

CSCI 5408

DATA MANAGEMENT AND  
WAREHOUSING

LAB - 5

Banner ID: B00984406

GitLab Assignment Link:

[https://git.cs.dal.ca/jems/csci5408\\_s24\\_b00984406\\_jems\\_patel.git](https://git.cs.dal.ca/jems/csci5408_s24_b00984406_jems_patel.git)

## Table of Contents

Screenshots of the step-by-step process followed to create the Apache Spark (GCP Dataproc) cluster and execute the job (Addition.jar) file on it.....	3
Report any challenges faced while executing the .jar file on the Apache spark cluster.....	8
Explanation of the Java Spark program with the screenshots of the code.....	9
References.....	10

## Screenshots of the step-by-step process followed to create the Apache Spark (GCP Dataproc) cluster and execute the job (Addition.jar) file on it.

Here are the steps that are taken to do this:

### Step:1

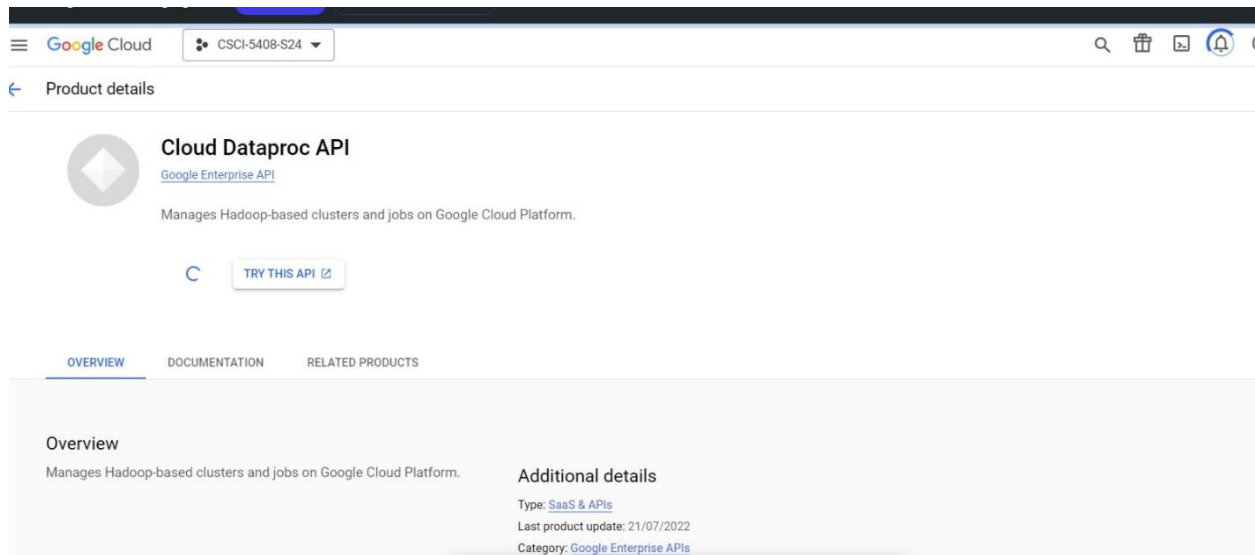


Figure 1.1: Enable the API

### Step:2

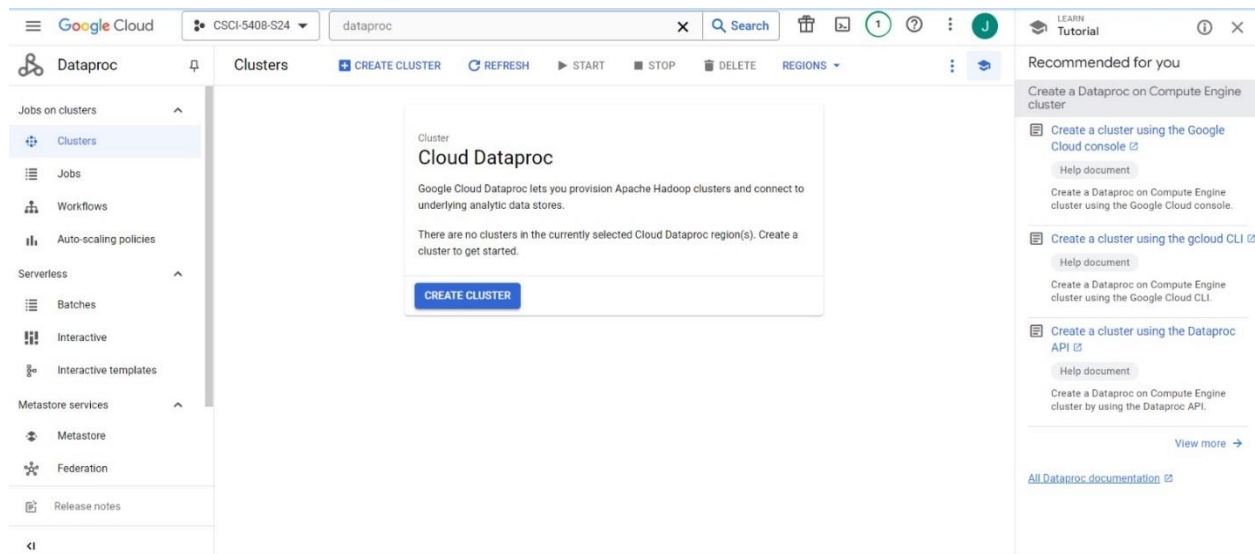


Figure 1.2: Searching the cloud Dataproc

### Step:3

Google Cloud console interface for creating a Dataproc cluster. The left sidebar shows navigation options: Jobs on clusters, Clusters, Jobs, Workflows, Auto-scaling policies, Serverless, Batches, Interactive, Interactive templates, Metastore services, Metastore, Federation, and Release notes. The main content area is titled 'Create a Dataproc cluster on Compute Engine' and includes steps: Set up cluster (active), Configure nodes (optional), Customise cluster (optional), and Manage security (optional). The 'Set up cluster' section contains the following fields:

- Name:** Cluster name \* (lab-5)
- Location:** Region \* (us-central1), Zone \* (Any)
- Cluster type:**
  - ☒ Standard (1 master, N workers)
  - ☐ Single Node (1 master, 0 workers)
  - ☐ High availability (3 masters, N workers)
- Versioning:** Image type and version (2.2-debian12)

Buttons: CREATE, CANCEL. Below the steps is an 'EQUIVALENT COMMAND LINE' dropdown.

