Description:

Raw logging data is continuously received and saved to a folder called **raw_logs** using plain text file format. Each subfolder is named after a job generating the logs. Each log file contains 3 lines of information:

- Version line (*v1*)
- Metadata line *{"batchWatermarkMs":0,"batchTimestampMs":1626616200136,...}*
- Commit information for each topic and its partition line in the following format: *“topicName” : {“partitionNumber” : “readOffset”}*

The list of topics of interest is saved in a file named ***topics***.

The number of partitions may change as topics can be repartitioned. For example, a topic named “developer.exception-handling” may be partitioned into 10 partitions, but after some time it may be re-partitioned into 20 partitions (this change would only affect newly generated log entries). The topics cannot be partitioned into more than 50 partitions. It is possible that some of the topics are partitioned into a single partition.

There is a need to visualize the progress of the jobs generating these logs - it should be possible to check the status of a particular job at any time period of choice.

Task:

Write a Spark job performing appropriate ETL operations and saving the transformed data into a single output file using a tabular file format of your choice. The contents of the output file should look something like this:

datetimeUtc	jobName	topicName	topicPartition	currentOffset
1999-07-19 05:30:23	some-job-1	some-topic-1	0	45
1999-07-19 05:30:23	some-job-1	some-topic-1	1	75
1999-07-19 05:30:23	some-job-1	some-topic-1	2	46
1999-07-19 05:30:23	some-job-1	some-topic-1	3	89
1999-07-19 05:30:23	some-job-1	some-topic-1	4	72
1999-07-19 05:30:23	some-job-2	some-topic-1	0	145
1999-07-19 05:30:23	some-job-2	some-topic-1	1	175
1999-07-19 05:30:23	some-job-2	some-topic-1	2	146
1999-07-19 05:30:23	some-job-2	some-topic-1	3	189
1999-07-19 05:30:23	some-job-2	some-topic-1	4	172
1999-07-19 05:45:23	some-job-1	...	...	...
...	...	...	...	...

Expectations:

- *pyspark* should be used for this task
- The code should be readable - it should be clear what the code does without running it
- Only native *python* libraries can be used (excluding *pyspark* itself)

Bonus points if you:

- Choose a *git* flow of your preference and use it to commit your work
- Write automated tests to test your code
- Format your code according to the *PEP-8* requirements