Cloud Dataprep Demo: Inconsistent company names

Standardising values in a text column

Celia Muriel – celiamuriel.com
Post, GitHub repository.

Introduction

This demo shows how to remove inconsistent data in a CSV file and load it to BigQuery. The source file has company names in a column. Their values are inconsistent (VANILLA Ltd, *** VANILLA LTD ***, vanilla ltd, vanila ltd, Vanilla Ltd., etc.). We are going to standardise the company names before uploading them to the target table.

There is no one single technique to achieve this task. We are going to use several out-of-the-box solutions implemented in Cloud Dataprep (by Trifacta), including clustering of values based on fuzzy matching.

This demo was done on June 13th, 2020, with the Generally Available features on the different services used for this exercise.

Prepare demo environment

Source file in Cloud Storage

Create a Cloud Storage bucket for the demo. Upload to the bucket the source file.

Target BigQuery table

Create the BigQuery dataset and table as shown below.

```
bq --location EU mk \
--dataset \
[PROJECT ID]:vanilla

bq mk \
--table \
[PROJECT ID]:vanilla.companies \
company_name:STRING,company_id1:STRING,company_id2:STRING,company_id3:STRI
NG,company_id4:STRING,country:STRING,town:STRING,zipcode:STRING
```

Enable and setup Cloud Dataprep

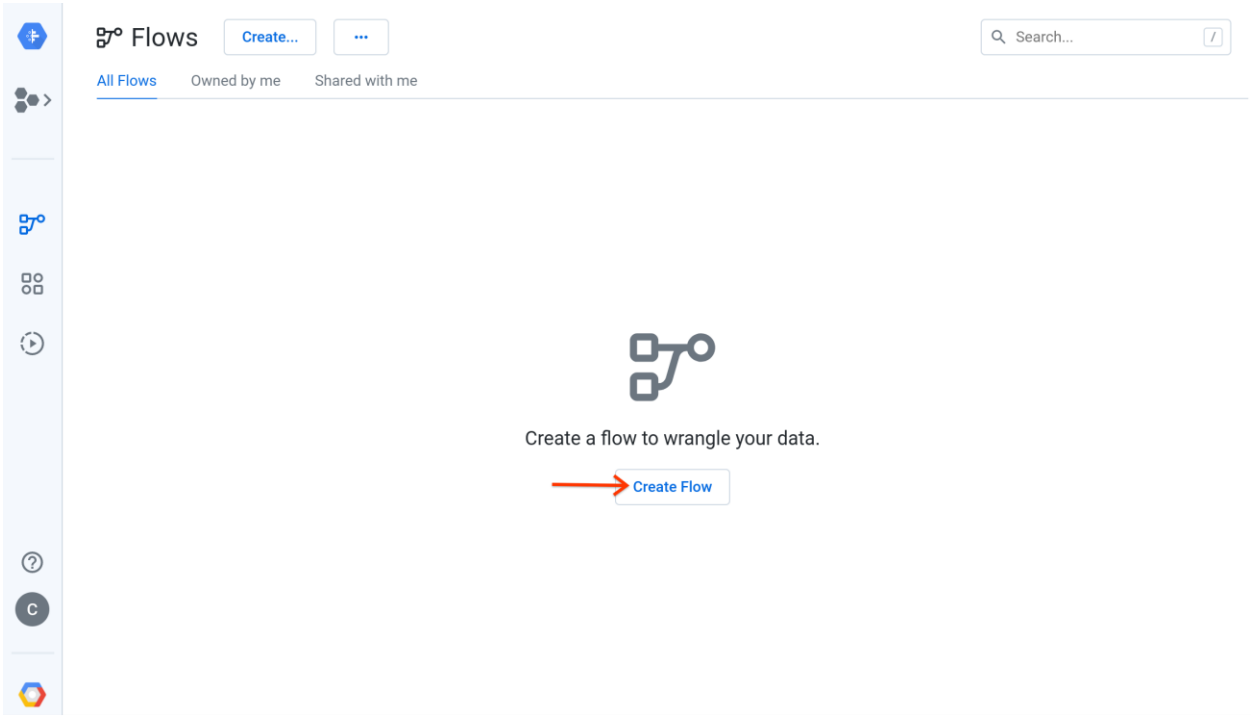
If we want to use Cloud Dataprep, we need our account to grant permissions to any of these roles: project editor, project owner or dataprep.user.

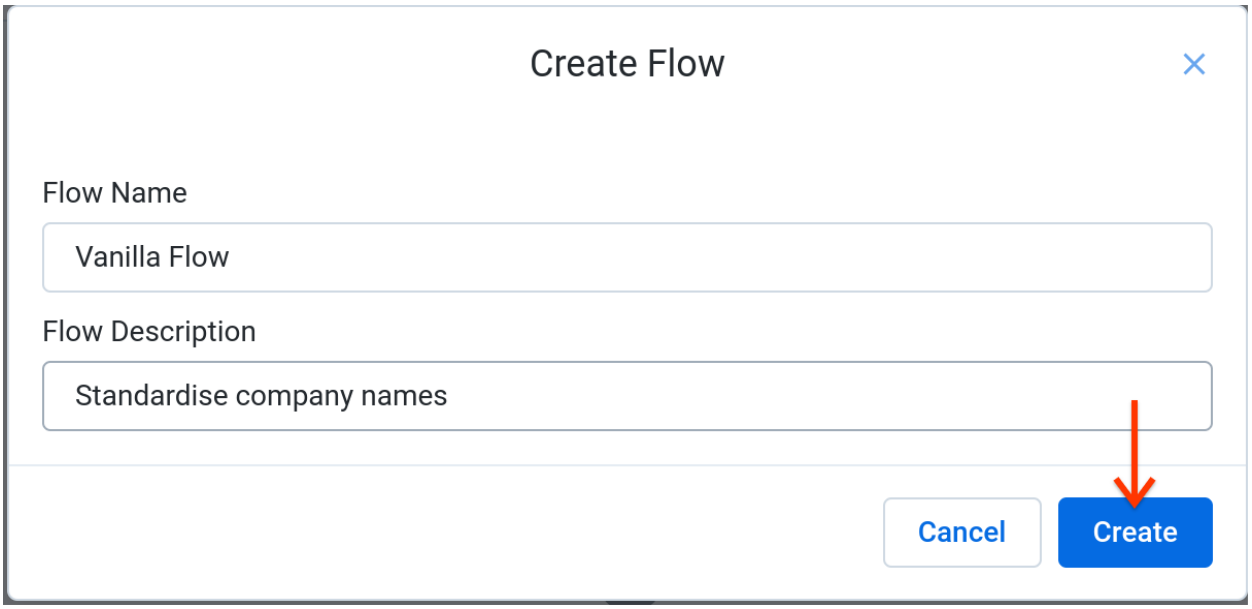
Then we need to enable Cloud Dataprep and set it up. If you are running your ecosystem outside the US, and you have requirements to be in a certain region, make sure that the Cloud Storage buckets Cloud Dataprep are set up according to your requirements.

Remove inconsistent values

Standardise company names

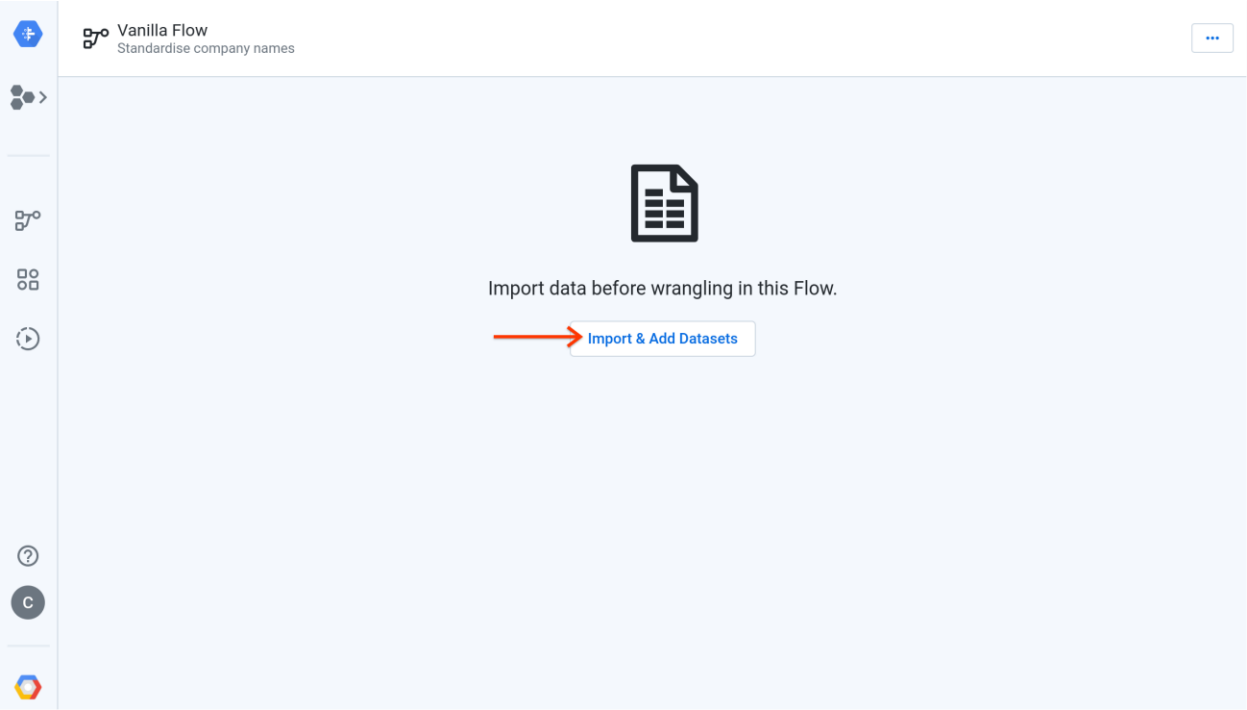
Once we have enabled and set up Cloud Dataprep, we create a flow.

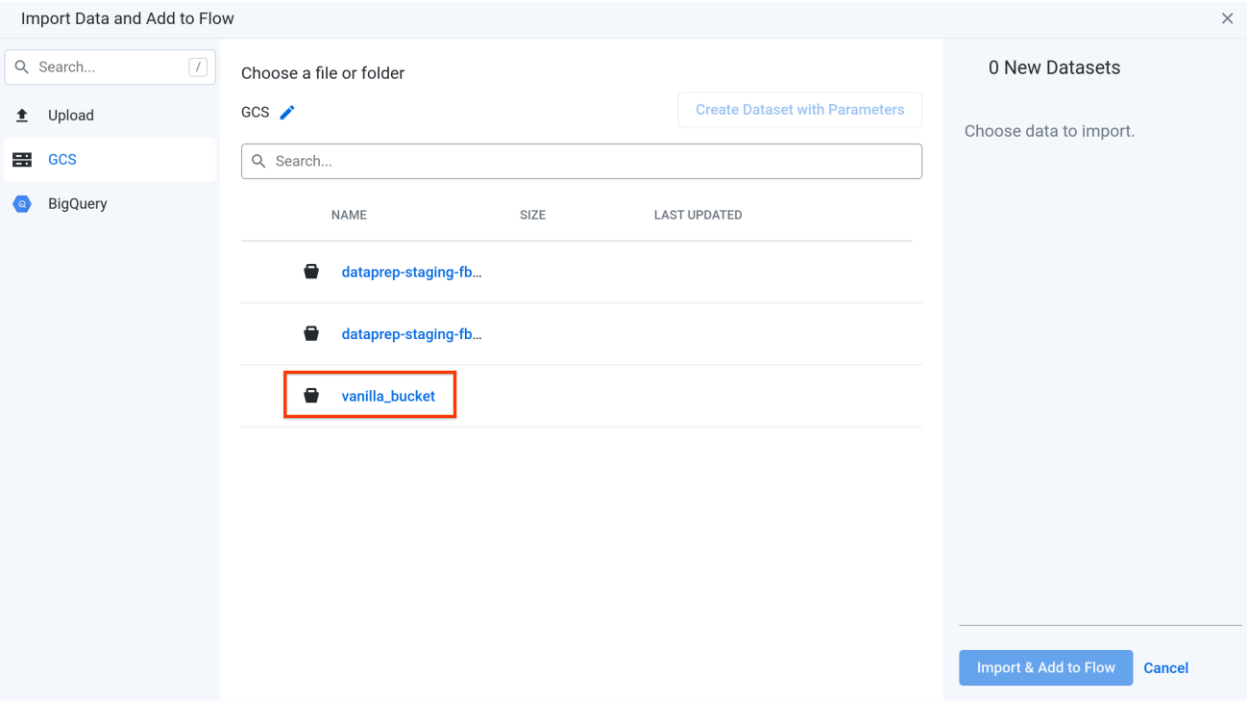


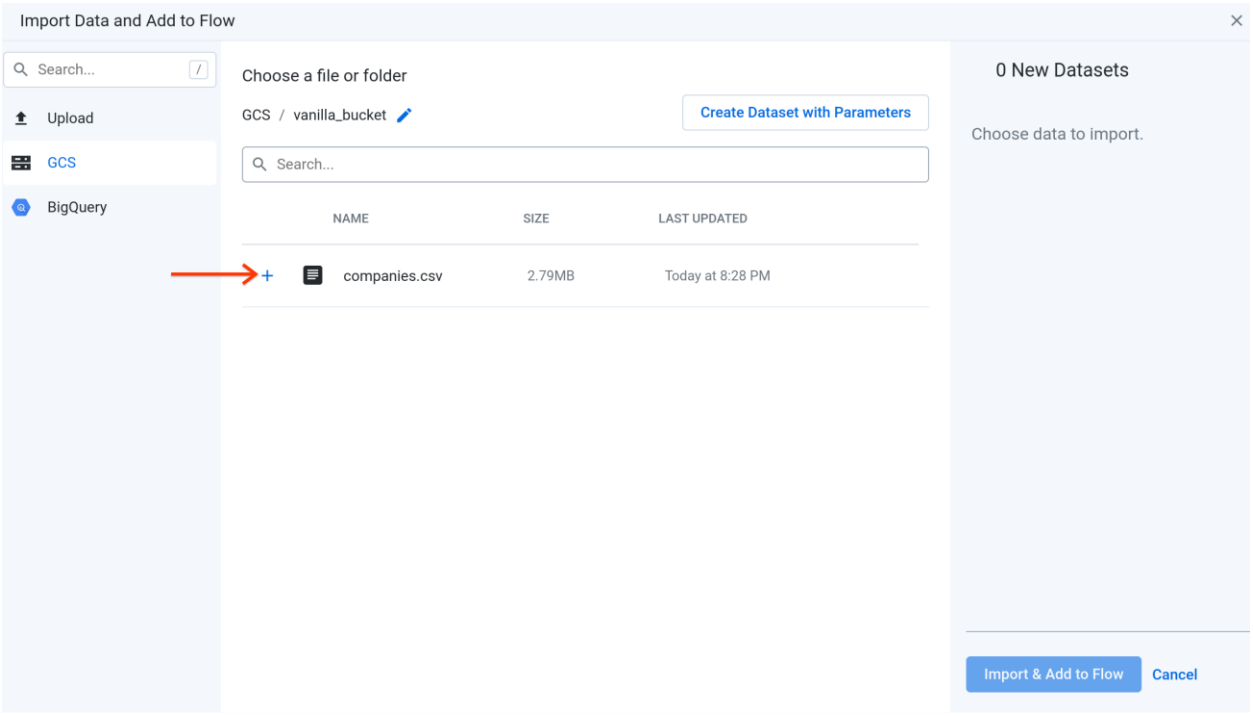


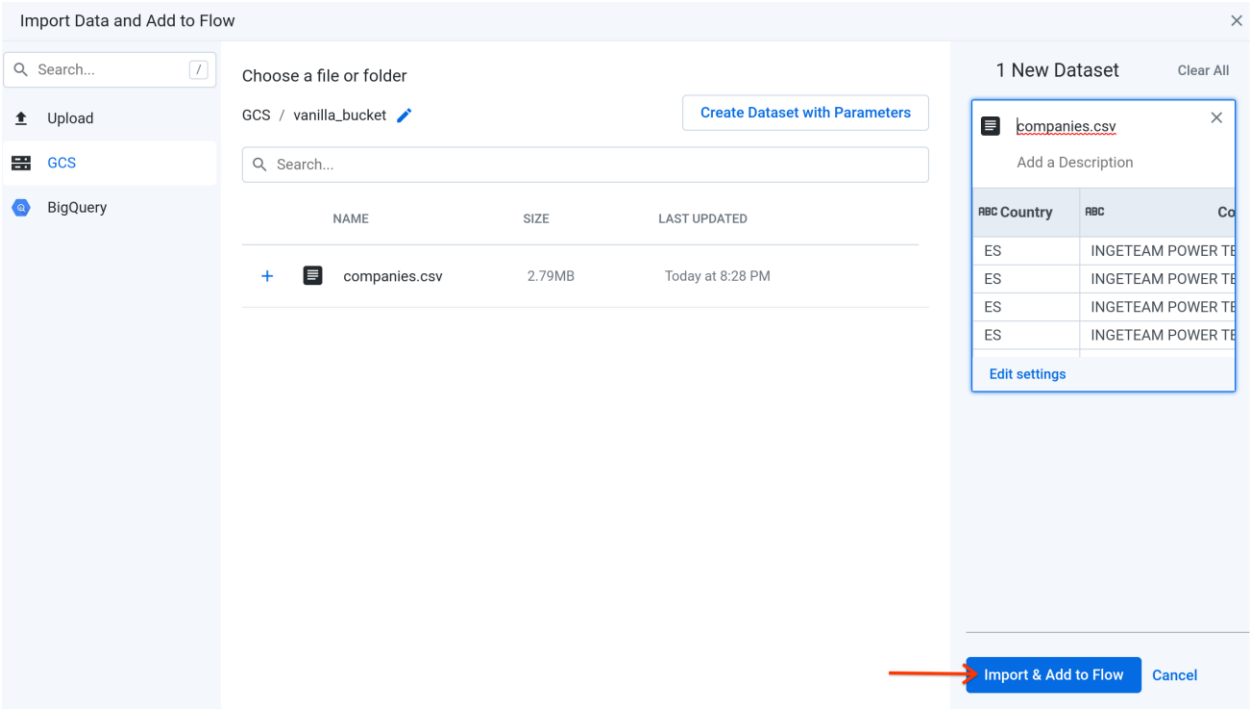
Then we add the companies.csv file as a dataset.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



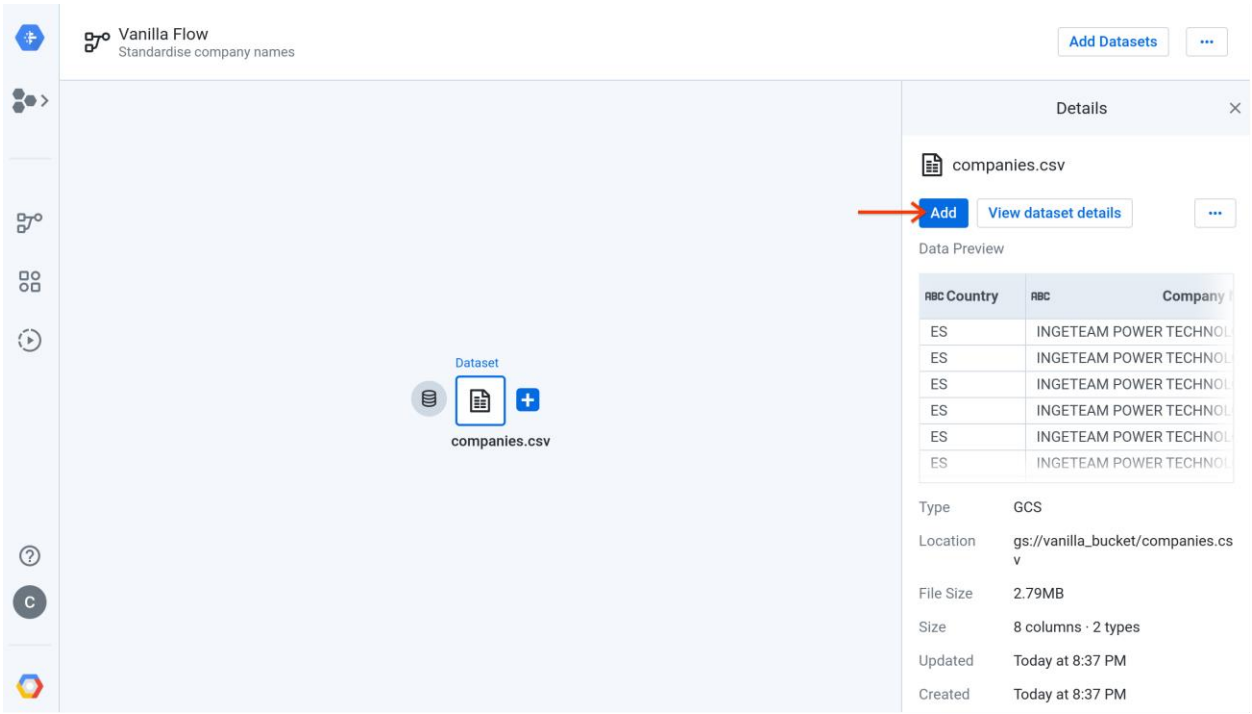


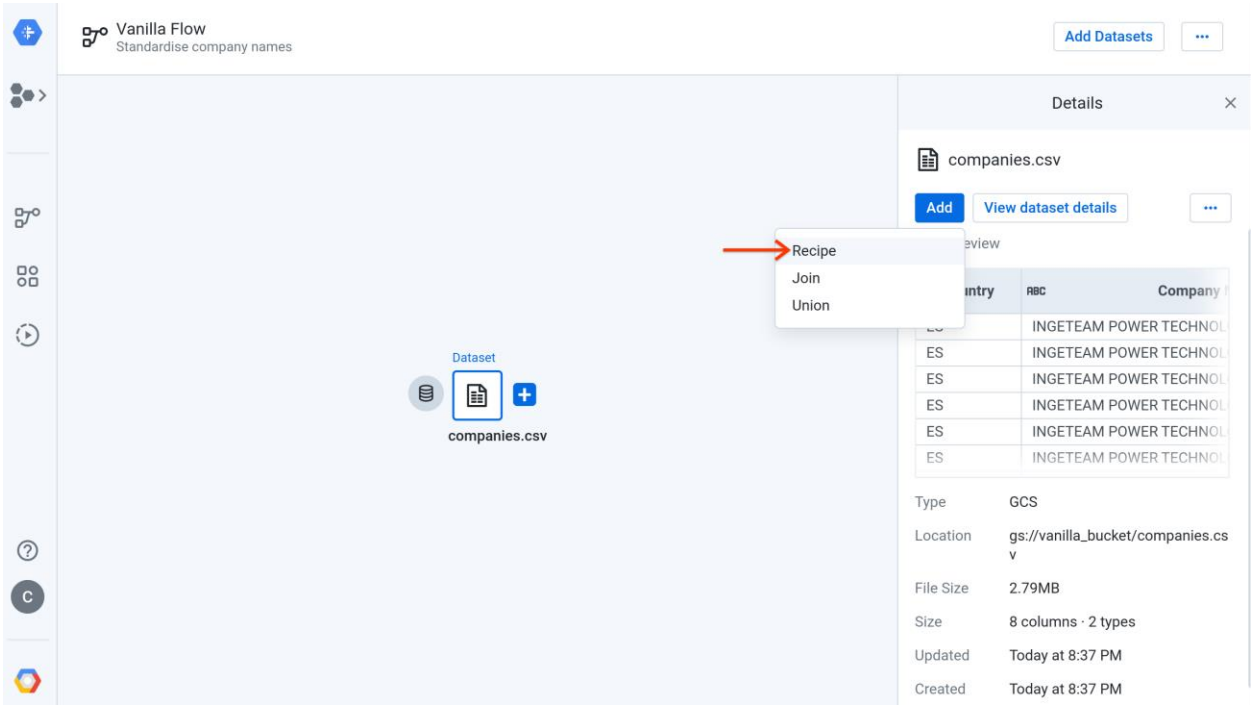


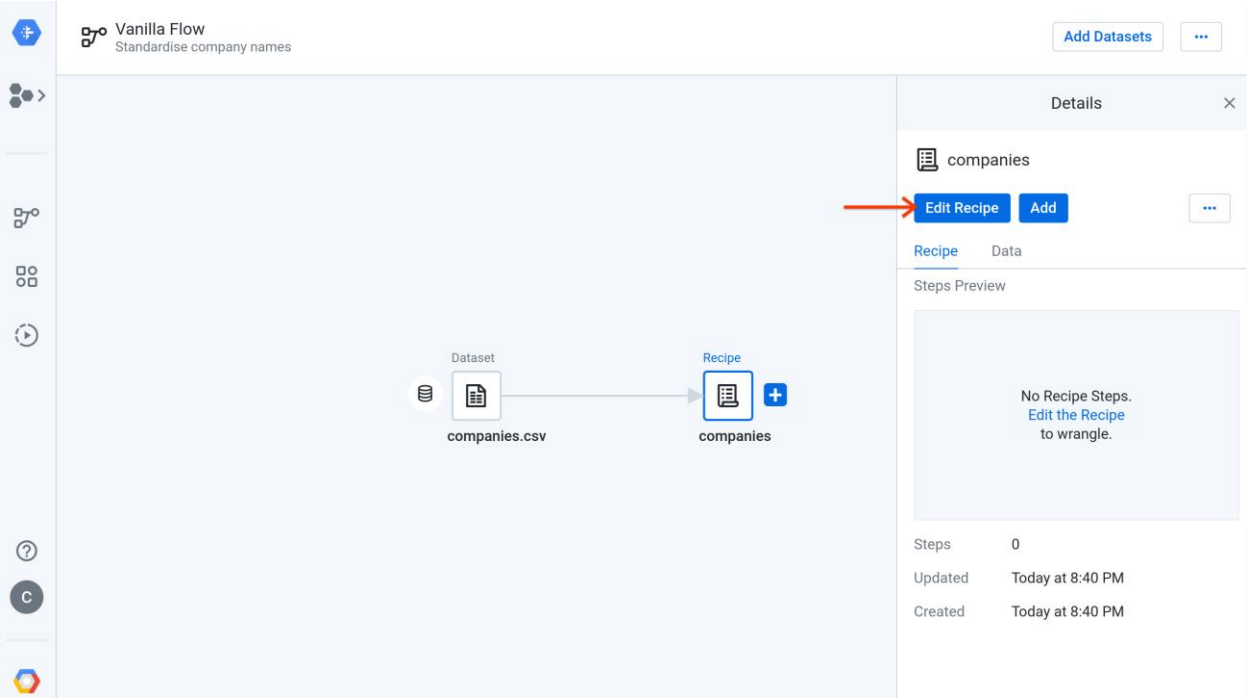


We must create a recipe to clean and prepare the data in companies.csv to upload it to BigQuery.

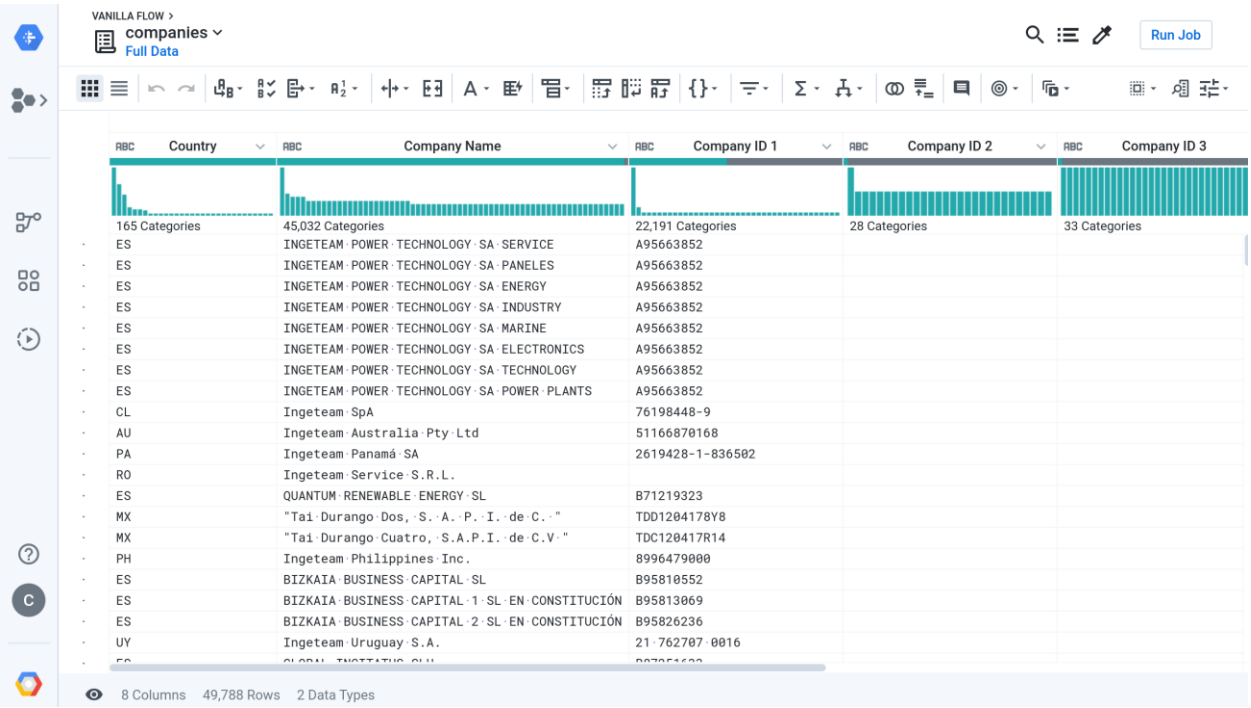
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



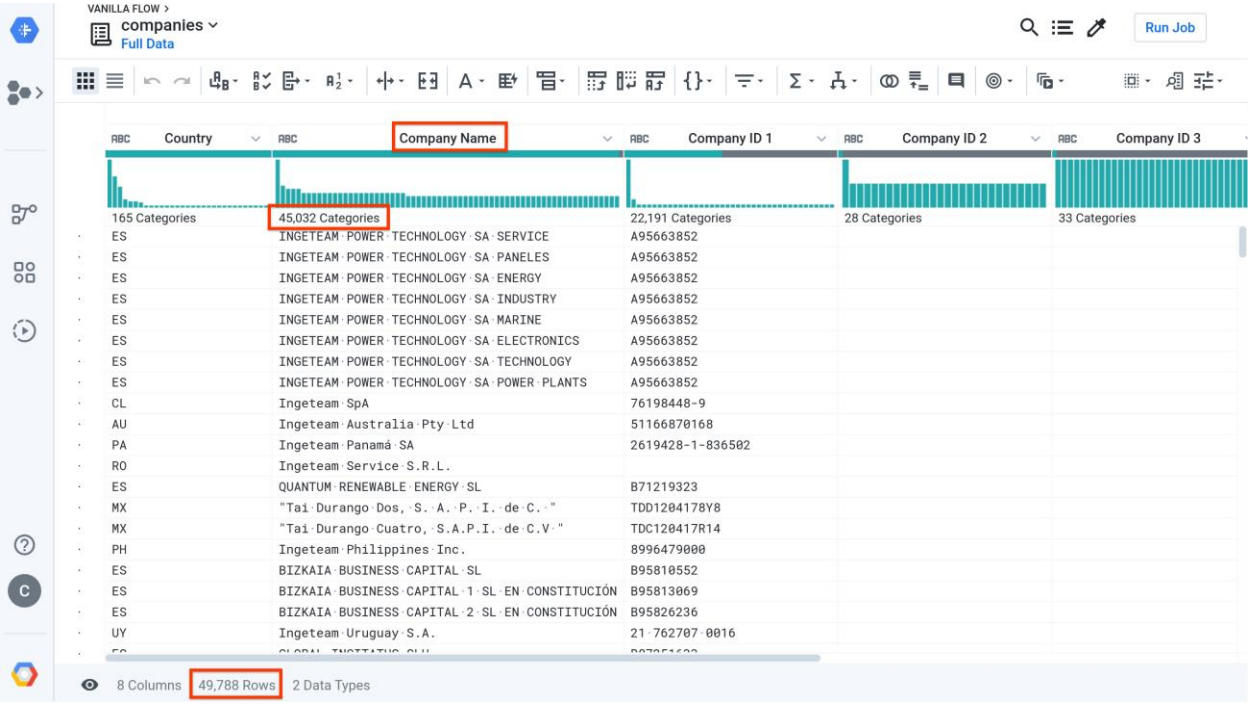




When we edit the recipe for the first time, it shows us the data in the CSV file. Dataprep allows to quickly analyse the data and add the transformations easily.