Figure 1.3: Setup the cluster

### Step:4

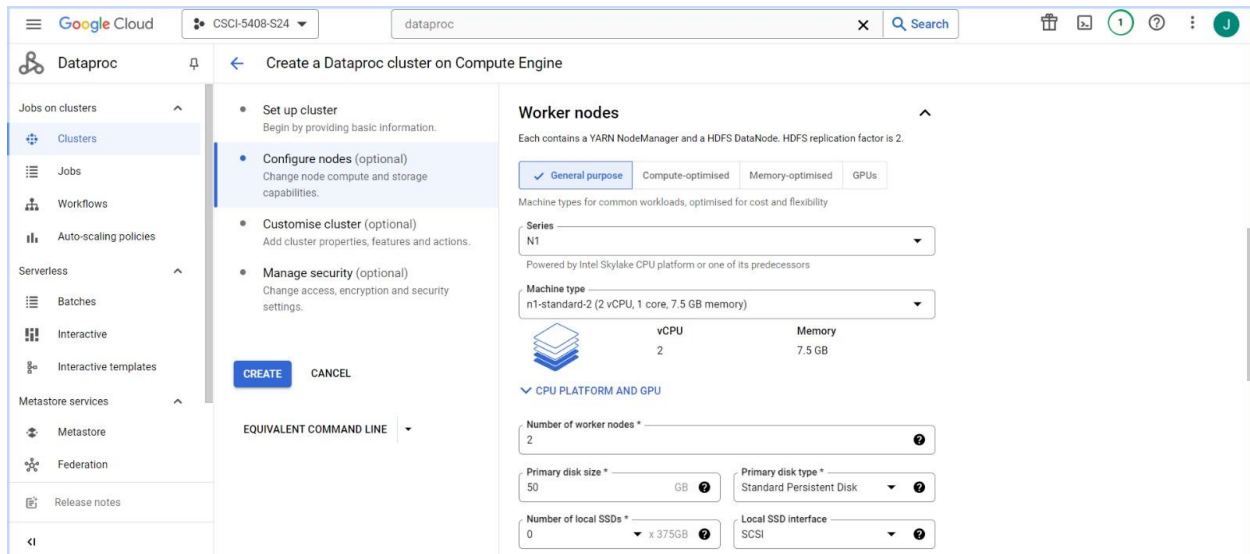
Google Cloud console interface for configuring the manager node. The left sidebar is the same as in Figure 1.3. The main content area is titled 'Create a Dataproc cluster on Compute Engine' and shows the 'Configure nodes (optional)' step active. The 'Manager node' section is expanded, showing the following configuration:

- Manager node:** Contains the YARN Resource Manager, HDFS NameNode and all job drivers.
- General purpose:** Selected tab (General purpose, Compute-optimised, Memory-optimised, GPUs).
- Machine type:** n1-standard-2 (2 vCPU, 1 core, 7.5 GB memory).
- CPU PLATFORM AND GPU:**
  - Primary disk size \* (50 GB)
  - Primary disk type \* (Standard Persistent Disk)
  - Number of local SSDs \* (0 x 375GB)
  - Local SSD interface (SCSI)

Buttons: CREATE, CANCEL. Below the steps is an 'EQUIVALENT COMMAND LINE' dropdown.

Figure 1.4: Configure the manager node

## Step:5



Google Cloud | CSCI-5408-S24 | dataproc

### Create a Dataproc cluster on Compute Engine

- Set up cluster  
Begin by providing basic information.
- Configure nodes (optional)**  
Change node compute and storage capabilities.
- Customise cluster (optional)  
Add cluster properties, features and actions.
- Manage security (optional)  
Change access, encryption and security settings.

**Worker nodes**  
Each contains a YARN NodeManager and a HDFS DataNode. HDFS replication factor is 2.

☒ General purpose ☐ Compute-optimised ☐ Memory-optimised ☐ GPUs

Machine types for common workloads, optimised for cost and flexibility

Series: N1  
Powered by Intel Skylake CPU platform or one of its predecessors

Machine type: n1-standard-2 (2 vCPU, 1 core, 7.5 GB memory)

