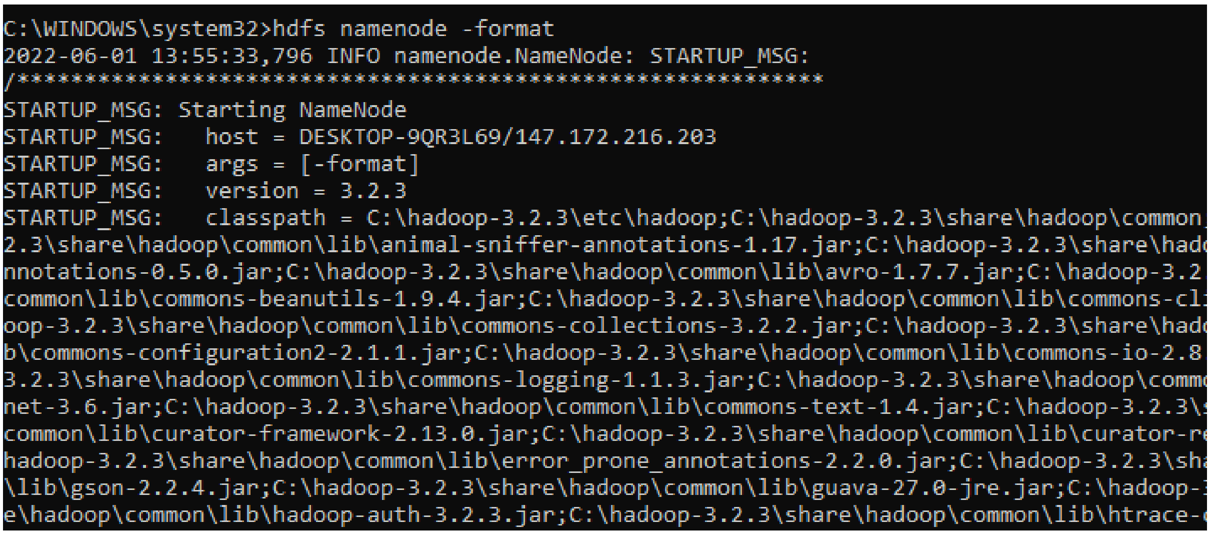
Sharon Laurance

Exercise 1: Preparing your Hadoop infrastructure

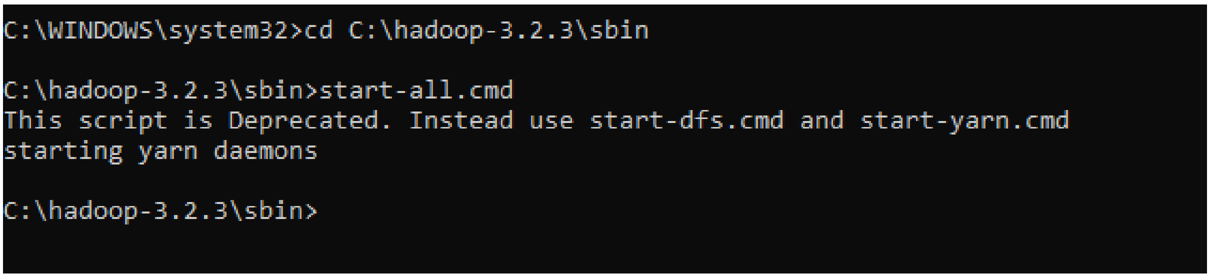
Exercise 1.1: Setting up a Hadoop infrastructure

Installation of Hadoop in Pseudo-Distributed mode

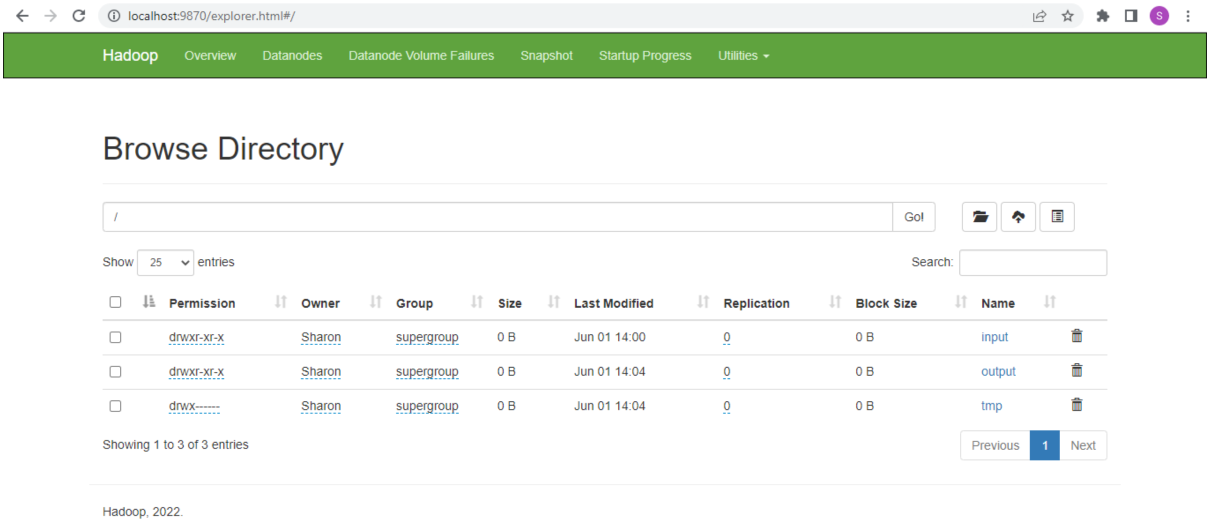
2. Initialize the HDFS file system by the command: hadoop namenode -format.



```
C:\WINDOWS\system32>hdfs namenode -format
2022-06-01 13:55:33,796 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = DESKTOP-9QR3L69/147.172.216.203
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.2.3
STARTUP_MSG:   classpath = C:\hadoop-3.2.3\etc\hadoop;C:\hadoop-3.2.3\share\hadoop\common
2.3\share\hadoop\common\lib\animal-sniffer-annotations-1.17.jar;C:\hadoop-3.2.3\share\had
nnotations-0.5.0.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\avro-1.7.7.jar;C:\hadoop-3.2
common\lib\commons-beanutils-1.9.4.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\commons-cl
oop-3.2.3\share\hadoop\common\lib\commons-collections-3.2.2.jar;C:\hadoop-3.2.3\share\had
b\commons-configuration2-2.1.1.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\commons-io-2.8
3.2.3\share\hadoop\common\lib\commons-logging-1.1.3.jar;C:\hadoop-3.2.3\share\hadoop\comm
net-3.6.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\commons-text-1.4.jar;C:\hadoop-3.2.3\
common\lib\curator-framework-2.13.0.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\curator-r
hadoop-3.2.3\share\hadoop\common\lib\error_prone_annotations-2.2.0.jar;C:\hadoop-3.2.3\sh
\lib\gson-2.2.4.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\guava-27.0-jre.jar;C:\hadoop-
e\hadoop\common\lib\hadoop-auth-3.2.3.jar;C:\hadoop-3.2.3\share\hadoop\common\lib\htrace-
```



```
C:\WINDOWS\system32>cd C:\hadoop-3.2.3\sbin

C:\hadoop-3.2.3\sbin>start-all.cmd
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\hadoop-3.2.3\sbin>
```



