 1 |  |  |
| fund\_raised\_millions | ☑ | 209 |  |  |
| percentage\_laid\_off |  |  | ☑ | 785 |

1. **Apply Data Cleaning Techniques**

We have three main tasks in this part:

**5.1 Remove Duplicates**

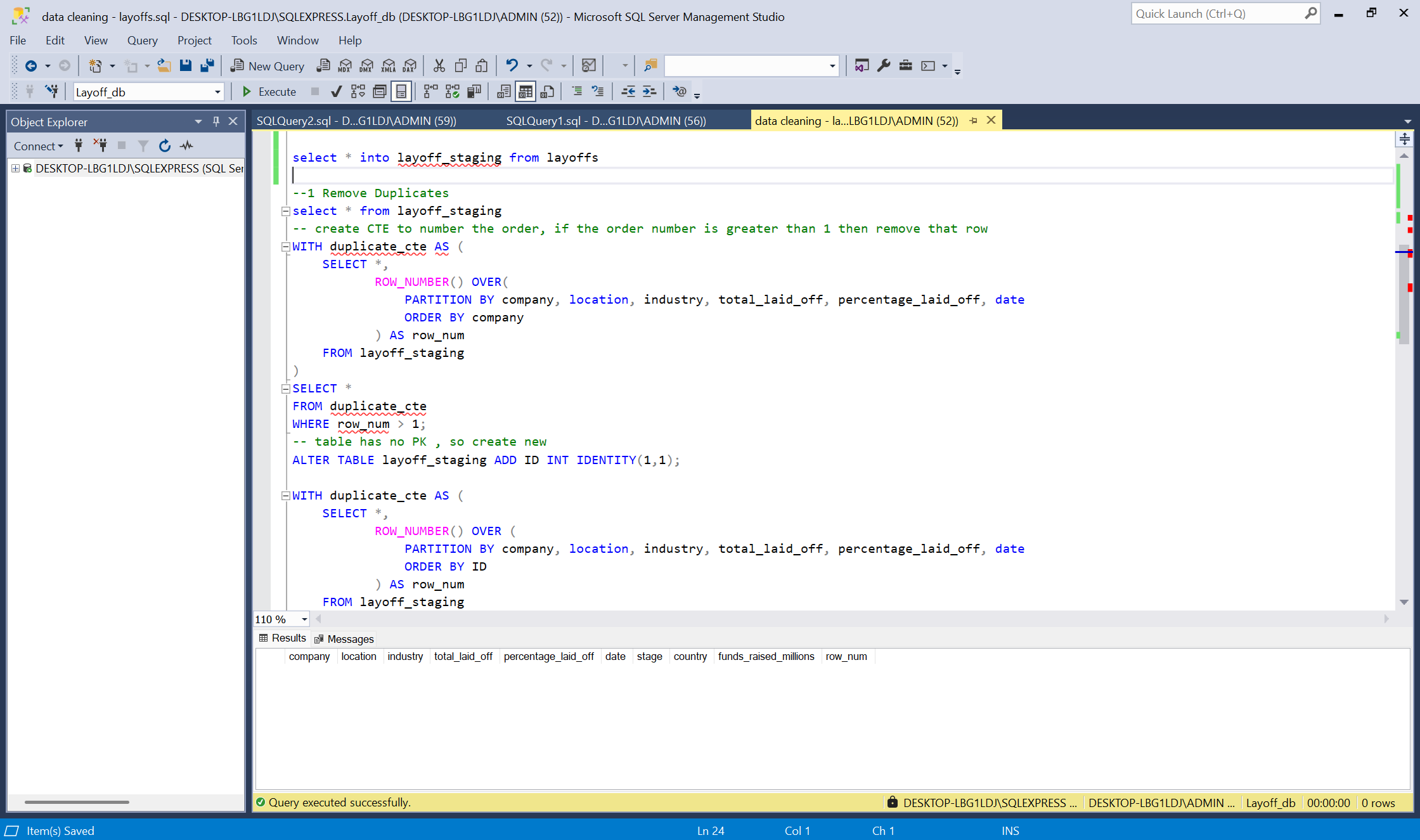


Figure 4: Remove duplicates

We apply CTE for remove duplicates instead of using nest queries. The first part is assigned value for all records in the table, if the record is unique, the row number is 1. If the record has duplicate values, the row number is 1, 2, 3 and so on. Because the table has no Primary Key, we will assign row number as Primary Key. After that, we apply CTE once again and delete every row which larger than 1. And the final output is nothing, which mean the duplicate values have been removed.

