CS4830 Big Data Laboratory

Jan – May 2021

Lab 3 - Assignment

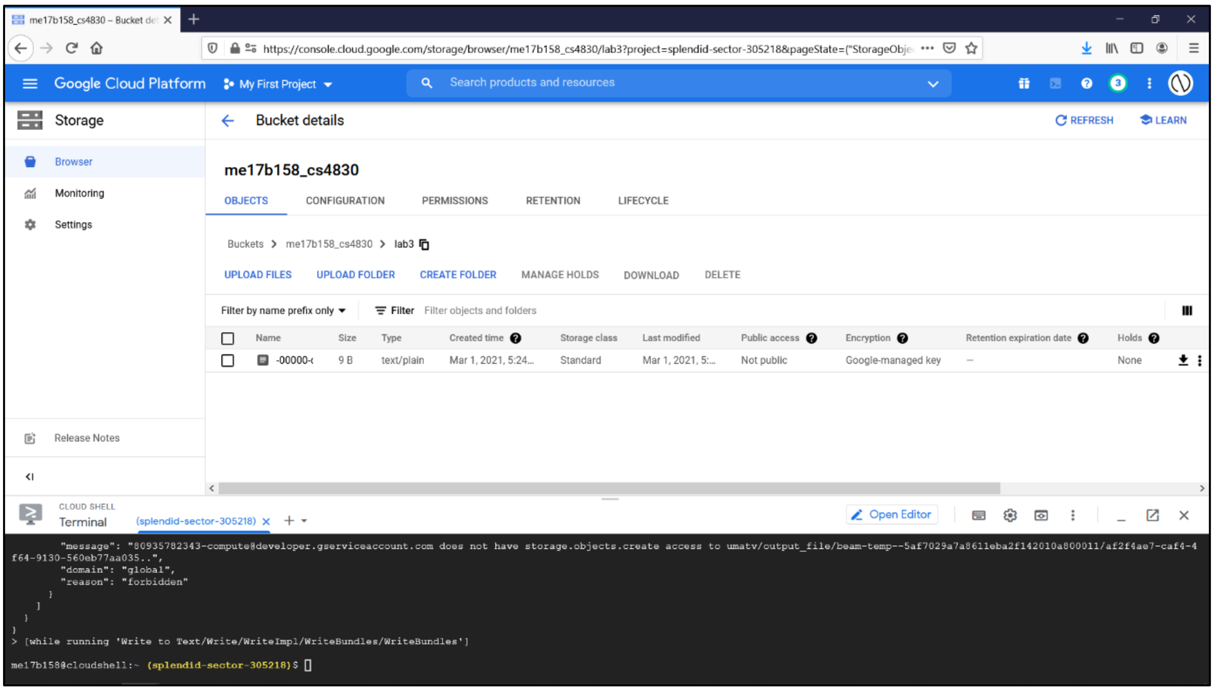
ME17B158 - Omkar Nath

1. Write a Python code to count lines of the file that is placed in the IITMBD bucket (gs://iitmbd/out.txt) using Dataflow and provide the screenshot of the file that is generated in your bucket. [2]

Attached Code: **“line_count.py”**

Generated File: **“Line_Count.txt”**

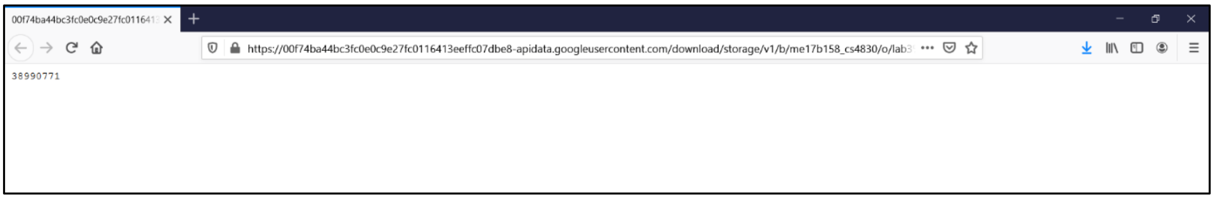
Screenshots of File:



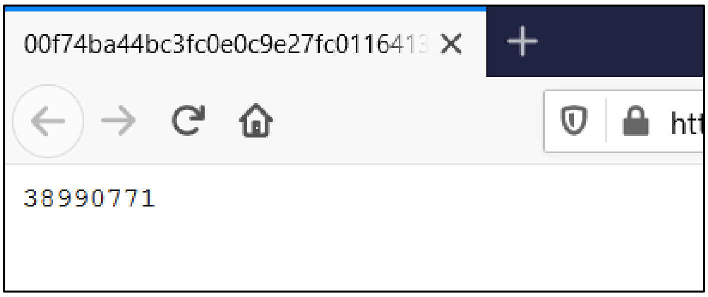
□	Name	Size	Type	Created time	Storage class	Last modified	Public access	Encryption	Retention expiration date	Holds
□	-00000-(	9 B	text/plain	Mar 1, 2021, 5:24...	Standard	Mar 1, 2021, 5:...	Not public	Google-managed key	–	None

```
        "message": "80935782343-compute@developer.gserviceaccount.com does not have storage.objects.create access to umatv/output_file/beam-temp--5af7029a7a8611eba2f142010a800011/af2f4ae7-caf4-4
f64-9130-560eb77aa035..",
        "domain": "global",
        "reason": "forbidden"
      }
    ]
  }
}
> [while running 'Write to Text/Write/WriteImpl/WriteBundles/WriteBundles']

me17b158@cloudshell:~ (splendid-sector-305218)$ 
```


00f74ba44bc3fc0e0c9e27fc011641
https://00f74ba44bc3fc0e0c9e27fc0116413eeffc07dbe8-apidata.googleusercontent.com/download/storage/v1/b/me17b158_cs4830/o/lab3

```
38990771
```



```
38990771
```

Python Code:

