Project - Introduction to Data Engineering

Objective:

Create a *simple Data Application* or a *simple REST API :* this must provide (a) simple service(s) to help users. You should use the knownledge you learned during the courses and the tools and the techniques you have seen during the courses to do the project. You will use the datasets you have seen during the courses and the TPs.

Examples of service:

- Show on a map the most popular restaurants/pubs/others around specific locations,
- Show the « bike popularity » of specific streets or places and show how it evolves throught time, display the results in an App or return results as a JSON (API),
- Help a new shop to pick up the best place to set up a shop,
- Your own idea ! (to get your own idea : you should first have a look at all the course's datasets and see what kind of information you can treat, join, mix and so on. You may need an external dataset that you can find on the internet in case you don't have enough interesting info to combine).

Deadlines:

1) Week 8, Thursday – Present a project proposal : Think of an interesting Data Application or REST API project proposal and make a very short presentation to describe your idea (max 3 slides). We will take 5 mins for each group to discuss the feasibility and we will discuss the interest of the proposal.

2) End of Trimestre (14th of June 23h59) – Upload the project zip : Compress in a zip archive all the project files you needed (.py files, .ipynb files, a docker-compose.yml file, eventual Dockerfiles, etc.) and upload the zip file to the ChaoXing platform before the deadline. In the zip archive, do not forget to *also upload a small video* (should not weight more than 20Mo) to make a demonstration of your application or API.

NOTE : DO NOT UPLOAD THE DATASETS (weibo, maituan, etc). If you have used external datasets, give URL addresses to download them.

Datasets:

Weibo Dataset, Maituan Dataset, etc. You can use all datasets you have been using during the courses/TPs.

Tools:

To make this Data APP or REST API you can use amongst the tools we have seen together : Pandas, Numpy, **Streamlit+Seaborn (*if you want to build an application*)**, **Flask (*if you want to build a REST API*)**, PySpark, MongoDB, ElasticSearch, etc.

Constraints:

The teaching team should be able to re-run all your cleaning, processing, storing code and be able to re-run and test your final API or App. Tips: design a docker-compose.yml file so we can easily run your App/API and test it.

Groups: 2 people max.

Rating: The project will count for 50% of the final mark (For the correction we will pay particular attention to how you clean, process and store your data as well as which tools you chose to perform these tasks. It is also very important that we can run and test your code).