**5.2 Standardazing the data**

We will select distinct value for each column to examine any unnormalized data.

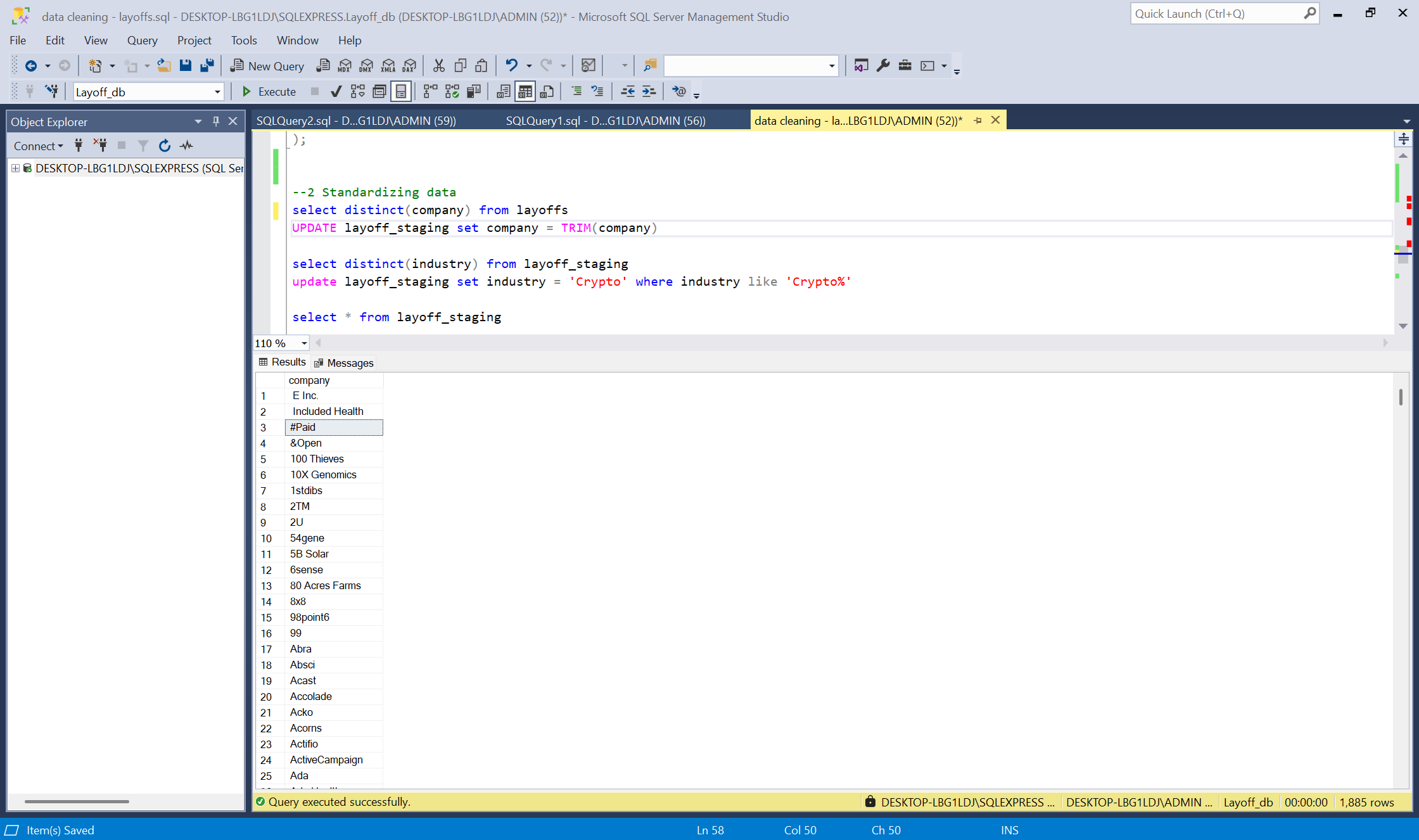


Figure 5: Normalized column “company”

This column is good, so we don’t need to worry about it.