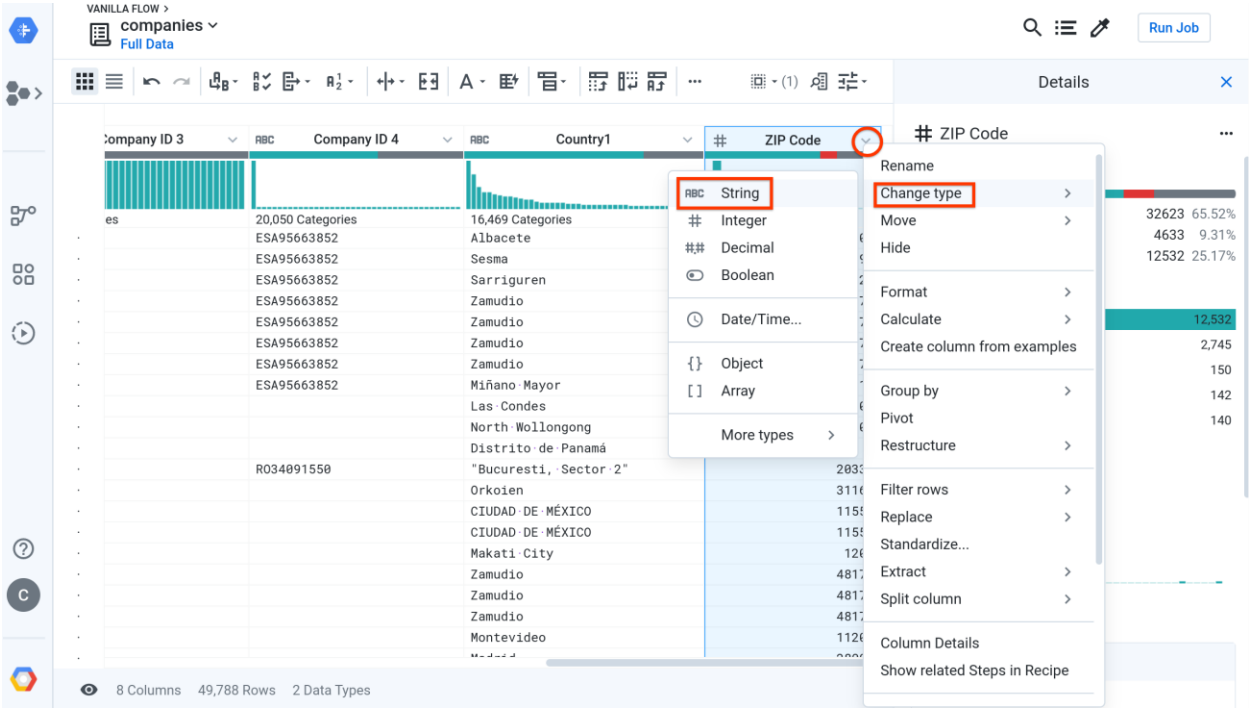


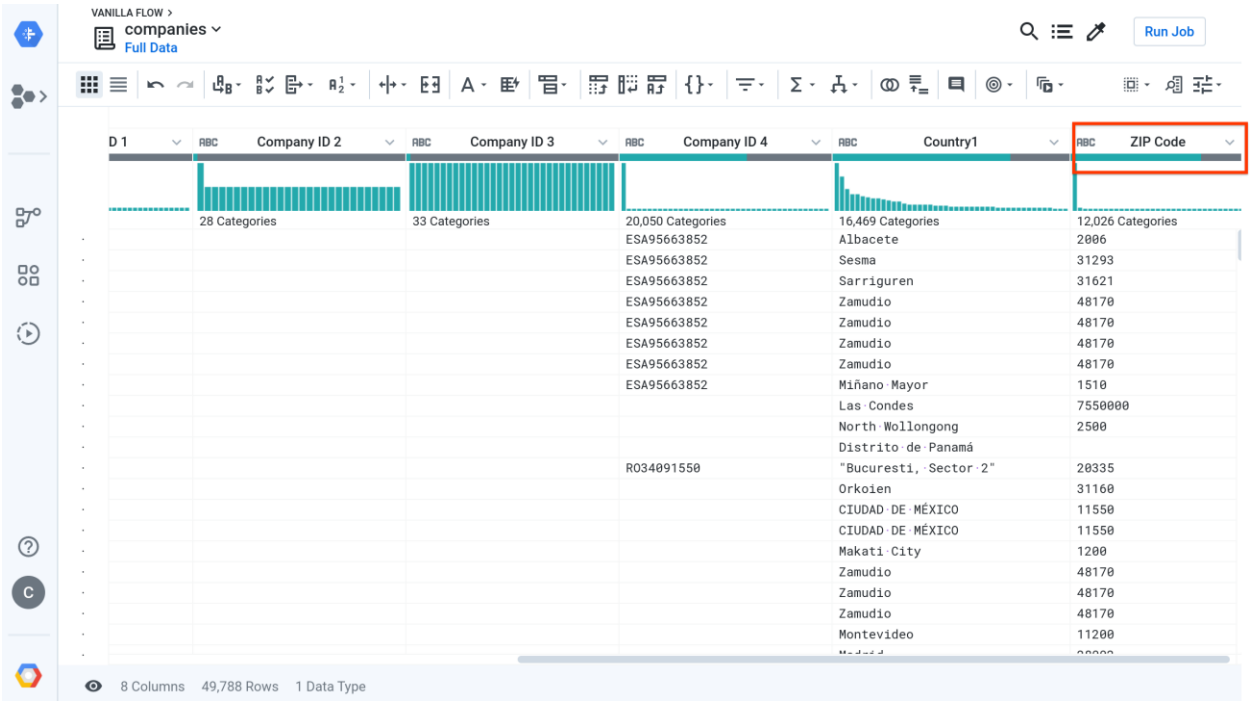
In this demo, we are going to clean and prepare three fields, but we are going to focus on the company name (Vanilla in companies.csv). See that when we open the CSV file before standardising the companies, there are 49,788 rows in the file and 45,032 different company names.



If we scroll right, we see that the quality bar for the ZIP code shows mismatched values. This is due to the fact that most of the ZIP codes in this file are Spanish, which means they are made of 5 digits. However, some companies are outside Spain and the ZIP codes may be alphanumeric. So we change the data type to string.

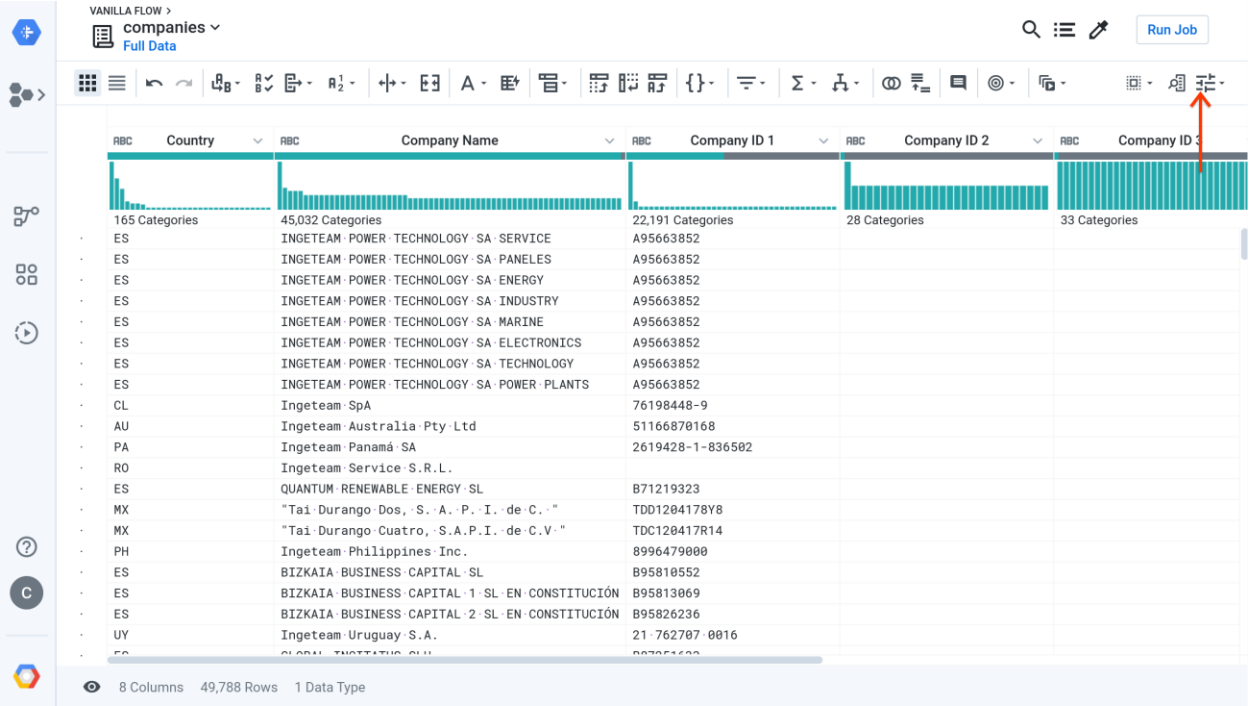
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com





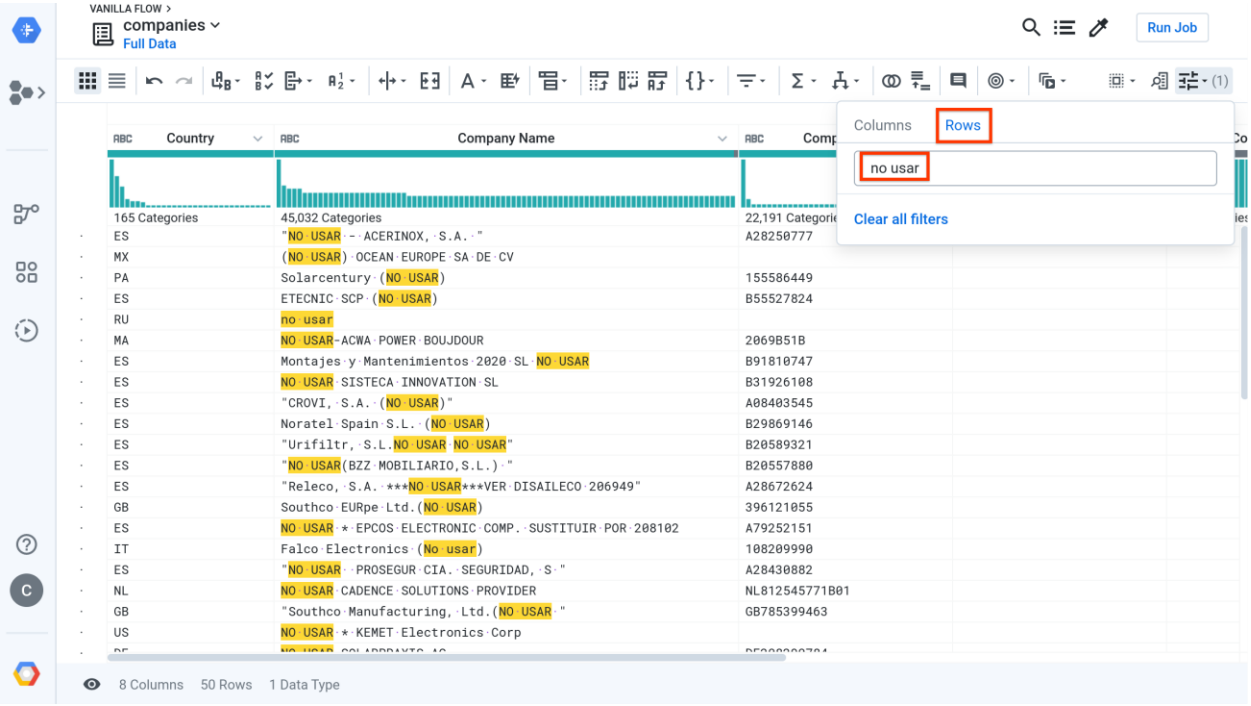
In the source application when someone wanted to indicate that a company name shouldn't be used, they have written "do not use" in Spanish and Italian within the company name. We are

going to remove all those strings as the company names are valid in our historical information and we should use an end date to mark the validity of a company. First we look for those rows.

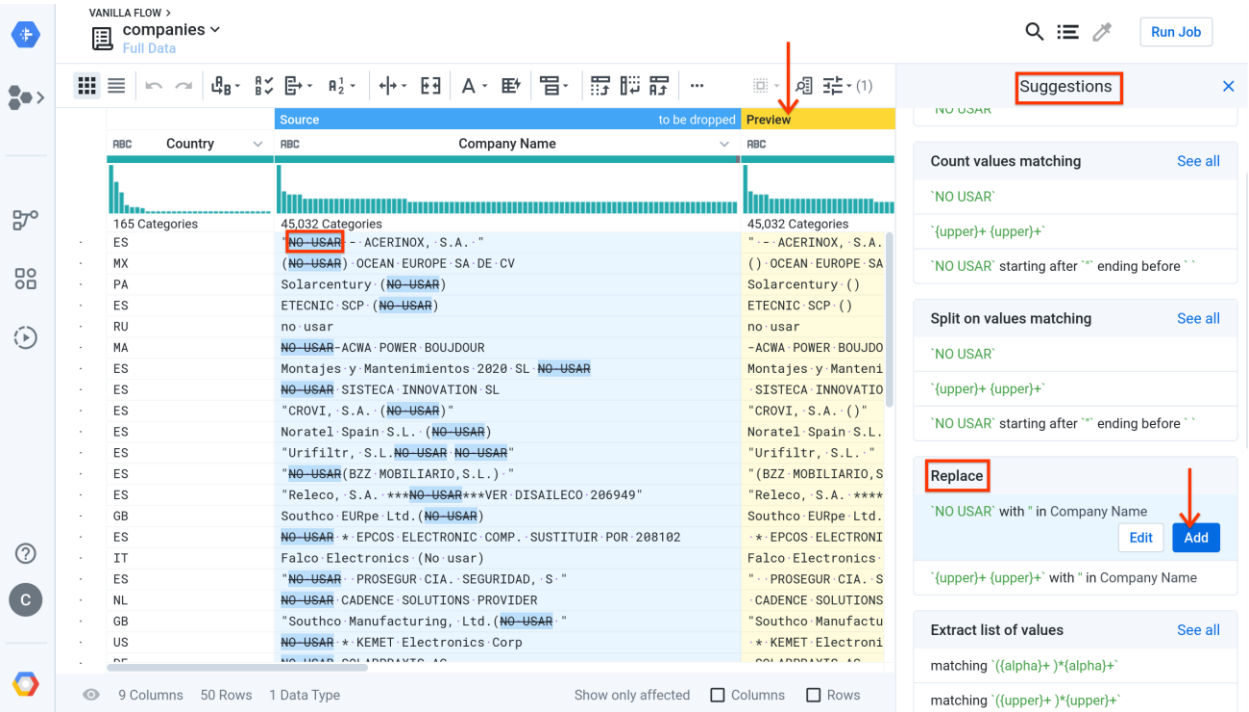


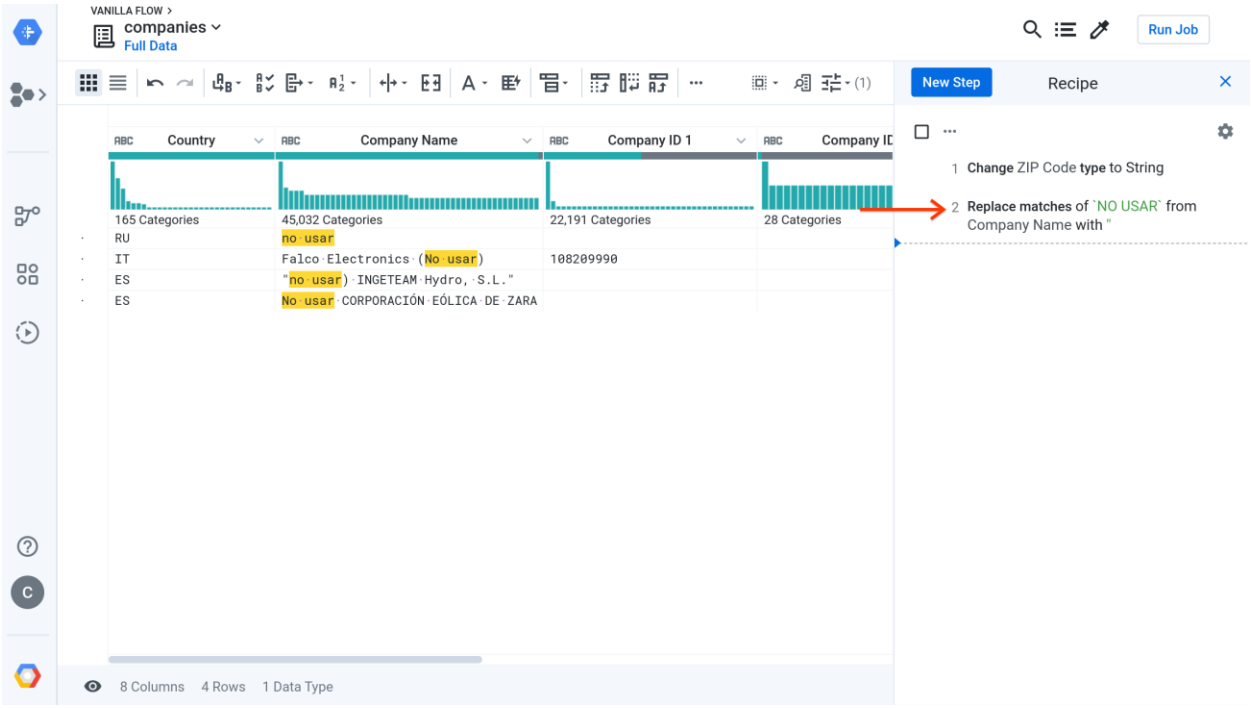
We search for “no usar” (do not use).

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

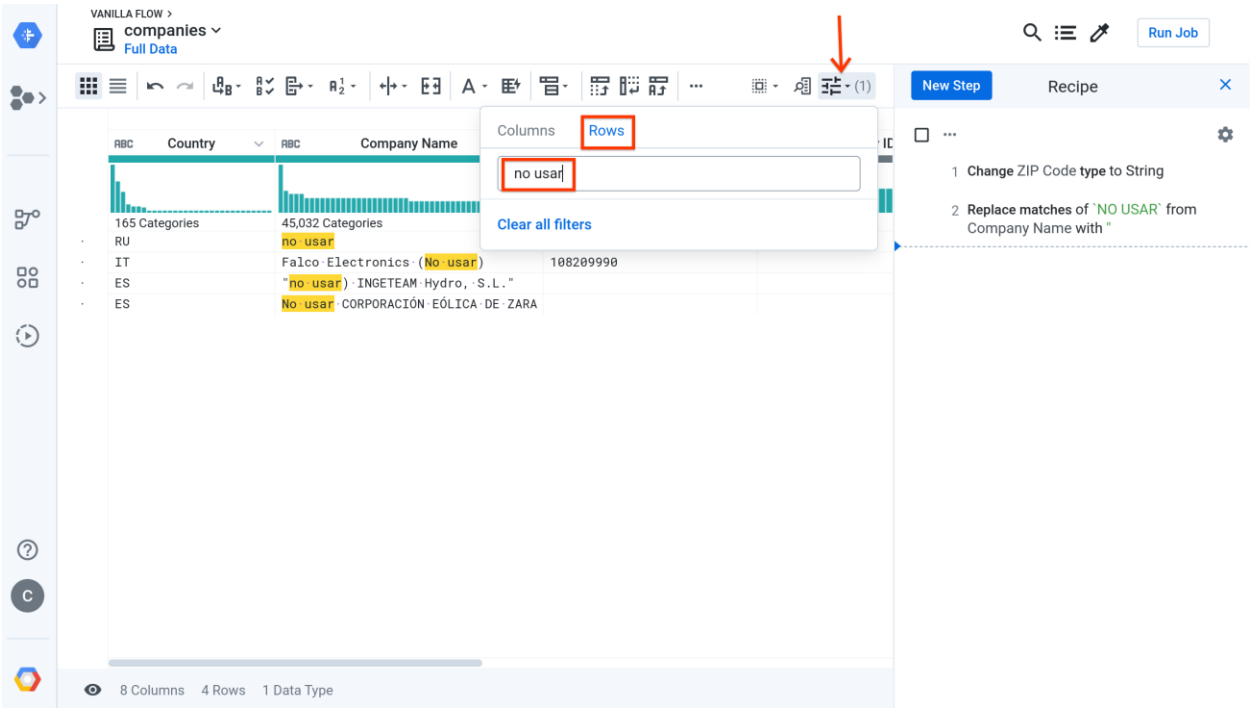


We highlight one of the “no usar” occurrences. A Suggestions wizard shows on the right. We choose Replace with ‘’. A yellow column appears which shows how the company names will look after the replacement. We click on “add”.

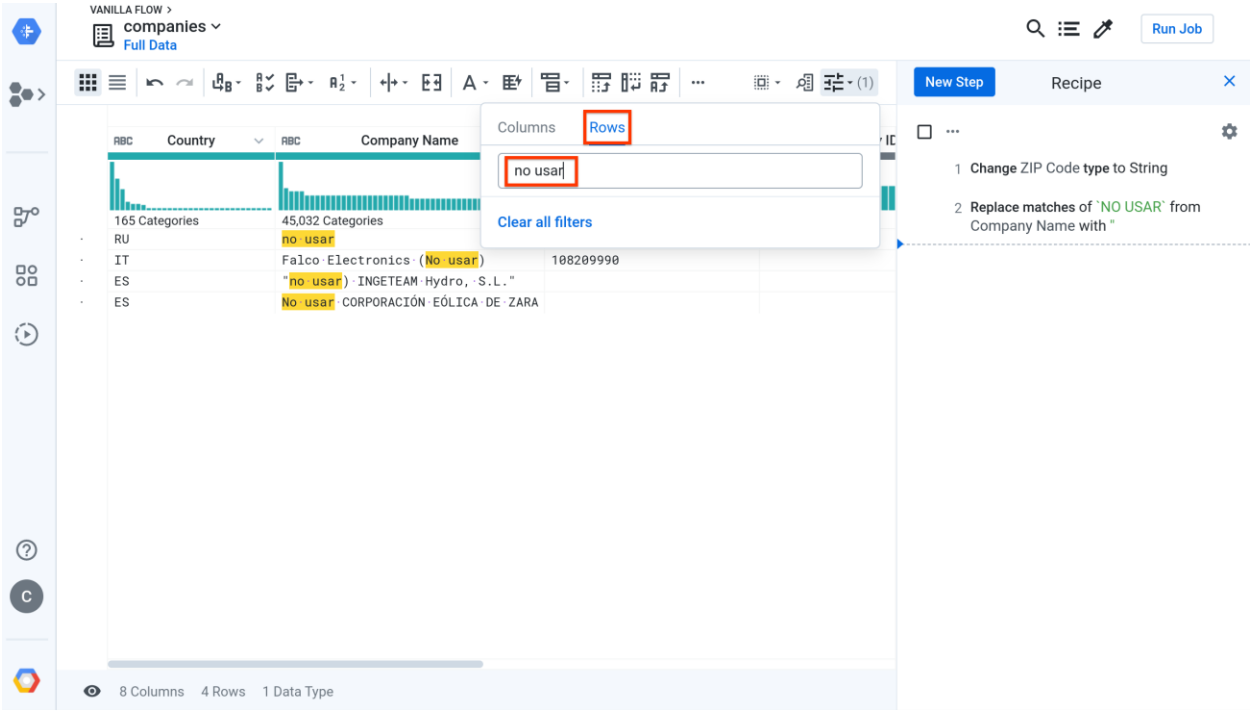




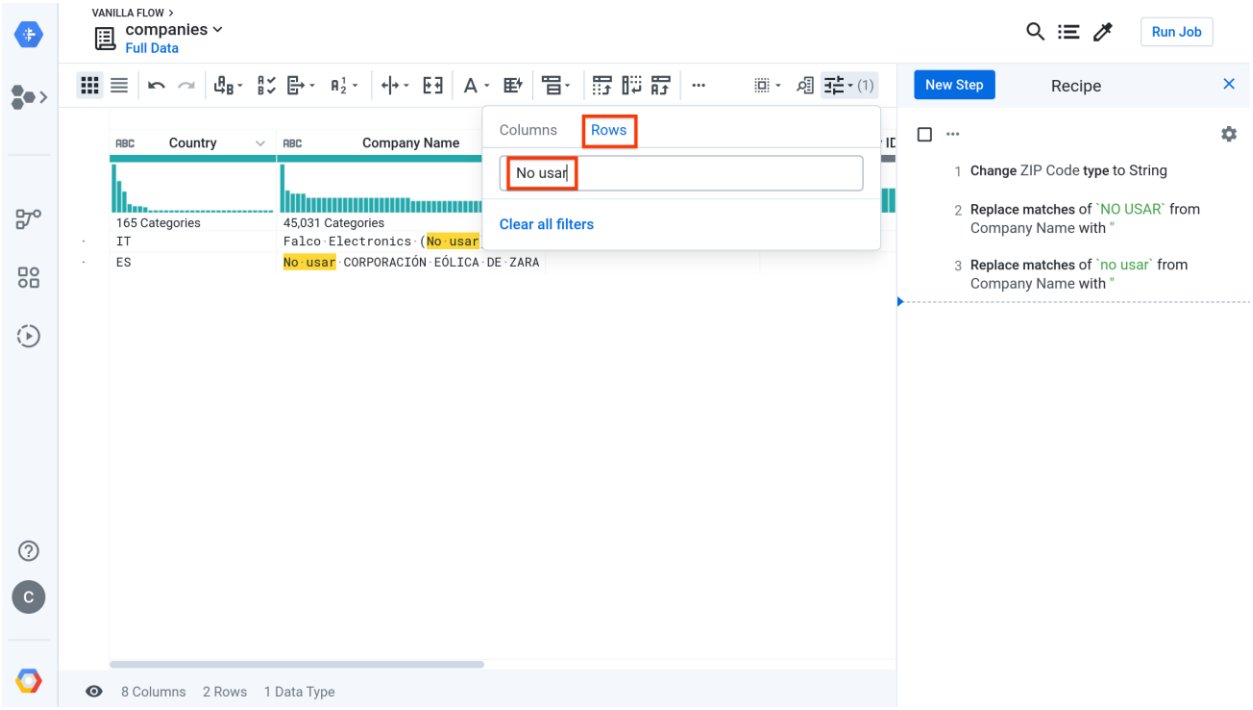
We click on the filter again and on “Clear all filters”.

