Meteostat JSON API

We are going to get historical weather data from the Meteostat JSON API
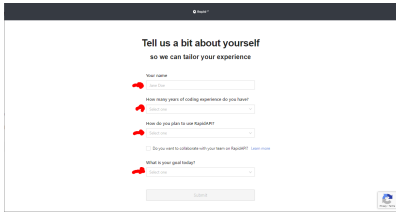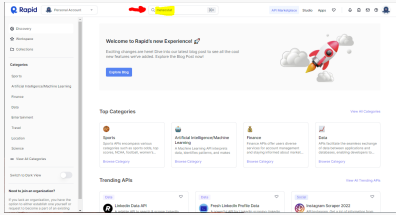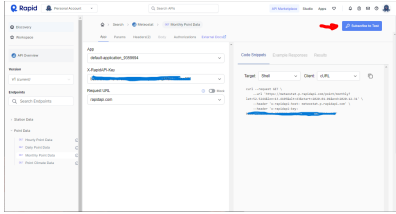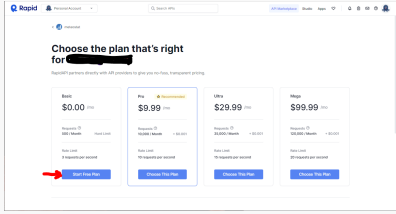
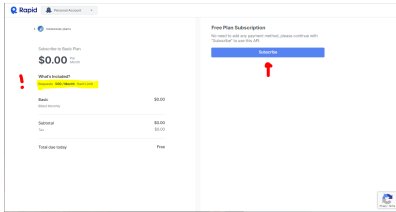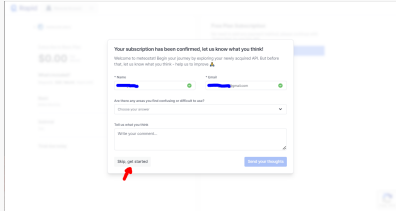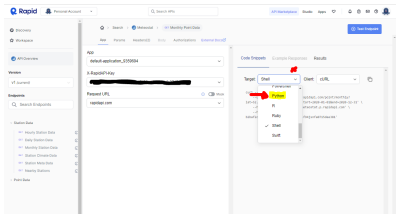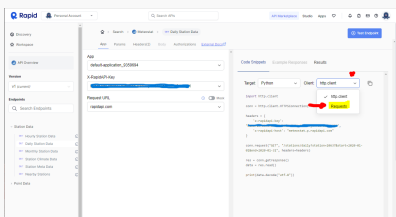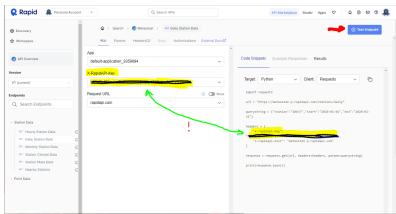Meteostat JSON API service is hosted on [RapidAPI](#), an API marketplace which allows developers to use thousands of APIs through the same platform.
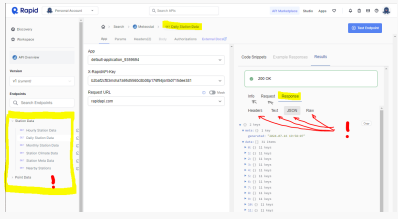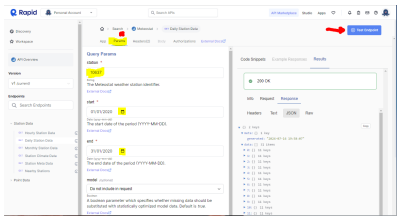
Meteostat JSON API Documentation: **https://dev.meteostat.net/api/**

Task #1:

Follow the **Sign Up** to [RapidAPI](#) instructions in order to get access to the Meteostat API.

STEP 1	Visit: https://rapidapi.com/auth/sign-up Enter a nickname, an email and a password to sign up.	
STEP 2	Check your email inbox for code Copy/Paste the code for verification.	
STEP 3	Fill out the fields.	
STEP 4	Search for '*meteostat*'. And select it.	
STEP 5	Click the **Subscribe to Test** button.	
STEP 6	Click **Start Free Plan** button under the basic plan.	

STEP 7	Note that it is for free. Note that it is limited to 500 requests per month. Click **Subscribe** button.	
STEP 8	Feel free to skip. Click **Skip, get started** button	
STEP 9	You see now an online interface API preview area. Select *Target:* **Python**	
STEP 10	Select *Client:* **Requests**	
STEP 11	Below you see now the python code. We will not need it as we are going to use a slightly different syntax. But for experiments you can check with this code for dafault syntax. However the headers are the mandatory part as it contains your API Key. Click **Test Endpoint** button (top/right) to get some data.	

STEP 12	On the right lower section we now have a Response. Expand the numbers to see the contents. Check out all those Respose parts (*Headers*, *Text*, *JSON*, *Raw*). Also see *Info* and *Request*. We just tested the **Daily Station Data** (see marked top/middle). On the left you see all other available endpoints.	
STEP 13	Switch to **Params**. Here you can adjust the period start and end also change the station id. How can we find the station we need? Check out the *Nearby Stations* endpoint.	

Another way of finding a station we look for is to enter Names or Coordinates here:

https://meteostat.net/en/

Finding a specific station id can get tricky and requires some patience and double checks. The API endpoint *Nearby Stations* finds multiple stations.

We could make our requests based on coordinates only, then it will return aggregated values from the nearby stations. However sometimes the aggregation causes missing data.

Hence we will address the exact station ids (will be given). And typically airport have their own weather stations.

Task #2

update your **.env** file with the `X-RapidAPI-Key = 'dfsd8w43roihlsdfu......whateveritis'`

Task #3:

- Check out the Meteostat JSON API documnetation: https://dev.meteostat.net/api/
- Find out what the call limits are for this API (Hint: it is not only the 500 requests per month)
  -> Consinder data retrieval needs (including testing) and how to fit within these limits.
- Check out the *Daily Data* and *Hourly Data*.
- Which parameter can we pass to the Request?
- Which data will get from the Response

Keep in mind that your API calls are limited!

When possible, separate code calling the API from other code working on the data.

How to check used API calls

To check the number of API calls so far: https://rapidapi.com/console/analytics/my-subscriptions/playground

Or on https://rapidapi.com/:

- click your Icon (Top-Right-Corner) >> Console
- Under Analytics switch from My APIs to **My Subscriptions** (My APIs is for API providers)
- Click **Analytics Playground** button

How to find meteostat interface on `rapidapi.com`

https://rapidapi.com/meteostat/api/meteostat

Or on https://rapidapi.com/:

- In the left Navigation select Workspace
- Scroll down to **My Subscriptions**
- CLick the link to meteostat