Browse Directory

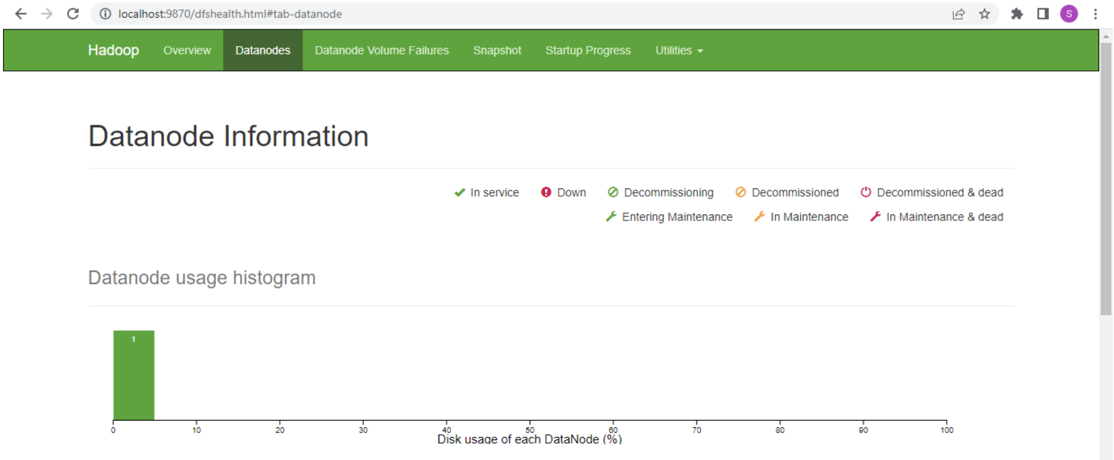
/

☐	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
☐	drwxr-xr-x	Sharon	supergroup	0 B	Jun 01 14:00	0	0 B	input
☐	drwxr-xr-x	Sharon	supergroup	0 B	Jun 01 14:04	0	0 B	output
☐	drwx------	Sharon	supergroup	0 B	Jun 01 14:04	0	0 B	tmp

Showing 1 to 3 of 3 entries

Previous 1 Next

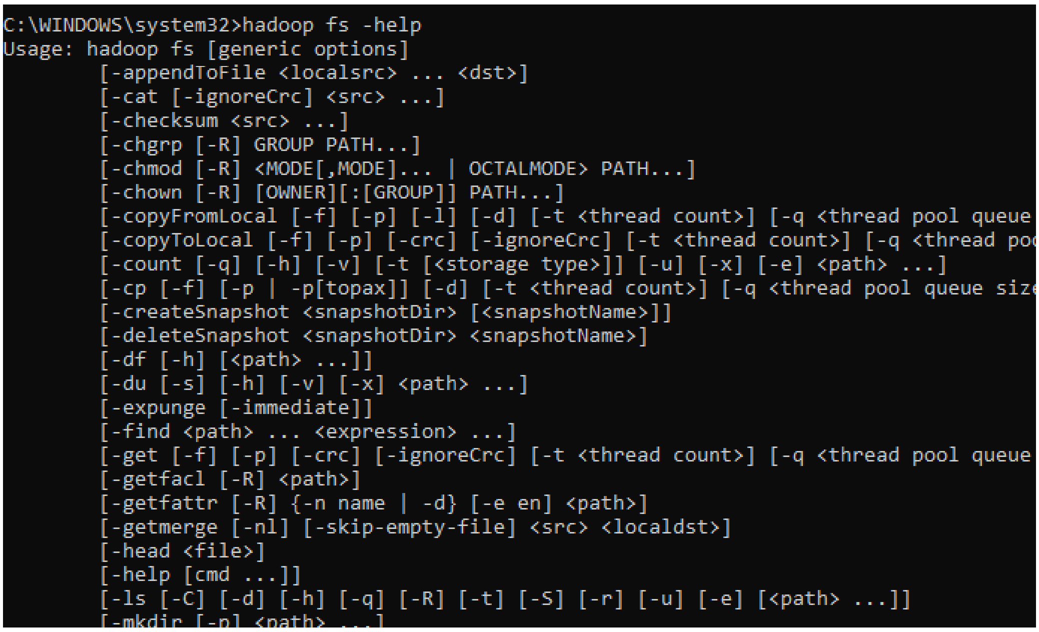
Hadoop, 2022.