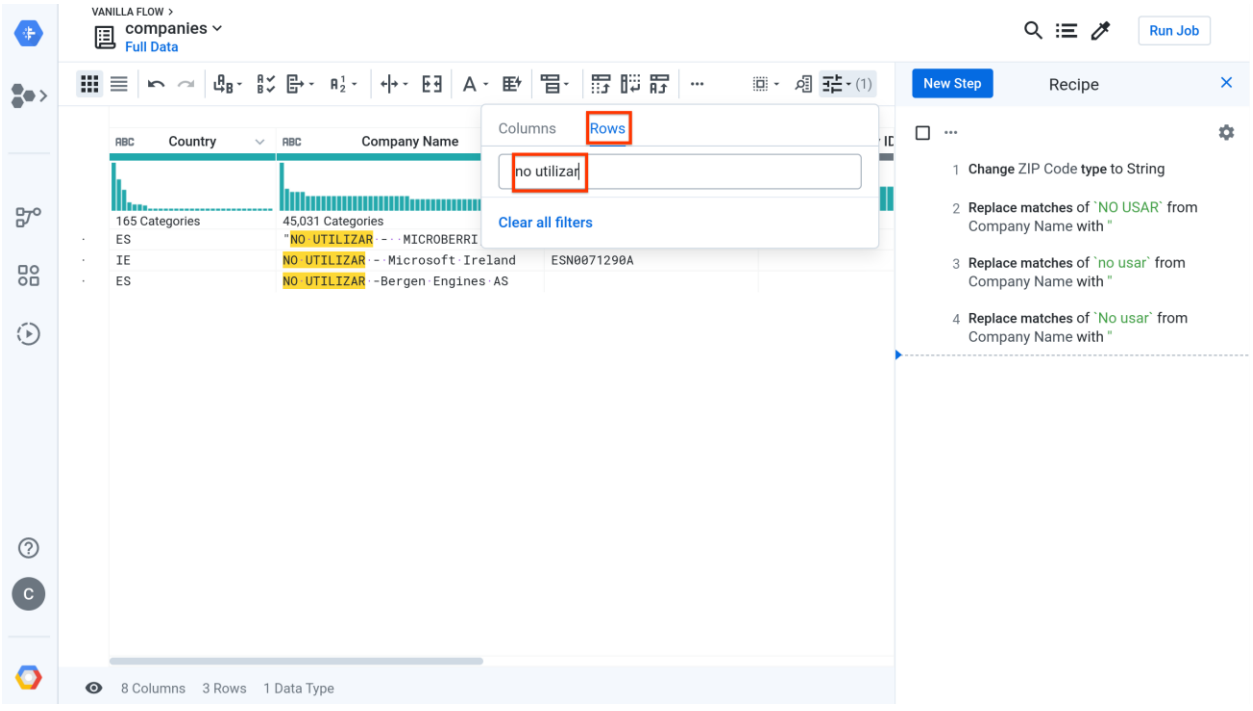


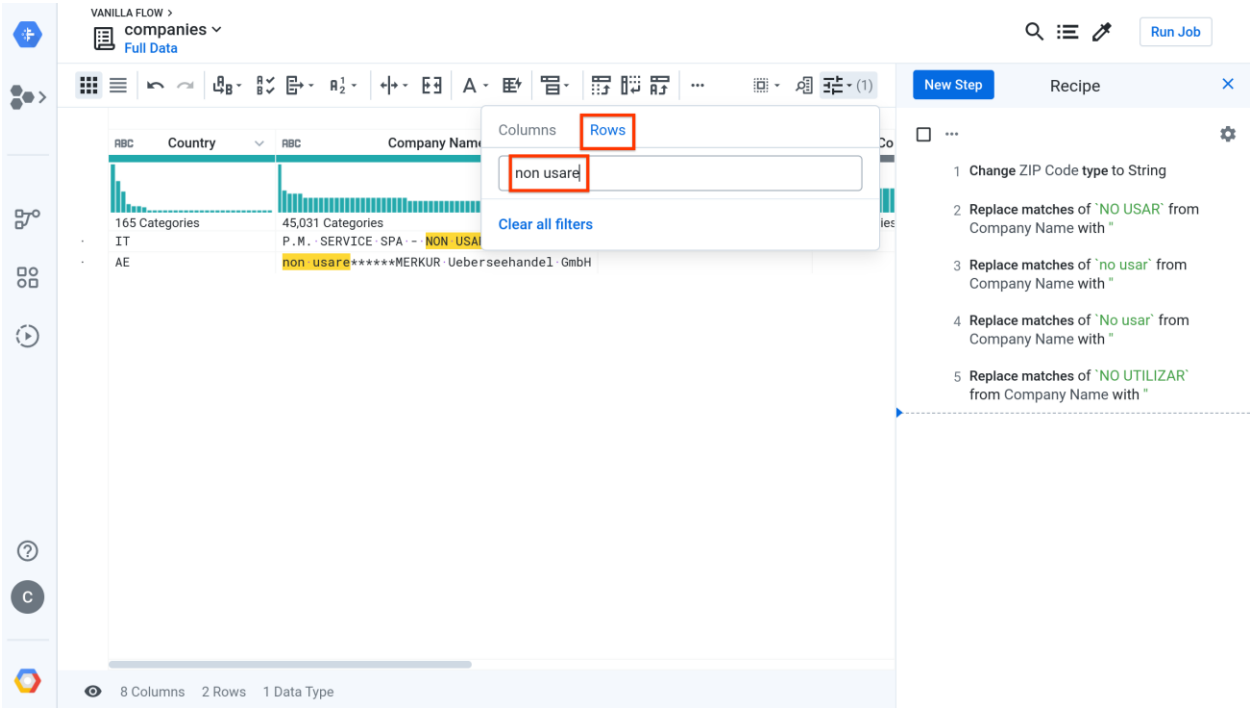
We continue searching for the “do not use” tags within the company name, and we remove them as done with “no usar”. Note that Cloud Dataprep is sensitive to capitalization. If we search for “do not use”, we find the records which have “DO NOT USE” and “do not use”. We must highlight and remove the occurrence in capital letters and the one in lowercase as well.

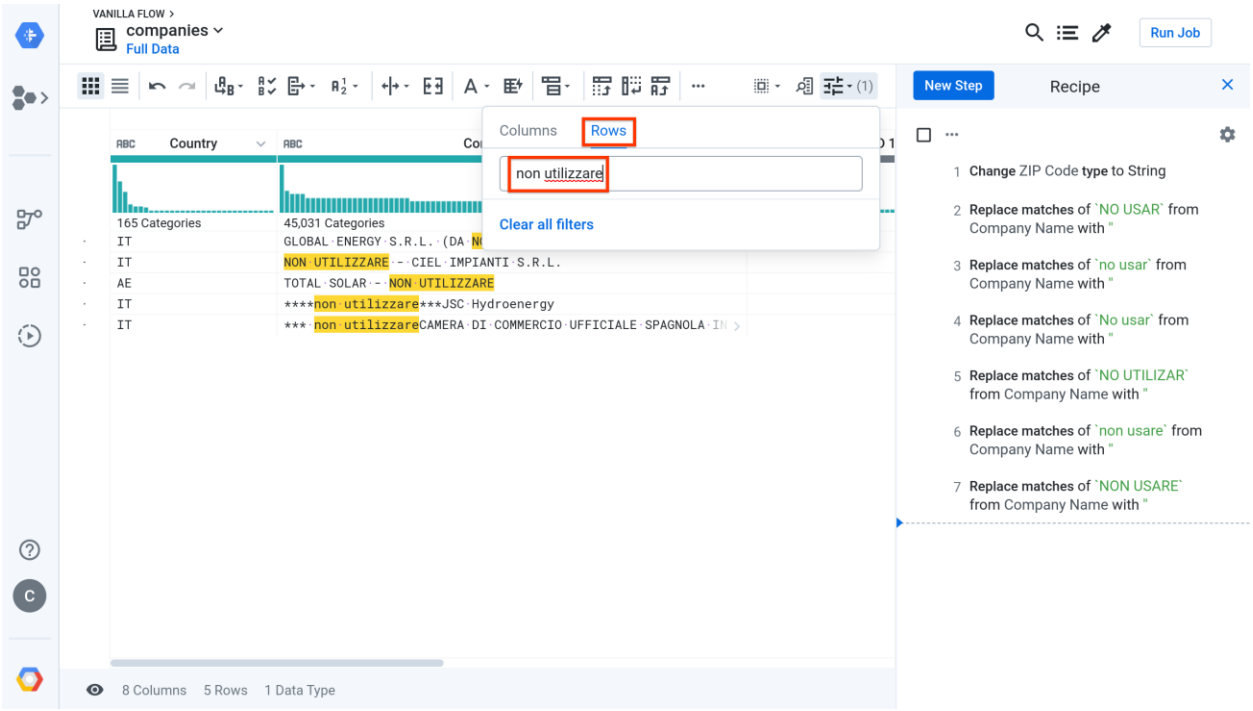


Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



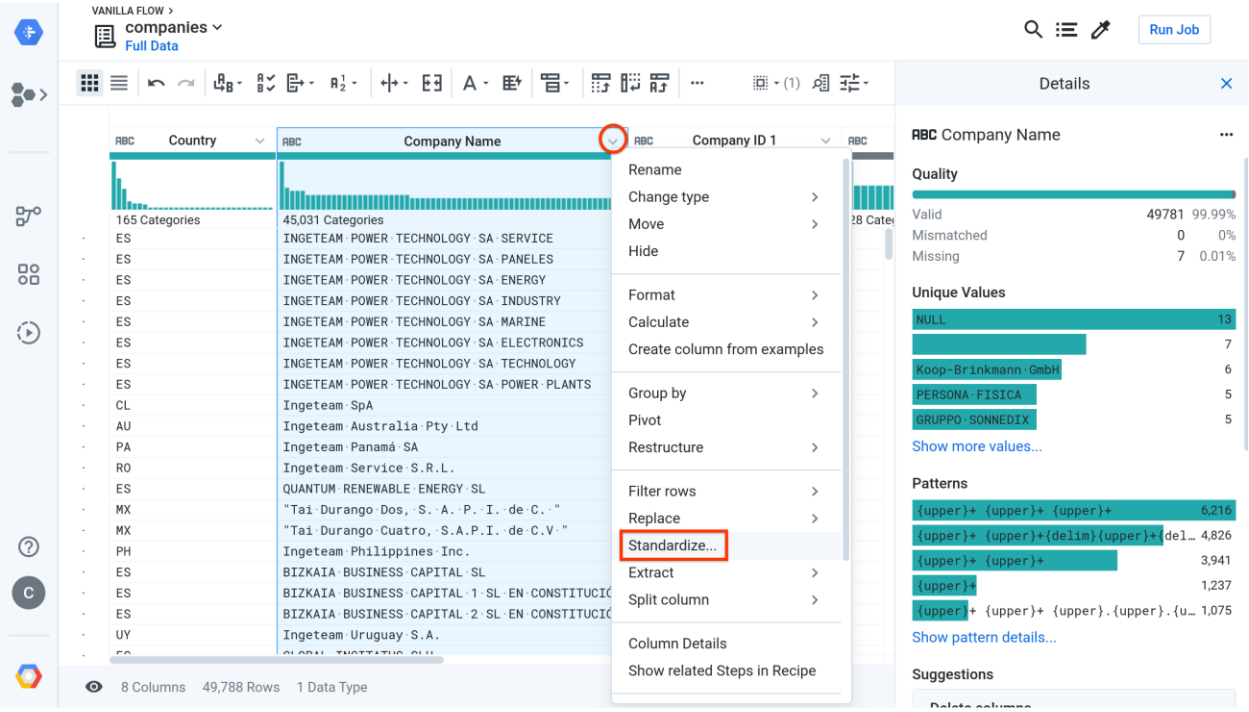






Now we standardise the company names.

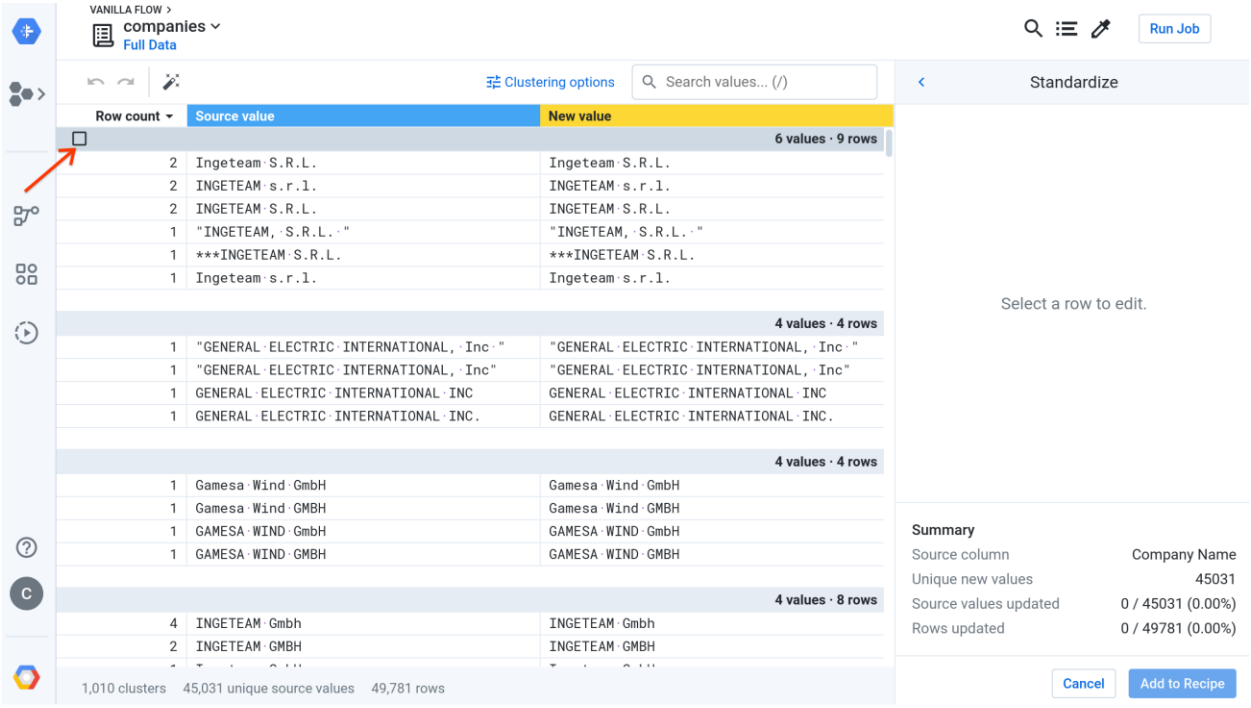
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



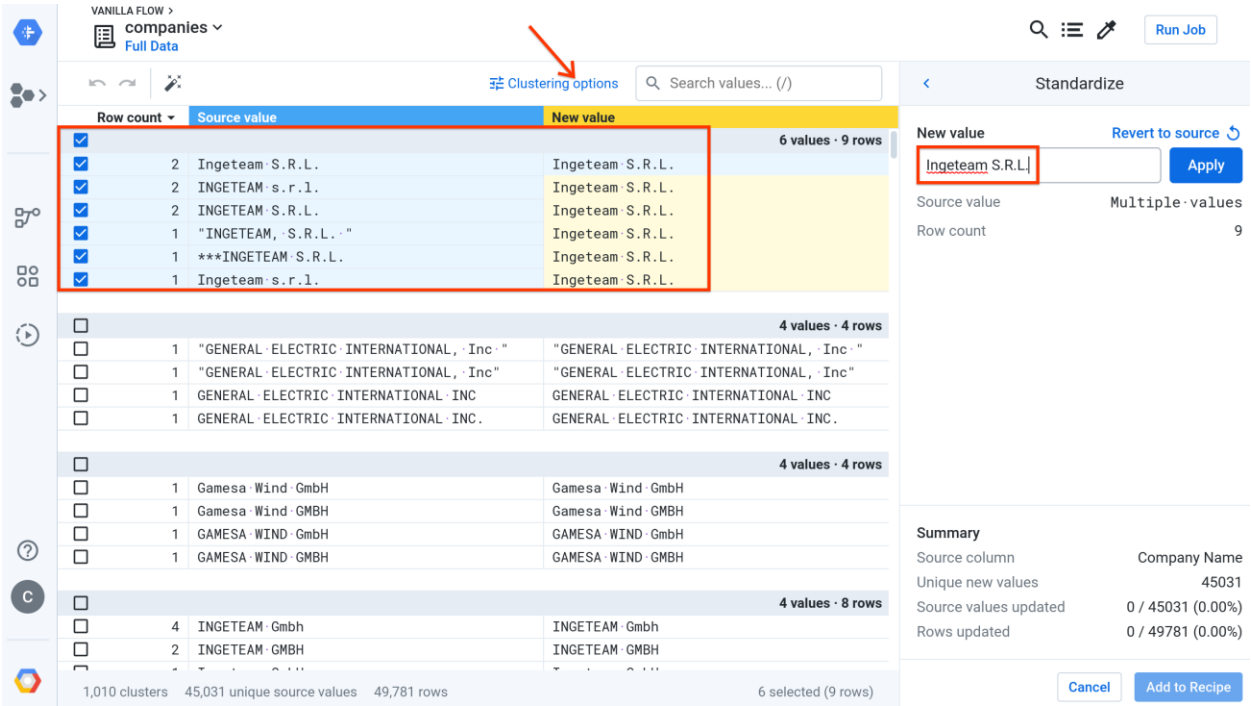
After clicking on “Standardize”, the company names are clustered by similar strings (the values have characters in common). This is a fuzzy matching method, as the company names within a cluster are not necessarily identical.

If all values in a cluster should have the same name, we select the cluster.

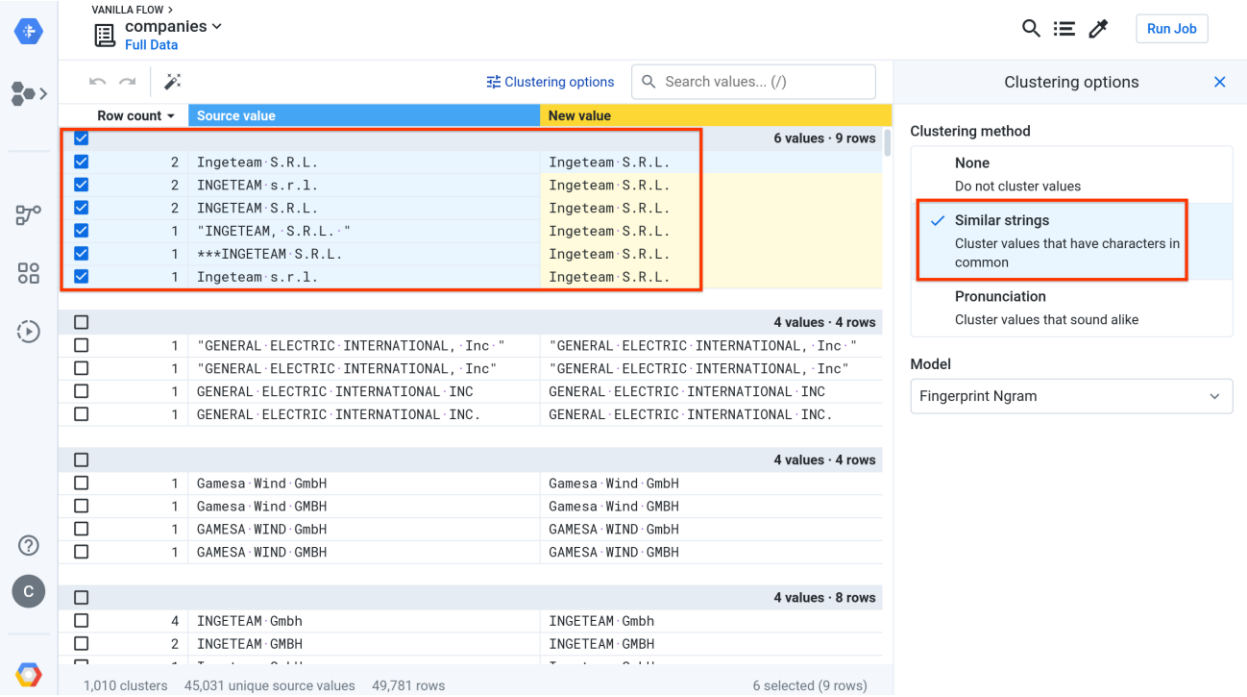
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



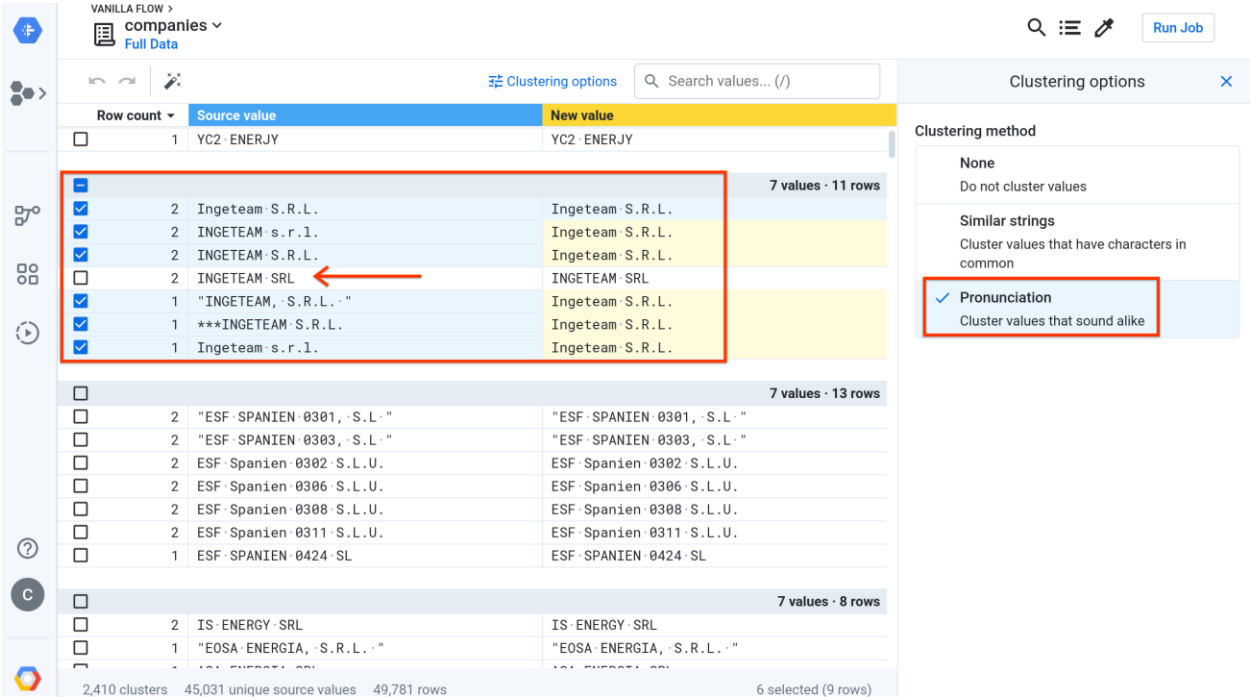
We type the new value all occurrences in the cluster should have. Then we click on “Clustering options”.

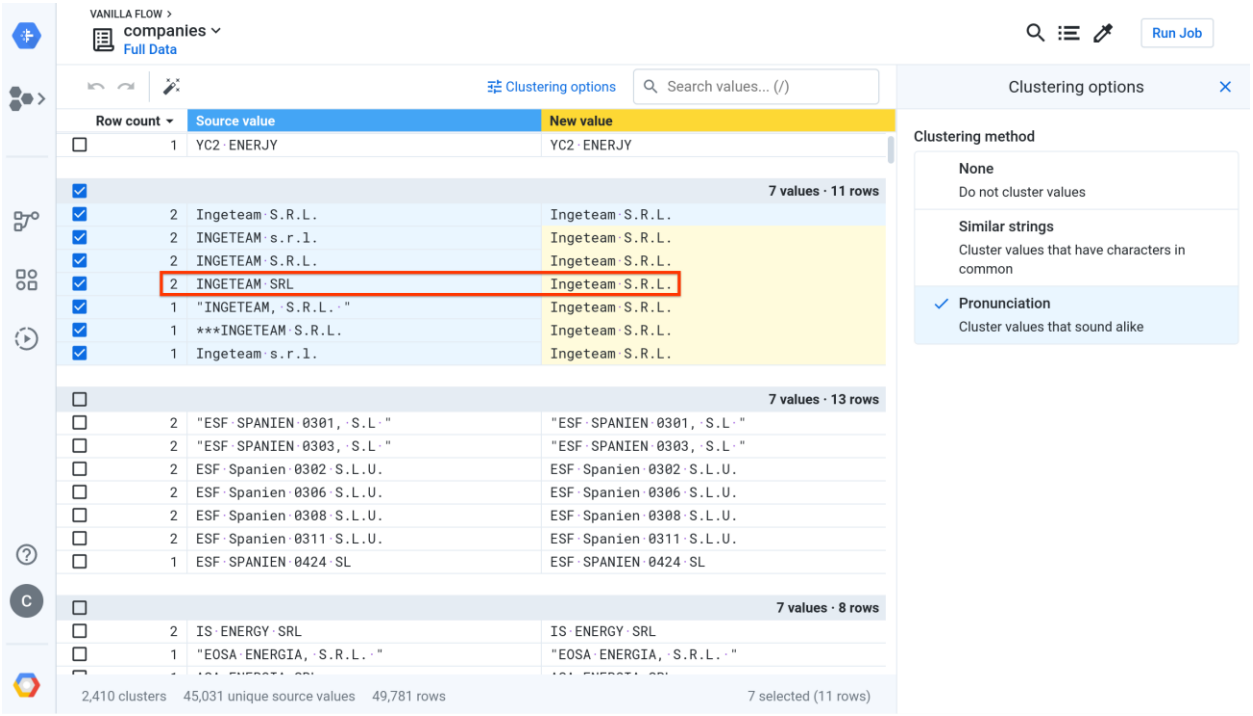


We confirm we were clustering on similar strings.



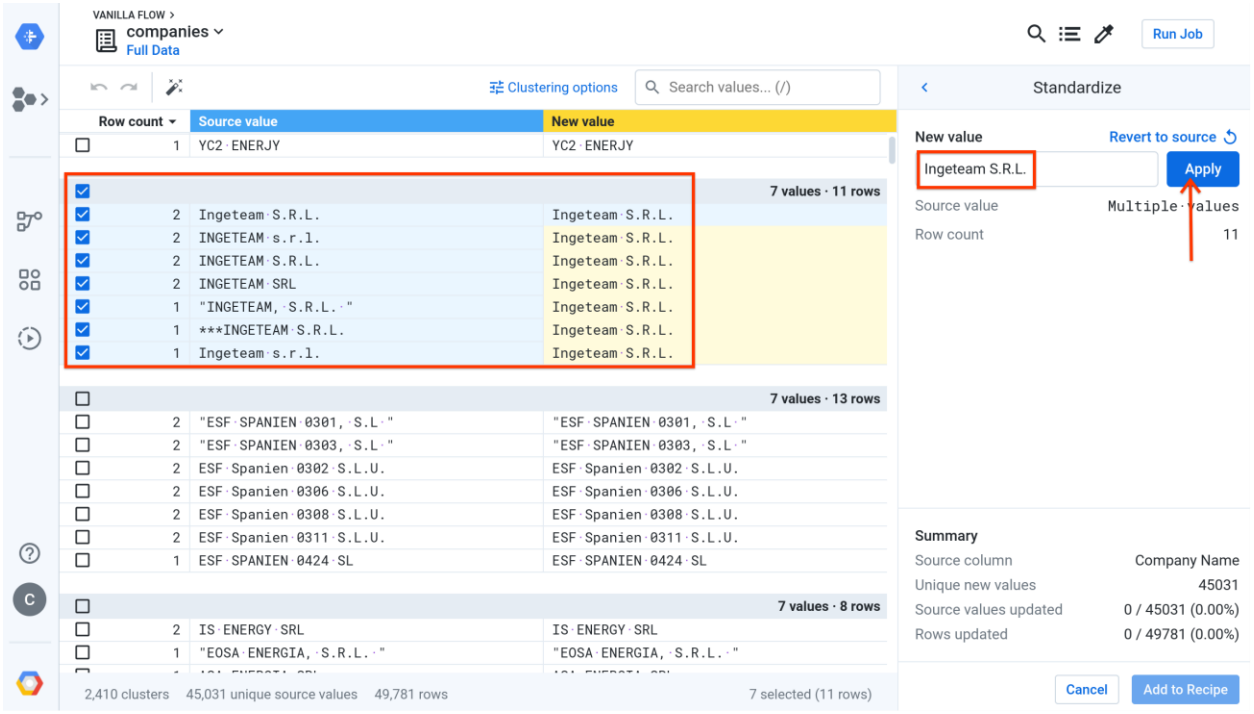
Now we cluster on similar pronunciation, that's to say, the values sound alike. We select uncheck values in the new cluster if they should have the same value as the cluster we have already selected with the similar string.

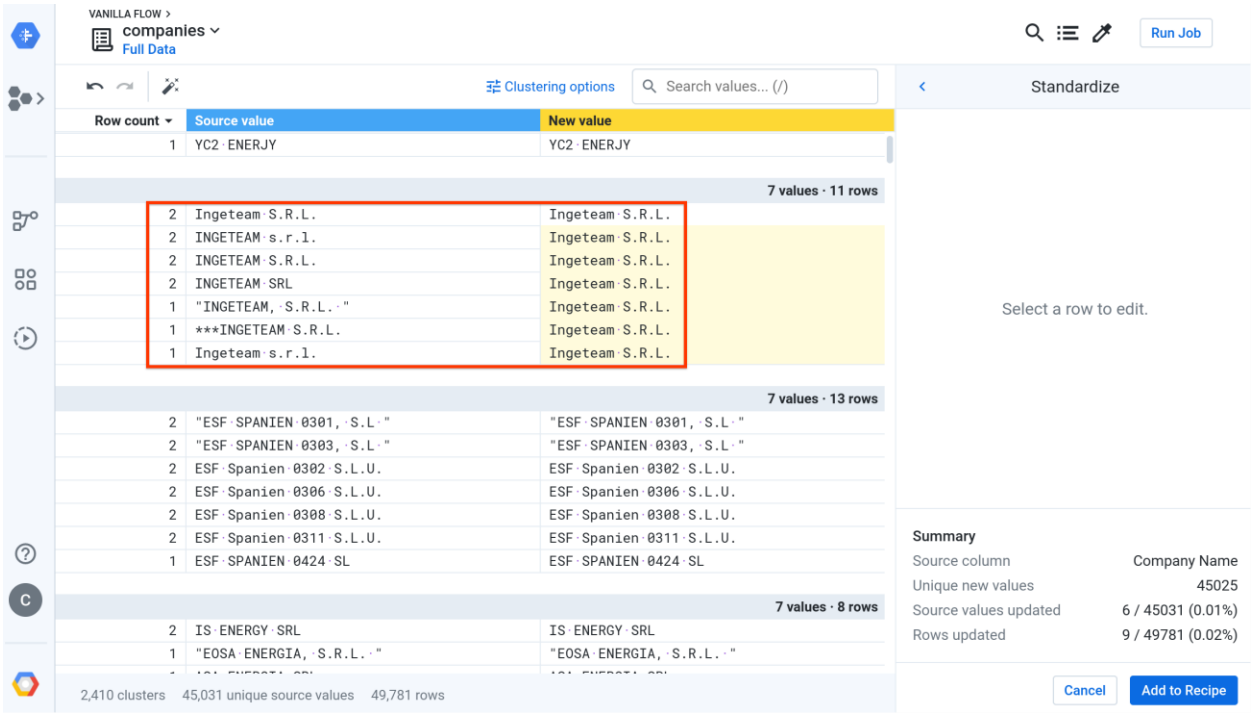




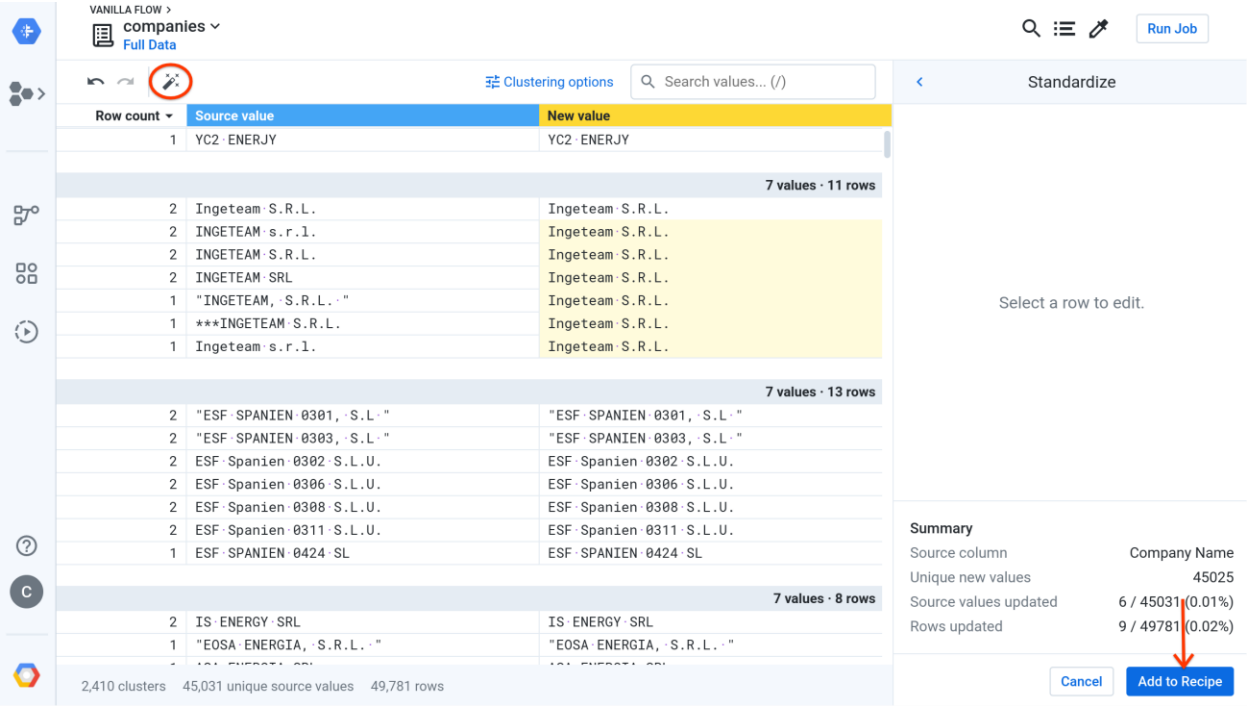
We apply the new value to all the occurrences we selected (similar string or pronunciation). Bear in mind that the Double Metaphone algorithm - used to cluster values with similar

phonetics - fails when there is a space among the characters. You should review all possible values for the company names in the CSV file used for this demo.