vCPU: 2 | Memory: 7.5 GB

**CPU PLATFORM AND GPU**

Number of worker nodes \*: 2

Primary disk size \*: 50 GB | Primary disk type \*: Standard Persistent Disk

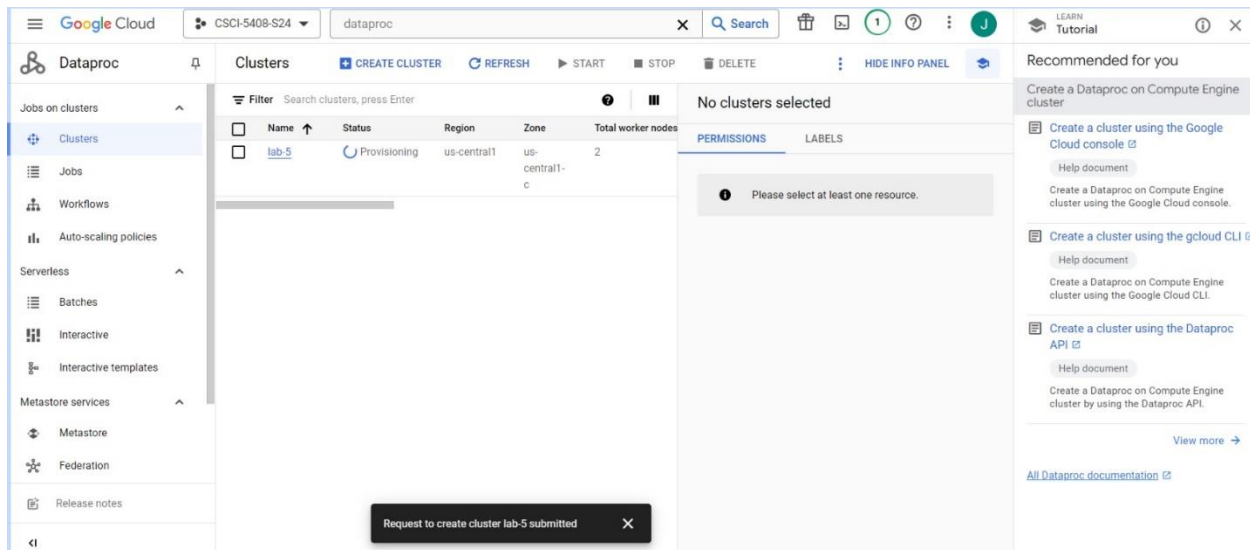
Number of local SSDs \*: 0 | Local SSD interface: SCSI

**CREATE** **CANCEL**

EQUIVALENT COMMAND LINE

Figure 1.5: Configure the worker node

## Step:6



Google Cloud | CSCI-5408-S24 | dataproc

### Clusters

**CREATE CLUSTER** **REFRESH** **START** **STOP** **DELETE** **HIDE INFO PANEL**

Filter: Search clusters, press Enter

Name	Status	Region	Zone	Total worker nodes
lab-5	Provisioning	us-central1	us-central1-c	2

**No clusters selected**

**PERMISSIONS** **LABELS**

Please select at least one resource.

**Request to create cluster lab-5 submitted**

#### Recommended for you

- Create a Dataproc on Compute Engine cluster
- Create a cluster using the Google Cloud console [Help document](#)  
Create a Dataproc on Compute Engine cluster using the Google Cloud console.
- Create a cluster using the gcloud CLI [Help document](#)  
Create a Dataproc on Compute Engine cluster using the Google Cloud CLI.
- Create a cluster using the Dataproc API [Help document](#)  
Create a Dataproc on Compute Engine cluster by using the Dataproc API.

[View more](#)

[All Dataproc documentation](#)

Figure 1.6: Creating the cluster

## Step:7

The screenshot shows the Google Cloud Dataproc console. The left sidebar contains navigation links for Clusters, Jobs, Workflows, Auto-scaling policies, Serverless, Batches, Interactive, Interactive templates, Metastore services, Metastore, Federation, and Utilities. The main panel displays the 'Cluster details' for a cluster named 'lab-5'. A message at the top indicates a failed validation of permissions for the default service account. Below this, a table lists the VM instances:

Name	Role	Machine type
lab-5-m	Master	n1-standard-2
lab-5-w-0	Worker	n1-standard-2
lab-5-w-1	Worker	n1-standard-2

The URL at the bottom of the console is: <https://console.cloud.google.com/dataproc/clusters/lab-5/instances?region=us-central1&project=lyrical-compass-425014-n3>

