

# S&DE Atelier - Visual Analytics

## Assignment 2 - part 2

Due May 8, 2023 @23:55

**Contacts:** [marco.dambros@usi.ch](mailto:marco.dambros@usi.ch) - [carmen.amenti@usi.ch](mailto:carmen.amenti@usi.ch)

The goal of this part of the second assignment is to use Elasticsearch and Kibana to solve different problems. The submitted zip file should be named as:

`SurnameName_Assignment2_part2.zip`. This should contain your solutions and the steps followed to arrive to these solutions.

The dataset you need for this exercise is in the csv named *restaurants\_extended.csv*.

### Exercise 2 - Data visualization (40 points)

The dataset for this exercise is similar to the one used in the previous exercise; it has some additional information.

#### Dashboard creation

The goal of this section is to create a dashboard in Kibana. We would like to visualize some information about the restaurants collected in the dataset.

Please create a dashboard that:

1. Shows some metrics: how many restaurants were reviewed, how many cities were covered and the total number of votes collected;
2. Provides a filter to choose the restaurant rating ( `AggregateRating` );
3. Shows the top 5 cities with respect to the average cost of all restaurants in that city. You should consider only the cities with a minimum number of restaurants equal to 25;
4. Shows the trend of restaurants review over time. Please make a distinction between **positive reviews** - rating from more than 3 to less than or equal to 5 - and **negative reviews** - rating from 0 to less than or equal to 3 - and provide a comparison between the previous and the **total reviews**;

5. Shows a map with cluster or grids of the location of all restaurants;
6. Shows a table view with details about all restaurants;  
💡 This should not be a table view in visualization but a saved search.
7. Shows a heat-map that gives information about the average of the number of votes for each continent and each `RatingText` type; the visualization should be shown in decreasing order. Please, solve this by using scripted fields.

## Canvas creation

The goal of this section is to create a canvas that shows the restaurants rating per city.

Please create a canvas that:

1. Allows filtering the data based on the city selected;
2. Shows the average cost of the restaurant for 2 people based on different rating categories ( `RatingText` );
3. Shows the percentages of cheap and expensive restaurants. Please consider cheap all restaurants whose `AverageCostForTwo` is less or equal than 300 (based on this assumption, the expensive ones have an average price from 300 and above);
4. Shows a table - with 5 rows per page - with the following information about the restaurants: the name, the average cost, and the rating text;
5. Shows some metrics about the restaurants: the number of votes, the average cost, and the average rating.

## Exercise 3 - Ingestion Plugin (30 points)

The goal of this exercise is to extend Elasticsearch by building an ingestion plugin (<https://www.elastic.co/guide/en/elasticsearch/plugins/current/index.html>). Ingestion plugins enrich the capabilities of Elasticsearch by providing additional logic at ingestion time.

The ingestion plugin to be implemented is a lookup plugin: given a field to operate on, and a lookup map, when ingesting documents the plugin will replace all instances of the keys with the values of the provided lookup map, in the context of the given field. The idea is to, for example, replace product codes with product names, or replace id-emails (used in some universities) with the corresponding name-emails. For example, if my

documents are about cars and I know that code C001 means tyre, when ingesting a document I want to replace all occurrences of C001 with the term tyre, so that a document with field = "Need to optimize the C001 temperature", would be indexed as "Need to optimize the tyre temperature".

To implement this plugin, you are provided with a skeleton repo, containing all the scaffolding and structure needed to create an ingestion plugin. You need to add/fill the classes as needed and implement tests.

The repo is available here: <https://gitlab.com/usi-si-teaching/msde/2022-2023/visual-analytics-atelier/elasticsearch-plugin/ingest-lookup-stub>

Make sure that the plugin works as expected by (1) installing it on your local instance of elasticsearch (2) setup a pipeline, (3) ingest a document, and (4) retrieve the document checking that the text was replaced as expected. The [README.md](#) file includes instructions on how to do that.