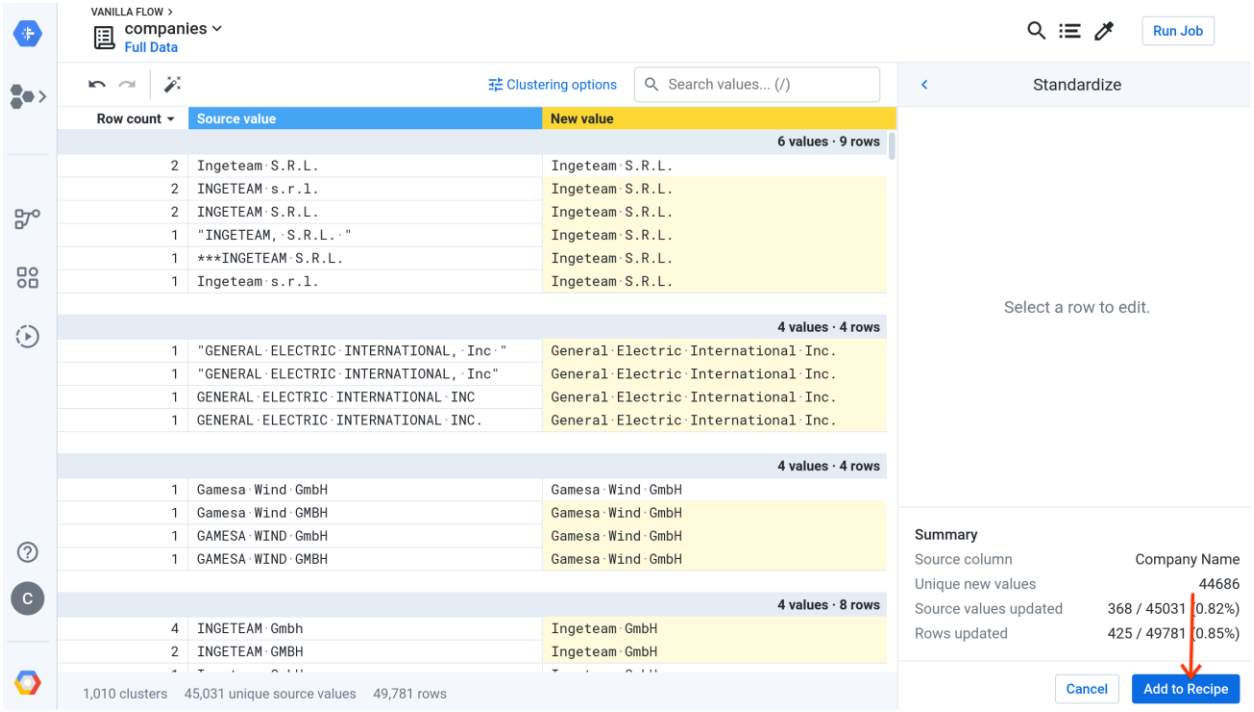


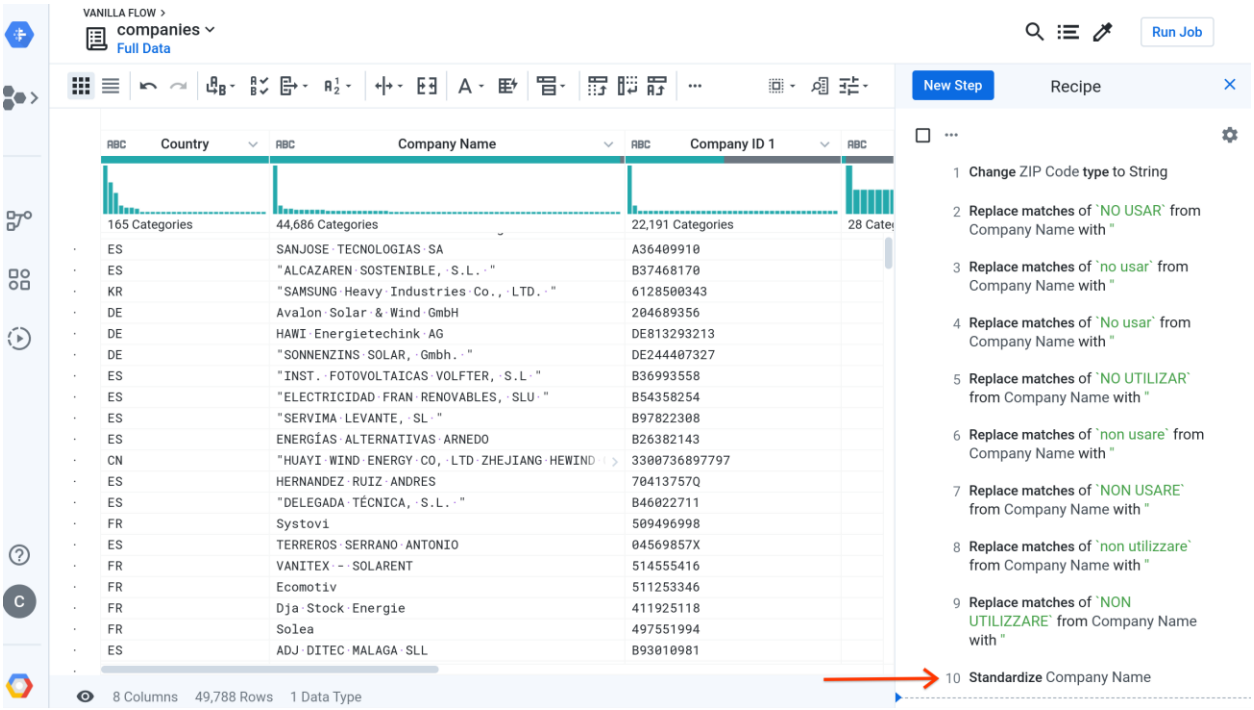
There is also a wizard button which will automatically standardise all values within each cluster to a suggested value based on occurrence frequency. Use it if you are fairly confident Cloud Dataprep will easily detect all changes which need to be done. Then you can review the changes before adding to the recipe.



Once you are pleased with all values in the CSV, click on “Save to recipe”.

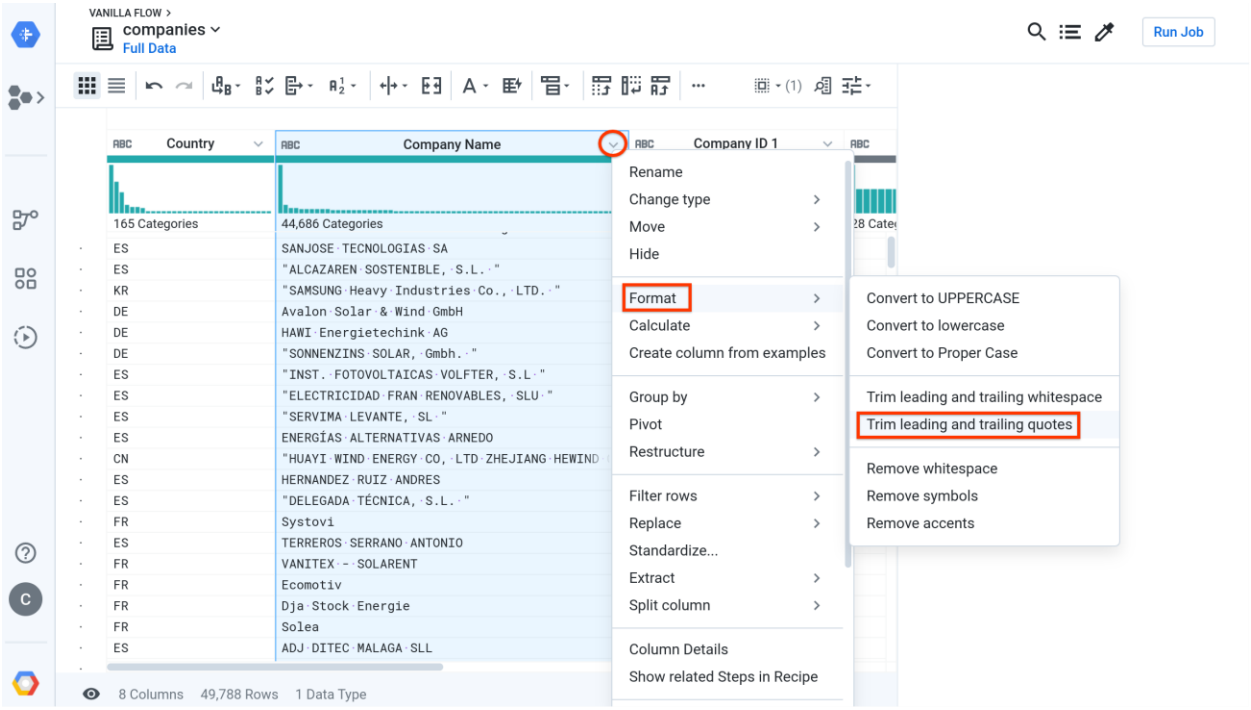
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

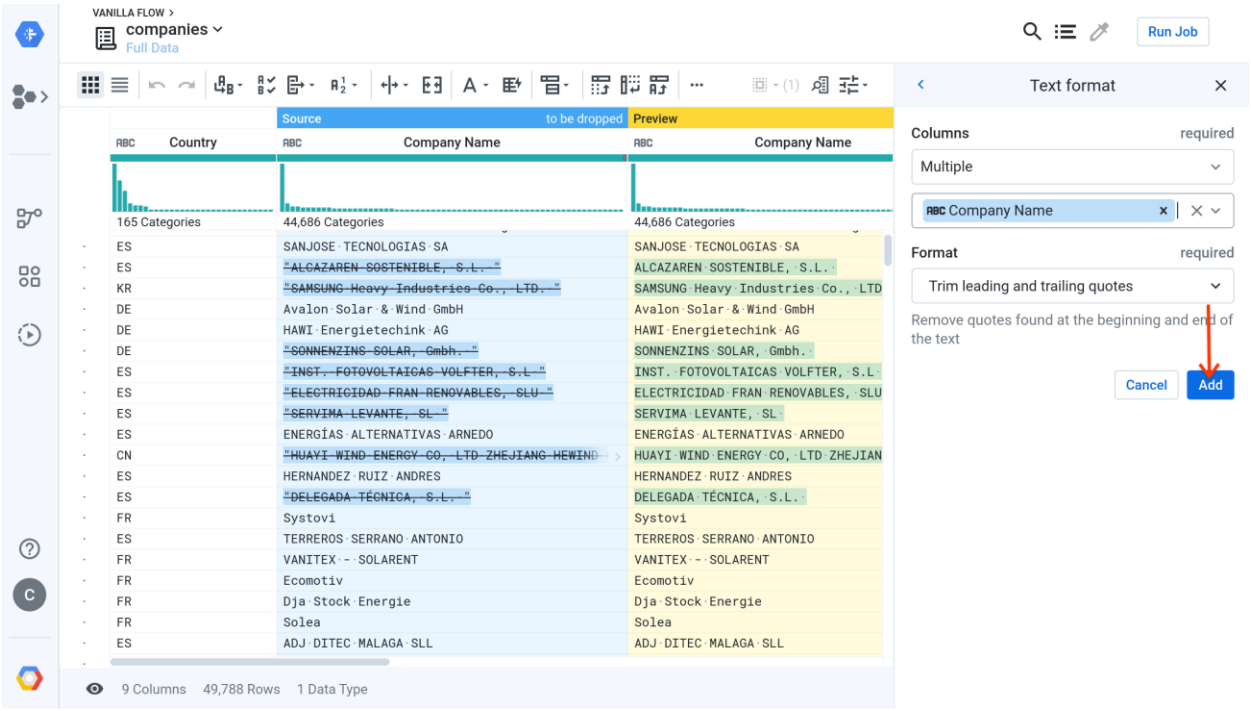




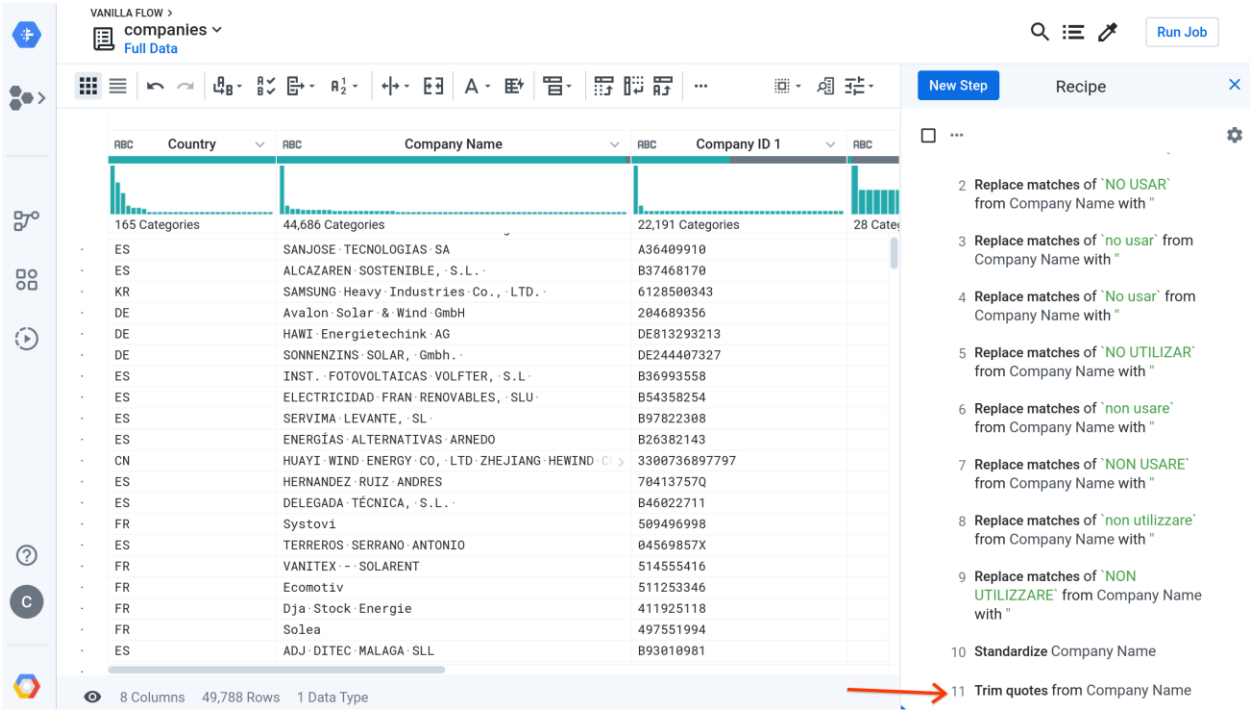
We remove the leading and trailing quotes in the company names.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

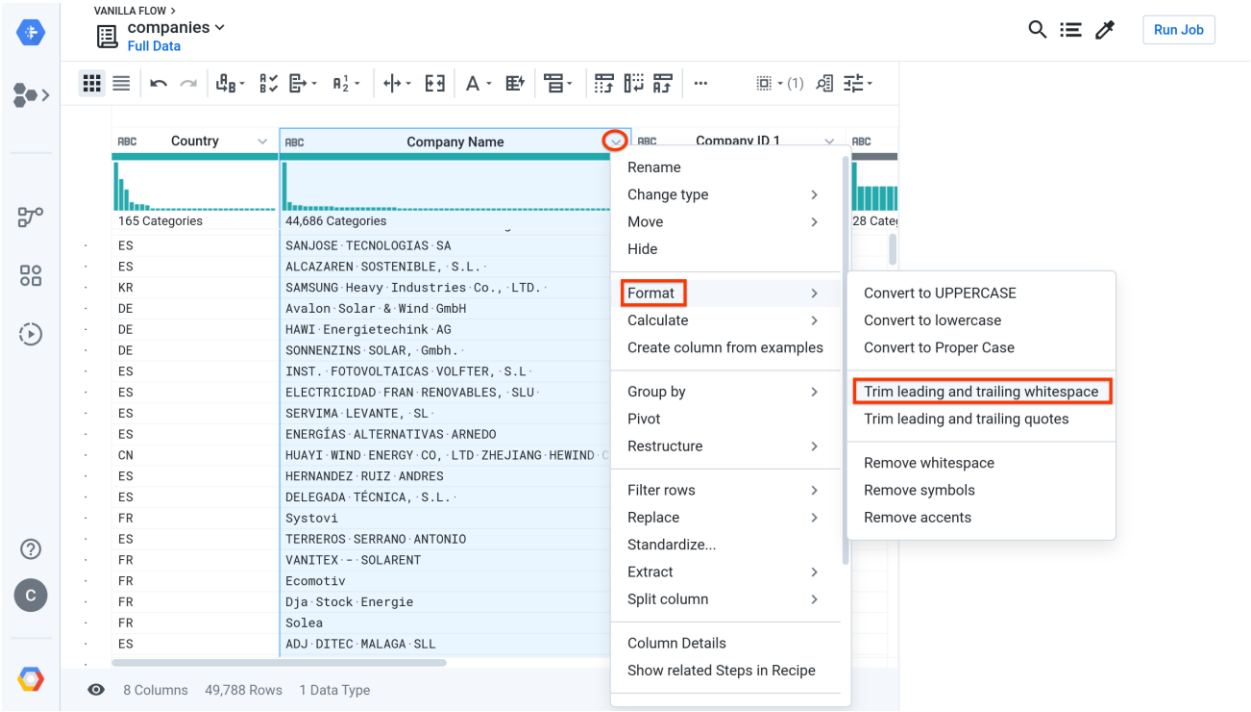


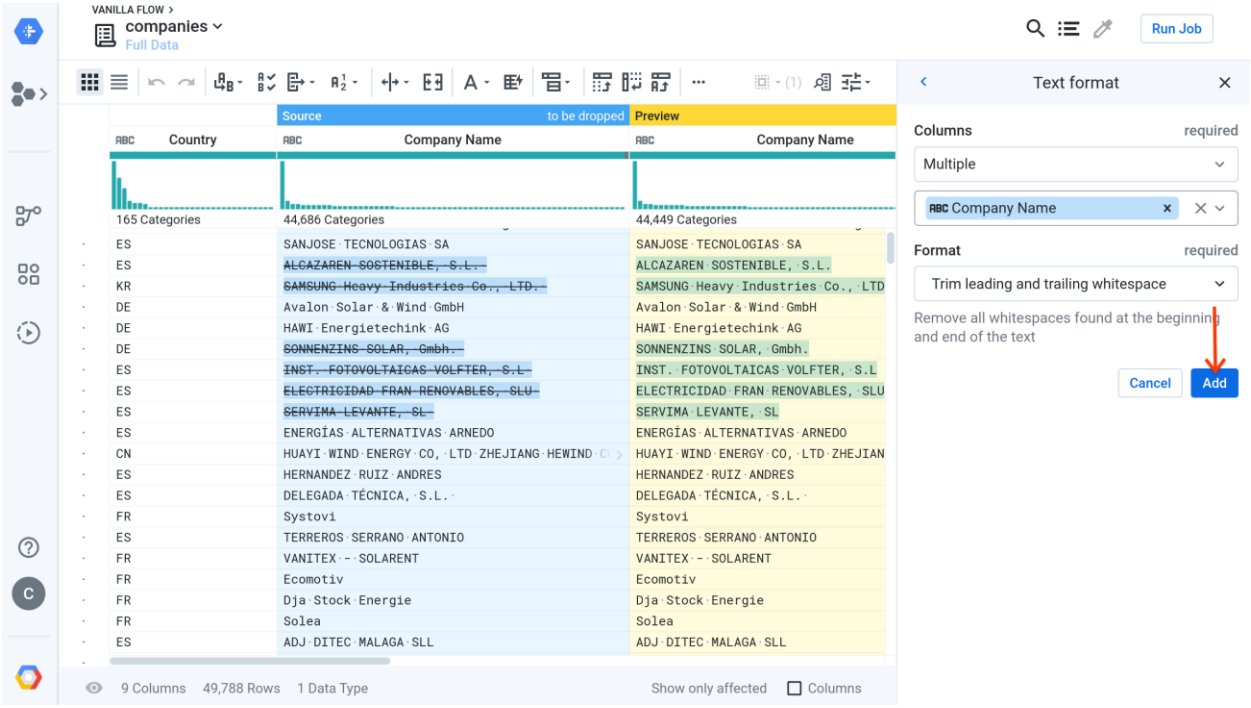


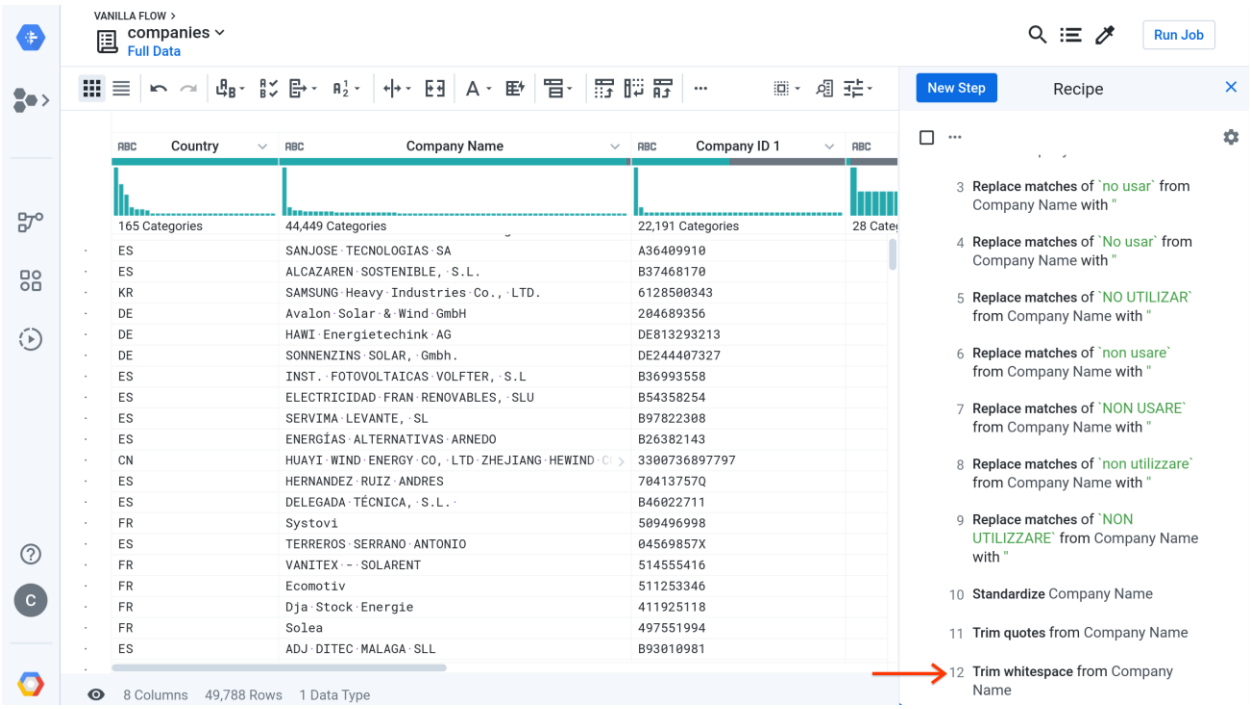
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



We remove the leading and trailing white spaces in the company names.

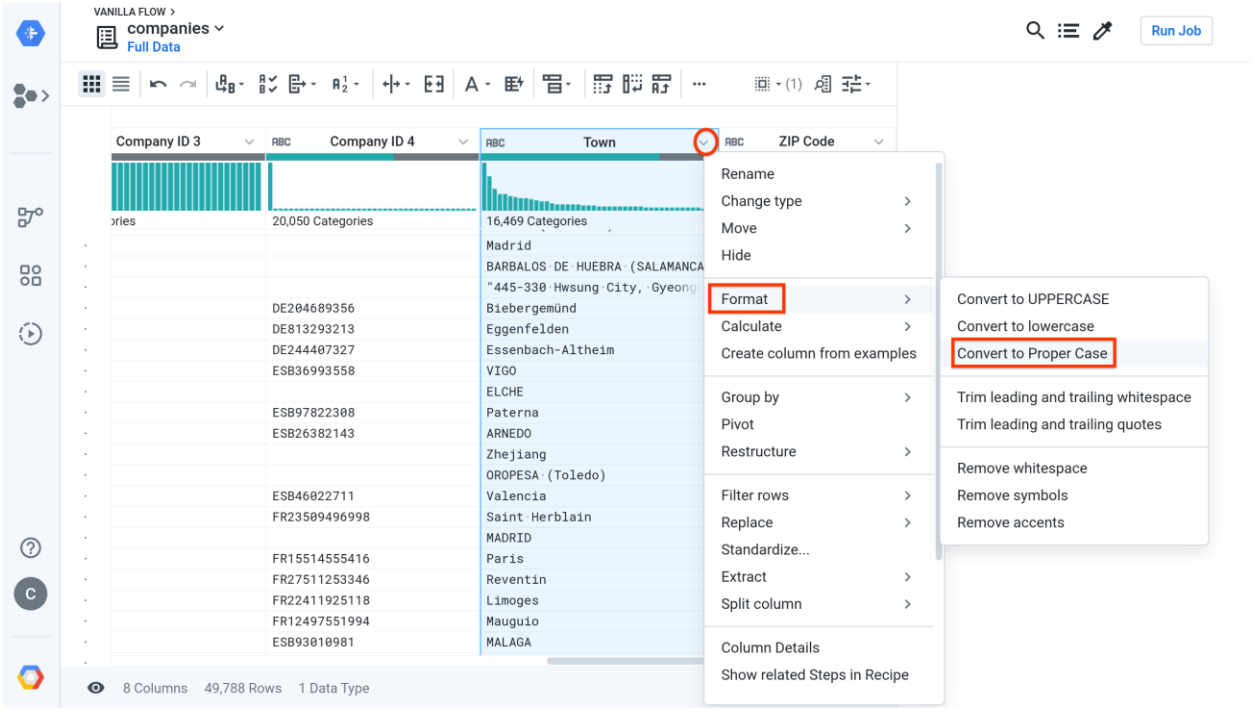


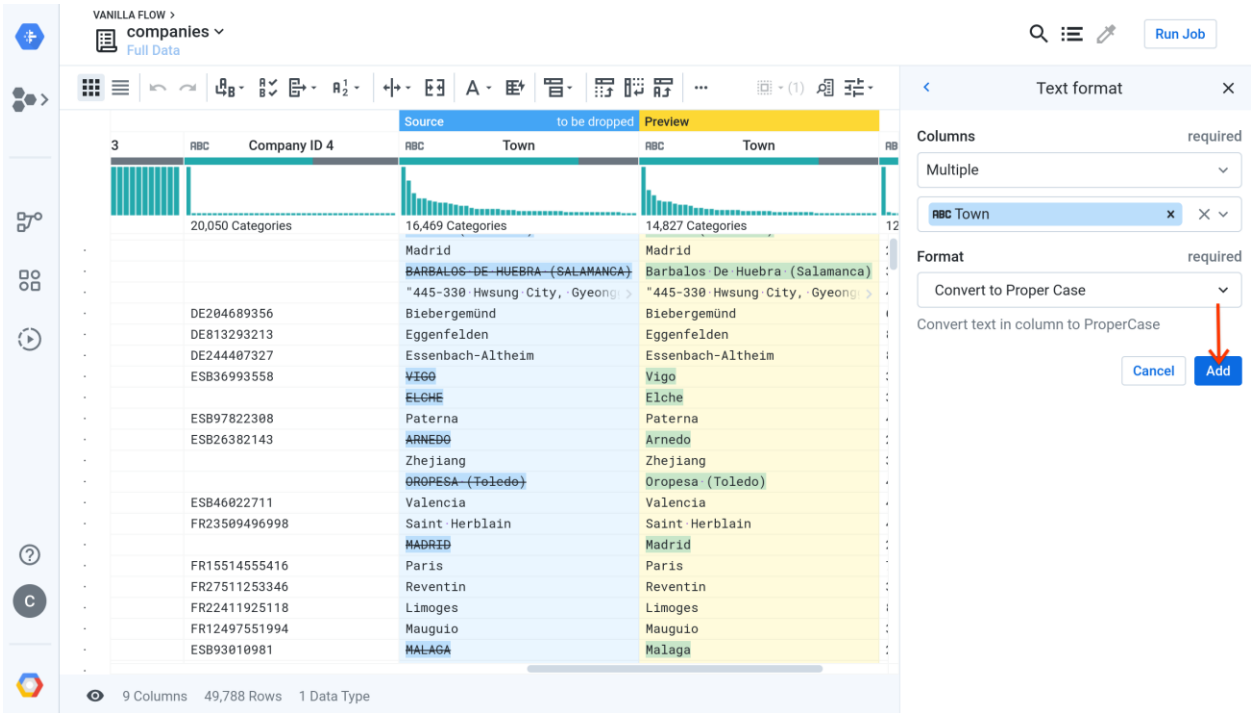




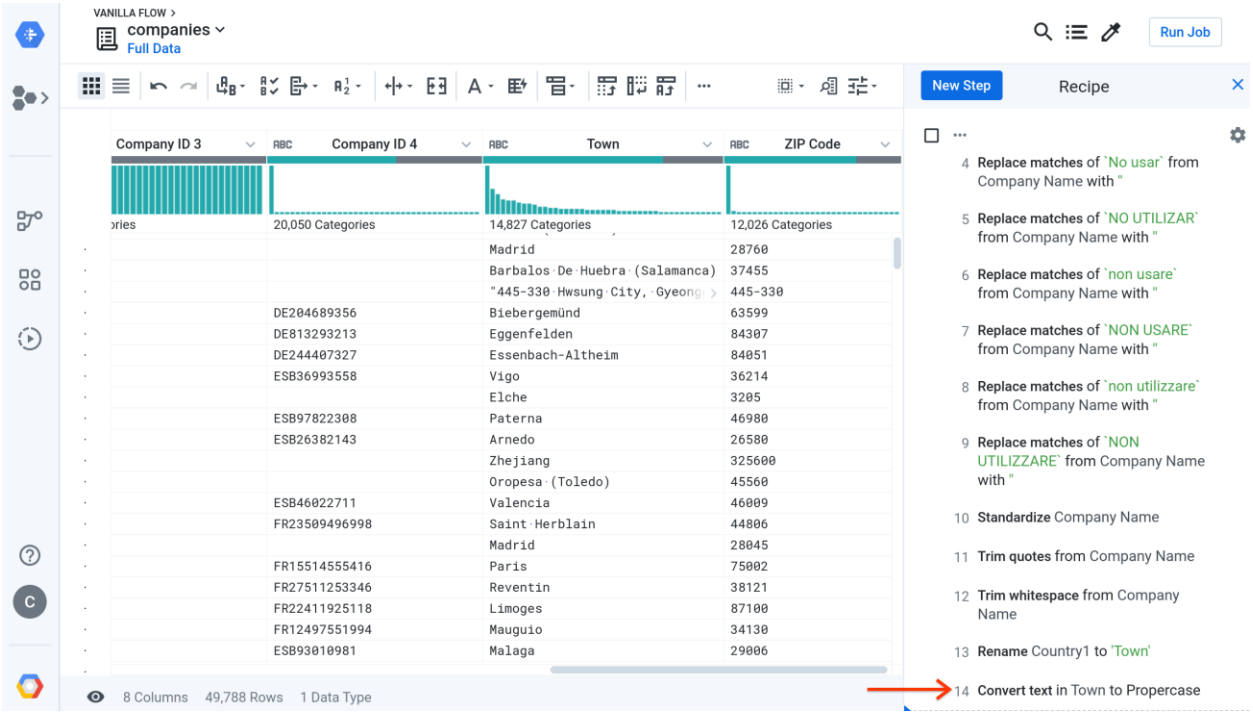
We change the format in the Town field to have the first letter in uppercase and the rest in lowercase.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