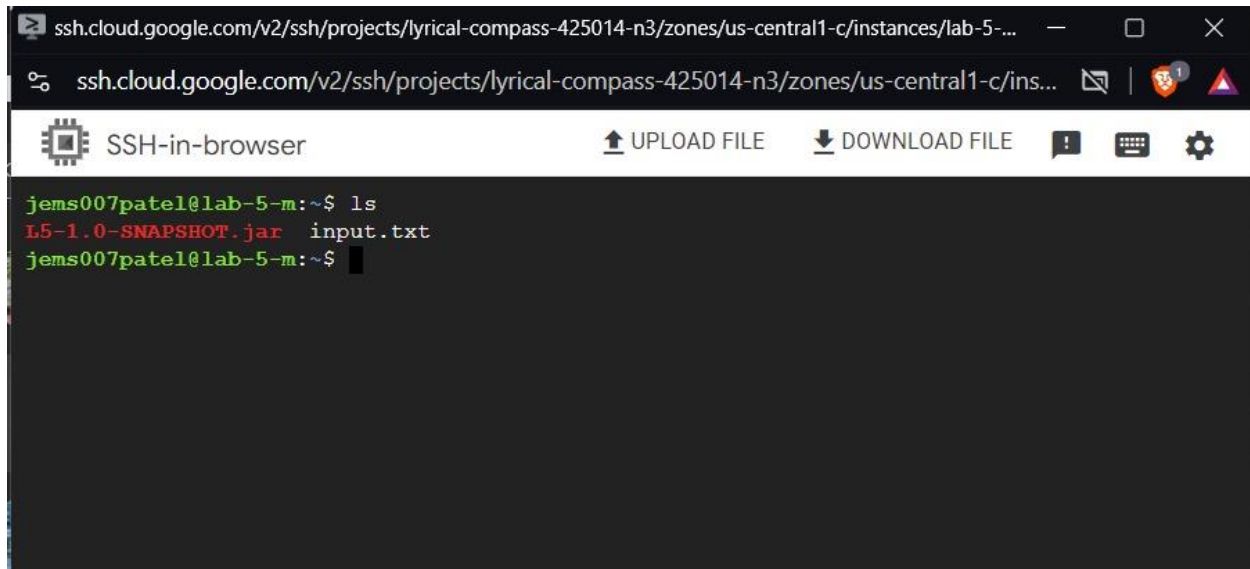
Figure 1.7: List of VM Instances

## Step:8

The screenshot shows an SSH-in-browser window with the address bar displaying `ssh.cloud.google.com/v2/ssh/projects/lyrical-compass-425014-n3/zones/us-central1-c/instances/lab-5-...`. The terminal shows the prompt `jems007pate1@lab-5-m: ~$`. An 'Upload' dialog box is open in the foreground, titled 'Upload', with the subtitle 'Upload files from your computer'. It contains a 'Choose Files' button, a text input field with the filename 'L5-1.0-SNAPSHOT.jar', and 'Cancel' and 'Upload files' buttons.

