# **OpenRefine Tutorial**

Here is the checklist of what we will do in this OpenRefine tutorial. You can **“Make a copy”** of this Google Docs to track what you have done.

### 1. Load data and create the project

* Open OpenRefine by running “openrefine.exe” from the “openrefine-3.8.2” folder.
* Create a new project by selecting This Computer > Choose File > Next ».
* Enable “Attempt to parse cell text into numbers” to try converting all numbers.
* Click on the “Create project »” button.

### 2. Clean the “university” column

We observe that university names contain special characters. For example, “Lumi%C3%A8re University Lyon 2” should be “Lumière University Lyon 2.”

* On the dropdown menu for the “university” column, select Edit cells > Transform…
* Use the expression value.unescape('url') in the expression textbox.
* Preview changes before clicking the OK button.

### 3. Clean the “country” column

Explore and correct the country name with the “Text Facet.”

* On the “university” dropdown menu, select Facet > Text Facet.
* Look on the “Facet / Filter” on the left panel. Have you noticed anything wrong with some country names?

Use the “Cluster & Merge” function to clean the country names.

* On the Text Facet, click the “Cluster” button at the top-right of the facet to group similar country names.
* Try out different different methods and keying functions. Which method should we use to merge the same country names?
* Verify that the merge from the algorithm is correct. If so, enable the “Merge?” checkbox. You can also manually edit the “New Cell Value” before the merge.
* Click on “Merge selected & re-cluster” to finalize the merge.

The “Cluster & Merge” function cannot detect many mistakes in country names. Therefore, we can change those country names manually.

* On the Text Facet, look at the country names that still haven’t been corrected from Cluster & Merge algorithms.
* Hover on that country's name and click “edit” on the right side.
* Input the new country name in the text box. Then, click “Apply” to change the country name.

### 4. Clean the “numStudents” column

We will explore the data in “numStudents” column by creating the numeric facet.

* On the numStudent column, click the dropdown menu and select Facet > Numeric facet.
* Looking at the histogram, have you noticed the outlier in this dataset? You can use the slider to filter the data.
* Also, check the rows that have non-numeric on this column. Have you noticed any patterns?

We will first handle the non-numeric values by converting some of them into numbers.

* While selecting only non-numeric rows, on the numStudents dropdown menu, choose Edit cells > Transform.
* Use this GREL expression to extract only numbers from the strings: value.replace(/[^0-9]/, "")
* Notice that the numbers we extract from the GREL expression are not numeric data types yet. On the dropdown menu, select Edit cell > Common transforms > To number.

### 5. Clean the “endowment” column

* **Create the numeric facet** for the “endowment” column. Again, select the “non-numeric” rows on the numeric facet and explore the patterns of those rows. Detect any typical pattern so that we can transform the data into the correct format.
* **Replace “US$”** with the empty string with this expression: value.replace("US $","").replace("US$", ""). Convert the string to numeric values using Edit cell > Common transforms > To number.
* **Replace “million”** with the empty string and recalculate the endowment with this expression: toNumber(value.replace("million", "").trim()) \* 1000000.
  + You can also use regex expression: toNumber(value.match(/(\s?\d+) million[^,]\*$/)[0]) \* 1000000
  + Note that the toNumber function has already converted the string to numeric values, so you don’t need to convert them to numeric data type again.
* **Remove non-numeric values** by selecting Edit cells > Common transforms > To null.

### 6. Clean the “established” column

* **Create the timeline facet** for the “established” column. On the university dropdown menu, select Facet > Timeline facet. Filter “non-time” values to explore the patterns. You can see that none of the values in this column are not datetime data types yet.
* **To convert those values into date time**, on the dropdown menu, select Edit cells > Common transforms > To date. Notice how many rows are successfully converted and look at those non-time values.
* **Transform the year to date format** using this transform expression: toDate(value.match(/.\*(\d{4}).\*/)[0]). OpenRefine assumes that the values we transformed are years (YYYY-01-01).
* **Create a new column to store the established year.** On the established dropdown menu, select Edit column > Add column based on this column… Name the new column “establishedYear” and extract the year from datetime using toNumber(value.toString('yyyy')).

### 7. Remove duplicate universities

You may also detect that there are many universities with duplicate rows. We are going to remove them.

* On the university column, click on Sort… in the dropdown menu. You will see the pop-up modal. Sort the university names alphabetically from a - z and click OK.
* Notice that the row indices remain the same, so the sort is not permanent. We need to reindex the rows before removing duplicated rows. On the top tab above the table, click Sort > Reorder rows permanently.
* To remove the duplicated row, we will keep only the top row for each university and remove the other rows below. To do so, click Edit cells > Blank down on the university column dropdown. This operation will keep the university name on the top row and remove others below.
* Next, we will select the duplicated rows by filtering the rows with the blank values. On the university column dropdown, click Facet > Customized facets > Facet by blank (null or empty string).
* The university facet will appear on the left tab of the page. Click on “true” to select only the rows with blank university values.
* To remove these duplicated rows, on the “All” column on the left-most side of the table, click Edit rows > Remove matching rows.

### 8. Export the cleaned data.

By default, the OpenRefine projects are [stored on your computer](https://openrefine.org/docs/manual/installing#:~:text=Depending%20on%20your%20version%20of,Local%20Settings%5CApplication%20Data%5COpenRefine). Your workspace folder location depends on your OS version.

Finally, we will export the data after cleaning and the log file from OpenRefine for reproducibility. The “Export” button is on the top menu of the web page.

* **Export the cleaned data:** Export > Comma-separated value.
* **Export the OpenRefine log for this project:** Export > OpenRefine project archive to file. The file will be downloaded in .tar.gz format. It contains both data and log files necessary to recreate the project.

### 9. What’s Next

* Think about something else to improve the quality of this dataset.
* Try column reconciliation with WikiData: Reconcile > Start reconciling…
* Geocoding names and addresses with OpenStreetMap Location API
  + API Documentation: <https://nominatim.org/release-docs/latest/api/Search/>
  + Request from the API: "https://nominatim.openstreetmap.org/search?format=json&amp;q=" + escape(value, "url")
  + Extract coordinates: value.parseJson()[0]['lat'] + ', ' + value.parseJson()[0]['lon']
* Try to use OpenRefine to clean other datasets: <https://evanwill.github.io/openrefine-b/content/3-demo.html>