DNEXT API - Code Example - Download a tradeMatrix

Below is a code snippet that can be used to download a tradeflow using the dnext API:

```python
import requests
import pandas as pd
from io import BytesIO
import time
import json


def _get_task_status(task_id, token):
    my_headers = {
        'Authorization': 'Bearer ' + token,
        'Content-Type': 'application/json'
    }

    success = False
    wait_time = 2

    while not success and wait_time < 10:
        time.sleep(wait_time)

        status_res = requests.get(f'https://api.dnext.io/v1.0/tasks/{task_id}', headers=my_headers)

        try:
            success = status_res.json()['status'] == 'SUCCEEDED'
        except:
            wait_time += 1
            continue

        if not success:
            wait_time += 1

    if success:
        return True
    else:
        return False


def download_tradeflow(code, token):

    my_headers = {
        'Authorization': 'Bearer ' + token,
        'Content-Type': 'application/json'
    }

    data = {
        'intraflows': {
            'strategy': 'include', # "include" or "exclude" the intraflows
        },
        'forecasts': {
            'strategy': 'include', # "include" or "exclude" the intraflows
            'list': ["string"] # Order by priority
        },
```

```
        'row': True,
        'refresh': False,
        'format': 'csv',
    }

    dl_url = f'https://api.dnext.io/v1.0/fundamentals/tradeflows/{code}/download'

    res = requests.post(dl_url, headers=my_headers, data=json.dumps(data))

    task_id = res.json()['task']['id']

    data_ready = _get_task_status(task_id, token)

    if data_ready:
        data_url = res.json()['result']['url']
        data_resp = requests.get(data_url)
        df = pd.read_csv(BytesIO(data_resp.content))

    else:
        df = None

    return df


df = download_tradeflow(<tradeflow_code>, <your_token>)
```