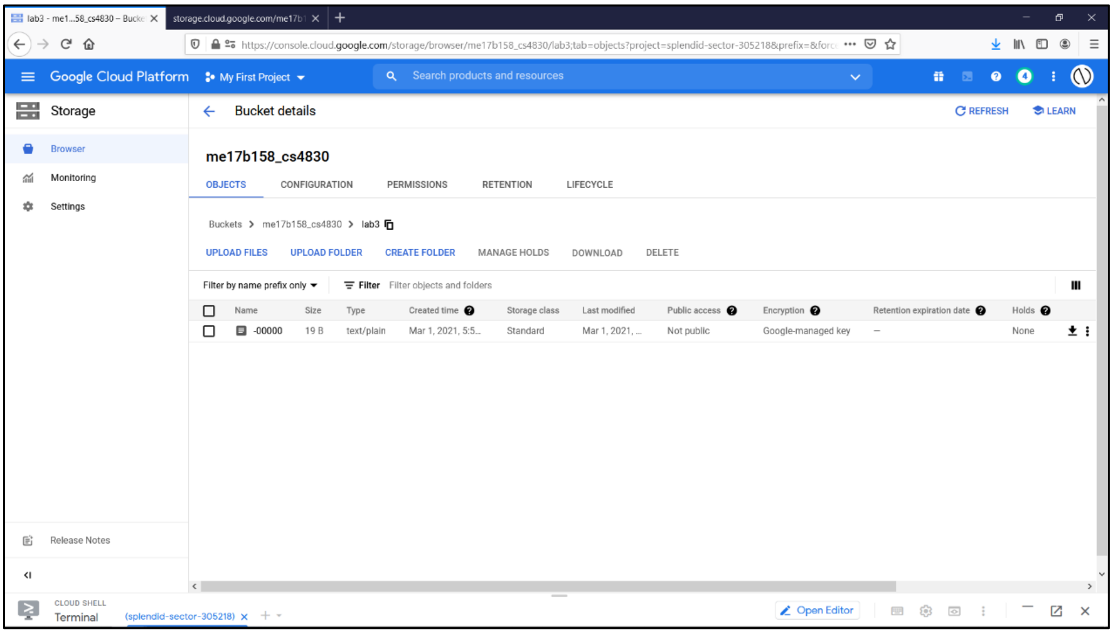
```
import apache_beam as beam
from apache_beam.io import ReadFromText
from apache_beam.io import WriteToText
from apache_beam.options.pipeline_options import PipelineOptions
from apache_beam.options.pipeline_options import GoogleCloudOptions
from apache_beam.options.pipeline_options import StandardOptions
options = PipelineOptions()
google_cloud_options = options .view_as(GoogleCloudOptions)
google_cloud_options.project = 'splendid-sector-305218'
google_cloud_options.region = "us-central1"
google_cloud_options.job_name = 'lab3q1'
google_cloud_options.temp_location = "gs://me17b158_cs4830/tmp"
options.view_as(StandardOptions).runner = 'DataflowRunner'
p = beam.Pipeline( options = options )
lines = p | 'Read' >> beam.io.ReadFromText( 'gs://iitmbd/out.txt' ) |'counting lines' >>
beam.combiners.Count.Globally(sum) | 'Write' >>
beam.io.WriteToText('gs://me17b158_cs4830/lab3/')
result = p.run()
```

2. Write a Python code to get the average number of words in a line of the file that is placed in the IITMBD bucket (gs://iitmbd/out.txt) using Dataflow and provide the screenshot of the file that is generated in your bucket. [4]

Attached Code: **"avg_words.py"**

Generated File: **"Average_Words.txt"**

Screenshots of File:



https://console.cloud.google.com/storage/browser/me17b158_cs4830/lab3;tab=objects?project=splendid-sector-305218&prefix=&forc

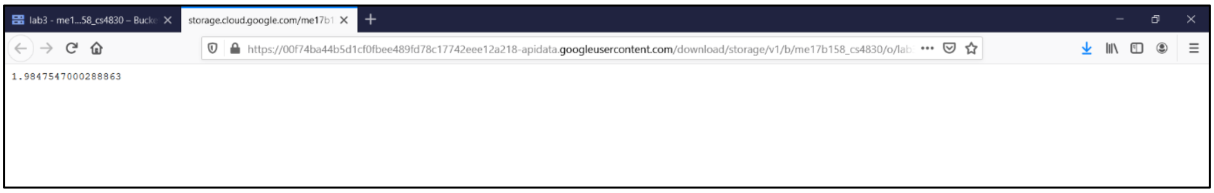
Bucket details

me17b158_cs4830

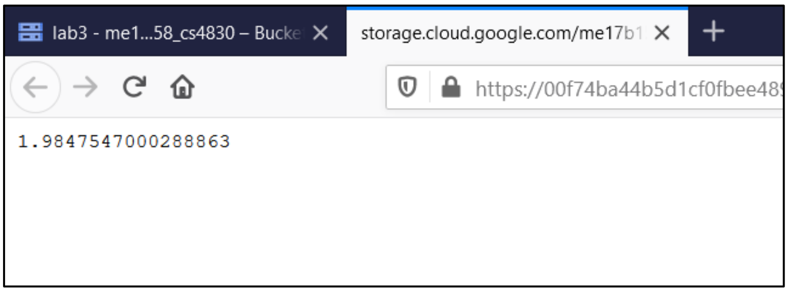
Buckets > me17b158_cs4830 > lab3

Filter by name prefix only

	Name	Size	Type	Created time	Storage class	Last modified	Public access	Encryption	Retention expiration date	Holds
□	-00000	19 B	text/plain	Mar 1, 2021, 5:5...	Standard	Mar 1, 2021, ...	Not public	Google-managed key	–	None


lab3 - me1...58_cs4830 – Bucke
storage.cloud.google.com/me17b1
https://00f74ba44b5d1cf0fbee489fd78c17742eee12a218-apidata.googleusercontent.com/download/storage/v1/b/me17b158_cs4830/o/lab

```
1.9847547000288863
```


lab3 - me1...58_cs4830 – Bucke
storage.cloud.google.com/me17b1
https://00f74ba44b5d1cf0fbee48

```
1.9847547000288863
```

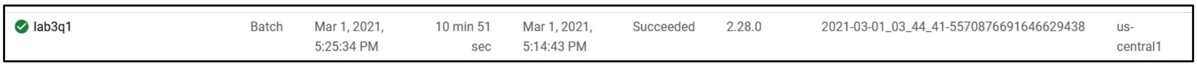
Python Code:

```
import re
import apache_beam as beam
from apache_beam.io import ReadFromText
from apache_beam.io import WriteToText
from apache_beam.options.pipeline_options import PipelineOptions
from apache_beam.options.pipeline_options import GoogleCloudOptions
from apache_beam.options.pipeline_options import StandardOptions
options = PipelineOptions()
google_cloud_options = options .view_as(GoogleCloudOptions)
google_cloud_options.project = 'splendid-sector-305218'
google_cloud_options.region = "us-central1"
google_cloud_options.job_name = 'lab3q2'
google_cloud_options.temp_location = "gs://me17b158_cs4830/tmp"
options.view_as(StandardOptions).runner = "DataflowRunner"
with beam.Pipeline(options=options) as p:
    avgwords = p | 'Read' >> ReadFromText ('gs://iitmbd/out.txt') | 'Counting words per line' >>
beam.Map (lambda x: len(re.split('[\s,;!]+',x))) | 'Taking mean' >>
beam.CombineGlobally((beam.transforms.combiners.MeanCombineFn()) | 'Write to Text' >>
WriteToText('gs://me17b158_cs4830/lab3/')
```