Datanode Information

✓ In service ● Down ⊘ Decommissioning ⊘ Decommissioned ⏻ Decommissioned & dead
Entering Maintenance In Maintenance In Maintenance & dead

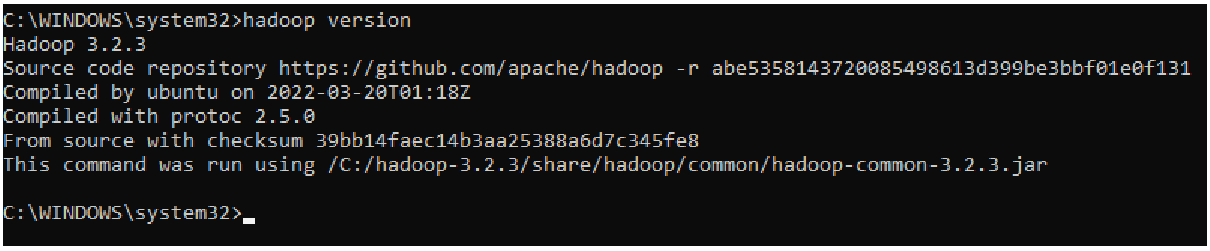
Datanode usage histogram



```
C:\WINDOWS\system32>hadoop fs -help
Usage: hadoop fs [generic options]
        [-appendToFile <localsrc> ... <dst>]
        [-cat [-ignoreCrc] <src> ...]
        [-checksum <src> ...]
        [-chgrp [-R] GROUP PATH...]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-copyFromLocal [-f] [-p] [-l] [-d] [-t <thread count>] [-q <thread pool queue
        [-copyToLocal [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread poo
        [-count [-q] [-h] [-v] [-t [<storage type>]] [-u] [-x] [-e] <path> ...]
        [-cp [-f] [-p | -p[topax]] [-d] [-t <thread count>] [-q <thread pool queue size
        [-createSnapshot <snapshotDir> [<snapshotName>]]
        [-deleteSnapshot <snapshotDir> <snapshotName>]
        [-df [-h] [<path> ...]]
        [-du [-s] [-h] [-v] [-x] <path> ...]
        [-expunge [-immediate]]
        [-find <path> ... <expression> ...]
        [-get [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread pool queue
        [-getfacl [-R] <path>]
        [-getfattr [-R] {-n name | -d} [-e en] <path>]
        [-getmerge [-nl] [-skip-empty-file] <src> <localdst>]
        [-head <file>]
        [-help [cmd ...]]
        [-ls [-C] [-d] [-h] [-q] [-R] [-t] [-S] [-r] [-u] [-e] [<path> ...]]
        [-mkdir [-p] <path> ...]
```

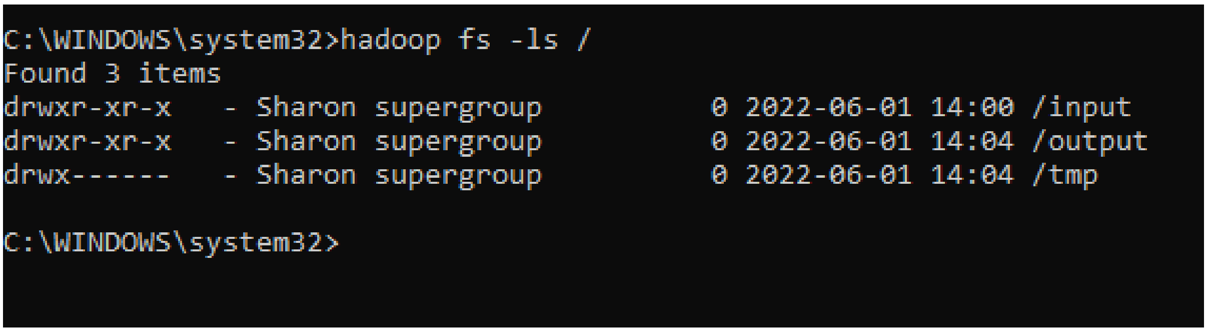
Exercise 1.2: Basic Hadoop operations

1. Check Hadoop version: hadoop version



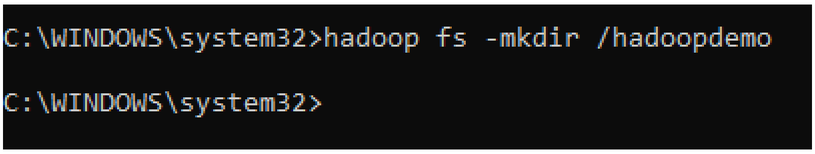
```
C:\WINDOWS\system32>hadoop version
Hadoop 3.2.3
Source code repository https://github.com/apache/hadoop -r abe5358143720085498613d399be3bbf01e0f131
Compiled by ubuntu on 2022-03-20T01:18Z
Compiled with protoc 2.5.0
From source with checksum 39bb14faec14b3aa25388a6d7c345fe8
This command was run using /C:/hadoop-3.2.3/share/hadoop/common/hadoop-common-3.2.3.jar

C:\WINDOWS\system32>_
```