Figure 1.8: Uploading the jar file of the java code

Step:9

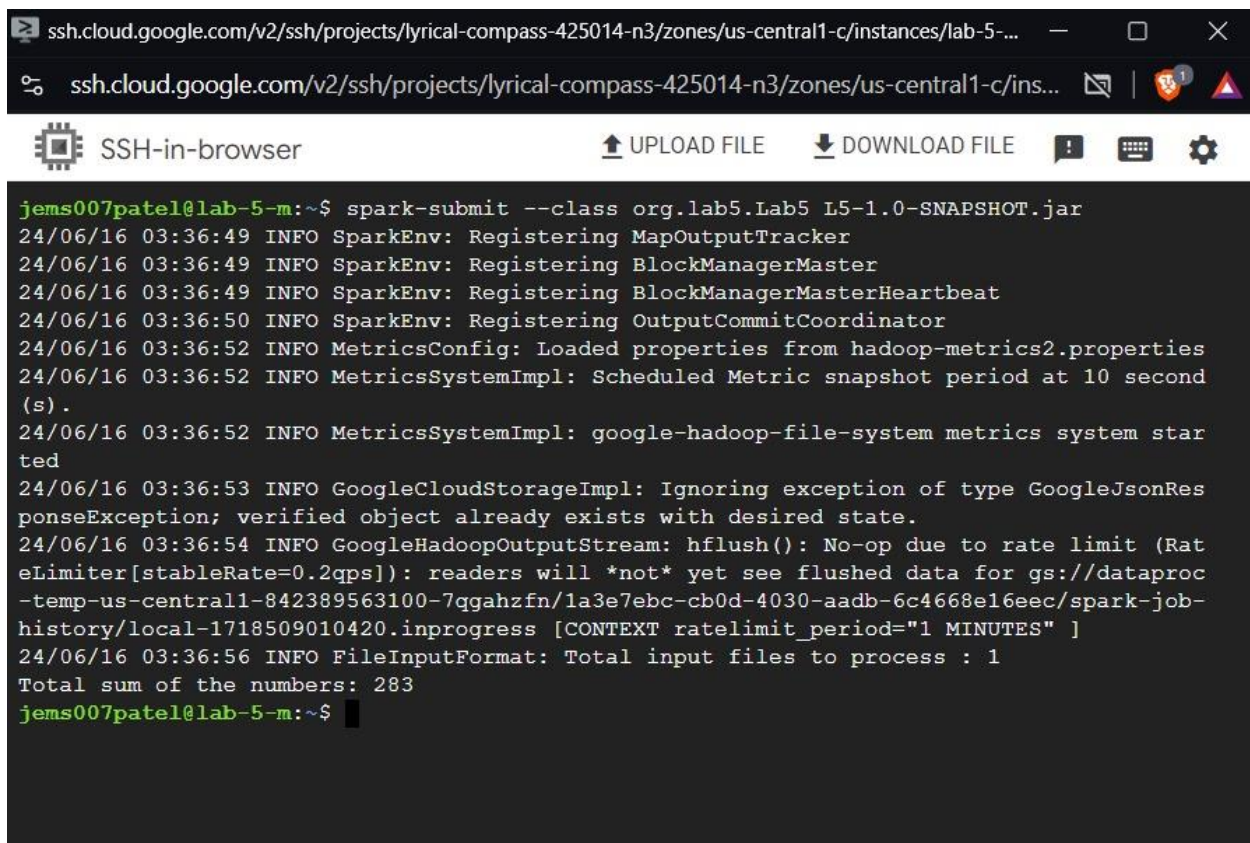


The screenshot shows a web browser window with the address bar displaying 'ssh.cloud.google.com/v2/ssh/projects/lyrical-compass-425014-n3/zones/us-central1-c/instances/lab-5-...'. The browser tab is titled 'SSH-in-browser'. Below the browser window, there is a terminal interface. The terminal shows the prompt 'jems007patel@lab-5-m:~\$' followed by the command 'ls'. The output of the command is 'L5-1.0-SNAPSHOT.jar' and 'input.txt'. The prompt then returns to 'jems007patel@lab-5-m:~\$'.

```
jems007patel@lab-5-m:~$ ls
L5-1.0-SNAPSHOT.jar  input.txt
jems007patel@lab-5-m:~$
```