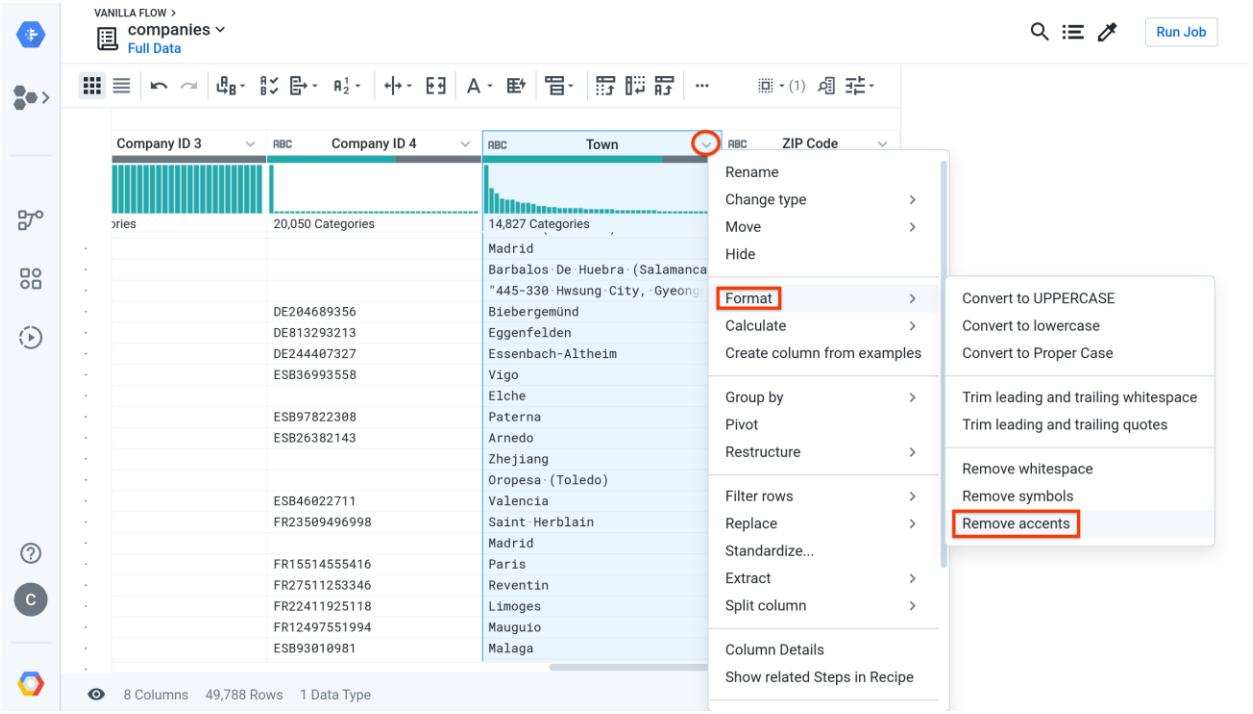




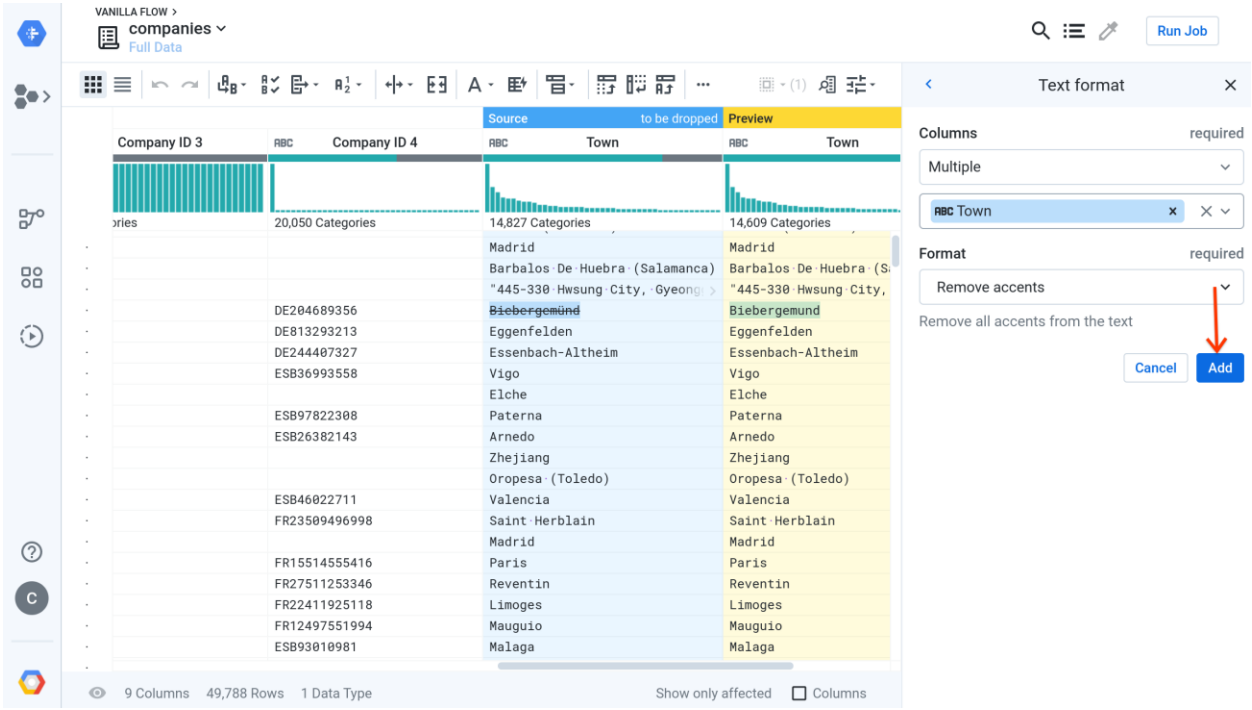
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

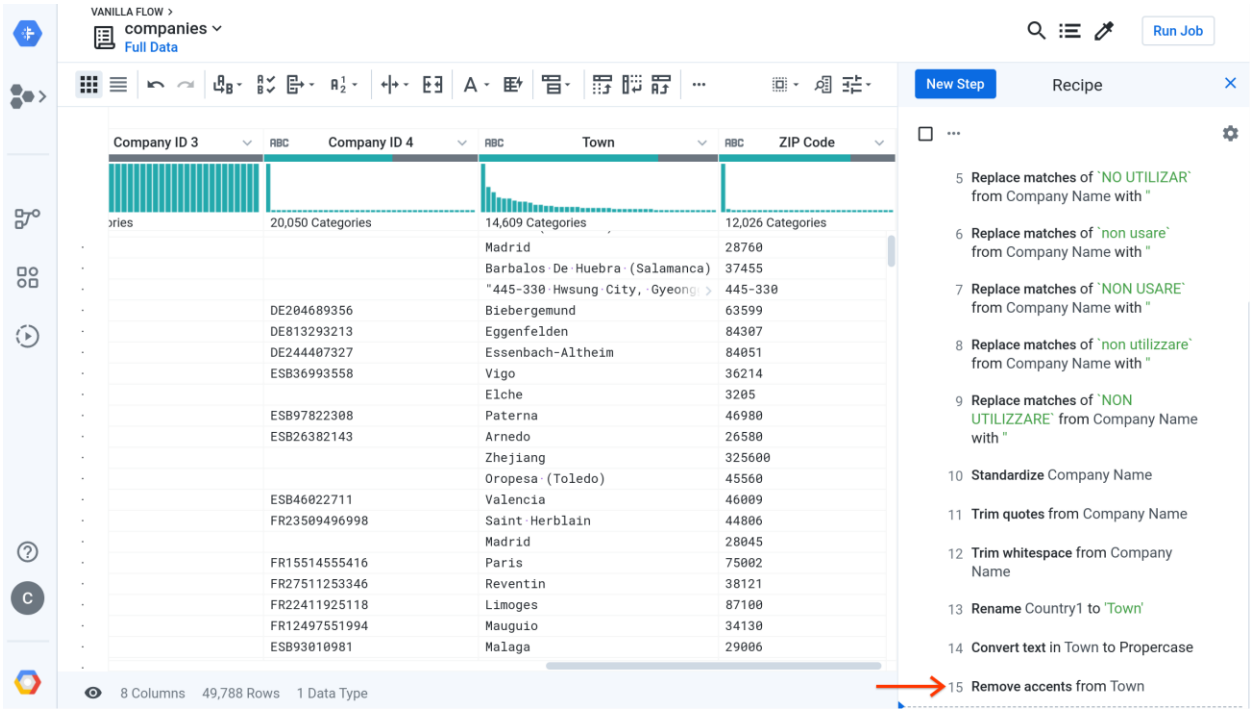


We remove accents.



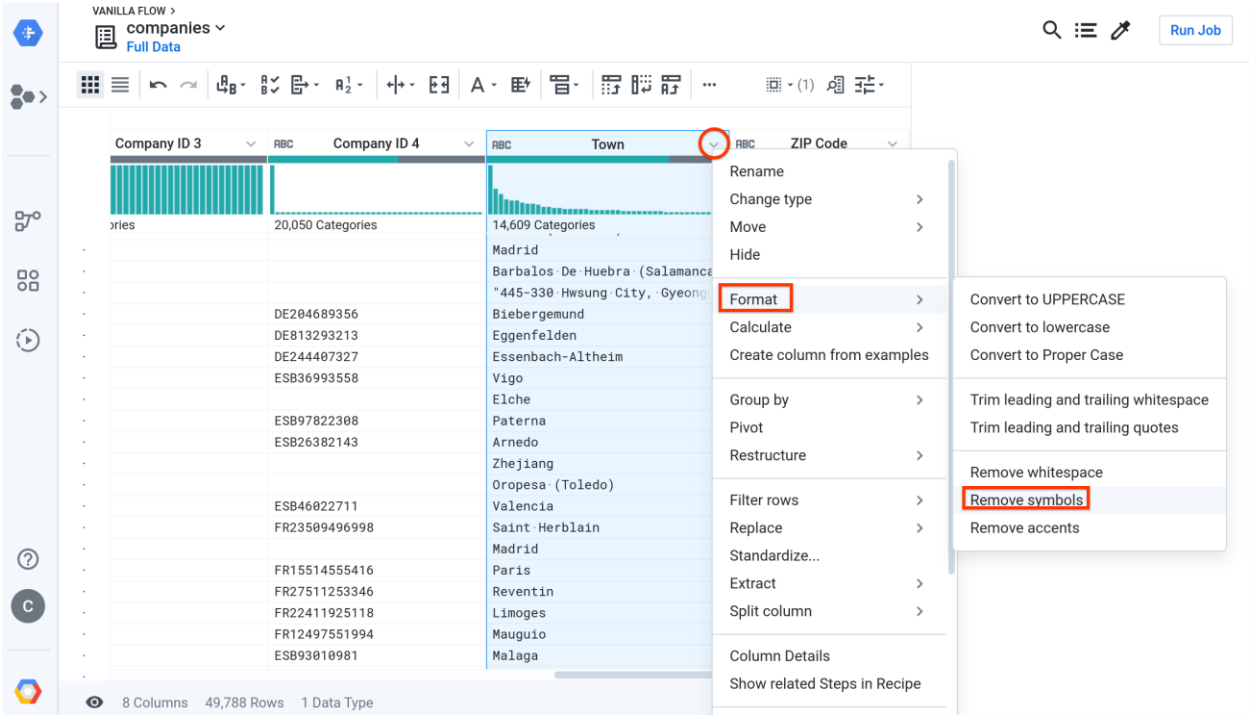
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

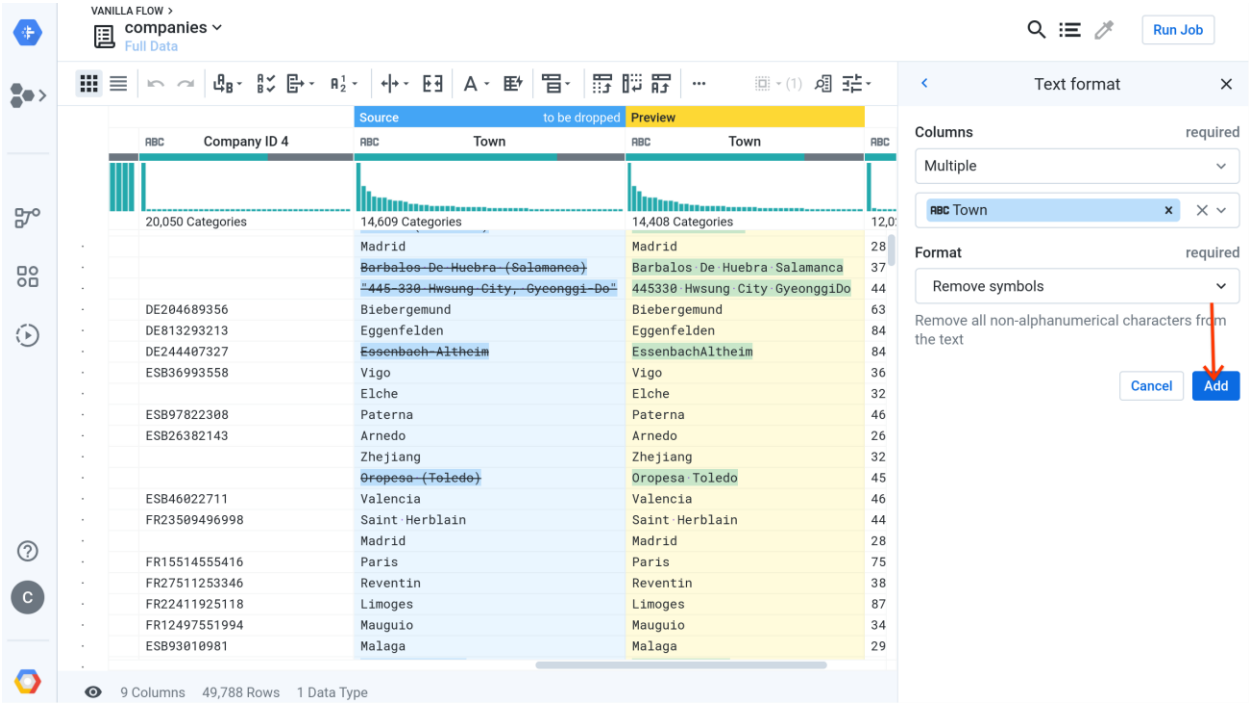




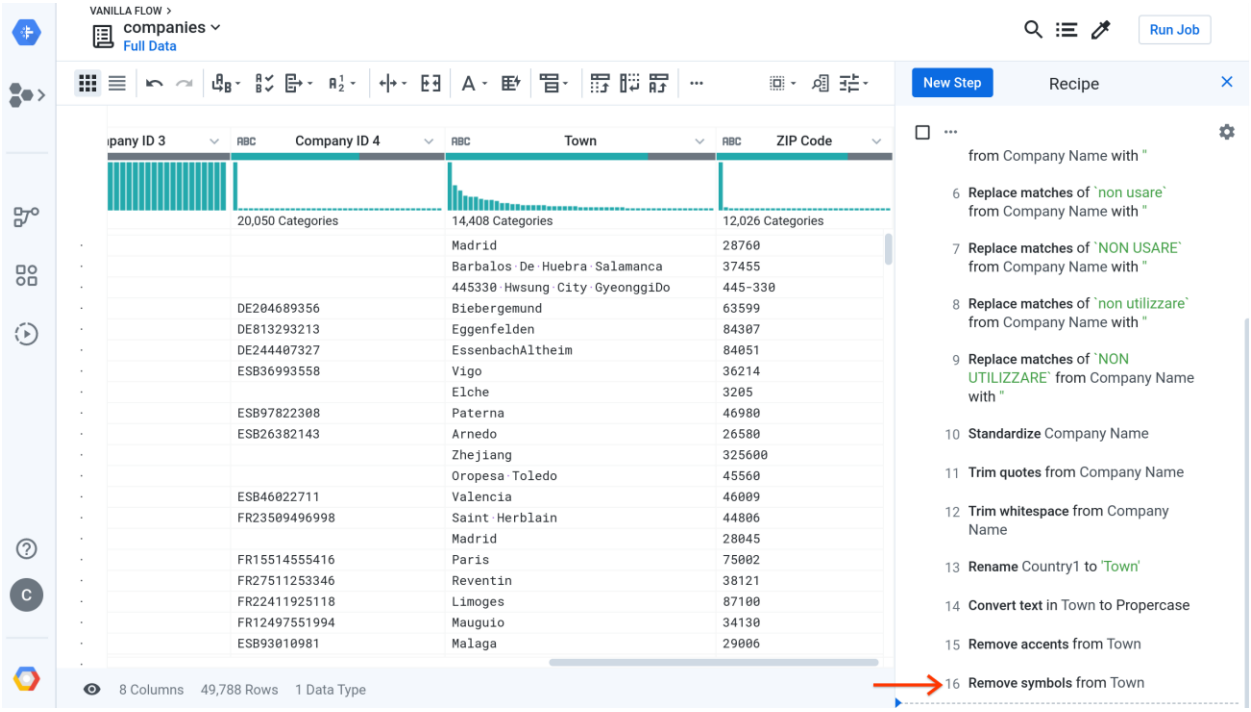
We remove symbols.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

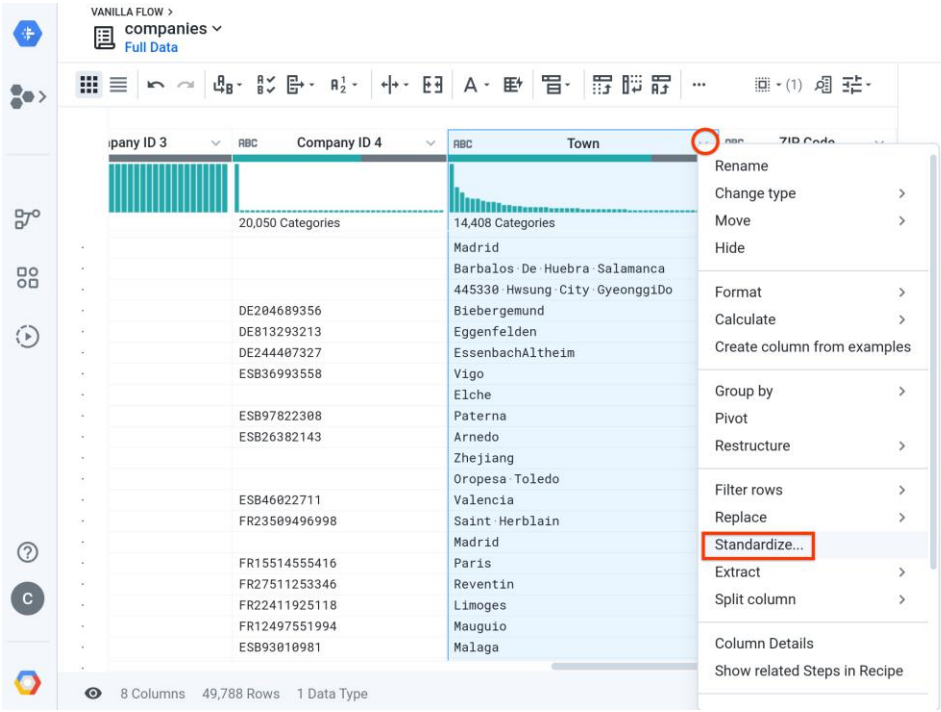




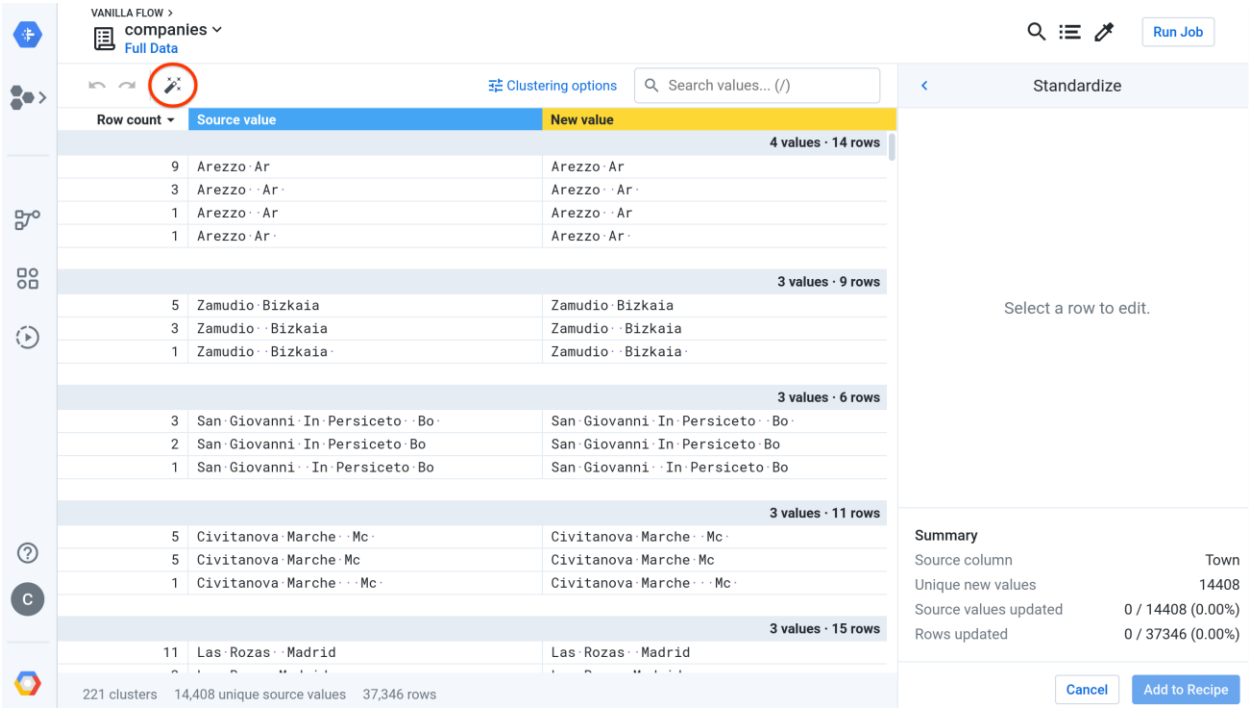
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

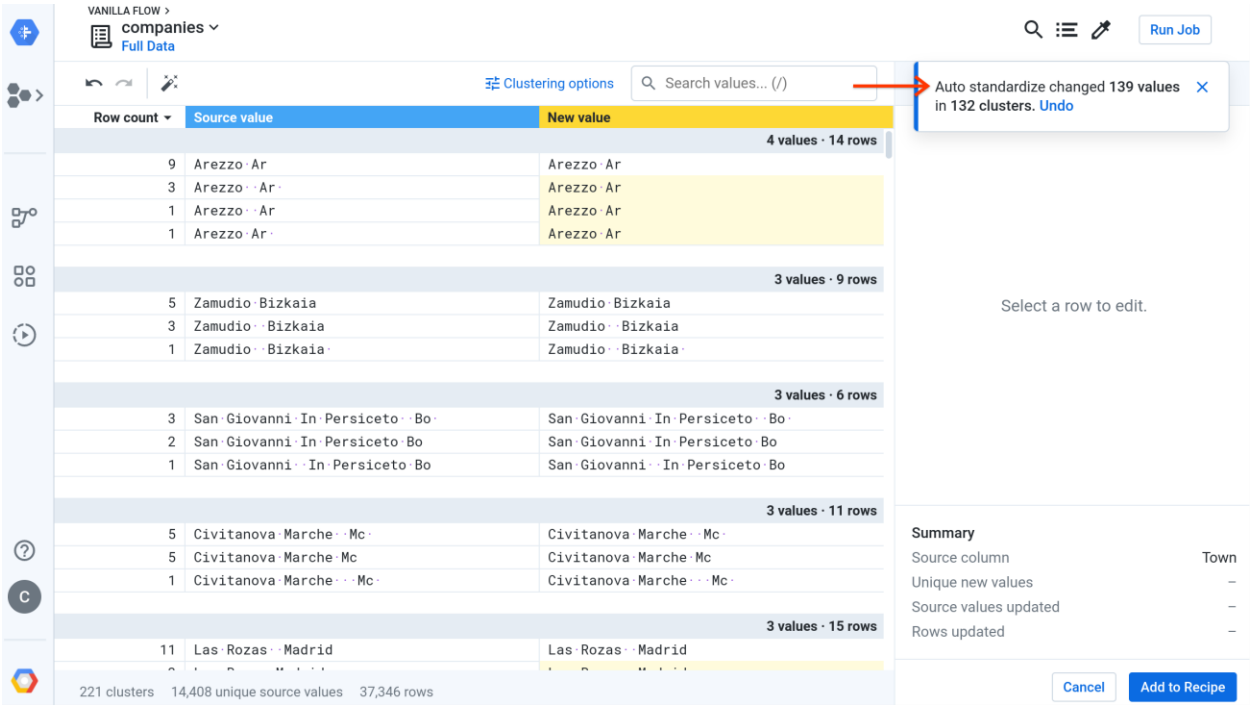


We standardise the Town field with the Wizard.