2. List files in HDFS: hadoop fs -ls /

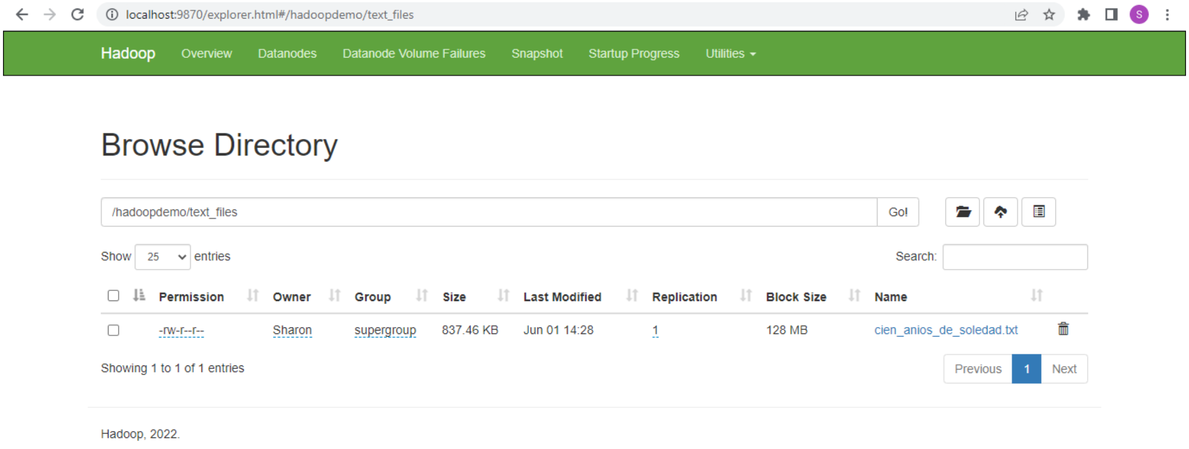


```
C:\WINDOWS\system32>hadoop fs -ls /
Found 3 items
drwxr-xr-x   - Sharon supergroup          0 2022-06-01 14:00 /input
drwxr-xr-x   - Sharon supergroup          0 2022-06-01 14:04 /output
drwx------   - Sharon supergroup          0 2022-06-01 14:04 /tmp

C:\WINDOWS\system32>
```

3. Create a hadoopdemo directory: hadoop fs -mkdir /hadoopdemo



```
C:\WINDOWS\system32>hadoop fs -mkdir /hadoopdemo

C:\WINDOWS\system32>
```



Browse Directory

/hadoopdemo/text_files

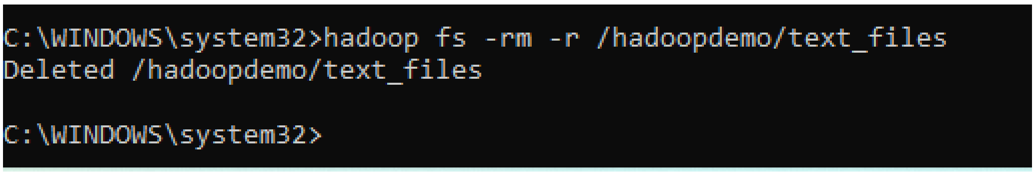
☐	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
☐	-rw-r--r--	Sharon	supergroup	837.46 KB	Jun 01 14:28	1	128 MB	cien_anios_de_soledad.txt

Showing 1 to 1 of 1 entries

Hadoop, 2022.

4. Create several sub-directories nested in hadoopdemo, e.g. text files, raw data

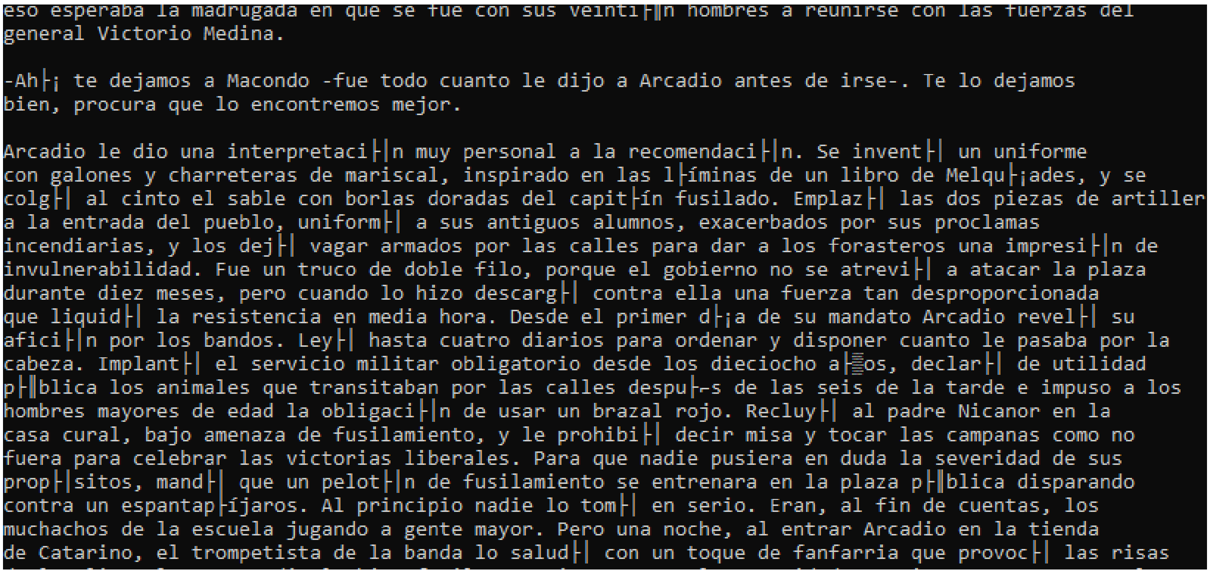
7. Remove the sub-directory hadoop fs -rm -r /hadoopdemo/text files



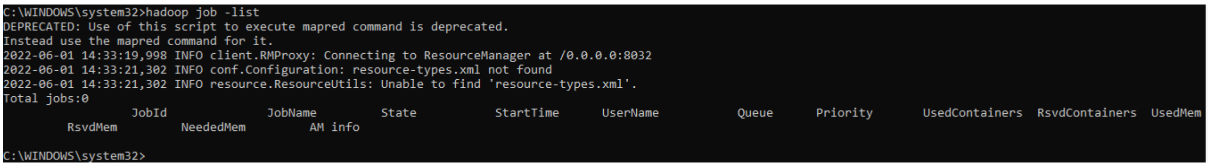
```
C:\WINDOWS\system32>hadoop fs -rm -r /hadoopdemo/text_files
Deleted /hadoopdemo/text_files

C:\WINDOWS\system32>
```