Figure 1.9: list the uploaded files

Step:10



The screenshot shows a web browser window with the address bar displaying 'ssh.cloud.google.com/v2/ssh/projects/lyrical-compass-425014-n3/zones/us-central1-c/instances/lab-5-...'. The browser tab is titled 'SSH-in-browser'. Below the browser window, there is a terminal interface. The terminal shows the prompt 'jems007patel@lab-5-m:~\$' followed by the command 'spark-submit --class org.lab5.Lab5 L5-1.0-SNAPSHOT.jar'. The output of the command is a series of log messages from SparkEnv, BlockManagerMaster, BlockManagerMasterHeartbeat, OutputCommitCoordinator, MetricsConfig, MetricsSystemImpl, and GoogleCloudStorageImpl. The final output is 'Total sum of the numbers: 283'. The prompt then returns to 'jems007patel@lab-5-m:~\$'.

```
jems007patel@lab-5-m:~$ spark-submit --class org.lab5.Lab5 L5-1.0-SNAPSHOT.jar
24/06/16 03:36:49 INFO SparkEnv: Registering MapOutputTracker
24/06/16 03:36:49 INFO SparkEnv: Registering BlockManagerMaster
24/06/16 03:36:49 INFO SparkEnv: Registering BlockManagerMasterHeartbeat
24/06/16 03:36:50 INFO SparkEnv: Registering OutputCommitCoordinator
24/06/16 03:36:52 INFO MetricsConfig: Loaded properties from hadoop-metrics2.properties
24/06/16 03:36:52 INFO MetricsSystemImpl: Scheduled Metric snapshot period at 10 second
(s).
24/06/16 03:36:52 INFO MetricsSystemImpl: google-hadoop-file-system metrics system star
ted
24/06/16 03:36:53 INFO GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonRes
ponseException; verified object already exists with desired state.
24/06/16 03:36:54 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (Rat
eLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data for gs://dataproc
-temp-us-central1-842389563100-7qgahzfn/1a3e7ebc-cb0d-4030-aadb-6c4668e16eec/spark-job-
history/local-1718509010420.inprogress [CONTEXT ratelimit_period="1 MINUTES" ]
24/06/16 03:36:56 INFO FileInputFormat: Total input files to process : 1
Total sum of the numbers: 283
jems007patel@lab-5-m:~$
```

Figure 1.10: Sum of the numbers

## ***Report any challenges faced while executing the .jar file on the Apache spark cluster.***

While executing the .jar file in the Apache Spark cluster, I encountered two issues.

The first issue was related to creating the cluster according to the provided steps. Despite following the instructions correctly, the process often failed after making me wait up to 30 minutes. I resolved this by changing the configuration for the worker node to N1, which eventually succeeded.

The second issue involved accessing the input file in the Java code. Although I uploaded the input.txt file via the SSH terminal, the code couldn't locate it. Upon investigation, I found that the file wasn't present in Hadoop. To address this, I first created the necessary directory using the following command:

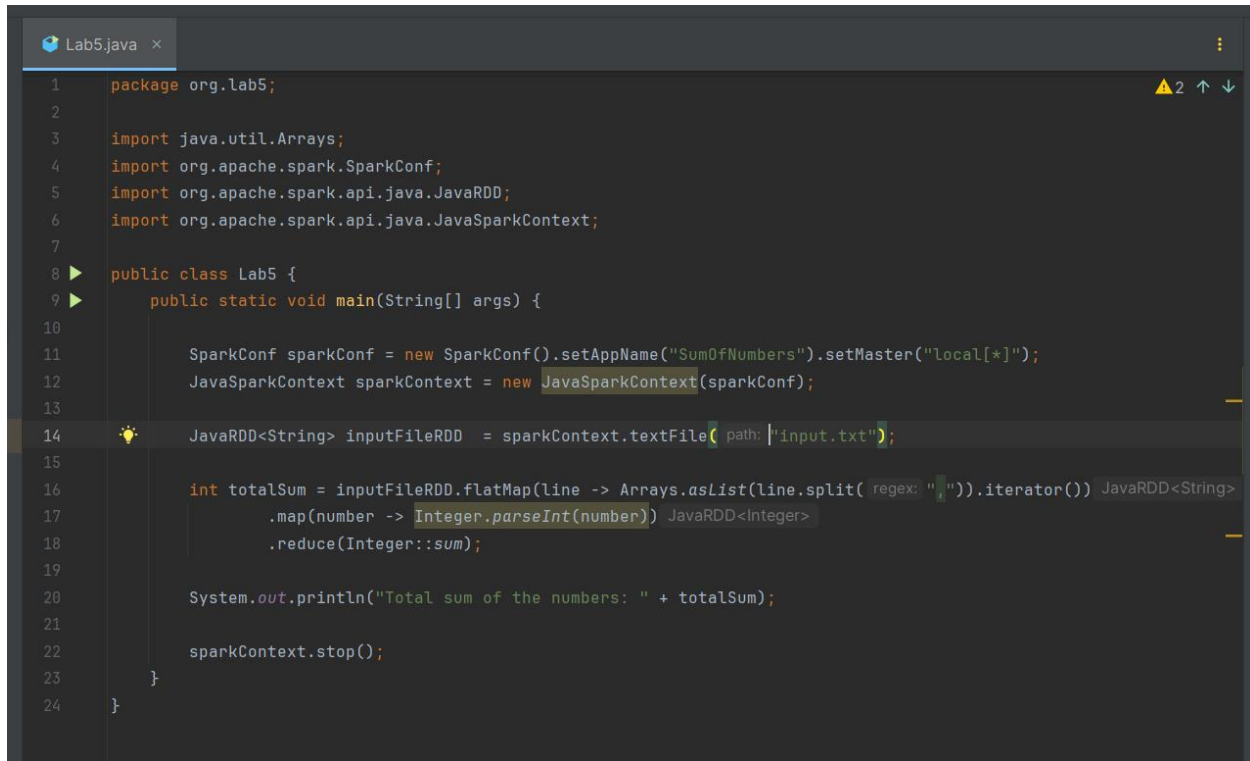
```
hadoop fs -mkdir -p hdfs://lab-5-m/user/jems007patel
```