3. Provide the screenshot for the execution graph created by Dataflow in the background for the pipeline object created for the questions 1 and 2. [2]

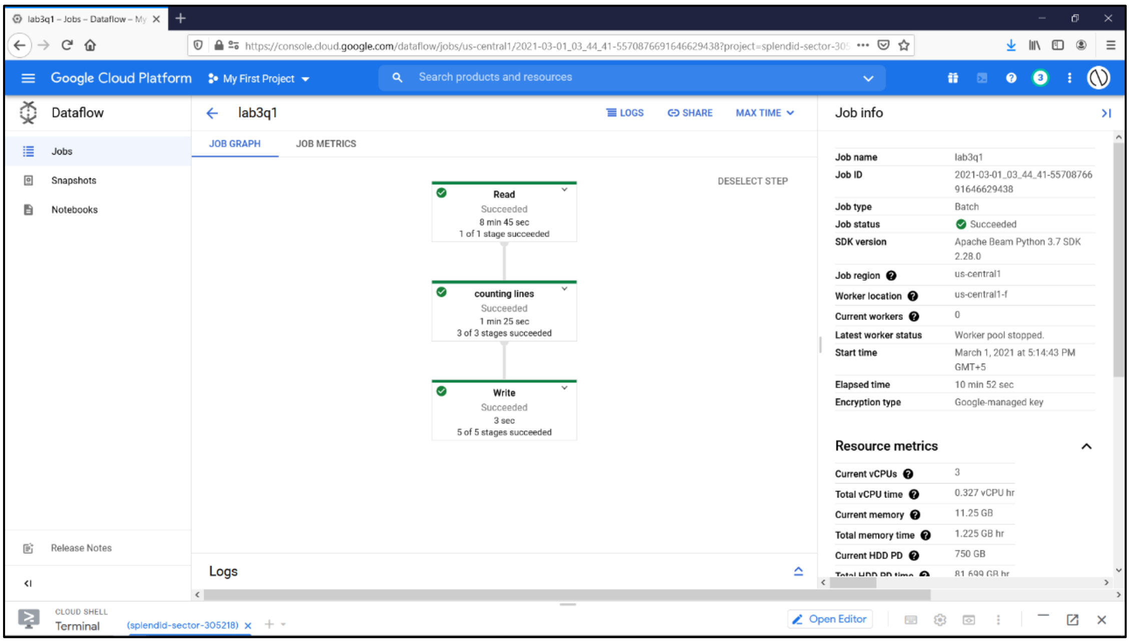
Execution Graph for Question 1:

Dataflow:



✓ lab3q1	Batch	Mar 1, 2021, 5:25:34 PM	10 min 51 sec	Mar 1, 2021, 5:14:43 PM	Succeeded	2.28.0	2021-03-01_03_44_41-5570876691646629438	us-central1

Overview:



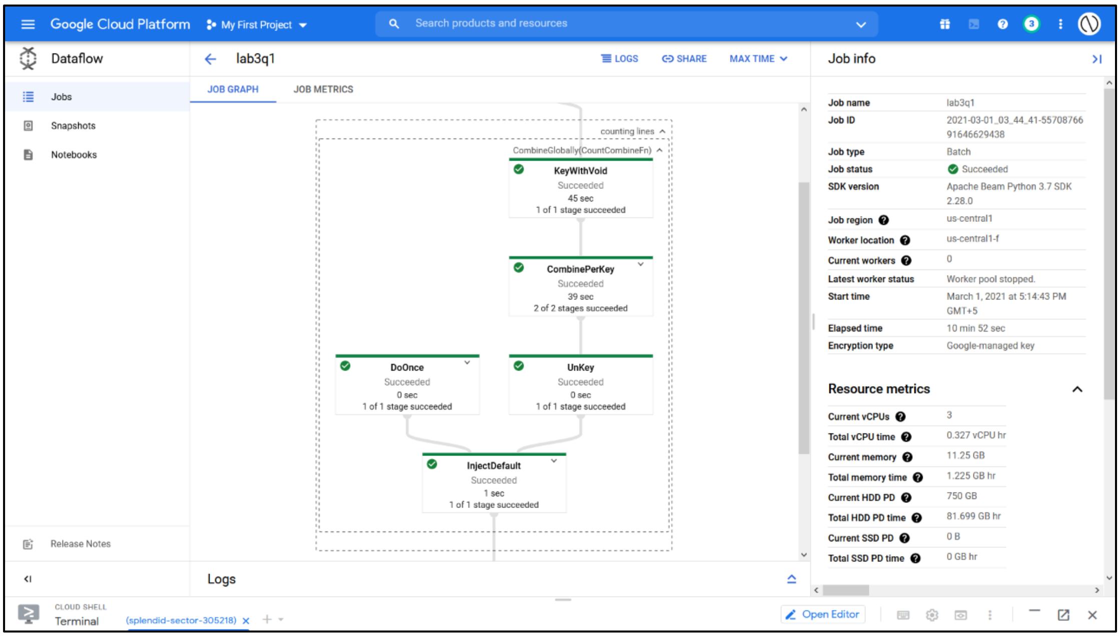
Job info

Job name	lab3q1
Job ID	2021-03-01_03_44_41-5570876691646629438
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:14:43 PM GMT+5
Elapsed time	10 min 52 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	3
Total vCPU time	0.327 vCPU hr
Current memory	11.25 GB
Total memory time	1.225 GB hr
Current HDD PD	750 GB
Total HDD PD time	81.699 GB hr

Counting Lines - Expanded:



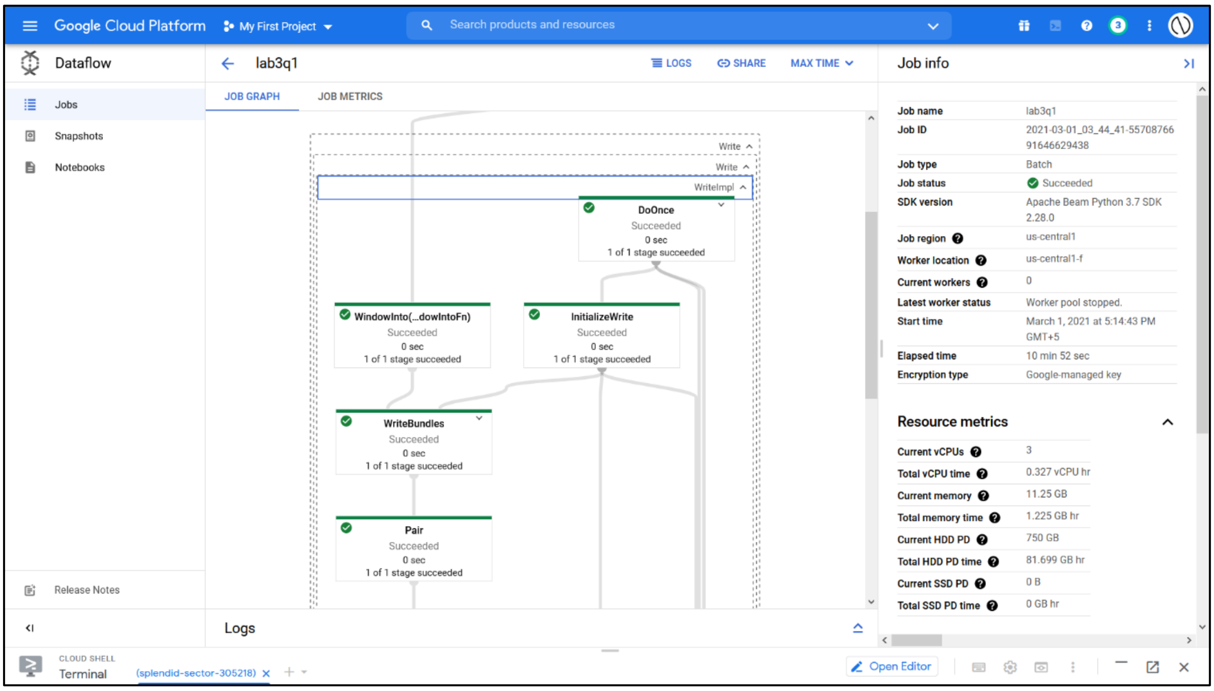
Job info

Job name	lab3q1
Job ID	2021-03-01_03_44_41-5570876691646629438
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:14:43 PM GMT+5
Elapsed time	10 min 52 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	3
Total vCPU time	0.327 vCPU hr
Current memory	11.25 GB
Total memory time	1.225 GB hr
Current HDD PD	750 GB
Total HDD PD time	81.699 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr

Write – Expanded:

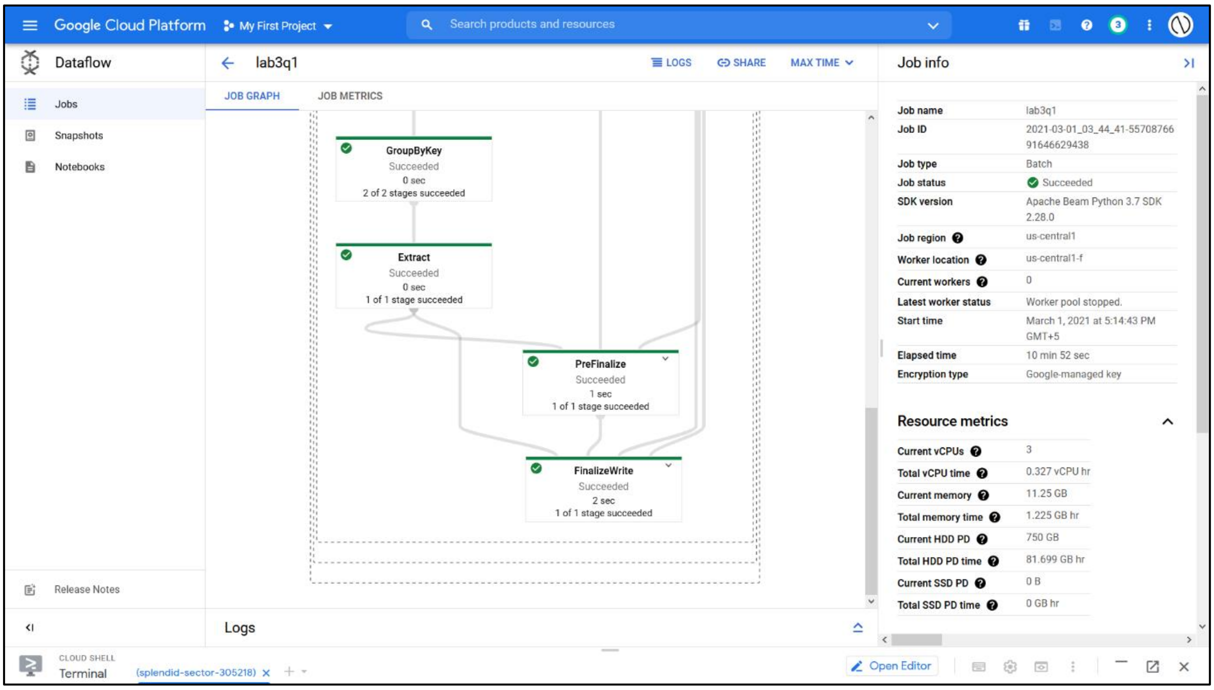


Job info

Job name	lab3q1
Job ID	2021-03-01_03_44_41-5570876691646629438
Job type	Batch
Job status	Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:14:43 PM GMT+5
Elapsed time	10 min 52 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	3
Total vCPU time	0.327 vCPU hr
Current memory	11.25 GB
Total memory time	1.225 GB hr
Current HDD PD	750 GB
Total HDD PD time	81.699 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr



Job info

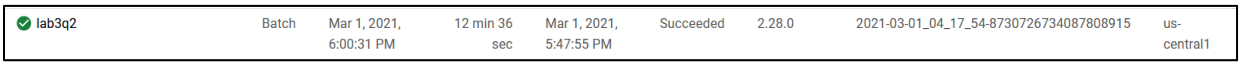
Job name	lab3q1
Job ID	2021-03-01_03_44_41-5570876691646629438
Job type	Batch
Job status	Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:14:43 PM GMT+5
Elapsed time	10 min 52 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	3
Total vCPU time	0.327 vCPU hr
Current memory	11.25 GB
Total memory time	1.225 GB hr
Current HDD PD	750 GB
Total HDD PD time	81.699 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr

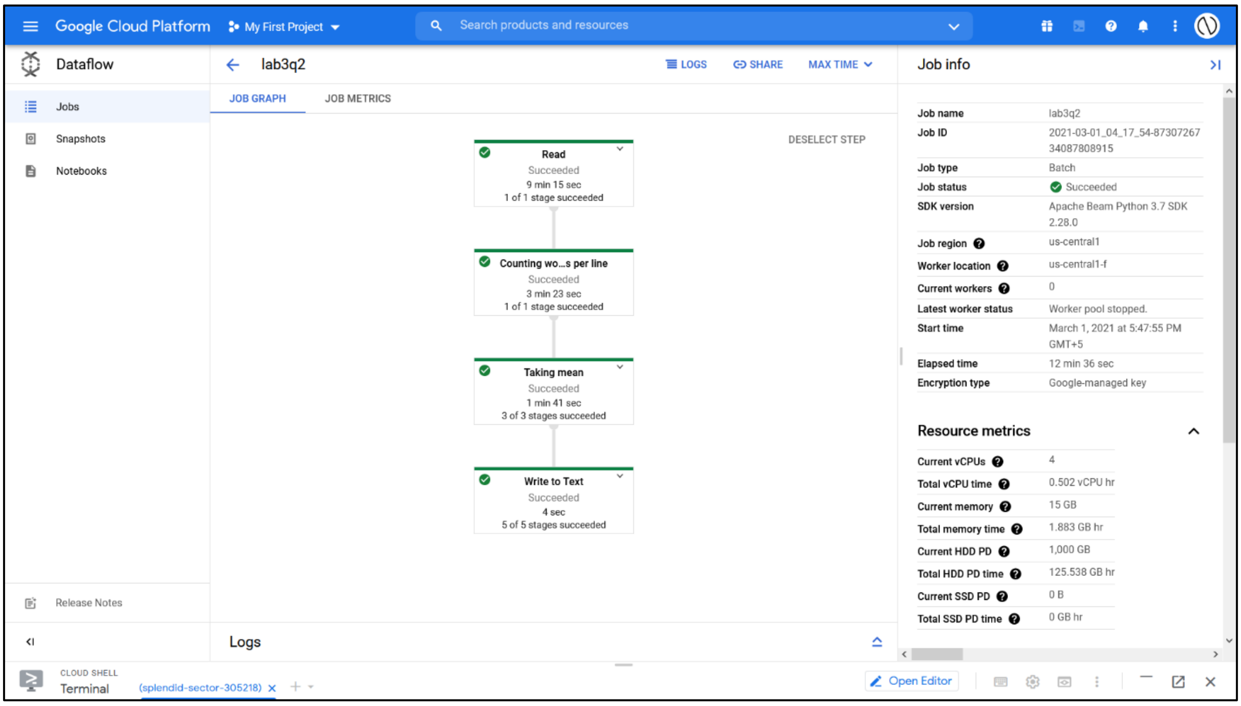
Execution Graph for Question 2:

Dataflow:



✓ lab3q2	Batch	Mar 1, 2021, 6:00:31 PM	12 min 36 sec	Mar 1, 2021, 5:47:55 PM	Succeeded	2.28.0	2021-03-01_04_17_54-8730726734087808915	us-central1

Overview:



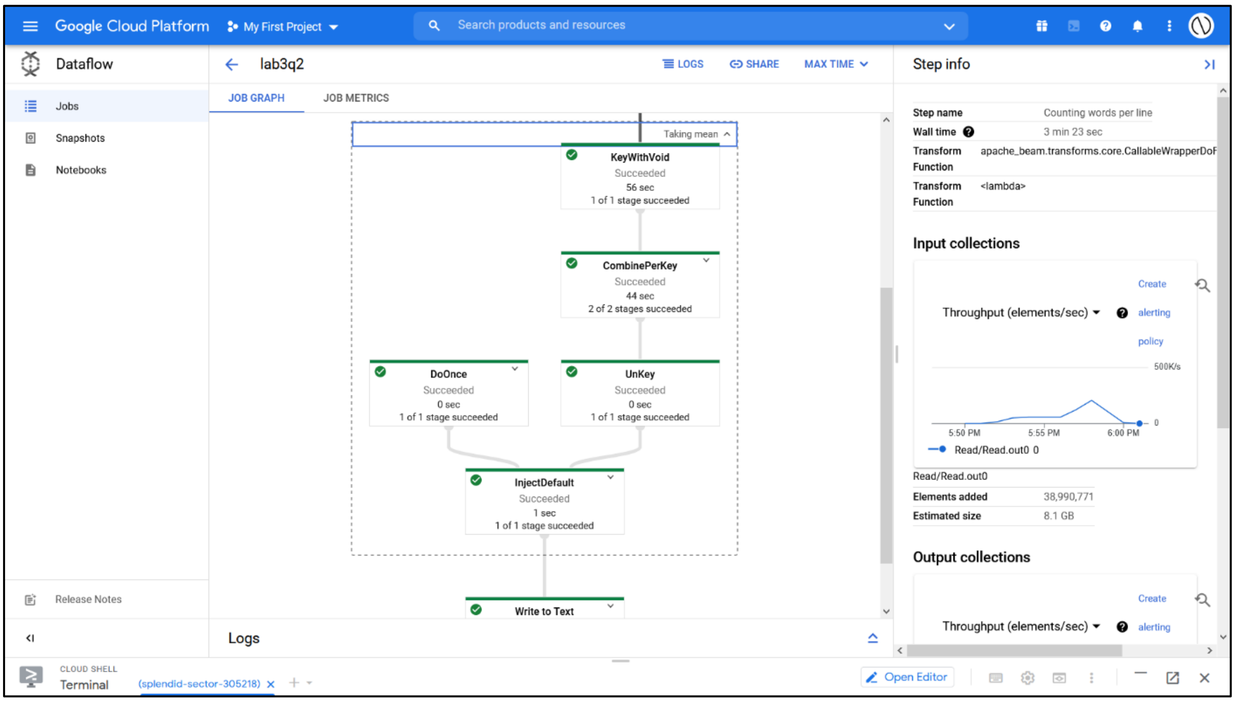
Job info

Job name	lab3q2
Job ID	2021-03-01_04_17_54-8730726734087808915
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:47:55 PM GMT+5
Elapsed time	12 min 36 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	4
Total vCPU time	0.502 vCPU hr
Current memory	15 GB
Total memory time	1.883 GB hr
Current HDD PD	1,000 GB
Total HDD PD time	125.538 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr

Taking Mean:



Step info

Step name	Counting words per line
Wall time	3 min 23 sec
Transform Function	apache_beam.transforms.core.CallableWrapperDoF
Transform Function	<lambda>

Input collections

Throughput (elements/sec)