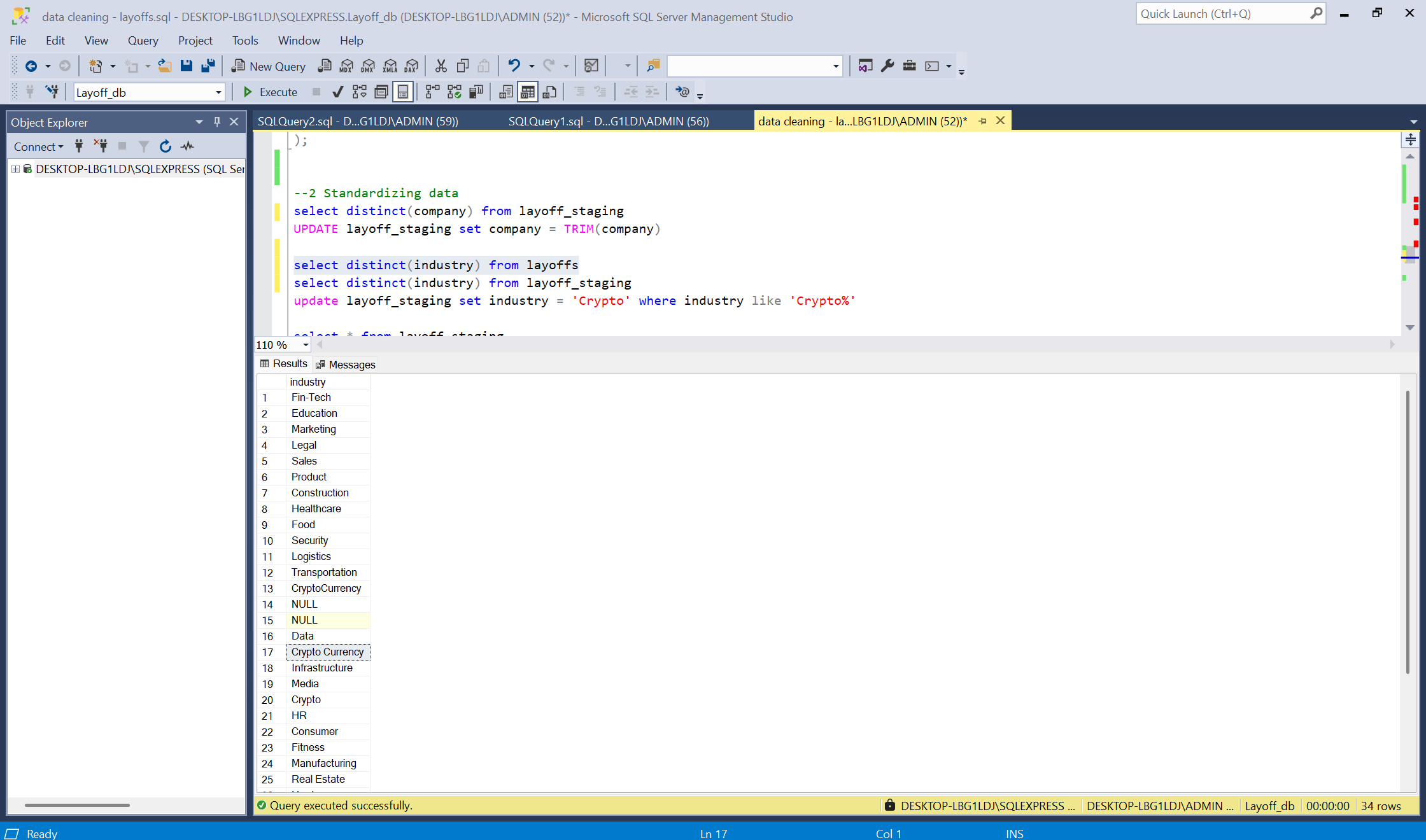