After this I copy the txt file into hadoop manually by using this given command and successfully run the jar file

```
hadoop fs -copyFromLocal ./input.txt user/jems007patel/input.txt
```



## Explanation of the Java Spark program with the screenshots of the code.



```
1 package org.lab5;
2
3 import java.util.Arrays;
4 import org.apache.spark.SparkConf;
5 import org.apache.spark.api.java.JavaRDD;
6 import org.apache.spark.api.java.JavaSparkContext;
7
8 public class Lab5 {
9     public static void main(String[] args) {
10
11         SparkConf sparkConf = new SparkConf().setAppName("SumOfNumbers").setMaster("local[*]");
12         JavaSparkContext sparkContext = new JavaSparkContext(sparkConf);
13
14         JavaRDD<String> inputFileRDD = sparkContext.textFile("input.txt");
15
16         int totalSum = inputFileRDD.flatMap(line -> Arrays.asList(line.split(" ")).iterator())
17                                     .map(number -> Integer.parseInt(number))
18                                     .reduce(Integer::sum);
19
20         System.out.println("Total sum of the numbers: " + totalSum);
21
22         sparkContext.stop();
23     }
24 }
```

Figure 1.11: Java Spark Code

Here is the step-by-step explanation for the above java code:

- To initialize this Spark application, a SparkConf object is created, where the property `conf.setAppName("SumOfNumbers")` sets the name of the application and `conf.setMaster("local")` specifies to run the application locally.
- Using this configuration, a `JavaSparkContext` is created, opening the gate to Spark functionalities.
- After that this program reads `input.txt` and convert file into an RDD, where an element in an RDD is equal to a line in the file.
- FlatMap Transformation Splits each line into numbers and then RDD of number is flattened out as an RDD of strings.
- Map Transformation Transforms all the string numbers, present in the input string, into integers.
- Reduce Action Reduces the RDD to single value by summing all the Integers.
- The sum of all these numbers is the total sum and it is printed in the console and shuts down the Spark context.

## References

- [1] "Apache Spark Documentation," Apache Spark, [Online]. Available: <https://spark.apache.org/docs/latest/>. [Accessed 16 June 2024].
- [2] " K. Saxena, "All You Need to Know About Google Cloud Dataproc," Medium, [Online]. Available: <https://medium.com/google-cloud/all-you-need-to-know-about-google-cloud-dataproc-23fe91369678>. [Accessed 16 June 2024].