Tempus Data Engineer Challenge

For this challenge, you will develop a simple Apache Airflow data pipeline that fetches data from News API, transforms the data into a tabular structure, and stores the transformed data on Amazon S3.

Apache Airflow

From the Apache Airflow documentation:

> Airflow is a platform to programmatically author, schedule and monitor workflows.

> Use airflow to author workflows as directed acyclic graphs (DAGs) of tasks. The airflow scheduler executes your tasks on an array of workers while following the specified dependencies. Rich command line utilities make performing complex surgeries on DAGs a snap. The rich user interface makes it easy to visualize pipelines running in production, monitor progress, and troubleshoot issues when needed.

> When workflows are defined as code, they become more maintainable, versionable, testable, and collaborative.

In order to facilitate the use of Airflow, we have included a Dockerfile and a docker-compose.yml that can be used to set up a local airflow development environment. **Make sure to have Docker and Docker Compose installed.**

From the root folder, you can execute the following command to run airflow:

```
docker-compose up --build
```

The Airflow UI/Admin Console should now be visible on http://localhost:8080.

DAGs

In order to build the data pipeline, it will be necessary to create a DAG. We have provided an example DAG, `dags/sample_dag.py`, that can be used as a reference. Further documentation can be found in the airflow tutorial and the airflow concepts pages.

To load a new DAG into airflow, simply create a new Python file in the `dags` folder that contains an airflow DAG object.

Python Packages

To install additional Python packages (boto3, pandas, requests, etc.), add them to `requirements.txt`.

Related Articles/Tutorials

- https://airflow.apache.org/index.html
- https://medium.com/@dustinstansbury/understanding-apache-airflows-key-concepts-a96efed52b1a
- https://speakerdeck.com/artwr/apache-airflow-dataengconf-sf-2017-workshop

- https://github.com/hgrif/airflow-tutorial

News API

A simple REST API that can be used to retrieve breaking headlines and search for articles. **A free News API account is required to obtain an API key.**

Route	Description
/v2/top-headlines	Returns live top and breaking headlines for a country, specific category in a country, single source, or multiple sources.
/v2/sources	Returns the subset of news publishers that top headlines are available from.

Amazon S3

A simple cloud storage service run by Amazon Web Services (AWS). **An AWS account is needed to use AWS S3. Furthermore, AWS has a free tier that can be used for this challenge.**

Amazon provides a Python SDK (**boto**), that provides an easy to use API for interacting with AWS S3.

Requirements

- □ Use Airflow to construct data pipeline
- □ Data pipeline must be scheduled to run once a day
- □ Data pipeline will:
  - □ Retrieve all English news sources
  - □ For each news source, retrieve the top headlines
    - □ Top headlines must be flattened into a CSV file. CSV Filename: `<pipeline_execution_date>_top_headlines.csv`
    - □ Result CSV must be uploaded to the following s3 location `<s3_bucket>/<source_name>`
- □ Bonus: Build a separate pipeline that uses the following keywords instead of English news sources: Tempus Labs, Eric Lefkofsky, Cancer, Immunotherapy

Rules of engagement

- We suggest that you establish a four hour timebox to complete the challenge.
- The solution must perform a Python transformation of the data; feel free to add any open-source libraries you wish and add additional output files.
- Please document changes required to make the solution resilient to failure by taking the following actions:
  - add developer-friendly requirements to functions
  - add comments in the main function that list failures that the solution should be designed to handle
- Please deliver your Python code via repo or zip ahead of the meeting.