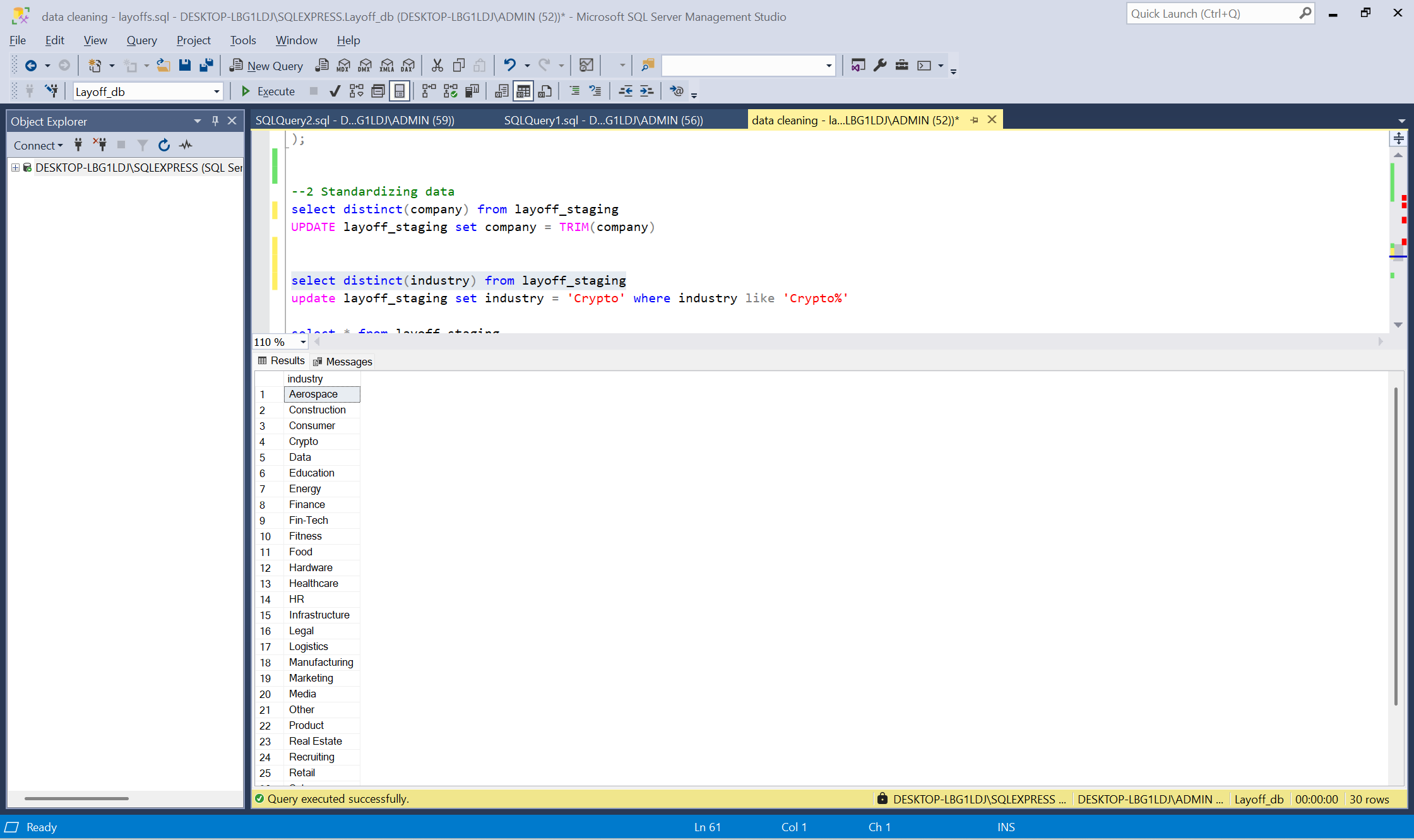


