Problem statement

We need to build an apache airflow data processing pipeline to analyse mobile app usage for a particular user with the help of the graph database, Neo4J in our case.

Let's simplify it by breaking the entire task in two smaller parts:

1. Data collection(storing data in JSON files)
2. Data transformation(transforming JSON files into Neo4J)

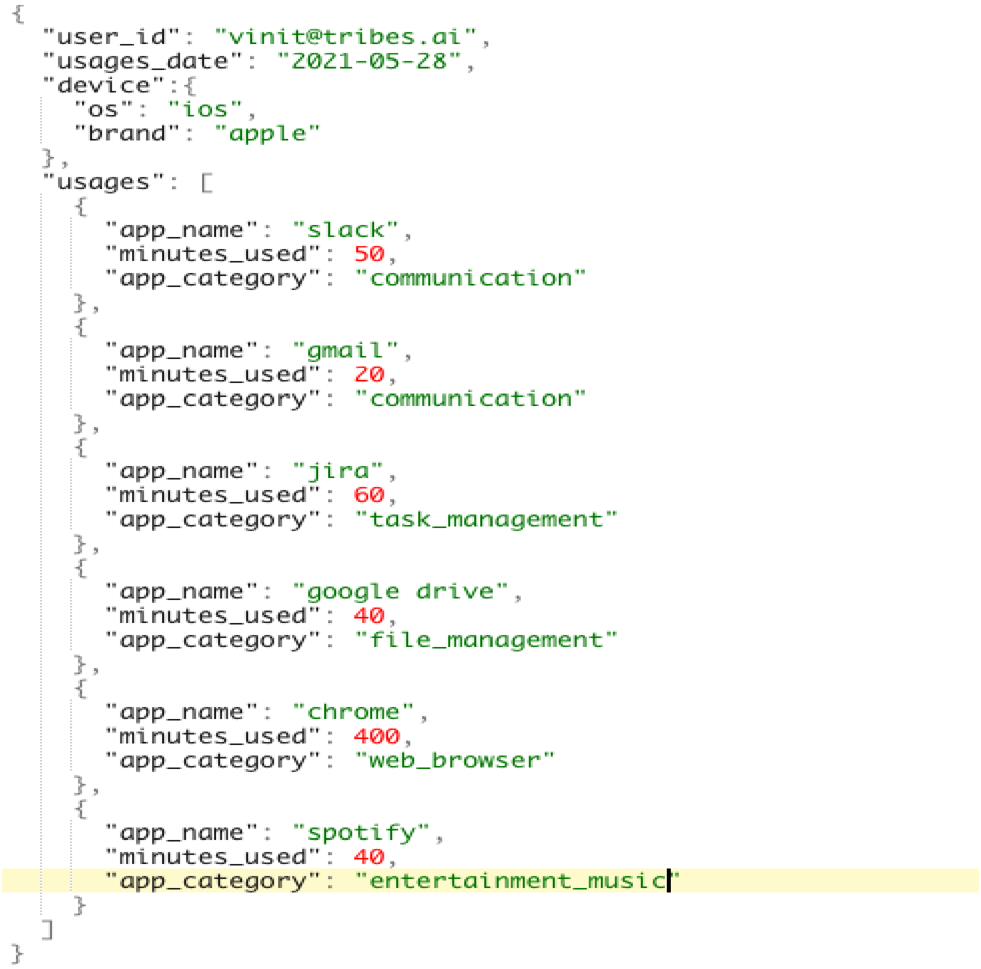
Let's get into details of both the steps of the airflow pipeline now :)

Step-1: Data collection

In the real world this data would be coming from the user's mobile phones but for the purpose of this assignment we would be mimicking this step, and we would need to generate a single JSON file for every user every day, so basically our pipeline would be running daily using airflow scheduler.

This would be the first part of the airflow pipeline you are developing as part of this assignment.

Please assume to have 5 users in our system for which we would be extracting data i.e. vinit@tribes.ai, guilermo@tribes.ai, christian@tribes.ai, elly@tribes.ai , {your_first_name}@tribes.ai.



```
{
  "user_id": "vinit@tribes.ai",
  "usages_date": "2021-05-28",
  "device":{
    "os": "ios",
    "brand": "apple"
  },
  "usages": [
    {
      "app_name": "slack",
      "minutes_used": 50,
      "app_category": "communication"
    },
    {
      "app_name": "gmail",
      "minutes_used": 20,
      "app_category": "communication"
    },
    {
      "app_name": "jira",
      "minutes_used": 60,
      "app_category": "task_management"
    },
    {
      "app_name": "google drive",
      "minutes_used": 40,
      "app_category": "file_management"
    },
    {
      "app_name": "chrome",
      "minutes_used": 400,
      "app_category": "web_browser"
    },
    {
      "app_name": "spotify",
      "minutes_used": 40,
      "app_category": "entertainment_music"
    }
  ]
}
```

Now for every user we would be producing minutes using a random value between 0 to 180 for a single day and the sum of usages of all platforms for a day for a user shouldn’t be exceeding 480 minutes(8 hours).

This step of the pipeline would be running daily and for the first time pipeline runs we should be populating data for the last 30 days.

2. Data transformation

In this step we would be loading the data generated from step-1 into a graph database(Neo4J).

In the neo4j we would be creating below nodes:

1. **Label**: User, **Properties**: {IdMaster: data[‘user_id’]}
2. **Label**: App, **Properties**: {IdMaster: data[‘app_name’], AppCategory: data[‘app_category’]}
3. **Label**: Device, **Properties**: {IdMaster: data[‘device’][‘os’]}
4. **Label**: Brand, **Properties**: {IdMaster: data[‘device’][‘brand’]}

Similarly for relationship we need to create relationship chain using above nodes:

User- [:USED] -> App - [:ON] -> Device - [:OF] -> Brand

The relationship will have following properties:
Type: USED, **Properties**: {TimeCreated: [‘Time of creation of relationship’]
TimeEvent: data[‘usages_date’]
UsageMinutes: data[‘minutes_used’]}
Type: ON, **Properties**: {TimeCreated: [‘Time of creation of relationship’]}
Type: OF, **Properties**: {TimeCreated: [‘Time of creation of relationship’]}

In this step of the pipeline all the new files populated from step-1 for all users(only the new files) should get transformed and get written in Neo4J.

Evaluation criteria

The goal of this assignment is to get a view on your hands-on "data engineering" skills. At our company, our data scientists and engineers collaborate on projects. Your main focus will be creating performant & robust data flows.
For a take-home-assignment, we cannot grant you access to our infrastructure.

We expect you to be good at:

- Adopting new technologies
- Documenting your code
- Problem solving
- Getting shit done attitude

In this exercise we expect you to demonstrate your ability to / knowledge of:

- PEP8 / Google python style guide
- Efficiently getting the job done
- Choose meaningful names for variables & functions

- Writing maintainable code (yes, you might need to document some steps)

Submission process

To submit your assignment please create a zip file, contents of which would be as below:

1. Screenshot of the airflow pipeline(a maximum of 1 screenshot)
2. Screenshots of loaded data in Neo4J(a maximum of 3 screenshots)
3. Your code with proper readme file about running it in local environment(please exclude the generated data file)

Once completed with the assignment please send it to the same email thread where the assignment is shared.

Also, we encourage you to ask questions if you have any queries using the email thread where we are sharing the assignment.