INSTRUCTIONS

The submission has 2 python files, config-parameters.py and Spark-With-Py.py

To be able to use maximum available resources on cluster to run Spark-With-Py.py we need to pass a few sparkContext configuration parameters and number of files to read in each iteration while submitting the job. config-parameters.py computes those parameters.

1. config-parameters.py and Spark-With-Py.py can be run only via command line
2. Logging is enabled only for Spark-With-Py.py and default logging level is ERROR
3. Required logging level for log file to capture time and other information is INFO. Nothing has been logged in any other logging mode.
4. Memory profiling is not done as that slows down execution time and is required only when program faces some issue that needs to be debugged.
5. To run config-parameters.py via command line, use the following:
    a. python config-parameters.py <number of nodes in cluster> <number of cores in each node> <per node memory in GB>
    b. For example: python config-parameters.py 6 16 64
6. To run Spark-With-Py.py via command line, use the following:
    a. spark-submit --master= <cluster URL> --num-executors <number of executors requested> --executor-cores <number of cores with each executor> --executor-memory <memory of each executor like 1g, 10g> <python file> <Data Folder with no slash at the end> <number of files to read in each iteration, preferably in multiples of 60>
    b. This will create empty log file
7. To be able to log into log file you need to provide logging level by adding the following to the above command line as the last argument:
    a. --log=<level of logging>
    b. e.g. --log=info
8. Level of logging has to be mentioned without quotes and levels of logging are debug, info, warning, error and critical. If incorrect logging level is provided, program will let you know and exit.