Figure 5: Normalized column “industry”

The name “CryptoCurrency” and “Crypto Currency” are not the same, we will transform them to “Crypto”.



This is the result after standardazing column “industry”. There are no missing values and industry “Crypto” is stanrdardized. To avoid repititve tasks, I wll skip other columns, you can see on the image to have more details.

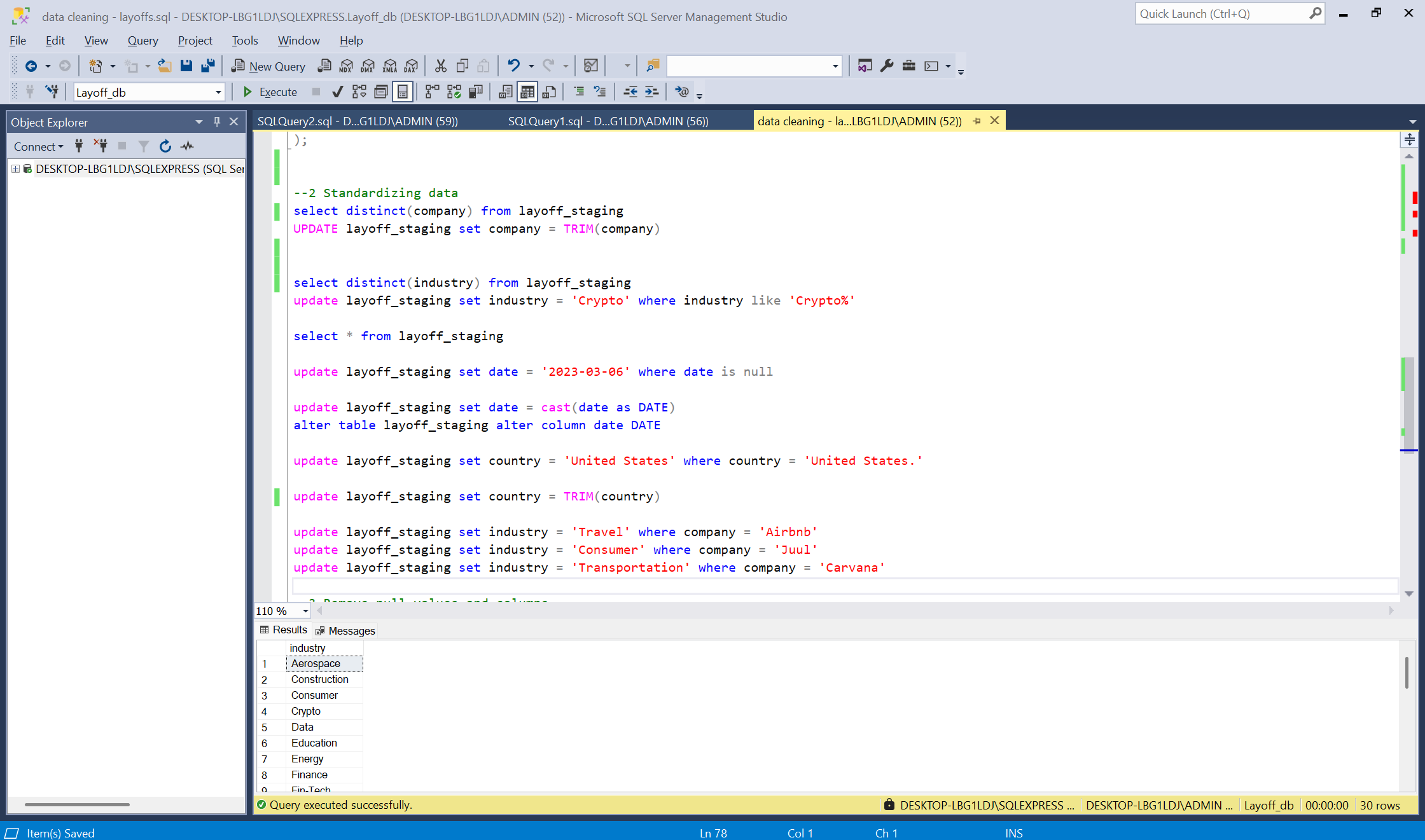


Figure 6: All columns needed to be standardized

**5.3 Remove missing values and invalid values**

The final part is removing missing values. You can read on part 4.2 to know how many columns have missing values and invalid values.