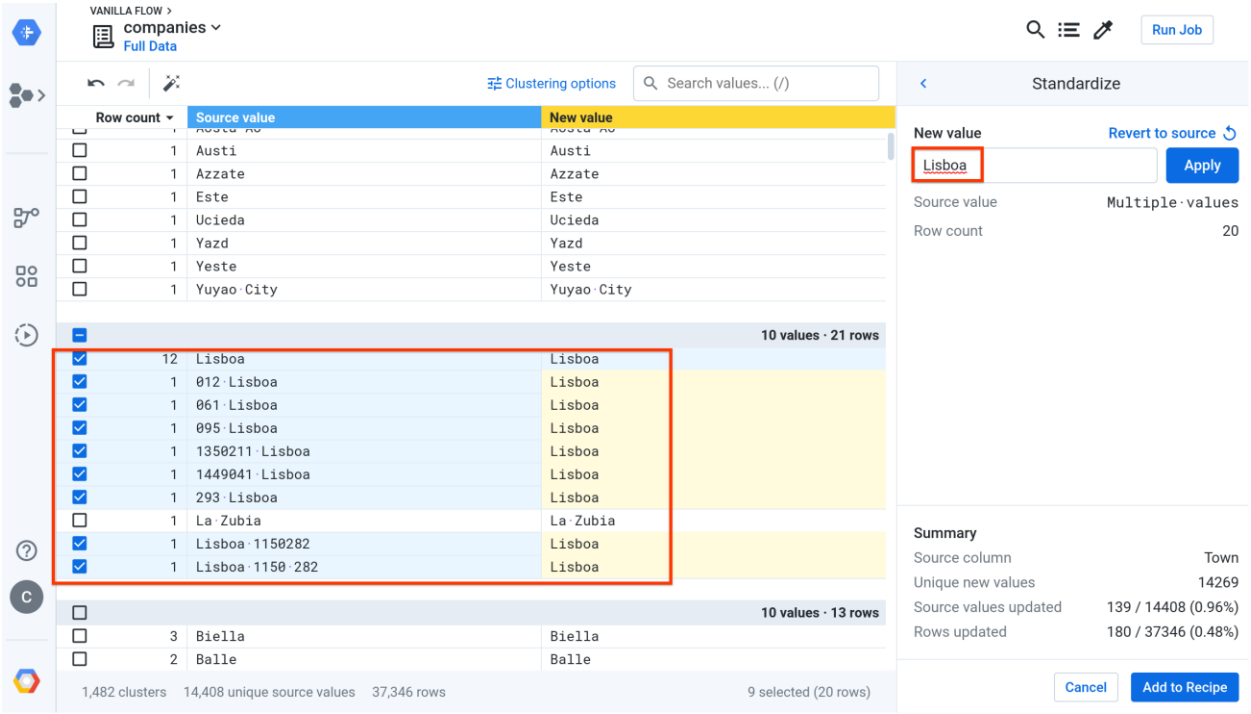
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



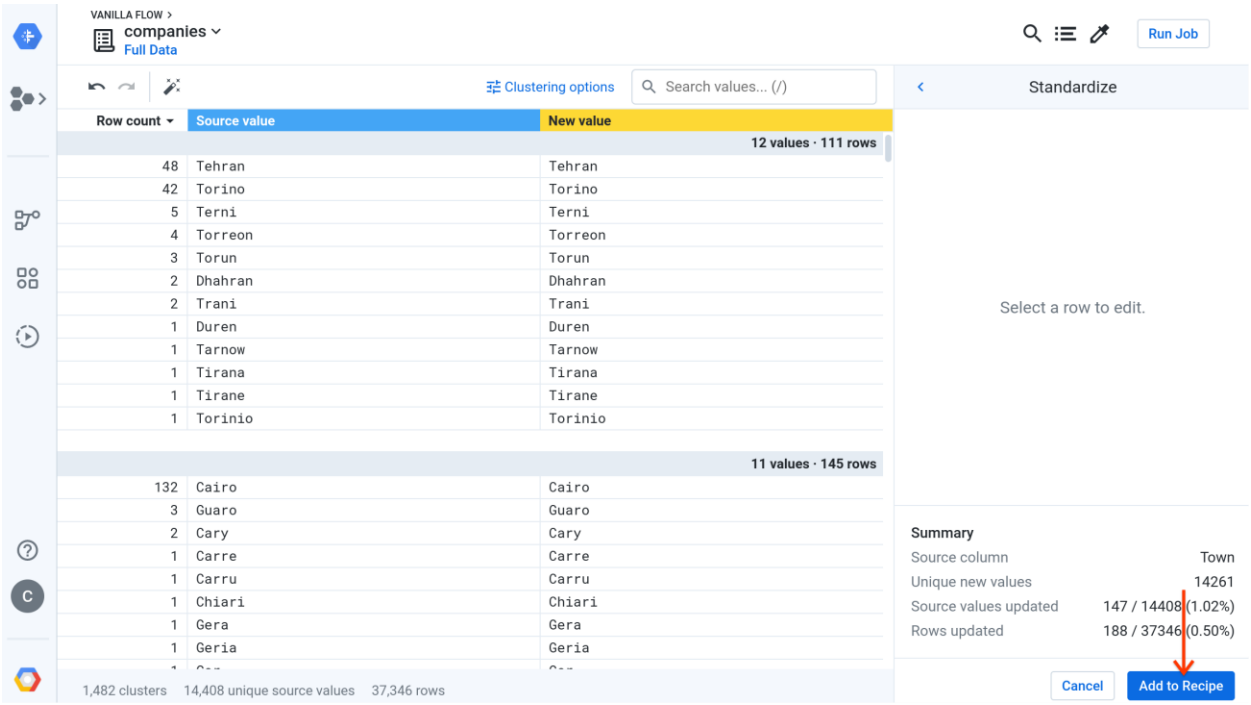


We review the town names after the automatic standardisation, and we manually modify the ones which require further work, relying on the clustering options.

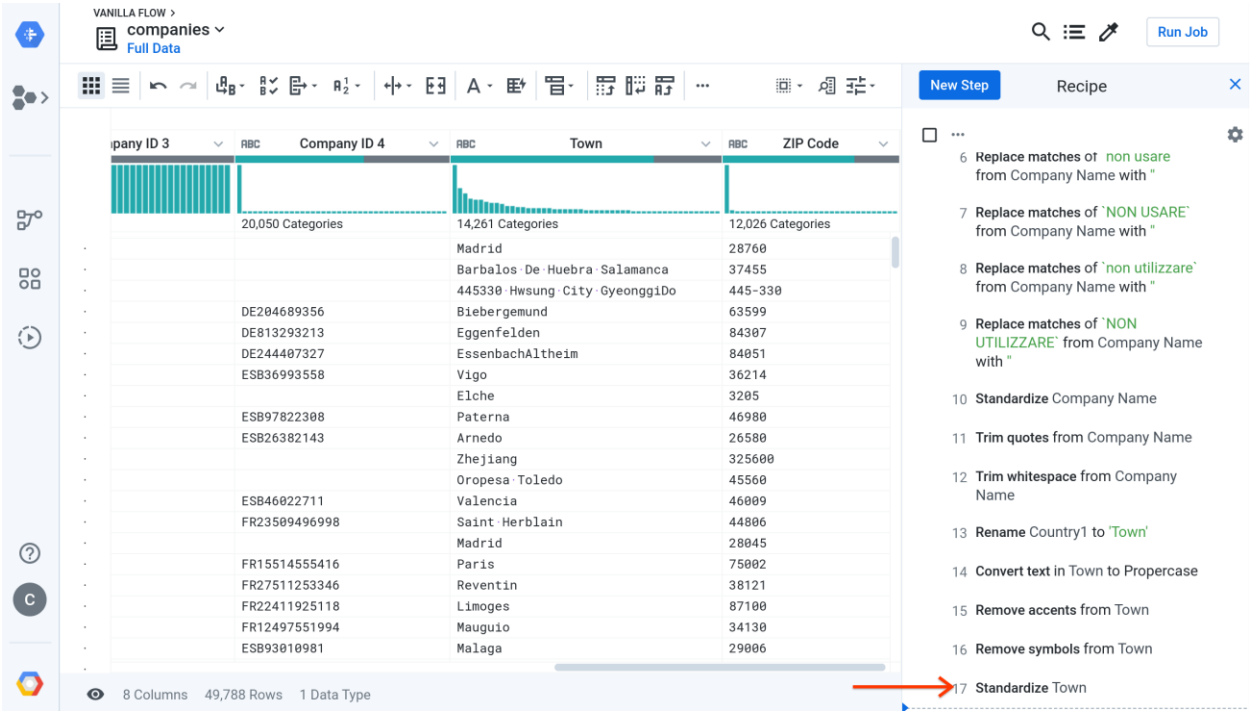
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



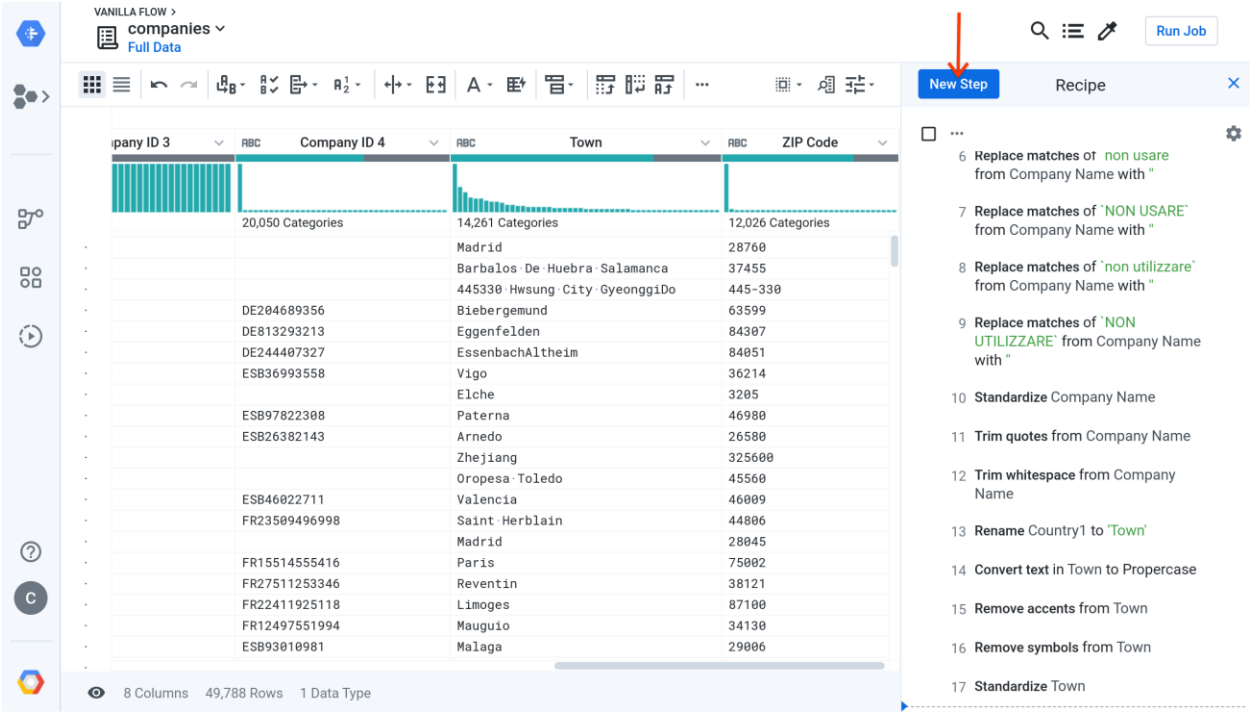
Once we are pleased with the town names, we “Add to Recipe”.



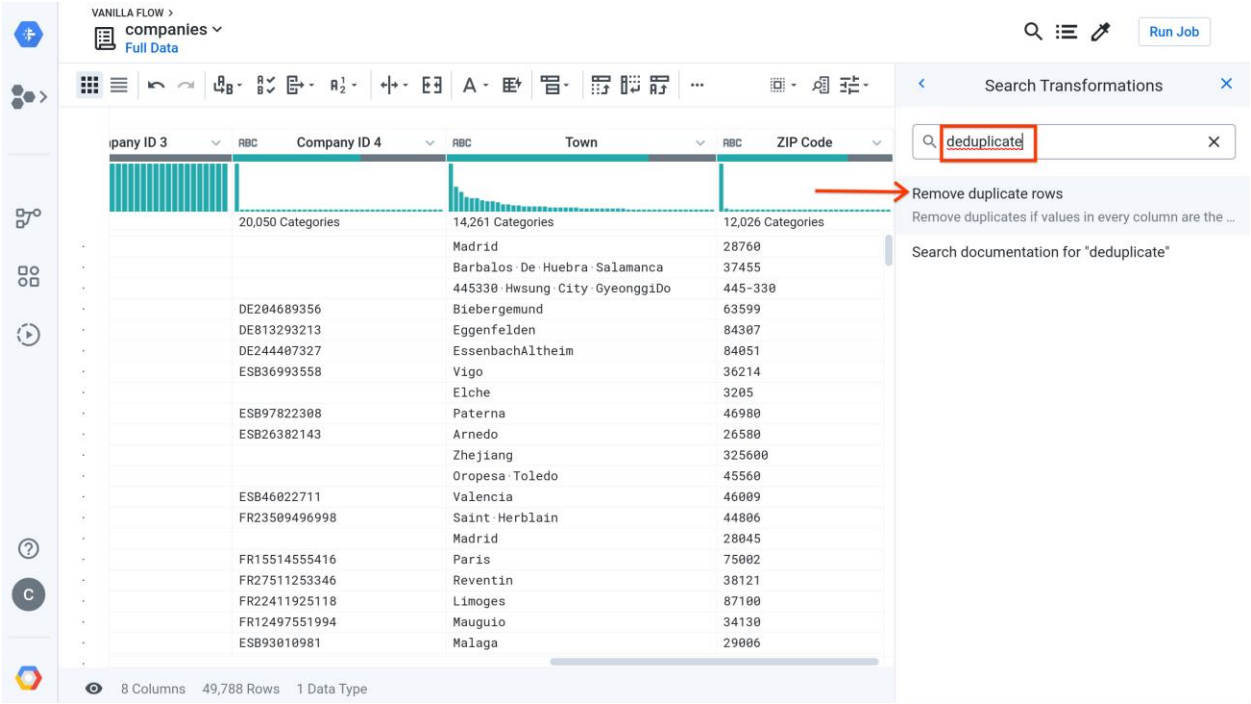
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

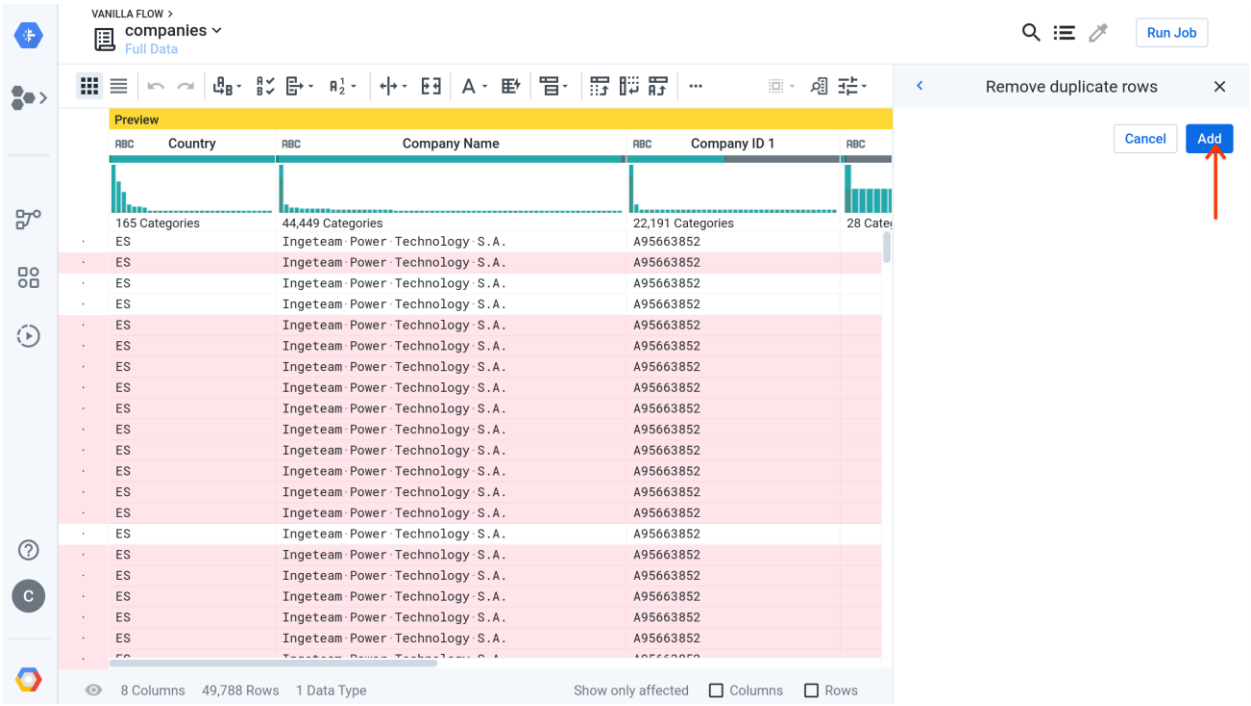


Our file should contain a row per company. We are going to deduplicate the rows if all fields have the same values.

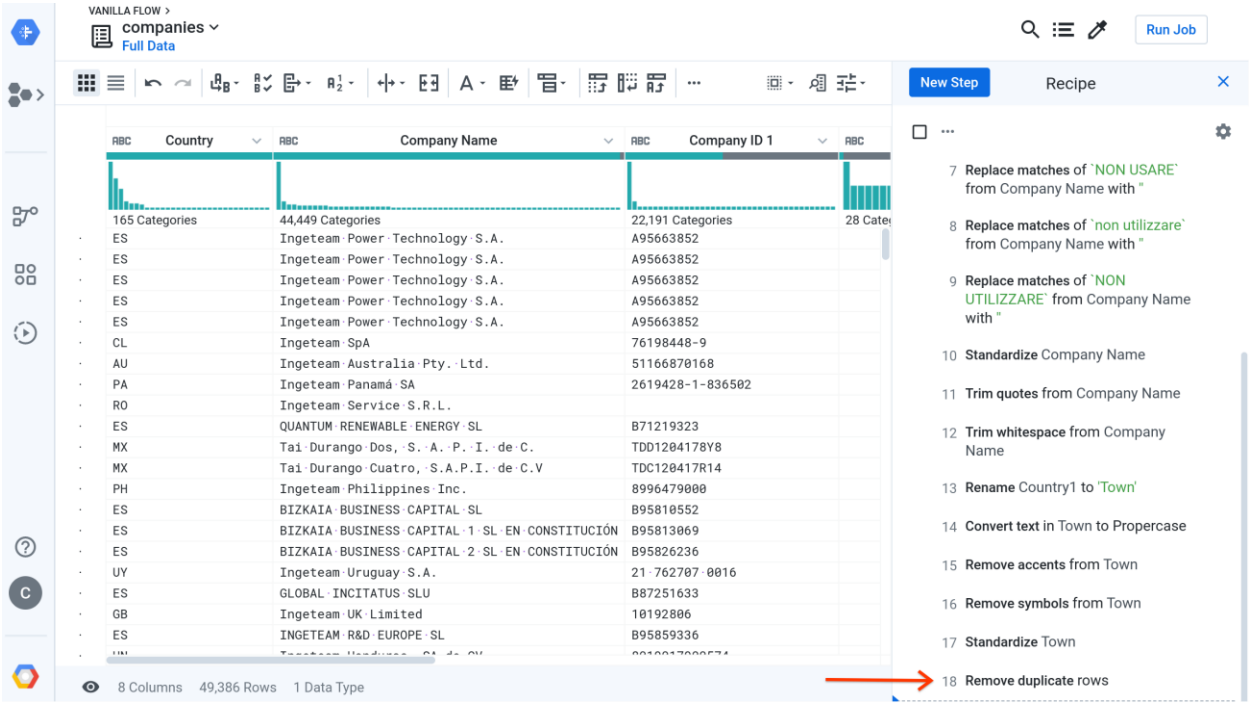


Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

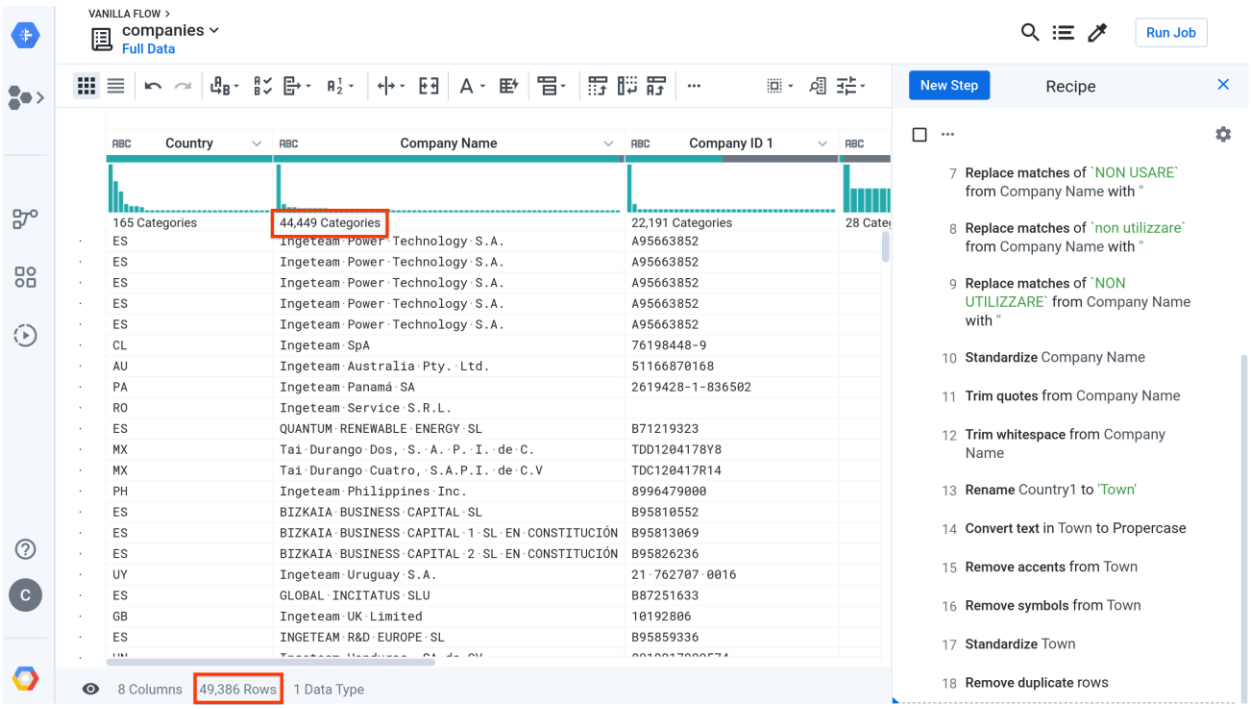




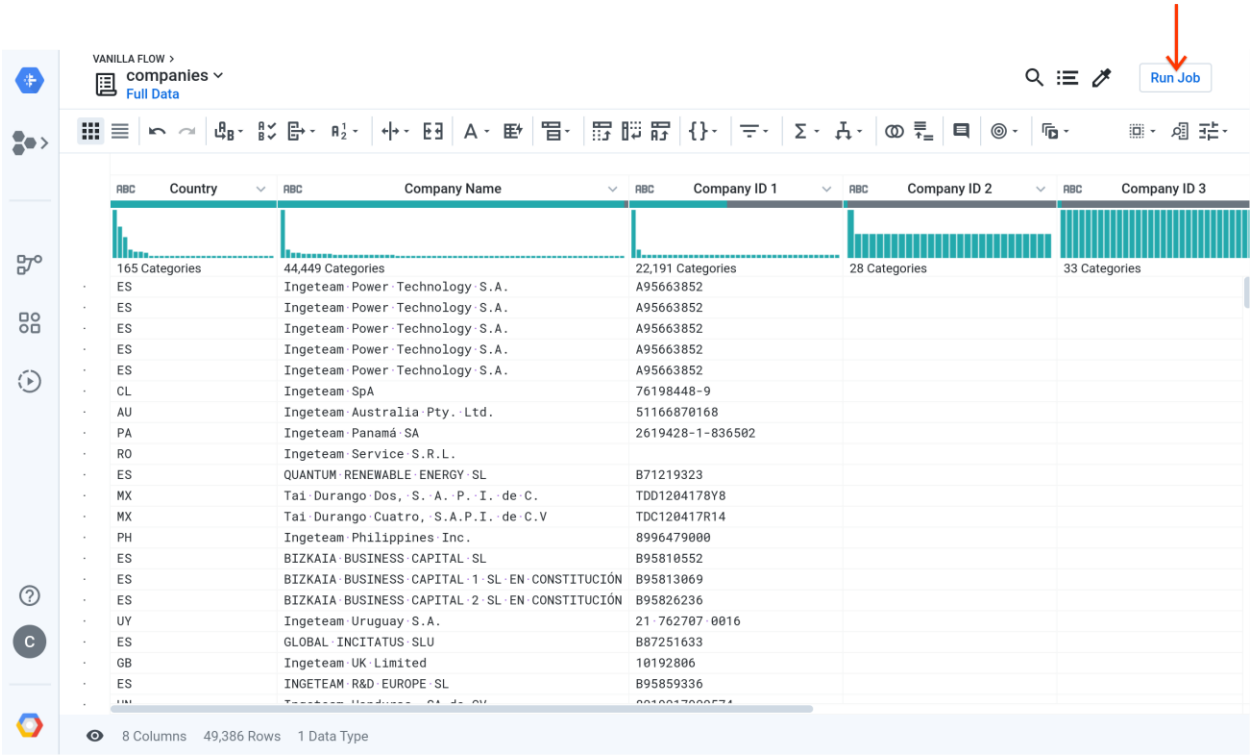
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



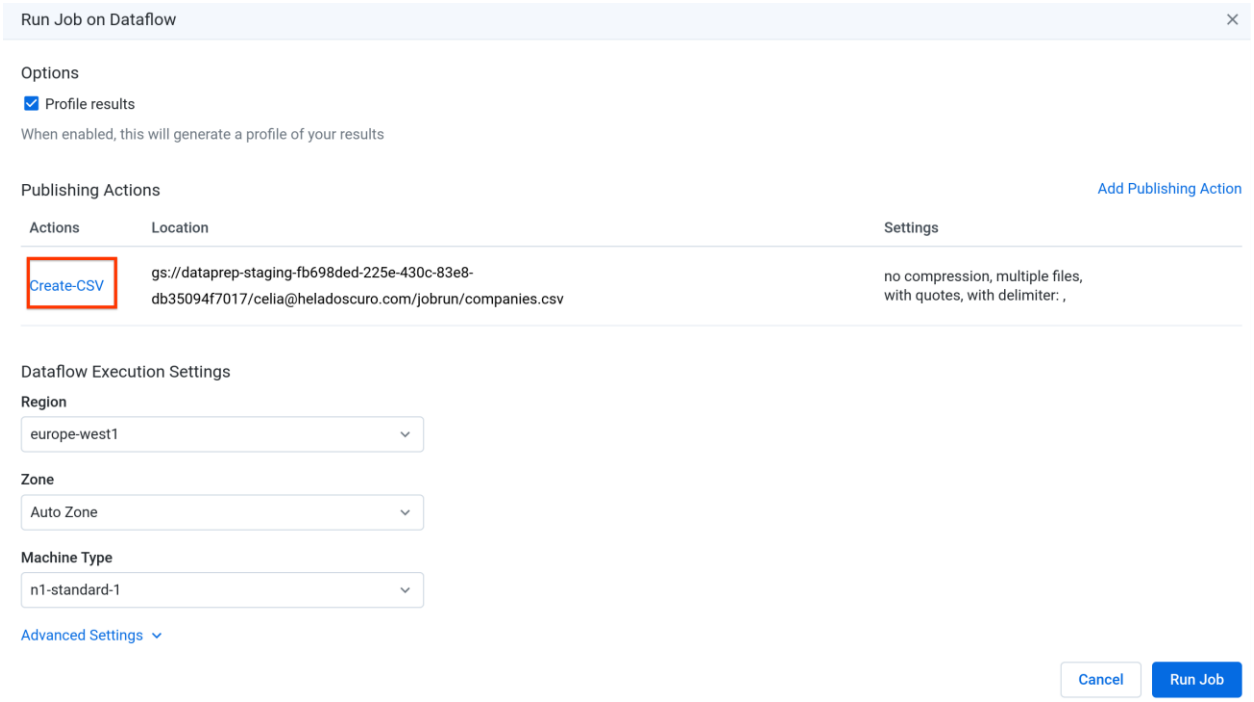
After all the transformations and cleaning we did on the data, we have 49,386 rows to upload and 44,449 different company names. There are duplicate company names because they have different IDs or they are placed on different cities.



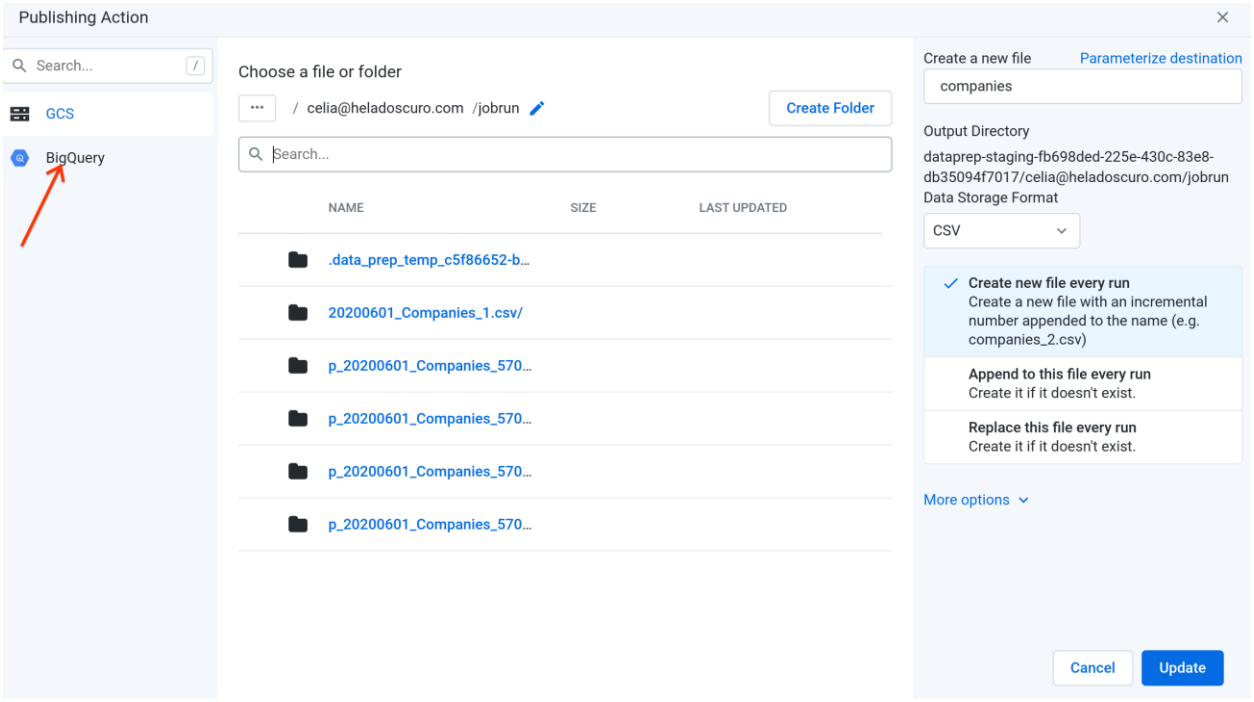
We click on “Run Job”.



