Streaming Data Project

For Skills Bootcamp: Software Developer/Coding Skills Graduates - Data Engineering

Context

From time to time, Northcoders may want to search media for relevant content. Relevant content can be saved for use by our marketing and careers teams. Unnecessary articles would be discarded.

High-level desired outcome

As a proof of concept, you are asked to create an application to retrieve articles from the Guardian API and publish it to a message broker so that it can be consumed and analysed by other applications.

The tool will accept a search term (e.g. "machine learning"), an optional "date_from" field, and a reference to a message broker. It will use the search terms to search for articles in the Guardian API. It will then post details of up to ten hits to the message broker.

For example, given the inputs:

- "machine learning"
- "date_from=2023-01-01"
- "Guardian_content" it will retrieve all content returned by the API and post up to the ten most recent items in JSON format onto the message broker with the ID "guardian_content".

Assumptions and Prerequisites

1. For this proof of concept, you will use only the free tier API key provided by the Guardian, and abide by the rate limits associated with it.
2. The library must be capable of being used in applications deployed in AWS.

Minimum viable product

The tool will publish data to the message broker in the following JSON format:

```
{
    "webPublicationDate": "2023-11-21T11:11:31Z",
    "webTitle": "Who said what: using machine learning to correctly attribute quotes",
    "webUrl": "https://www.theguardian.com/info/2023/nov/21/who-said-what-using-machine
}
```


These fields are the minimum required. Others may be added at your discretion.

Non-functional requirements

- The tool should be written in Python, be unit tested, PEP-8 compliant, and tested for security vulnerabilities.
- The code should include documentation.
- No credentials are to be recorded in the code.
- The complete size of the module should not exceed the memory limits for Python Lambda dependencies

Performance criteria

The tool is not expected to handle more than 50 requests per day to the API.

Data should not be persisted in the message broker longer than three days.

Possible extensions

It would be helpful to include a field called (for example) "content_preview" in the message that displays the first few lines of the article content, perhaps the first 1000 characters or so.

Non-binding tech suggestions

It is expected that the message broker employed will be AWS Kinesis. If you choose to use an alternative such as Kafka or RabbitMQ, it will have to be accommodated within the AWS Free Tier

This application is intended to be deployed as a component in a data platform. However, for demonstration purposes, you may want to be able to invoke it from your local command line.

Due date

To be advised, but not later than four weeks from commencement.