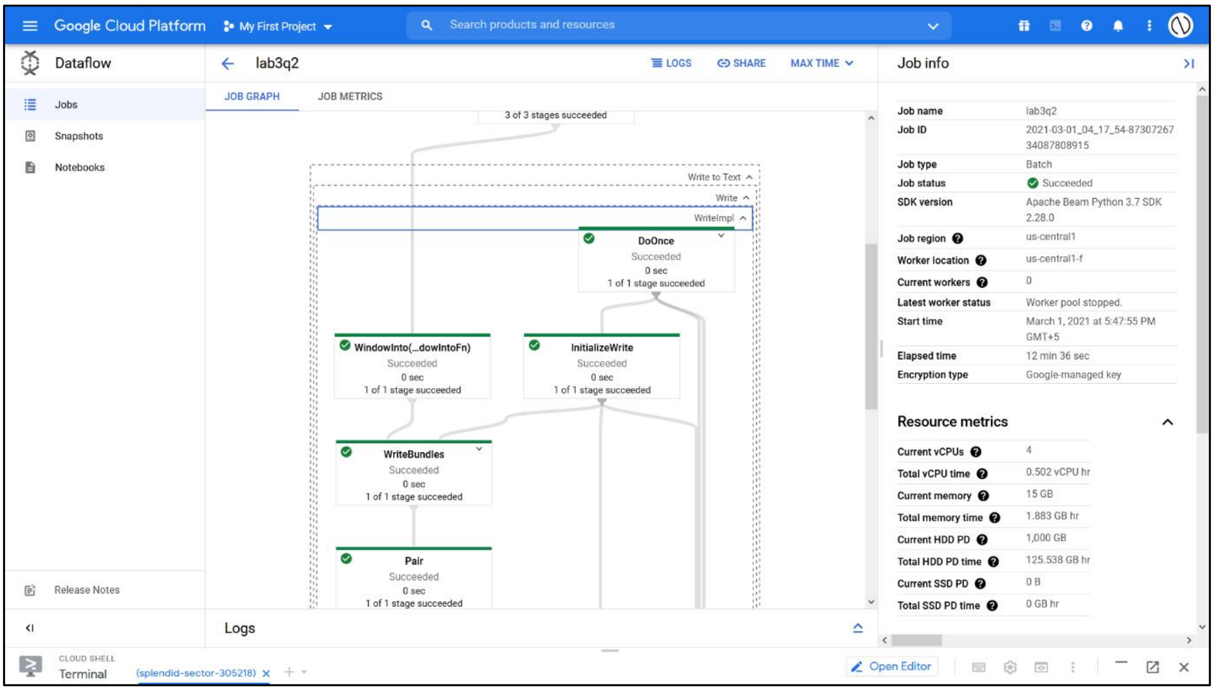
Read/Read.out0

Elements added	38,990,771
Estimated size	8.1 GB

Output collections

Throughput (elements/sec)

Write to text:

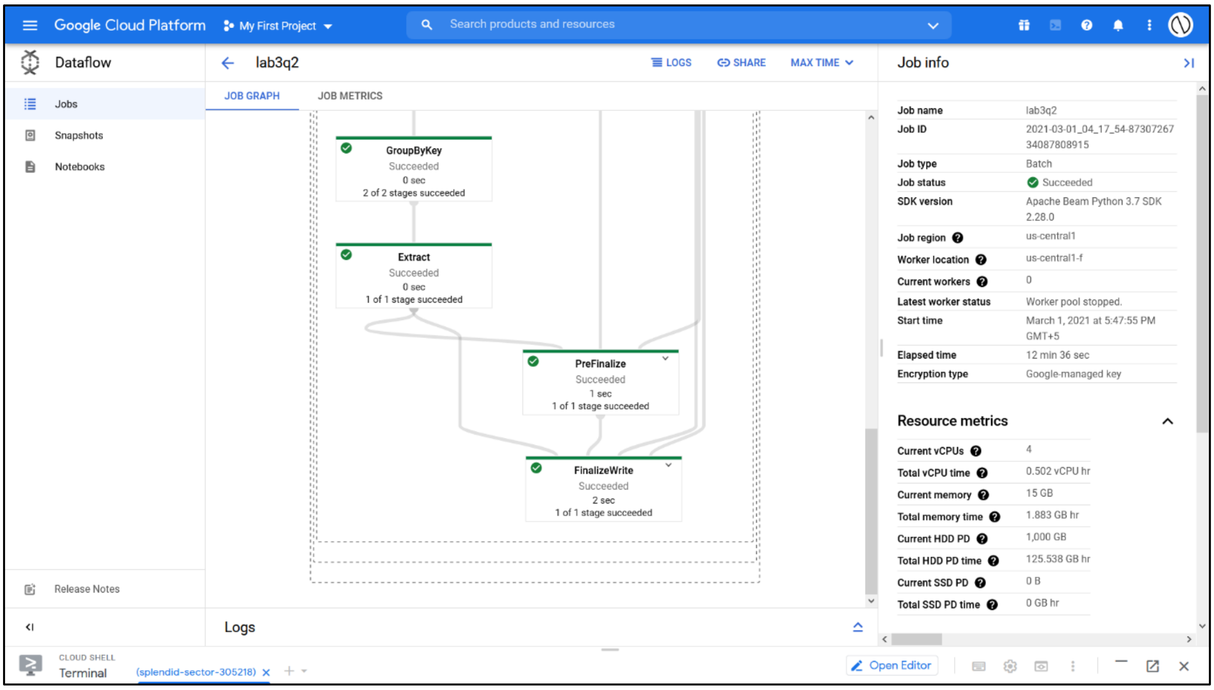


Job info

Job name	lab3q2
Job ID	2021-03-01_04_17_54-87307267 34087808915
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:47:55 PM GMT+5
Elapsed time	12 min 36 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	4
Total vCPU time	0.502 vCPU hr
Current memory	15 GB
Total memory time	1.883 GB hr
Current HDD PD	1,000 GB
Total HDD PD time	125.538 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr



Job info

Job name	lab3q2
Job ID	2021-03-01_04_17_54-87307267 34087808915
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 5:47:55 PM GMT+5
Elapsed time	12 min 36 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	4
Total vCPU time	0.502 vCPU hr
Current memory	15 GB
Total memory time	1.883 GB hr
Current HDD PD	1,000 GB
Total HDD PD time	125.538 GB hr
Current SSD PD	0 B
Total SSD PD time	0 GB hr

4. Explain the pipeline used in the first two questions. What issues did you face while trying to make the code work for the first two questions and how did you resolve them? [2]

Usage of Google Cloud:

The apache beam pipeline is used in the Google cloud storage for running applications that involve the processing of large amounts of data. The broad steps for this are as follows:

1. Locate the data to be processed
2. Create a python code to perform the required function.
3. Use Google Cloud functions for more efficient processing.
4. Launch the process on Google Cloud platform.
5. Output is written to the specified location.

Pipeline for first question:

To count the number of lines:

1. Write the python script with the tasks to be performed.
    a. Read the text from out.txt given to us
    b. Count the global number of lines in the file
    c. Write the result to the output file.
2. Launch a virtual environment
3. Run the python script.