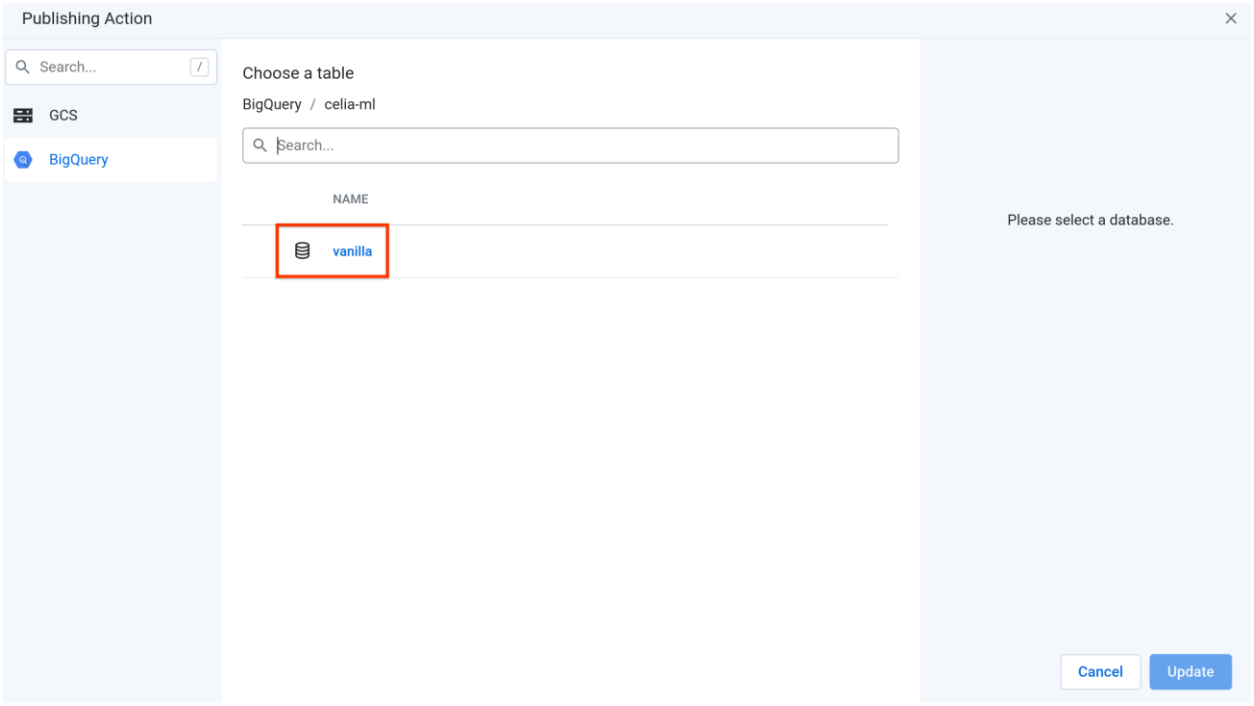
We click on “Create-CSV”. We are going to change this publishing action to append to our BigQuery table.



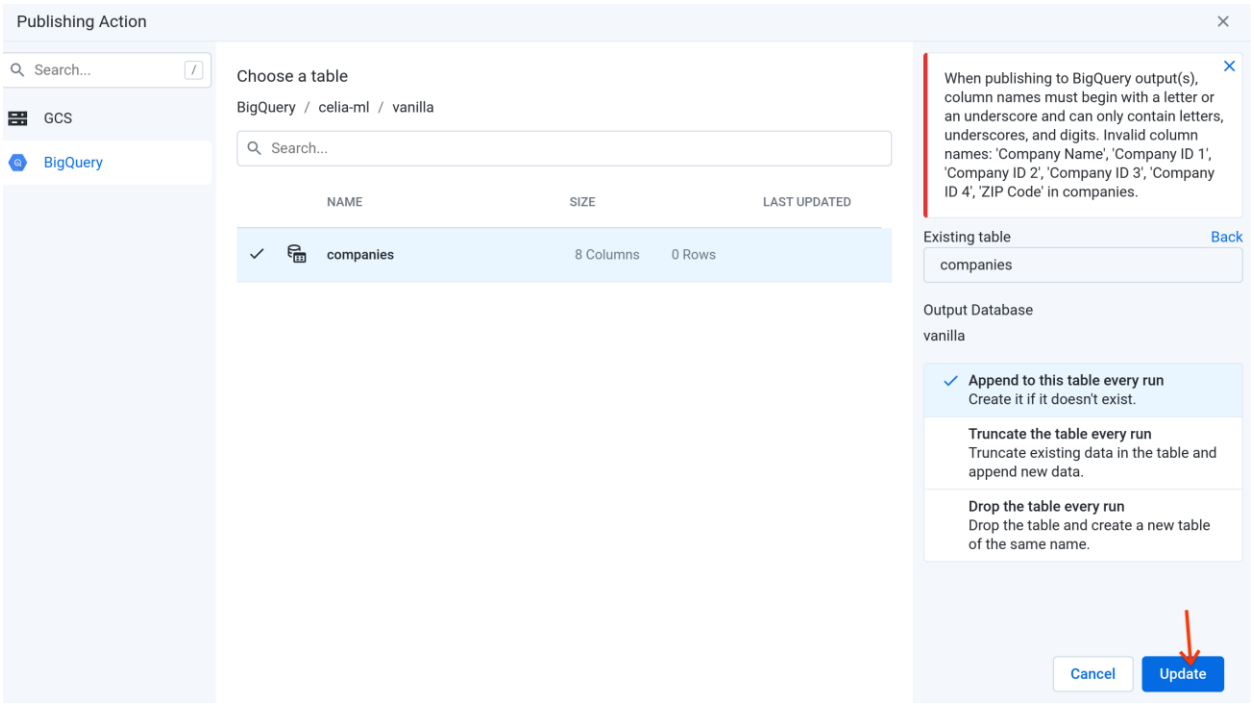
We select BigQuery

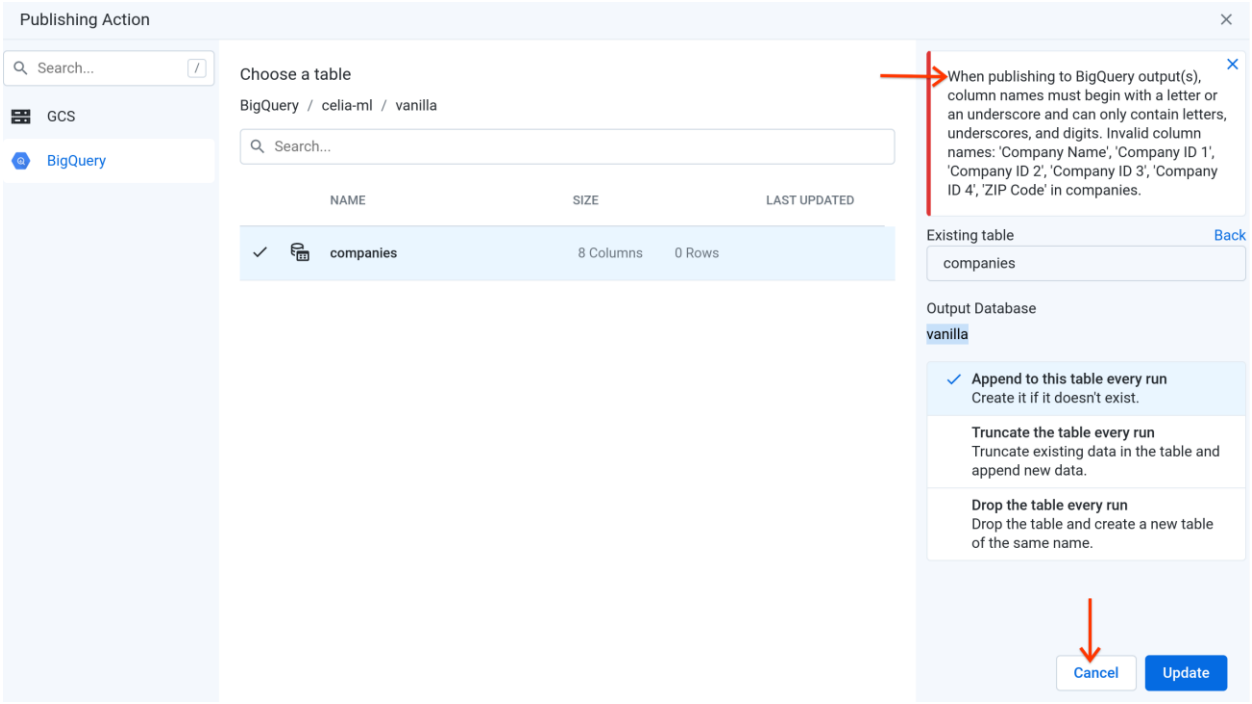


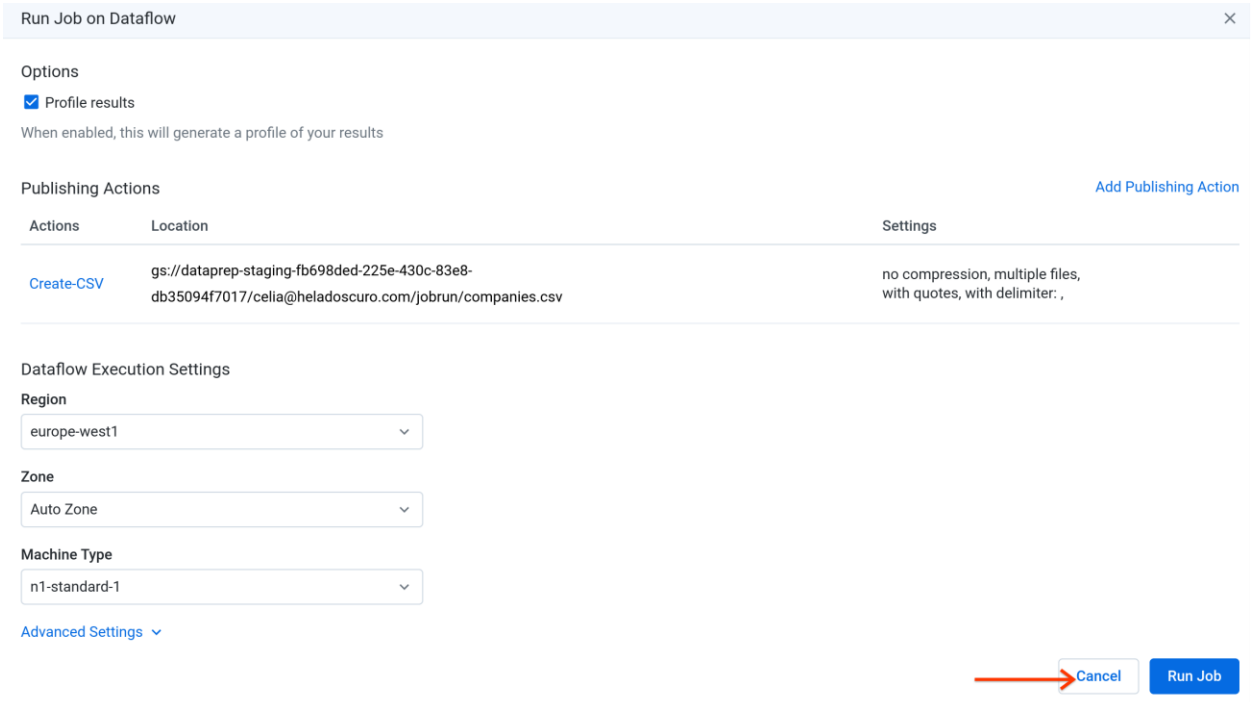
We click on our BigQuery dataset.



We try to append the data in our CSV file to our BigQuery Table. But we can't click on "Update" because we have an error message which says that the fields in the CSV file can't have spaces. So we cancel the publishing action and we come back to our recipe to edit the column names.

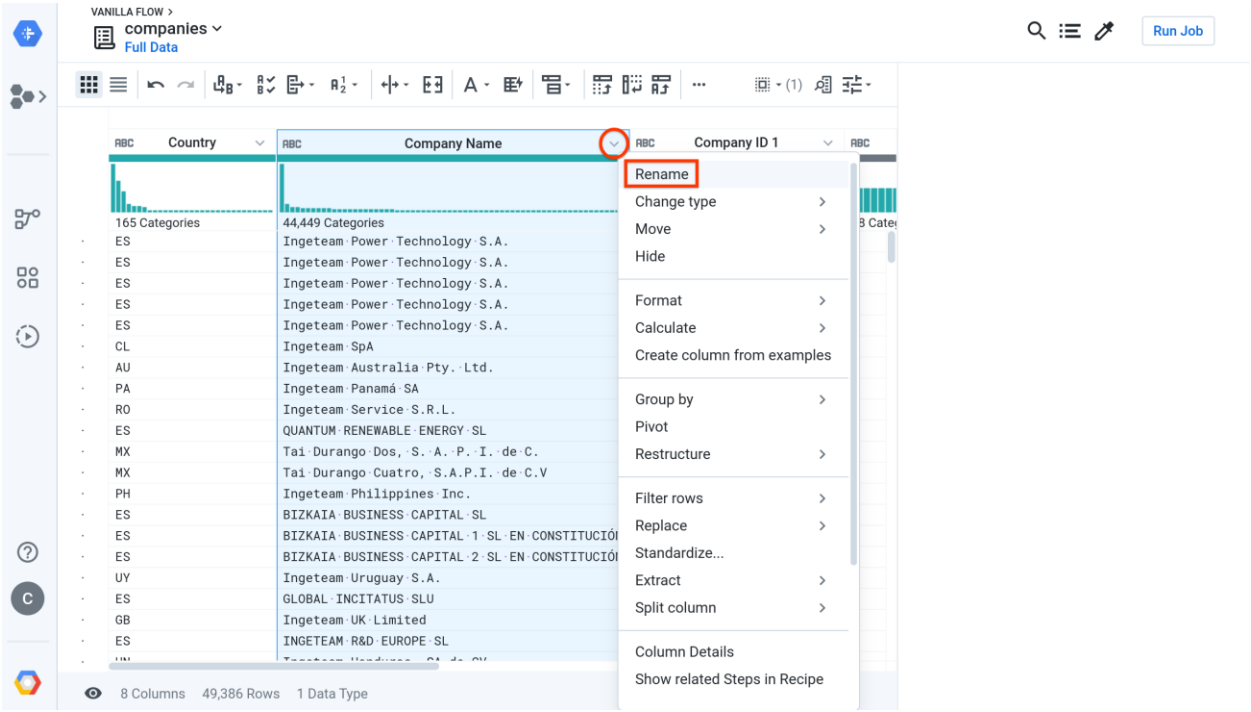


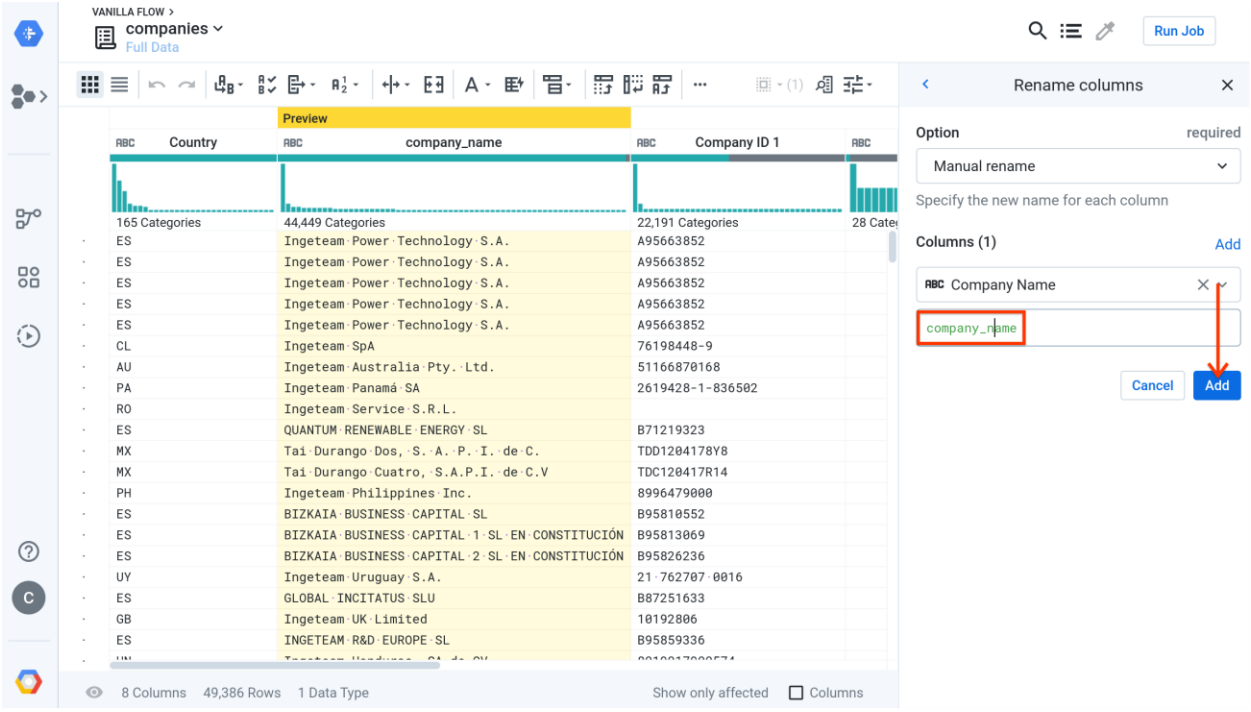




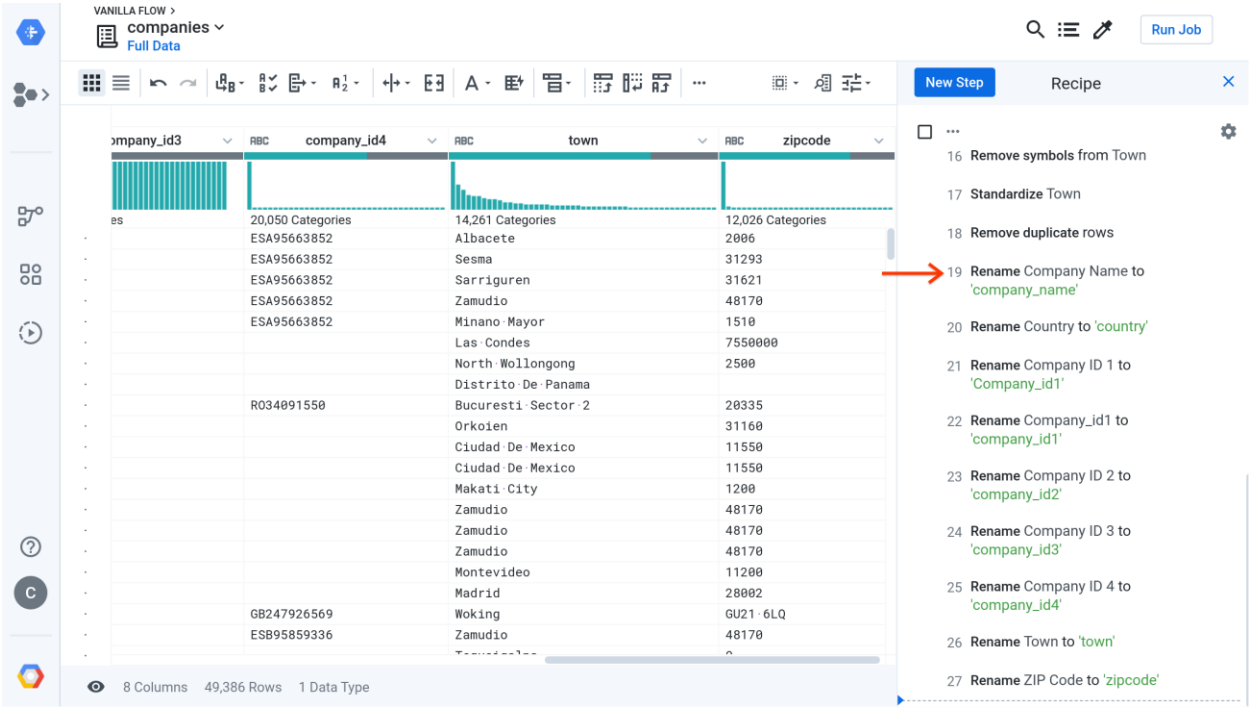
We rename the CSV column names.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com

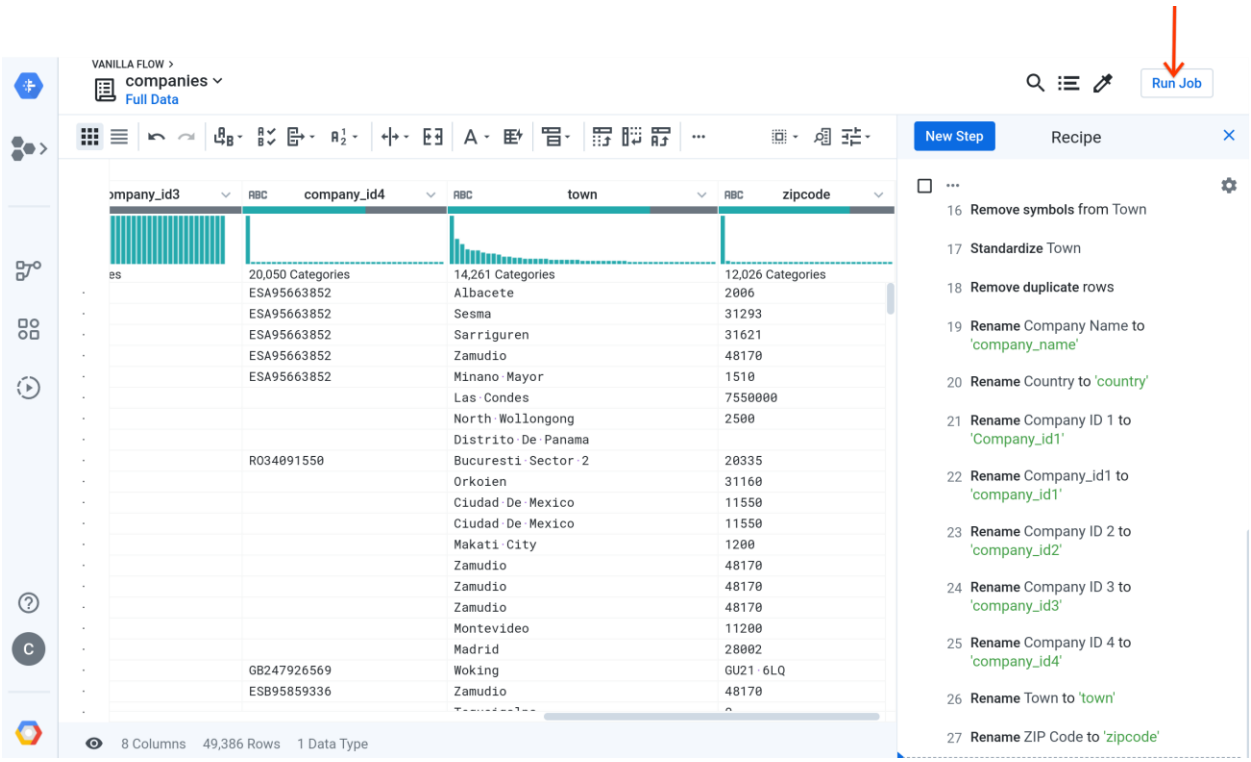




Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



Once all the CSV columns have the right name format, we click on “Run Job” again.



Inconsistent company names demo
Standardising values in a text column

Celia Muriel – celiamuriel.com

Run Job on Dataflow

Options

Profile results

When enabled, this will generate a profile of your results

Publishing Actions

Add Publishing Action

Actions	Location	Settings
Create-CSV	gs://dataprep-staging-fb698ded-225e-430c-83e8-db35094f7017/celia@heladoscuro.com/jobrun/companies.csv	no compression, multiple files, with quotes, with delimiter: ,

Dataflow Execution Settings

Region: europe-west1

Zone: Auto Zone

Machine Type: n1-standard-1

Advanced Settings

Cancel | Run Job

Publishing Action

Search...

GCS

BigQuery

Choose a file or folder

/ celia@heladoscuro.com /jobrun

Create Folder

Search...

NAME	SIZE	LAST UPDATED
.data_prep_temp_c5f86652-b...		
20200601_Companies_1.csv/		
p_20200601_Companies_570...		
p_20200601_Companies_570...		
p_20200601_Companies_570...		
p_20200601_Companies_570...		

Create a new file

Parameterize destination

companies

Output Directory

dataprep-staging-fb698ded-225e-430c-83e8-db35094f7017/celia@heladoscuro.com/jobrun

Data Storage Format

CSV

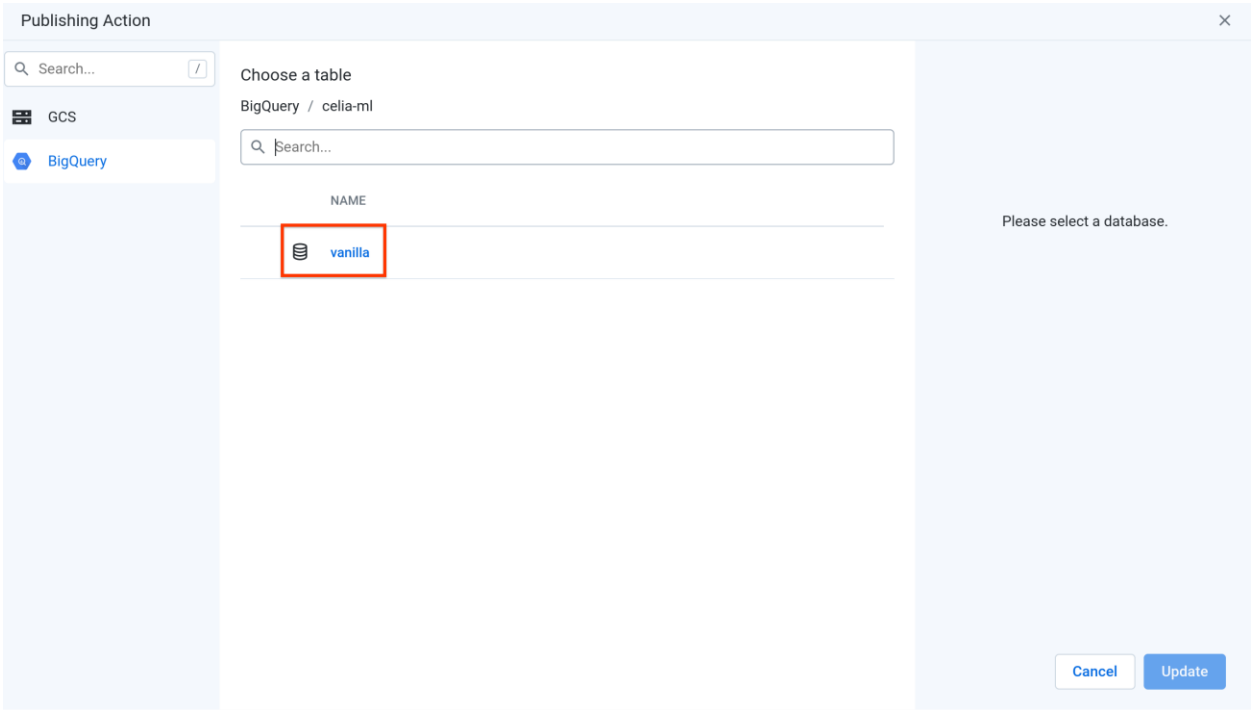
Create new file every run
Create a new file with an incremental number appended to the name (e.g. companies_2.csv)

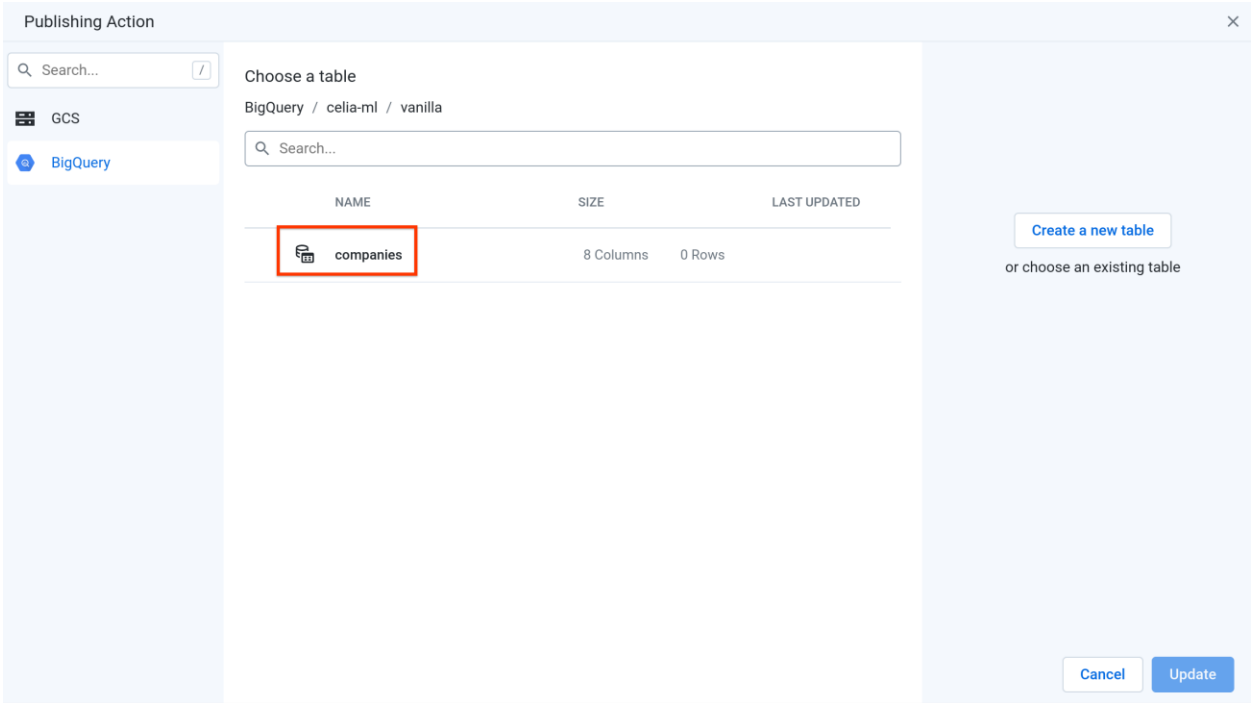
Append to this file every run
Create it if it doesn't exist.

Replace this file every run
Create it if it doesn't exist.