8. Change the content of file.txt in the local system and overwrite it in Hadoop hadoop fs -put -f file.txt /hadoopdemo/text files

9. Read the content of the file: hadoop fs -cat /hadoopdemo/text files/file.txt



```
eso esperaba la madrugada en que se fue con sus veinti├║n hombres a reunirse con las fuerzas del
general Victorio Medina.

-Ah├¡ te dejamos a Macondo -fue todo cuanto le dijo a Arcadio antes de irse-. Te lo dejamos
bien, procura que lo encontremos mejor.

Arcadio le dio una interpretaci├│n muy personal a la recomendaci├│n. Se invent├│ un uniforme
con galones y charreteras de mariscal, inspirado en las l├íminas de un libro de Melqu├¡ades, y se
colg├│ al cinto el sable con borlas doradas del capit├ín fusilado. Emplaz├│ las dos piezas de artiller
a la entrada del pueblo, uniform├│ a sus antiguos alumnos, exacerbados por sus proclamas
incendiarias, y los dej├│ vagar armados por las calles para dar a los forasteros una impresi├│n de
invulnerabilidad. Fue un truco de doble filo, porque el gobierno no se atrevi├│ a atacar la plaza
durante diez meses, pero cuando lo hizo descarg├│ contra ella una fuerza tan desproporcionada
que liquid├│ la resistencia en media hora. Desde el primer d├¡a de su mandato Arcadio revel├│ su
afici├│n por los bandos. Ley├│ hasta cuatro diarios para ordenar y disponer cuanto le pasaba por la
cabeza. Implant├│ el servicio militar obligatorio desde los dieciocho a├▒os, declar├│ de utilidad
p├║blica los animales que transitaban por las calles despu├ęs de las seis de la tarde e impuso a los
hombres mayores de edad la obligaci├│n de usar un brazal rojo. Recluy├│ al padre Nicanor en la
casa cural, bajo amenaza de fusilamiento, y le prohibi├│ decir misa y tocar las campanas como no
fuera para celebrar las victorias liberales. Para que nadie pusiera en duda la severidad de sus
prop├│sitos, mand├│ que un pelot├│n de fusilamiento se entrenara en la plaza p├║blica disparando
contra un espantap├íjaros. Al principio nadie lo tom├│ en serio. Eran, al fin de cuentas, los
muchachos de la escuela jugando a gente mayor. Pero una noche, al entrar Arcadio en la tienda
de Catarino, el trompetista de la banda lo salud├│ con un toque de fanfarria que provoc├│ las risas
```



```
C:\WINDOWS\system32>hadoop job -list
DEPRECATED: Use of this script to execute mapred command is deprecated.
Instead use the mapred command for it.
2022-06-01 14:33:19,998 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-01 14:33:21,302 INFO conf.Configuration: resource-types.xml not found
2022-06-01 14:33:21,302 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
Total jobs:0
                  JobId              JobName          State           StartTime       UserName           Queue       Priority       UsedContainers  RsvdContainers  UsedMem
         RsvdMem         NeededMem          AM info

C:\WINDOWS\system32>
```

5. Transfer and store a data file from local systems to Hadoop: hadoop fs -put file.txt /hadoopdemo/text files

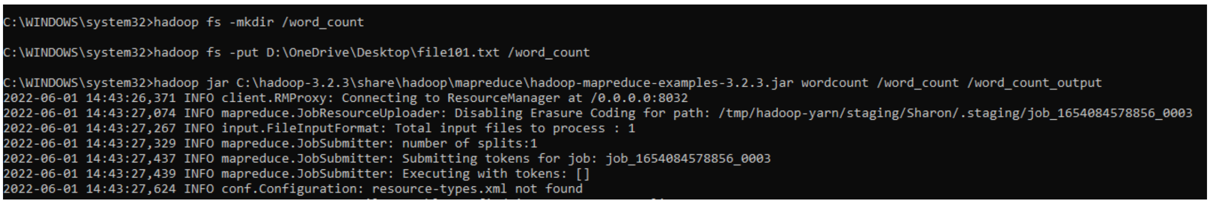
Exercise 1.3: WordCount MapReduce example

1. First a directory is created named word_count using command: hadoop fs -mkdir word_count
2. Then text dataset used in this exercise is put in the folder using command: hadoop fs -put D:\OneDrive\Desktop\file101.txt /word_count
3. Now we will use the map-reduce that already comes with the Hadoop installation and is present at the path "C:\hadoop-3.2.3\share\hadoop\mapreduce" and help to load the word count program
4. The output of the program is stored in /word_count /word_count_output and from there we can see count of individual words at the URL "http://localhost:9870/explorer.html#/word_count/output1"

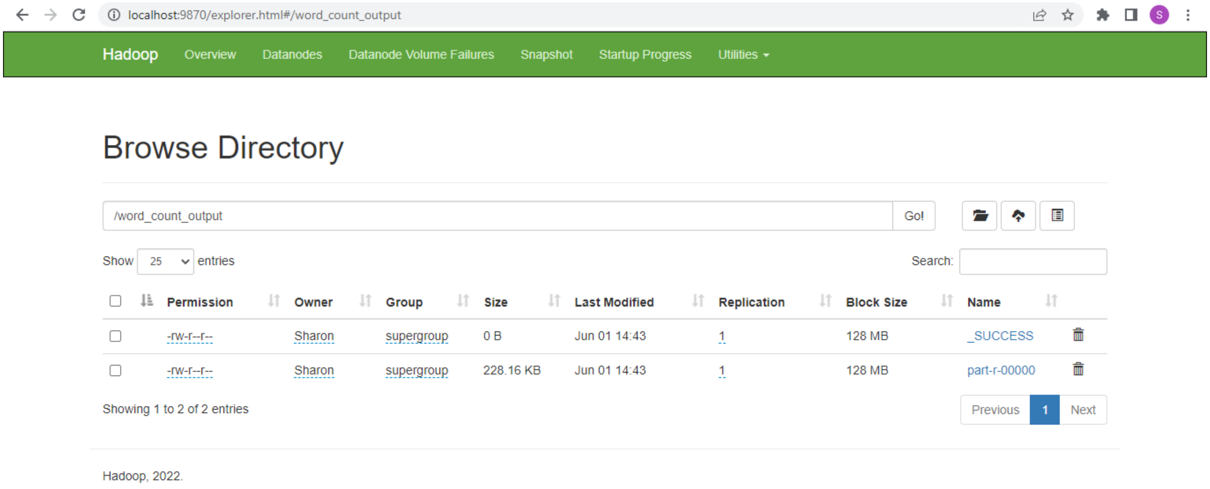
Commands Used are:

hadoop fs -put D:\OneDrive\Desktop\file101.txt /word_count

hadoop jar C:\hadoop-3.2.3\share\hadoop\mapreduce\hadoop-mapreduce-examples-3.2.3.jar wordcount /word_count /word_count_output



```
C:\WINDOWS\system32>hadoop fs -mkdir /word_count

C:\WINDOWS\system32>hadoop fs -put D:\OneDrive\Desktop\file101.txt /word_count

C:\WINDOWS\system32>hadoop jar C:\hadoop-3.2.3\share\hadoop\mapreduce\hadoop-mapreduce-examples-3.2.3.jar wordcount /word_count /word_count_output
2022-06-01 14:43:26,371 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-01 14:43:27,074 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654084578856_0003
2022-06-01 14:43:27,267 INFO input.FileInputFormat: Total input files to process : 1
2022-06-01 14:43:27,329 INFO mapreduce.JobSubmitter: number of splits:1
2022-06-01 14:43:27,437 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654084578856_0003
2022-06-01 14:43:27,439 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-01 14:43:27,624 INFO conf.Configuration: resource-types.xml not found
```



Browse Directory

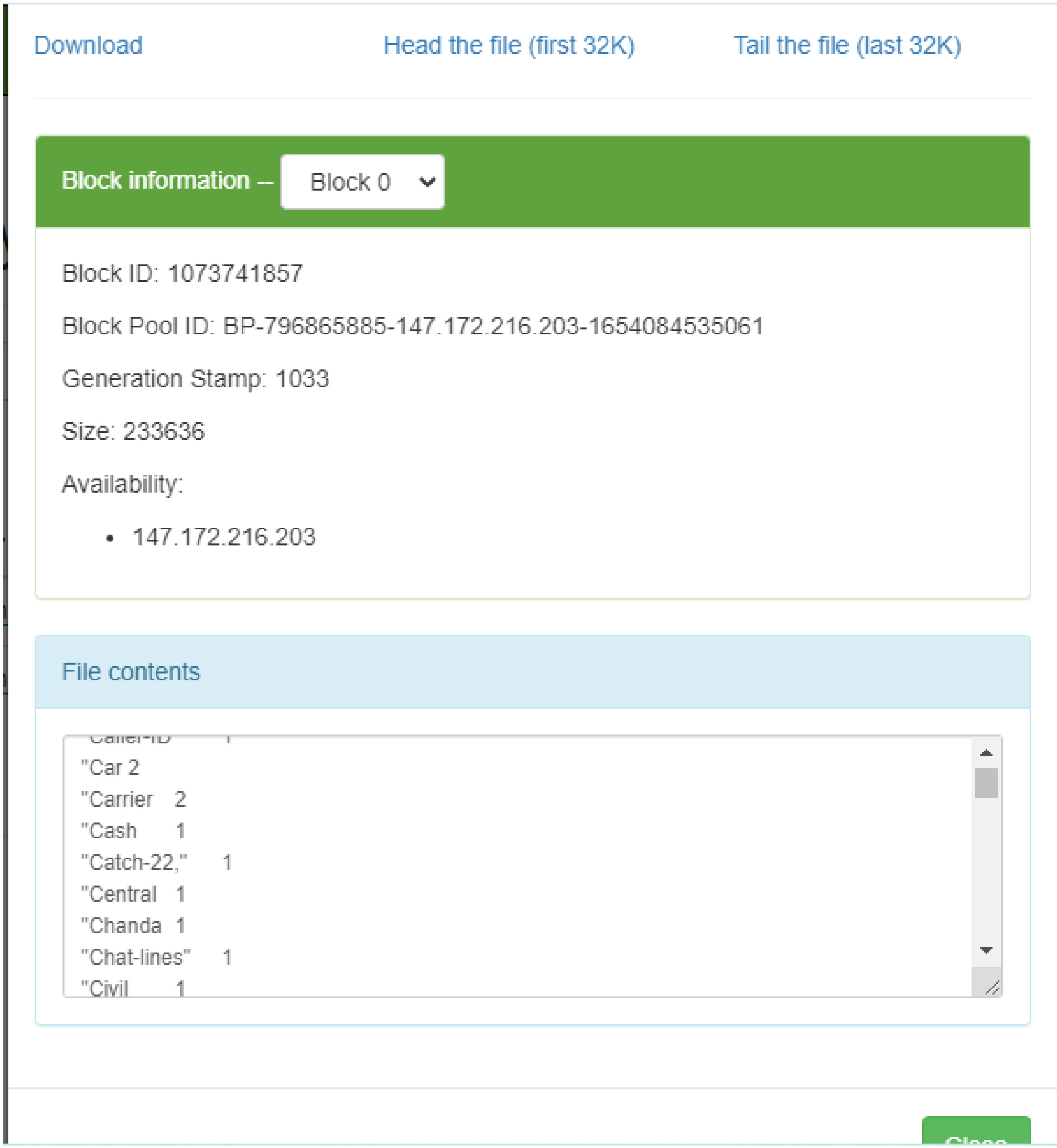
/word_count_output Go!