Pipeline for second question:

To count average number of words per line:

1. Write the python script with the tasks to be performed.
    a. Read the text from out.txt given to us
    b. Count the number of words per line
    c. Take the global average over all the lines
    d. Write the result to the output file.
2. Launch a virtual environment
3. Run the python script.

Challenges faced:

1. Understanding how to use the Google Cloud Platform, as well as the purpose of various functions.
   This was resolved by going through the various associated documentation

2. Various permissions had to be given, and additional add on packages had to be added.
   These were added as per the given prompts.

3. Specifically, for the addition of the Data Analytics Package, it took about ten minutes to add the package.

Had to wait that long for the change to be reflected.

4. Finding appropriate Cloud Functions, especially for use in the second questions. Going through the various relevant documentation was able to resolve this problem

5. The dataflow was not being triggered for Q5 by having the file in the same bin. A second bin had to be created in which the file had to be moved into to trigger the dataflow.

5. [Bonus] Trigger a dataflow using GCF for any one of the first two questions. [2]

Attached Code: **“main.py”**

Attached Requirements: **“requirements.txt”**

Generated File: **“Count_Lines.txt”**

Python Code:

```
def dataflow_count_lines(data, context):
  from uuid import uuid4
  import apache_beam as beam
  from apache_beam.io import ReadFromText
  from apache_beam.io import WriteToText
  from apache_beam.options.pipeline_options import PipelineOptions
  from apache_beam.options.pipeline_options import GoogleCloudOptions
  from apache_beam.options.pipeline_options import StandardOptions
  file_path = f"gs://{data['bucket']}/{data['name']}"
  unique_id = f"{data['name'].split('.')[0]}-{uuid4()}"
  output_path = f"gs://me17b158_cs4830/lab3/"
  options = PipelineOptions()
  google_cloud_options = options.view_as(GoogleCloudOptions)
  google_cloud_options.project = "splendid-sector-305218"
  google_cloud_options.job_name = f"{unique_id}"
  google_cloud_options.temp_location="gs://me17b158_cs4830/tmp/"
  options.view_as(StandardOptions).runner = "DataflowRunner"
  google_cloud_options.region="us-central1"
  with beam.Pipeline(options=google_cloud_options) as p:
    lines = p | 'Read' >> ReadFromText(file_path)
    counts = lines | 'Count elements' >> beam.combiners.Count.Globally()
    output = counts
    output | 'Write' >> WriteToText(output_path)
```

Requirements:

apache-beam[gcp]

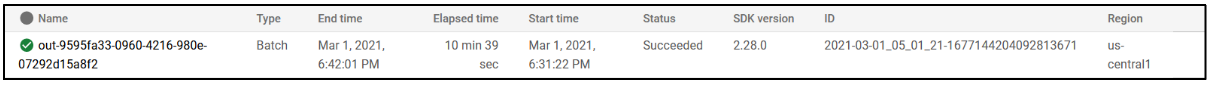
Triggering the Google Cloud Dataflow:

gcloud functions deploy dataflow_count_lines --runtime python37 --timeout 540 --trigger-resource gs://me17b158_cs4830_2 --trigger-event google.storage.object.finalize

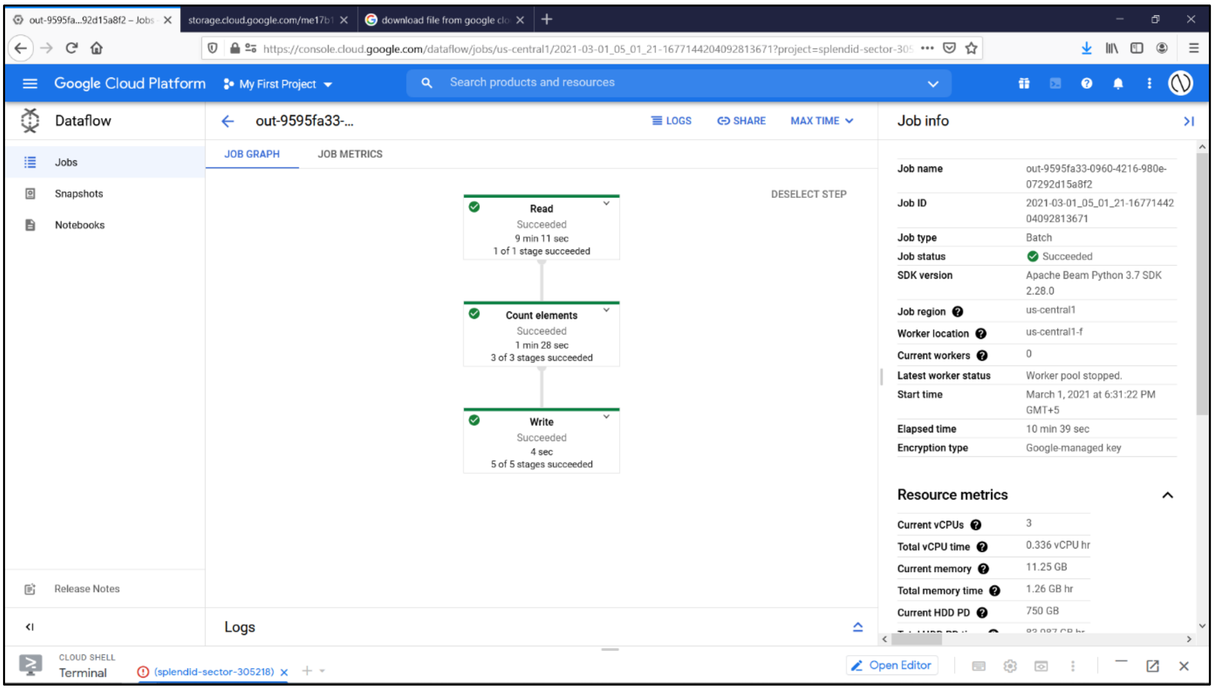
Copying the file (hence triggering dataflow):

gsutil cp gs://iitmbd/out.txt gs://me17b158_cs4830_2/

Screenshots of created Process.



