

Dataset creation task

Data extraction

You will create a dataset for an analytics company that wants to track the circulating supply of some cryptocurrency tokens. We have previously analysed a cryptocurrency token called Fluence.

The CTO (chief technology officer) of your company tells you that there is an API service called Alchemy (<http://alchemy.com>) that you can use to obtain token data. Now, the CTO would like you to datamine the Fluence token from a public data source, and analyse it.

1. Go to Coingecko and search for the Fluence token. You should be able to find the "contract address" on the Ethereum blockchain.
2. Sign up for an account on Fluence, obtain an API key, and create a new app. Since the Fluence token is on the Ethereum blockchain, make sure your app connects to the Ethereum mainnet only. When you create the app, you will obtain an API address which you use to download data.
3. The API has many **endpoints**, which can give you different types of data. Alchemy's Token API allows you to extract all transactions for Fluence's contract address. Your CTO informs you that Fluence is an ERC20 token, so you should specify in the API input that you want only ERC20 transactions (otherwise you will be downloading a lot of other data you don't need).
4. The API output is in JSON format. Put all the data rows you have downloaded into a large CSV file.
5. Import this CSV file into a suitable database. You can use any database you like, but your CTO recommends sqlite3.
6. Calculate the circulating supply in SQL (you previously did it in Pandas). Since you downloaded the full history of the coin, the circulating supply should be very close to the number reported on Coingecko. The CTO recommends sqlite3 because you can run the queries in a Python file / Jupyter notebook without installing additional software.

Notes

1. Alchemy has many tutorials and extensive documentation to help you. If you don't understand, forget about the task and try the tutorials first
2. Same for Sqlite3.
3. In Alchemy's token API, you can download up to 1000 transactions at a time. Every time you call the API, it returns a "next page" code. You need to enter this code the next time you call the API, so that it downloads the next 1000 data rows (instead of the first 1000 data rows all the time). Remember to remove duplicates from your table later!
4. Be very careful about calling API endpoints too quickly. Check how frequently you can call the API, and make sure you stay inside the limit. Usually, you will be blocked if you call the API too often.