Task

Make a simple web service that will solve at least one of the four offered tasks for an authorized user.

You will have to:

1. Download the latest Basic (free) version of geolocation data (currently it is: simplemaps_worldcities_basicv1.74.zip) from https://simplemaps.com/data/world-cities
2. Import data from a CSV file. The CSV file will contain data about countries, cities, population, and geolocations of those cities.
3. Create a database whose table will contain:
    a. Users
    b. Cities
    c. Countries
4. Create an unauthorized API for users' registration. It is necessary that the API receives a username and password and that if the username is available it returns a token[1] as if the user is logged in.
5. Create an unauthorized API for logging in of a registered user. The form should be the same as for users' registration but it will return a token if the user exists and his password is correct and if that is not the case it will return a message about the error with a suitable HTTP status code.
6. All other APIs must be authorized, i.e. require that the token is provided in the header of each message.
7. Create a script that will read the CSV file and populate tables of Cities and Countries having in mind referential integrity between the countries and their cities.
8. Create an API that returns all countries and allows searches by a parameter.
9. Create an API that returns all cities in a selected country and allows searches by a parameter.

Complete at least 1 of the following 4 tasks:

A. Create an authorized GET - API that will calculate and return two nearest[2] and two farthest cities for the selected country (API expects country ID in parameter).

---

[1] For token in (3. and 4.), you can use whatever you want, e.g. a number, hash, UUID, but the best solution would be to use JWT. if you decide to use an ID as a token, you can store this id into Users table or, for example, you can create a specific table Sessions.
[2] For distance on earth you can use Haversine given here https://en.wikipedia.org/wiki/Haversine_formula

B. Create an authorized GET - API that will find a cluster of three nearest cities for the selected country.
C. Create an authorized GET - API that will calculate the northernmost, easternmost, southernmost, and westernmost cities for the selected country and return information about their distance from each other.
D. Create an authorized GET - API that will, for the set of selected countries, return information about the city with the largest population, the city with the smallest population, as well as the total number of residents in all cities for each country in the set.

Stack

The acceptable programming stack is:

- Python 3+, or Javascript (Node.js) for code
- Any framework is acceptable (flask, fast API, Django, Express, etc…)
- Any SQL or NoSQL database

As an additional plus, we will consider
- Container usage (Docker, Podman,..)
- If you deploy your solution somewhere online and provide us access to the API and instructions for testing.

Delivery

You should open a private repository on GitHub (named with [your_first_and_last_name]-dc-march2022), and push your solution to that repo.

Provide a README file where you will precisely describe the process of requirements and installation for your solutions.

Invite our user digital-cube to participate, send us an email that you finished a task, and we will be able to fetch your code.