● Name	Type	End time	Elapsed time	Start time	Status	SDK version	ID	Region
✓ out-9595fa33-0960-4216-980e-07292d15a8f2	Batch	Mar 1, 2021, 6:42:01 PM	10 min 39 sec	Mar 1, 2021, 6:31:22 PM	Succeeded	2.28.0	2021-03-01_05_01_21-1677144204092813671	us-central1


out-9595fa...92d15a8f2 – Jobs
storage.cloud.google.com/me17b1
download file from google clo
https://console.cloud.google.com/dataflow/jobs/us-central1/2021-03-01_05_01_21-1677144204092813671?project=splendid-sector-305
Google Cloud Platform
My First Project
Search products and resources
Dataflow
Jobs
Snapshots
Notebooks
Release Notes
out-9595fa33-...
LOGS
SHARE
MAX TIME
JOB GRAPH
JOB METRICS
DESELECT STEP
Logs
CLOUD SHELL
Terminal
(splendid-sector-305218)
Open Editor

Read
Succeeded
9 min 11 sec
1 of 1 stage succeeded

Count elements
Succeeded
1 min 28 sec
3 of 3 stages succeeded

Write
Succeeded
4 sec
5 of 5 stages succeeded

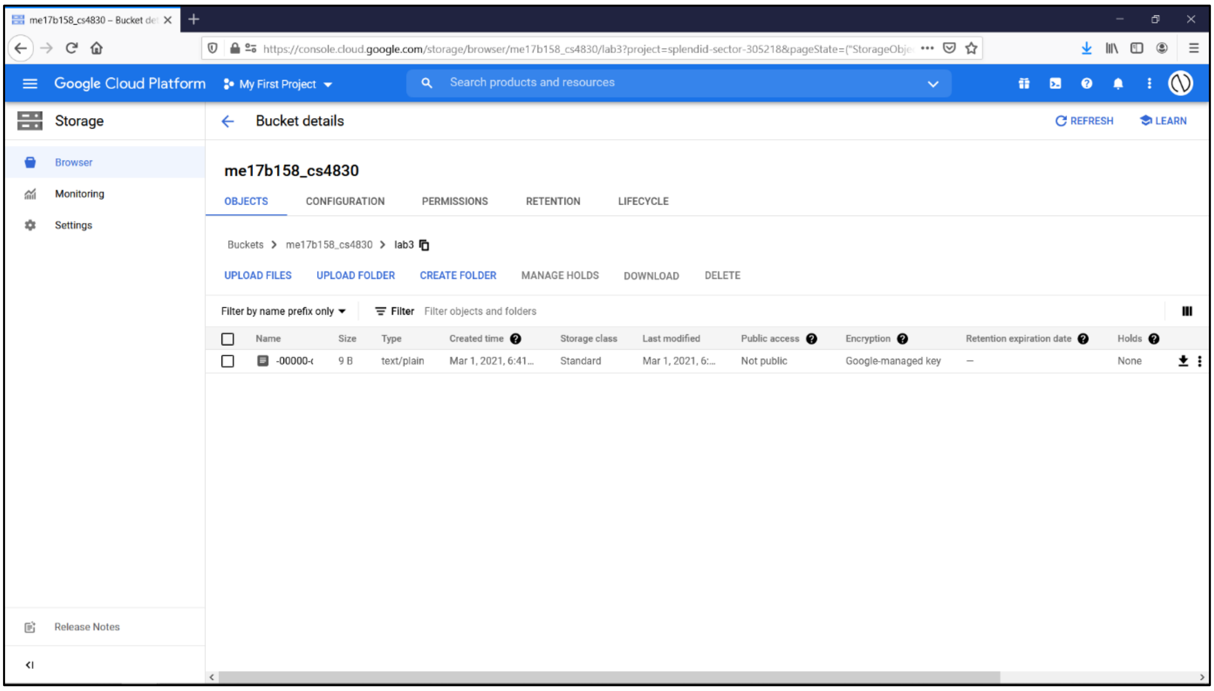
Job info

Job name	out-9595fa33-0960-4216-980e-07292d15a8f2
Job ID	2021-03-01_05_01_21-1677144204092813671
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.7 SDK 2.28.0
Job region	us-central1
Worker location	us-central1-f
Current workers	0
Latest worker status	Worker pool stopped.
Start time	March 1, 2021 at 6:31:22 PM GMT+5
Elapsed time	10 min 39 sec
Encryption type	Google-managed key

Resource metrics

Current vCPUs	3
Total vCPU time	0.336 vCPU hr
Current memory	11.25 GB
Total memory time	1.26 GB hr
Current HDD PD	750 GB

Screenshots of File:

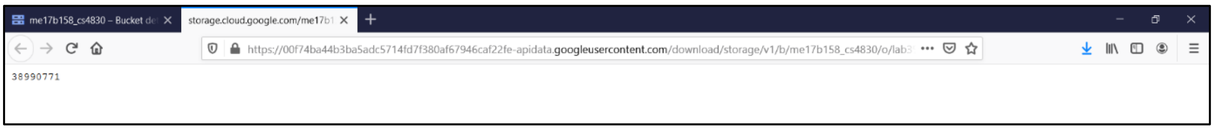


me17b158_cs4830 – Bucket de
https://console.cloud.google.com/storage/browser/me17b158_cs4830/lab3?project=splendid-sector-305218&pageState=("StorageObje
Google Cloud Platform
My First Project
Search products and resources
Storage
Browser
Monitoring
Settings
Release Notes
Bucket details
REFRESH
LEARN
OBJECTS
CONFIGURATION
PERMISSIONS
RETENTION
LIFECYCLE
UPLOAD FILES
UPLOAD FOLDER
CREATE FOLDER
MANAGE HOLDS
DOWNLOAD
DELETE
Filter by name prefix only
Filter
Filter objects and folders

me17b158_cs4830

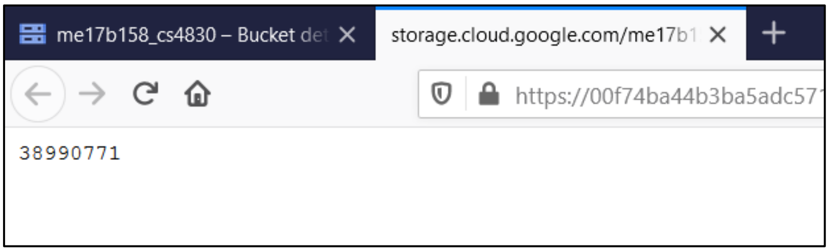
Buckets > me17b158_cs4830 > lab3

	Name	Size	Type	Created time	Storage class	Last modified	Public access	Encryption	Retention expiration date	Holds
□	-00000-(	9 B	text/plain	Mar 1, 2021, 6:41…	Standard	Mar 1, 2021, 6:…	Not public	Google-managed key	–	None



me17b158_cs4830 – Bucket de
storage.cloud.google.com/me17b1
https://00f74ba44b3ba5adc5714fd7f380af67946caf22fe-apidata.googleusercontent.com/download/storage/v1/b/me17b158_cs4830/o/lab3

```
38990771
```



me17b158_cs4830 – Bucket de
storage.cloud.google.com/me17b1
https://00f74ba44b3ba5adc571

```
38990771
```

The results obtained is the same as in Q1.

Create a PDF file containing answers to the above questions. Zip it along with the output files (for the dataflow task) and your Python files. Then, submit this zip file on moodle.