Show 25 entries Search:

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	-rw-r--r--	Sharon	supergroup	0 B	Jun 01 14:43	1	128 MB	_SUCCESS
□	-rw-r--r--	Sharon	supergroup	228.16 KB	Jun 01 14:43	1	128 MB	part-r-00000

Showing 1 to 2 of 2 entries

Hadoop, 2022.



Block information -- Block 0

Block ID: 1073741857

Block Pool ID: BP-796865885-147.172.216.203-1654084535061

Generation Stamp: 1033

Size: 233636

Availability:

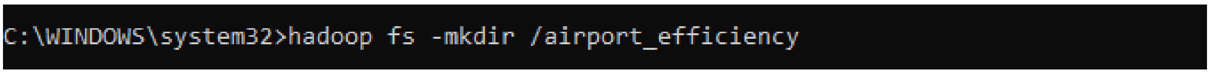
- 147.172.216.203

File contents

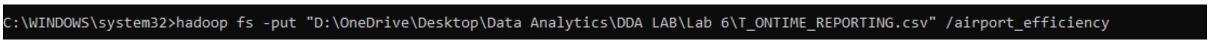
```
"Caller-ID	1
"Car	2
"Carrier	2
"Cash	1
"Catch-22,"	1
"Central	1
"Chanda	1
"Chat-lines"	1
"Civil	1
```

Exercise 2: Analysis of Airport efficiency with Map Reduce

1. First a directory is created named “airport_efficiency” using command: hadoop fs -mkdir /airport_efficiency
2. Then csv file is put in the folder using command: hadoop fs -put D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\T_ONTIME_REPORTING.csv /airport_efficiency
3. Now we will use custom mapper and reducer functions
4. The output of the program is stored in /airport_efficiency/output2 and from there we can see output at the URL “http://localhost:9870/explorer.html#/airport_efficiency/output”



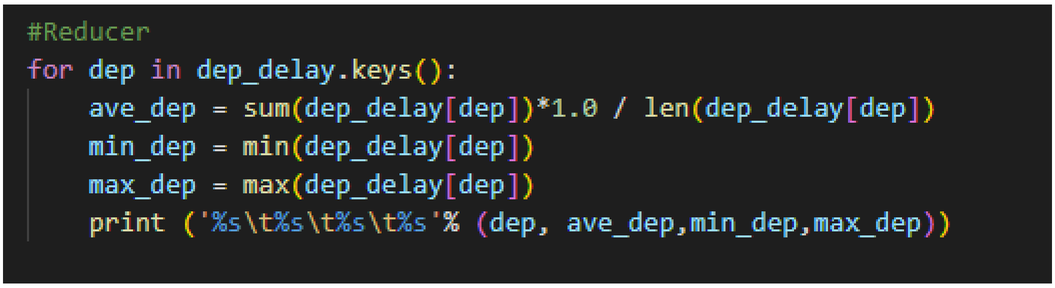
```
C:\WINDOWS\system32>hadoop fs -mkdir /airport_efficiency
```



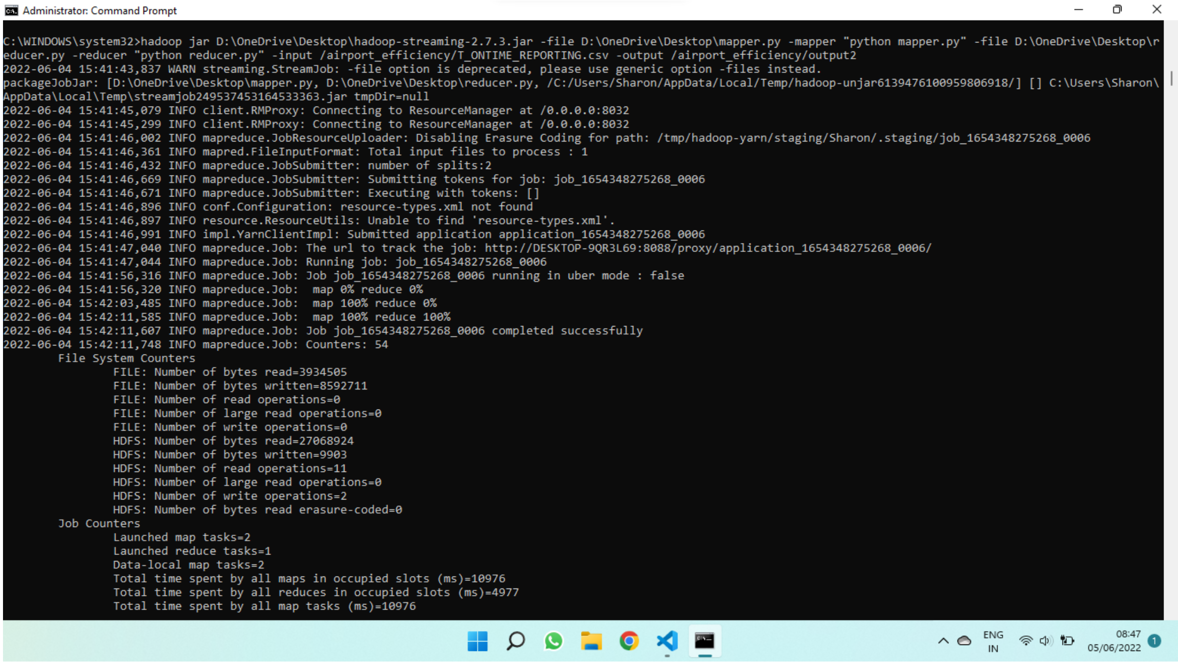
```
C:\WINDOWS\system32>hadoop fs -put "D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\T_ONTIME_REPORTING.csv" /airport_efficiency
```

1. Computing the maximum, minimum, and average departure delay for each airport

- For the mapper function each line is read from the csv file and split into tokens.
- Attributes Origin and Departure Delay is read and printed to the reducer.
- The reducer takes this input and checks whether departure delay has a value.
- Dictionary dep_delay stores origin as the key and delay as the value. This ensures unique airports.
- The key of the dictionary dep_delay is iterated and average, min and max values are found and printed on the final output console.

