

FirstAPI :

For this project, we'll ask you to create an API from scratch with NodeJS & ExpressJS.

We don't want you to rush and try to go as fast as you can, we want you to be as serious as you can. No matter your actual Back-end skills, we are here to try, fail and learn.

Even if we advice you to read all the subject before start, we want you to process step by step and little by little (not too slowly of course...).

For the ones will only do the API you will work the C6;C7. For thoses who finished and have time, you can work the front to connect all the routes to a beautifull web page, and you will work the C2;C3 as well

Step 1 : Create a simple express server which send "Simple Text" when you make a /GET on "/" path, status 200.

Step 2 : Create a new route /GET on "/teachersName" who send country, status 200.

Step 3 : Create a file country.json and put all the information of <https://restcountries.eu/rest/v2/all> (Ctrl-c Ctrl-v, are accepted for this step, only).

Step 4 : Create a new simple express server (the goal of make you create a new server is only to make you practice !), which will send the content of country.json when /GET on "/all", status 200.

Step 5 : Create a new route /GET on "/names/all" which will send only the names of every country on the country.json file (Loop), status 200.

Step 6 : Create a new route /GET on "/names/all" which will send only the names of every country on the country.json file (Map), status 200.

Step 7 : Create a new route /GET on "/capitals/all" which will send only the capitals names of every country on the country.json file (Loop), status 200.

Step 8 : Create a new route /GET on "/capitals/all" which will send only the capitals names of every country on the country.json file (Map), status 200.

Step 9 : Create a new simple express server with /GET on "/country/:name" which will send you all the informations about this specific country (case insensitive) (if you find it..., status 200) (if not, you will send a 404 Not Found).

Step 10 : Create a new route /GET on "/regions/:regionName" which will send you, as an array, all the countries who are located on this specific region (case insensitive) (if you find some..., status 200) (if not, you will send a 404 Not Found).

Step 11 : Create a new route /GET on "/subregion/:subregionName" which will send you, as an array, all the countries who are located on this specific subregion (case insensitive) (if you find some..., status 200) (if not, you will send a 404 Not Found).

Step 12 : Create a new route /GET on "/currencies/:currency" which will send you, as an array, all the countries who have the same currency as the one asked (case insensitive) (if you find some..., status 200) (if not, you will send a 404 Not Found).

Step 13 : Download PostMan and try all your previous routes with this software.

Step 14 :

Create a new route /PUT on "/countries/:countryName" which will update a country with the field you want to change it :

- If I send /PUT "/countries/France" with {region: "Bretagne", subregion: "Finister"}, you will modify the France country, return this country and save those informations on the JSON file

SCREEN 1

Step 15 : Create a new route /DELETE on "/countries/:countryName" which will delete from the JSON file the specific country you asked (case insensitive) (if you find it..., status 200) (if not, you will send a 404 Not Found).

Step 16 : Create a new route /POST on "countries/:countryName" with at least "name, alpha2Code, alpha3Code, capital, region, subregion, population, denonym, nativeName, flag" as body to create and insert the new country on the JSon file.

Step 17 : Insert the new country created by the step 13 in the alphabetical order in the JSON file.

Step 18 : Create a new express server which take all the previous steps and combine them in one API.

Step 19 : Use Express Router in a separated file.

Step 20 : Develop a front-end part which consume our own API.

To do this project, you will be alone, BUT, we want you to work with your team mates, we want you to help each others. (Not by giving the solutions, but by asking questions to each others !)