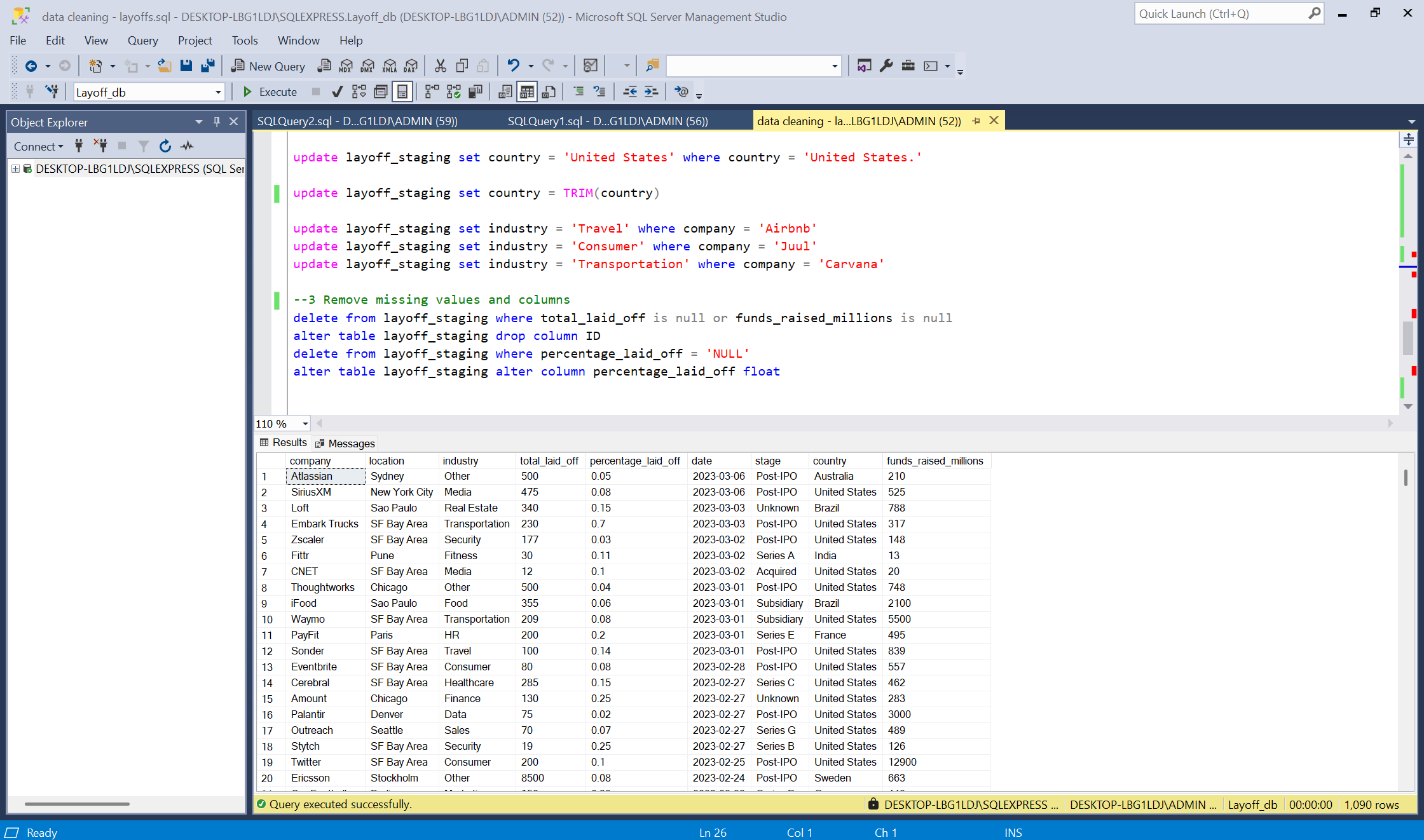


Figure 7: Final result

Now we have a table to compare total rows and columns before and after cleaning and transform dataset.

|  |  |  |
| --- | --- | --- |
|  | **Before** | **After** |
| **Rows** | 2361 | 1090 |
| **Columns** | 9 | 9 |

1. **Tools and best practices**

* Use **temporary tables** or **views** for intermediate cleaning steps.
* Using CTE and row\_number() function to remove duplicate values.
* Replace missing values with the current and relevant data, don’t delete them if you don’t have to. Using text functions to implement.
* Document every step for transparency and reproducibility.

1. **Conclusion**

Data cleaning in SQL is a critical process that ensures the accuracy and consistency of organizational data. By applying SQL functions and best practices, analysts and database administrators can maintain high-quality data, minimize errors, and improve decision-making efficiency.