1)

```
#!/usr/bin/env python


import pyspark
import sys

if len(sys.argv) != 3:
  raise Exception("Exactly 2 arguments are required: <inputUri> <outputUri>")

inputUri=sys.argv[1]
outputUri=sys.argv[2]

sc = pyspark.SparkContext()

lines = sc.textFile(sys.argv[1])
temp = lines.flatMap(lambda line: [(line.split())[1]]).filter(lambda word: word[0].isdigit())
words = temp.map(lambda word: '-'.join([str((6*(int(word[0:2])/6))),str(((6*(int(word[0:2])/6)))+int(6))]))
wordCounts = words.map(lambda word: (word, 1)).reduceByKey(lambda count1, count2: count1 + count2)
wordCounts.coalesce(1).saveAsTextFile(sys.argv[2])
```

The above is the code used for the assignment. The python file for the same is attached with the submission zip file.

The output obtained by running the Dataproc job is:

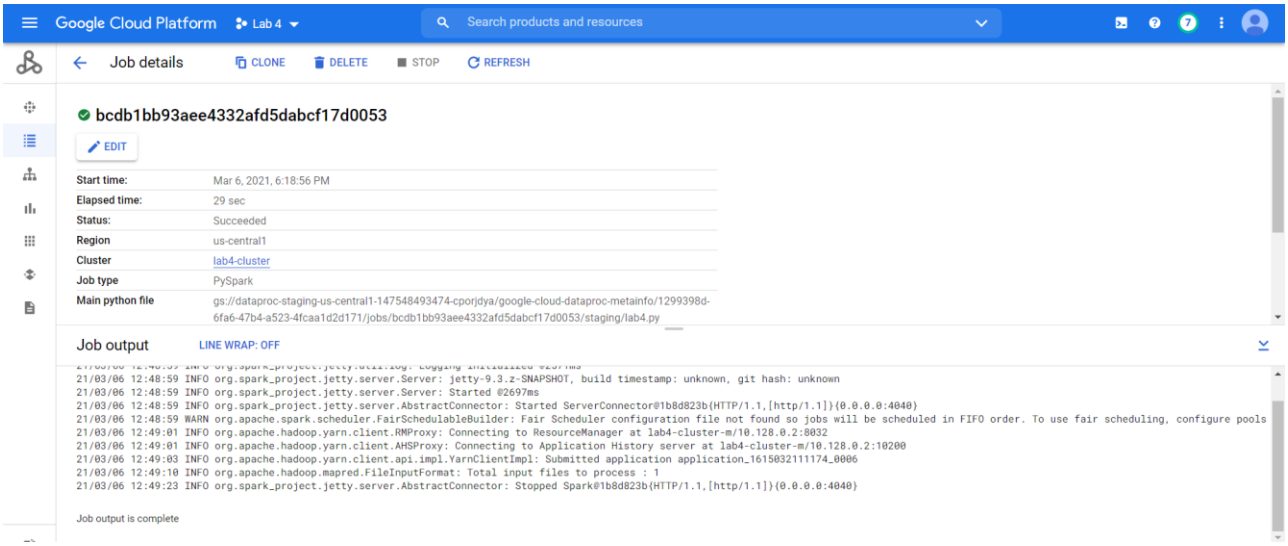
('0-6', 501103)

('12-18', 499906)

('6-12', 499409)

('18-24', 499583)

Cluster name: lab4-cluster **Region:** us-central1 **Output path:** gs://naveenrd_lab4/output/output.txt



2)

a) **HDFS (Hadoop Distributed File System):** HDFS is used for storage permissions in Hadoop cluster. It is designed to work on a distributed file system design. HDFS is capable of handling larger size data with high volume velocity and variety makes Hadoop work more efficient and reliable with easy access to all its components. HDFS stores the data in the form of the block where the size of each data block is 128MB in size which is configurable means you can change it according to your requirement in *hdfs-site.xml* file in your Hadoop directory. HDFS provides high availability and fault tolerance. In HDFS data is stored in distributed format across various nodes. It provides scalability to scaleup or scale down nodes as per requirement. Using HDFS there is no fear of data loss because it provides replication. When a node fails the data is transferred across the nodes to restore.

b) **Hive:** It is a data warehouse and ETL (Extraction, Transform & Load) tool which provides an SQL-like interface between the user and the HDFS. It facilitates reading, writing and handling wide datasets that are stored in distributed storage and queried by SQL syntax. Hive is built to manage and query only structured data which is residing under tables.

c) **Pig:** Pig is a high-level platform or tool which is used to process the large datasets. It provides a high-level of abstraction for processing over the MapReduce. It provides a high-level scripting language, known as *Pig Latin* which is used to develop the data analysis codes. First, to process the data which is stored in the HDFS, the programmers will write the scripts using the Pig Latin Language. Internally *Pig Engine* (a component of Apache Pig) converted all these scripts into a specific map and reduce task. But these are not visible to the programmers in order to provide a high-level of abstraction. Pig Latin and Pig Engine are the two main components of the Apache Pig tool. The result of Pig always stored in the HDFS.

d) **YARN (Yet Another Resource Negotiator):** YARN architecture separates resource management layer form the processing layer. YARN also allows different data processing engines like graph processing, interactive processing, stream processing as well as batch processing to run and process data stored in HDFS (Hadoop Distributed File System) thus making the system much more efficient. Through its various components, it can dynamically allocate various resources and schedule the application processing. For large volume data processing, it is quite necessary to manage the available resources properly so that every application can leverage them.