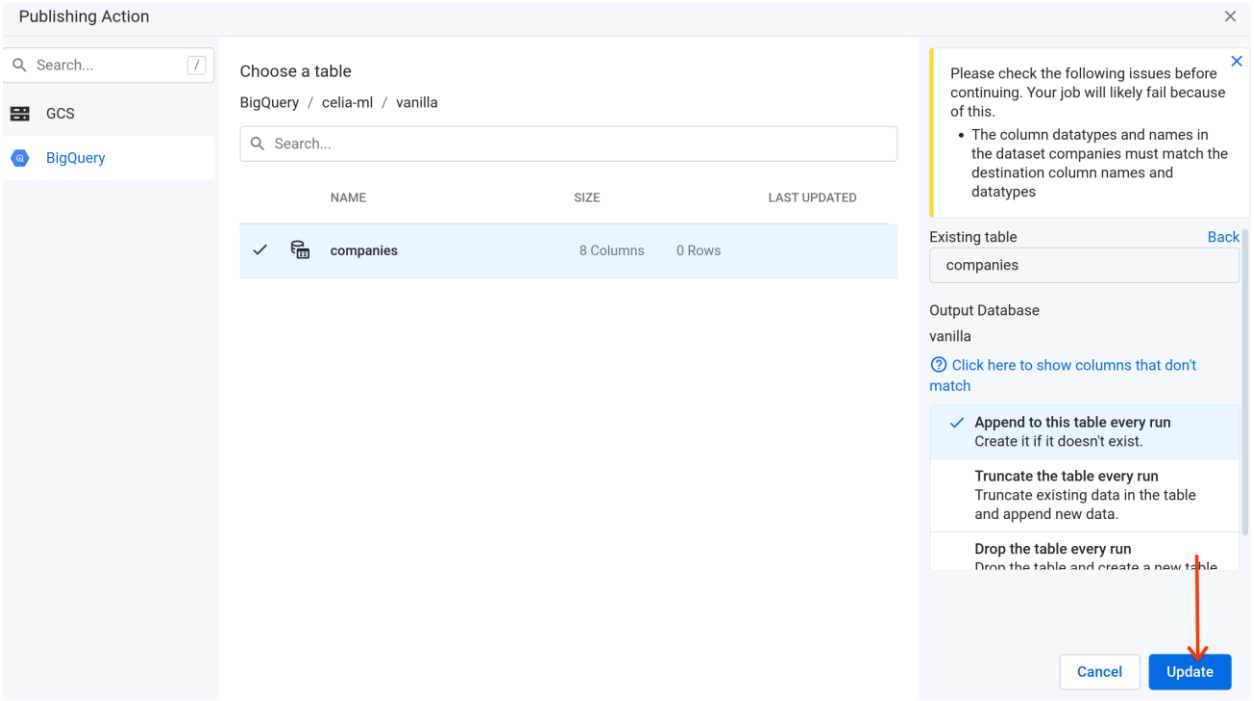
More options

Cancel | Update

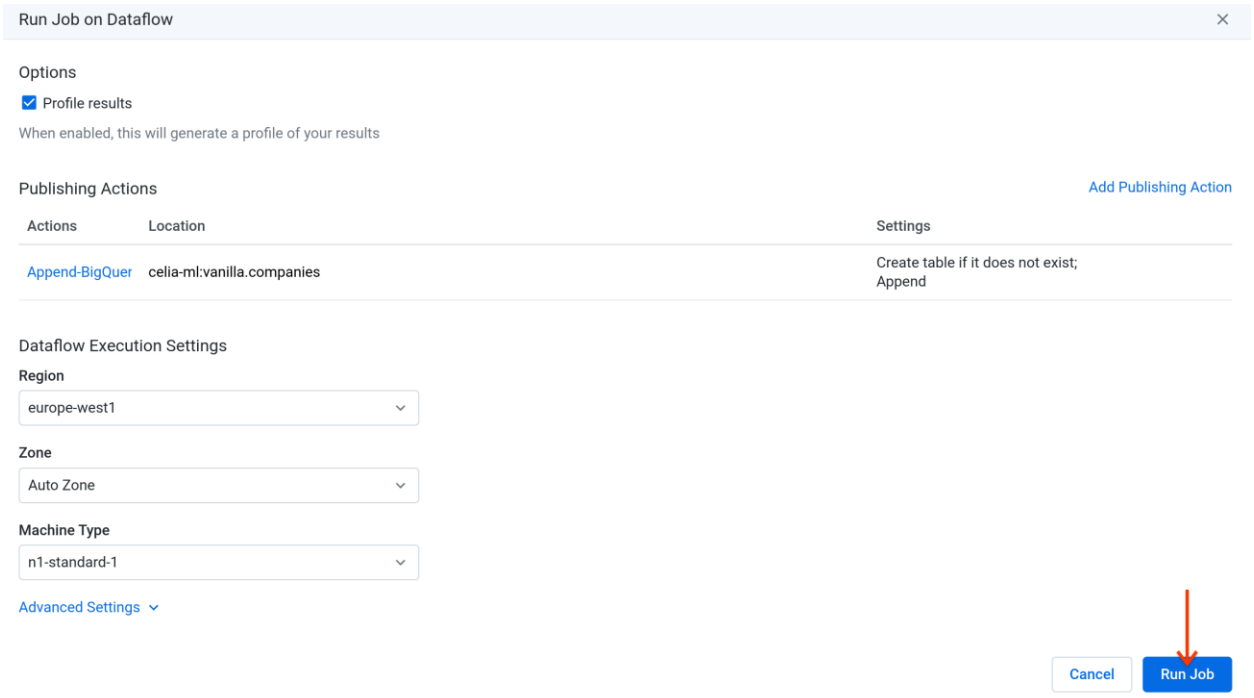




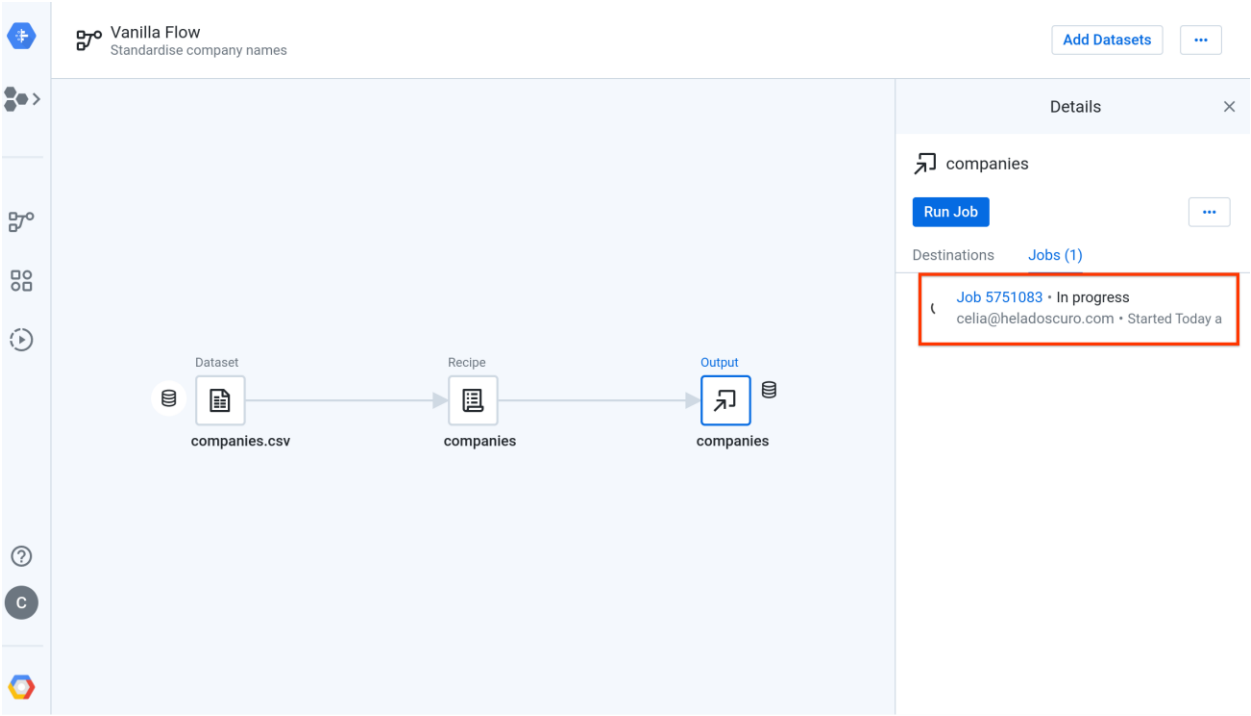
We select “append to this table” and we click on “Update”.



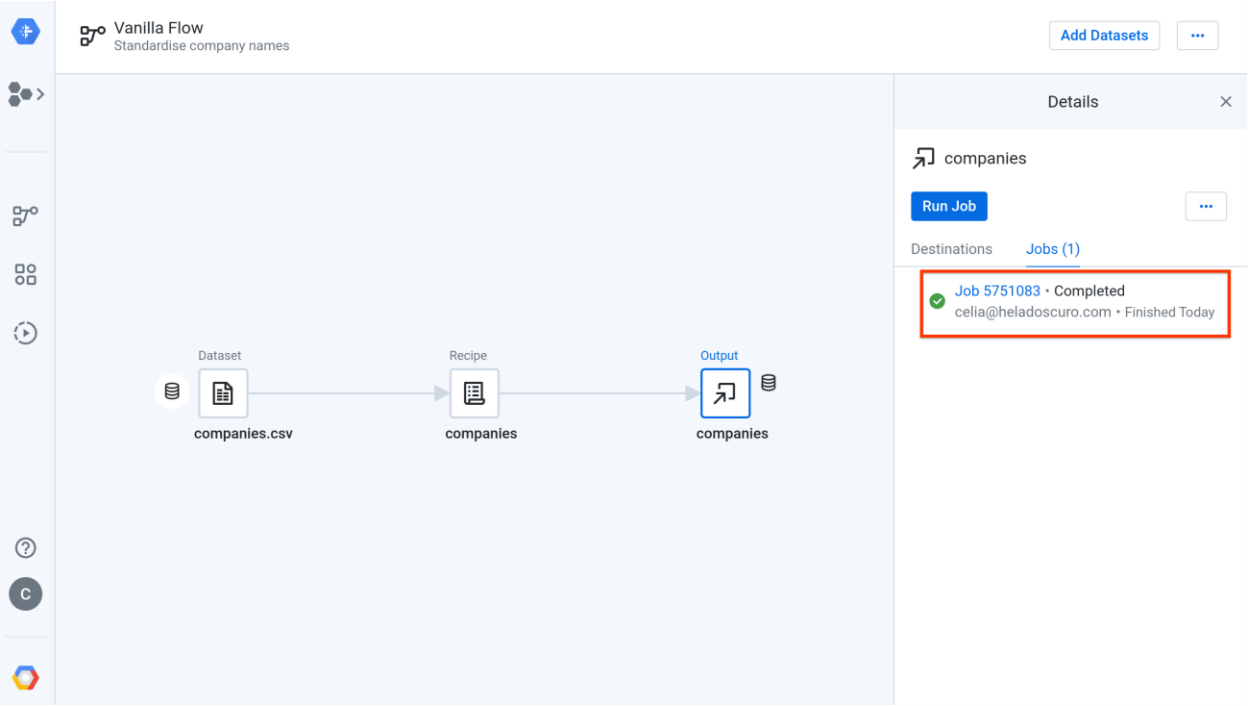
Click on “Run Job”.



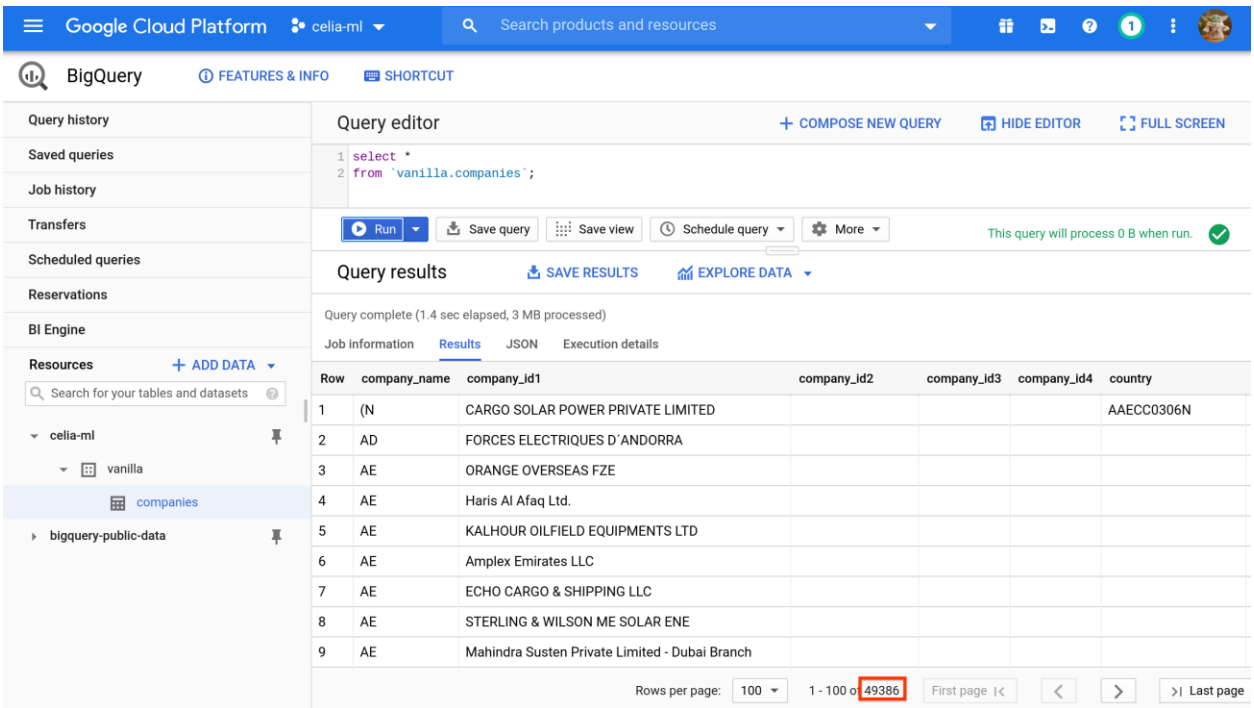
Cloud Dataprep is the combination of Trifacta software for data preparation and Cloud Dataflow to upload the data. When we hit on “Run Job”, we trigger a Cloud Dataflow job which actually copies the data from the CSV file to the BigQuery table.



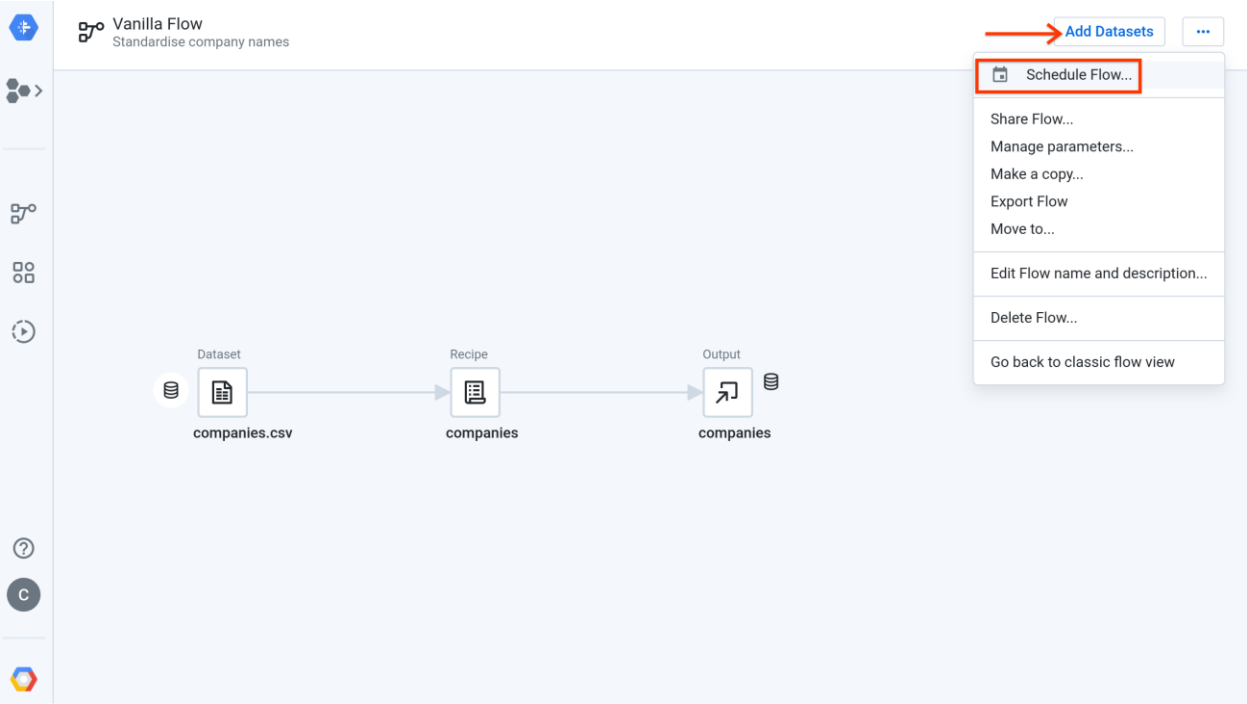
Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com



Once the job completes successfully, we check in BigQuery that we upload the standardised companies data into our table.



If we want to schedule our job to run regularly to upload the data from our CSV file to our BigQuery Table, we come back to the Dataprep flow. We open it, and we click on the 3 dots on the top right corner. We click on “Schedule Flow”.



Recipe and flow

The complete recipe and flow with all transformations done to prepare the companies.csv data to load are available in these links.

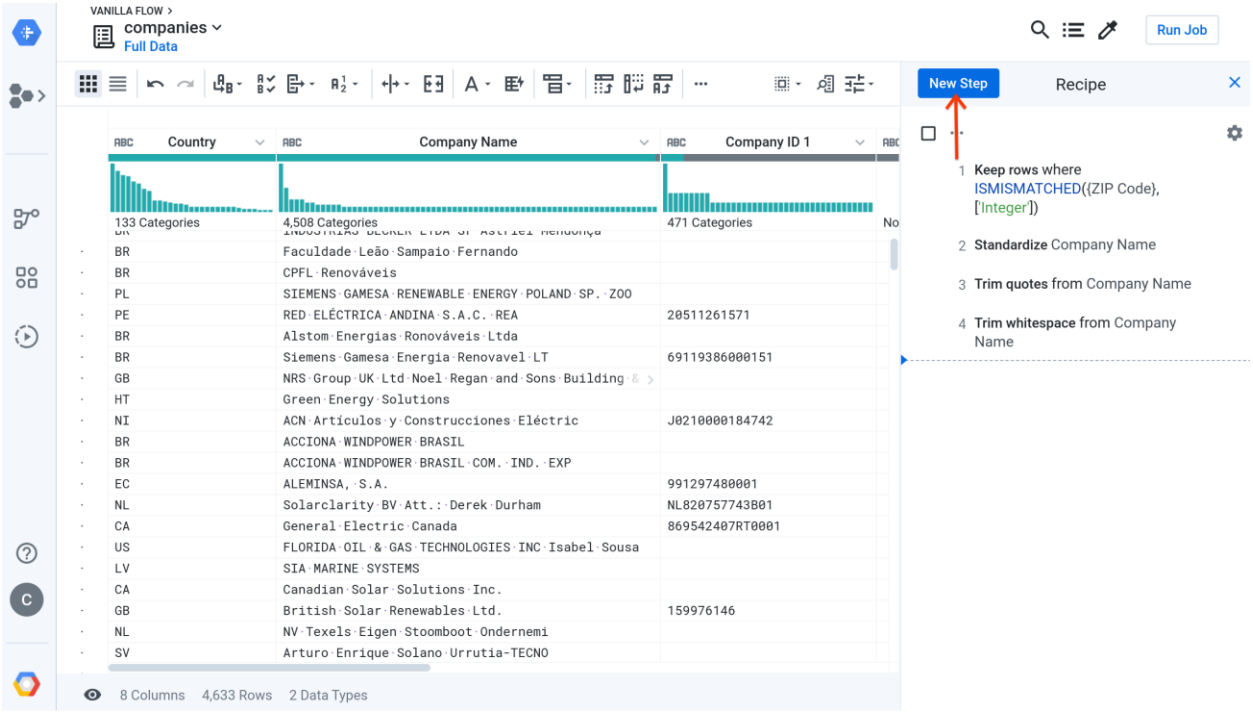
Other option

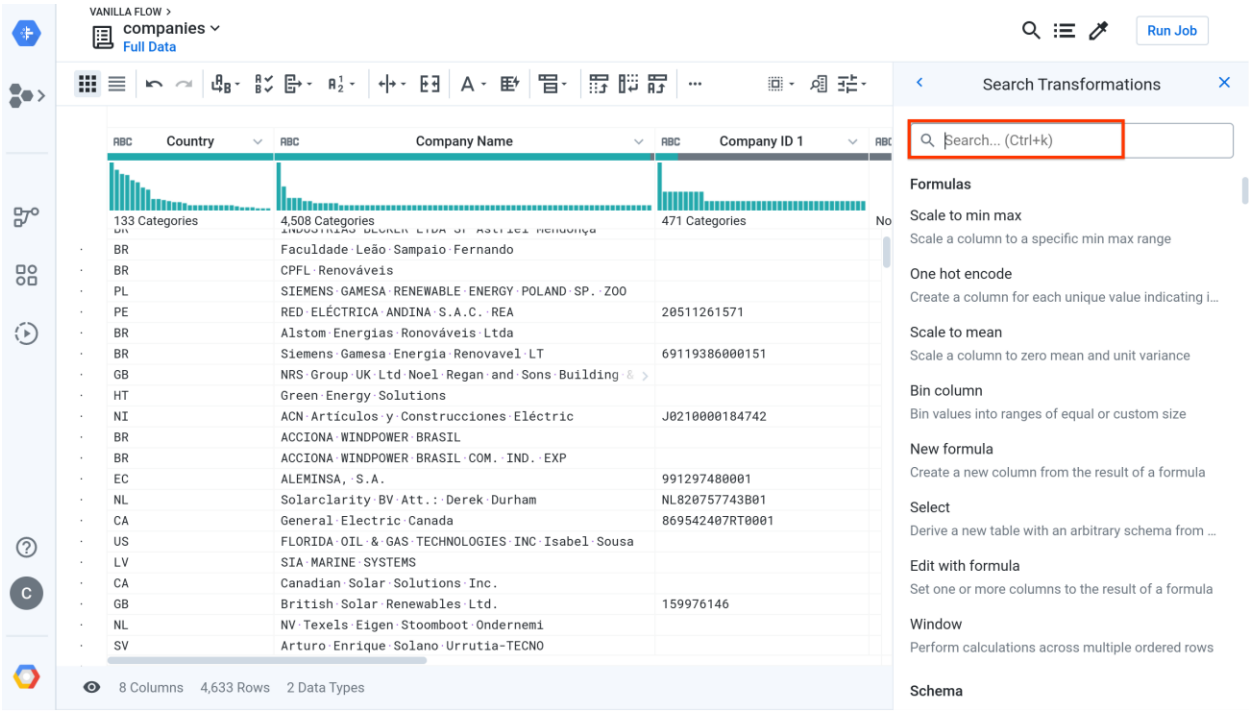
If we have a master table in BigQuery with the standardised company names, we can also create a flow in Cloud Dataprep where we join the source CSV file and the master table. Then we can compare records using the DOUBLEMETAPHONEEQUALS function.

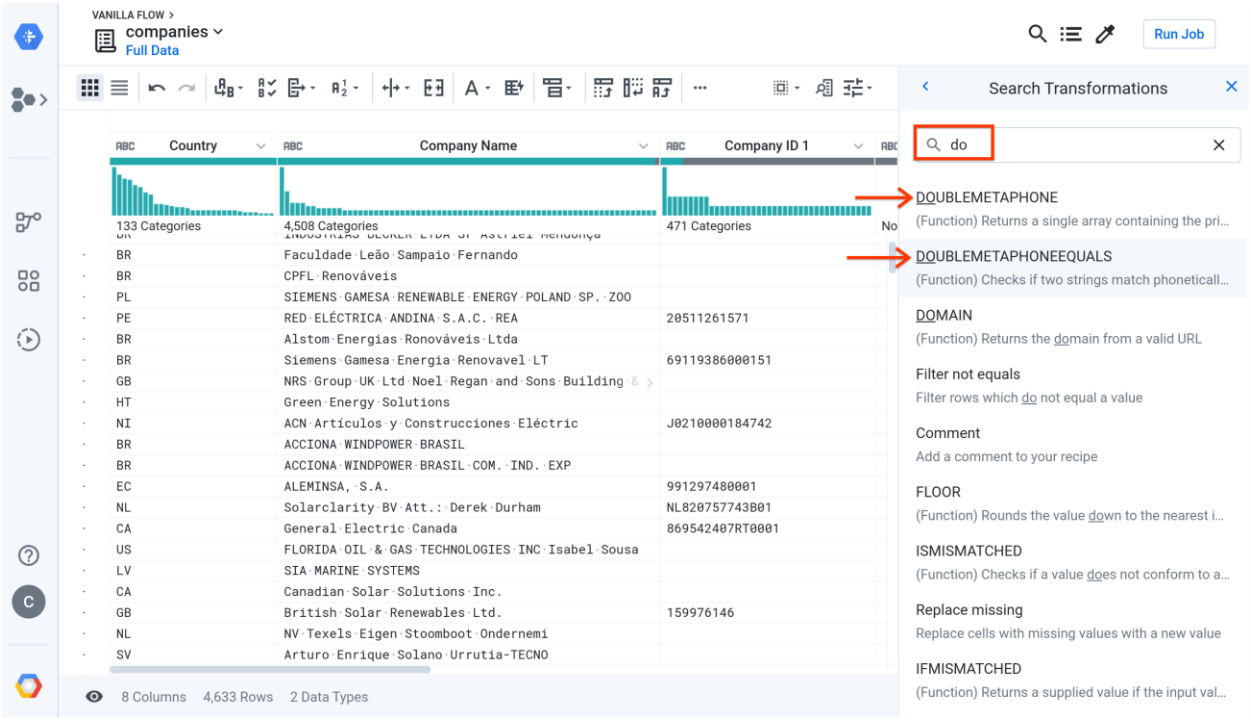
The double metaphone algorithm is based on phonetic coding. It encodes an English word phonetically by reducing them to a combination of 12 consonant sounds. It also attempts to encode non-English words. It returns two codes if a word has two plausible pronunciations.

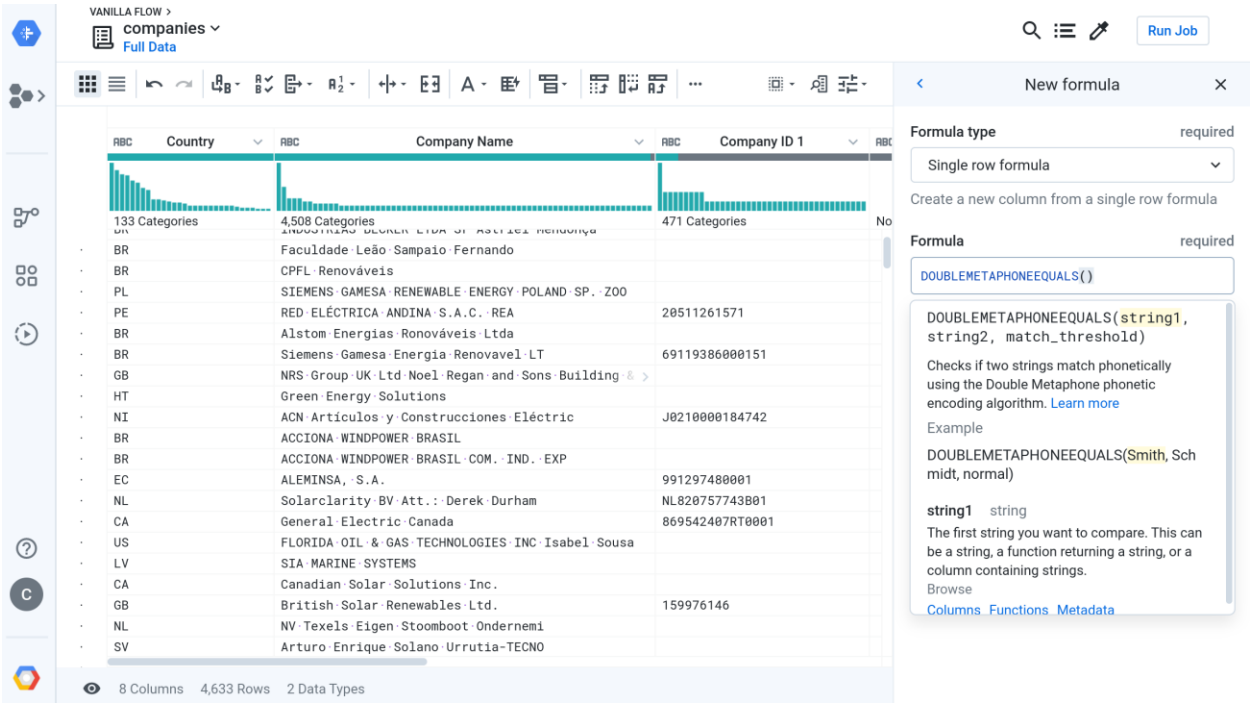
If we want to add a function as the DOUBLEMETAPHONEEQUALS, we must open the recipe again, and proceed as shown in the screenshots below.

Inconsistent company names demo
Standardising values in a text column
Celia Muriel – celiamuriel.com









Other fuzzy matching techniques, such as the Levenshtein Distance, must be implemented by the user as functions in Trifacta to use them.

References

Trifacta

Advanced Data Cleanup Techniques using Cloud Dataprep. Cloud Next '19. Google Cloud Platform. Accessed November 7th, 2021.

Quickstart. Cloud Dataprep by Trifacta. Google Cloud. Accessed November 7th, 2021.

Enabling and Disabling Dataprep by Trifacta. Cloud Dataprep by Trifacta. Google Cloud. Accessed November 7th, 2021.

User Profile Page. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

Overview of Standardization. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

DOUBLEMETAPHONE Function. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

DOUBLEMETAPHONEEQUALS Function. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

Wrangle Language. Documentatio for Trifacta Wrangler. Accessed November 7th, 2021.

Compare Strings. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

Join Window. Documentation for Trifacta Wrangler. Accessed November 7th, 2021.

David McNamara. December ’19 Wrangler Release – Rapid Target Fuzzy Match, UI Improvements, Downloadable Profiles. Trifacta Blog. December 18th, 2019. Accessed November 7th, 2021.

Bertrand Cariou. New AI-driven features in Dataprep enhance the wrangling experience. Google Cloud Blog. April 8th, 2020. Accessed November 7th, 2021.

Other

Celia Muriel. Fuzzy Matching or approximate string matching. Available on December 13th, 2021.

Using the bq command-line tool. BigQuery. Google Cloud. Accessed November 7th, 2021.

Dataflow. Google Cloud. Accessed November 7th, 2021.