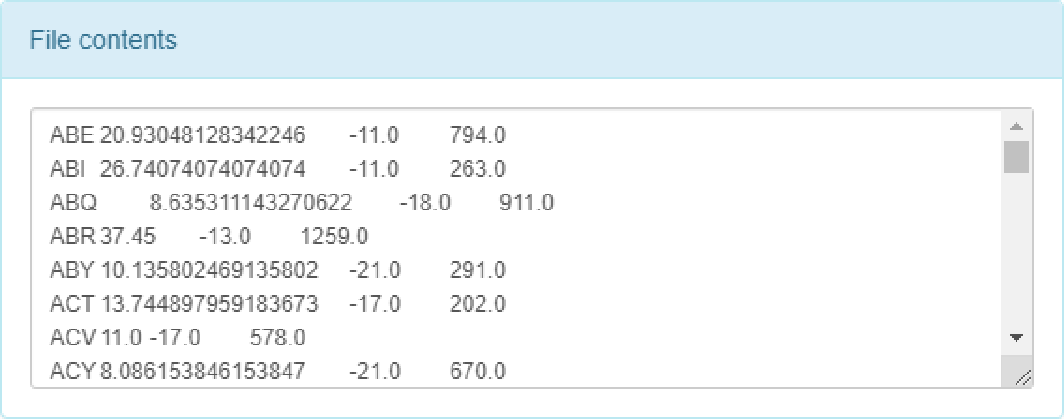


```
#Reducer
for dep in dep_delay.keys():
    ave_dep = sum(dep_delay[dep])*1.0 / len(dep_delay[dep])
    min_dep = min(dep_delay[dep])
    max_dep = max(dep_delay[dep])
    print ('%s\t%s\t%s\t%s'% (dep, ave_dep,min_dep,max_dep))
```



```
C:\WINDOWS\system32>hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\mapper.py -mapper "python mapper.py" -file D:\OneDrive\Desktop\r
educer.py -reducer "python reducer.py" -input /airport_efficiency/T_ONTIME_REPORTING.csv -output /airport_efficiency/output2
2022-06-04 15:41:43,837 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [D:\OneDrive\Desktop\mapper.py, D:\OneDrive\Desktop\reducer.py, /C:/Users/Sharon/AppData/Local/Temp/hadoop-unjar6139476100959806918/] [] C:\Users\Sharon\
AppData\Local\Temp\streamjob2495374531645333363.jar tmpDir=null
2022-06-04 15:41:45,079 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-04 15:41:45,299 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-04 15:41:46,002 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654348275268_0006
2022-06-04 15:41:46,361 INFO mapred.FileInputFormat: Total input files to process : 1
2022-06-04 15:41:46,432 INFO mapreduce.JobSubmitter: number of splits:2
2022-06-04 15:41:46,669 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654348275268_0006
2022-06-04 15:41:46,671 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-04 15:41:46,896 INFO conf.Configuration: resource-types.xml not found
2022-06-04 15:41:46,897 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-06-04 15:41:46,991 INFO impl.YarnClientImpl: Submitted application application_1654348275268_0006
2022-06-04 15:41:47,040 INFO mapreduce.Job: The url to track the job: http://DESKTOP-9QR3L69:8088/proxy/application_1654348275268_0006/
2022-06-04 15:41:47,044 INFO mapreduce.Job: Running job: job_1654348275268_0006
2022-06-04 15:41:56,316 INFO mapreduce.Job: Job job_1654348275268_0006 running in uber mode : false
2022-06-04 15:41:56,320 INFO mapreduce.Job:  map 0% reduce 0%
2022-06-04 15:42:03,485 INFO mapreduce.Job:  map 100% reduce 0%
2022-06-04 15:42:11,585 INFO mapreduce.Job:  map 100% reduce 100%
2022-06-04 15:42:11,607 INFO mapreduce.Job: Job job_1654348275268_0006 completed successfully
2022-06-04 15:42:11,748 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=3934505
                FILE: Number of bytes written=8592711
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=27068924
                HDFS: Number of bytes written=9903
                HDFS: Number of read operations=11
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=10976
                Total time spent by all reduces in occupied slots (ms)=4977
                Total time spent by all map tasks (ms)=10976
```

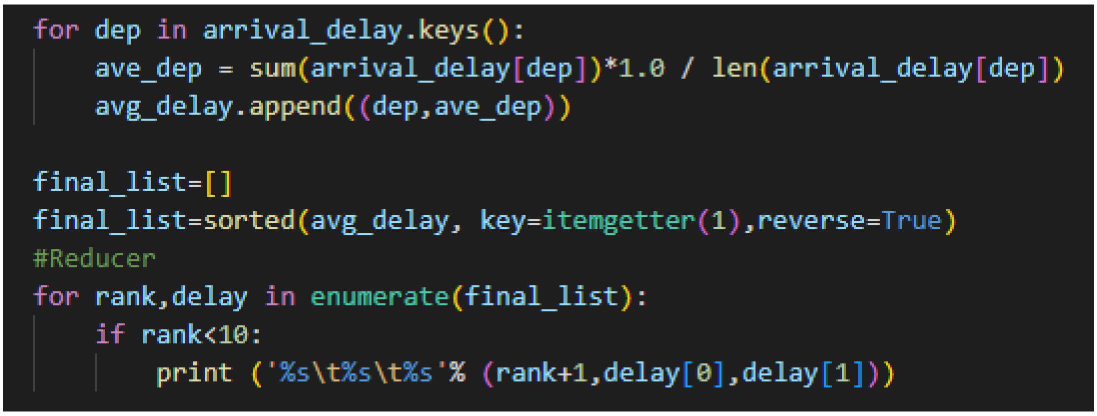
Airport, Average, Minimum, Maximum


File contents

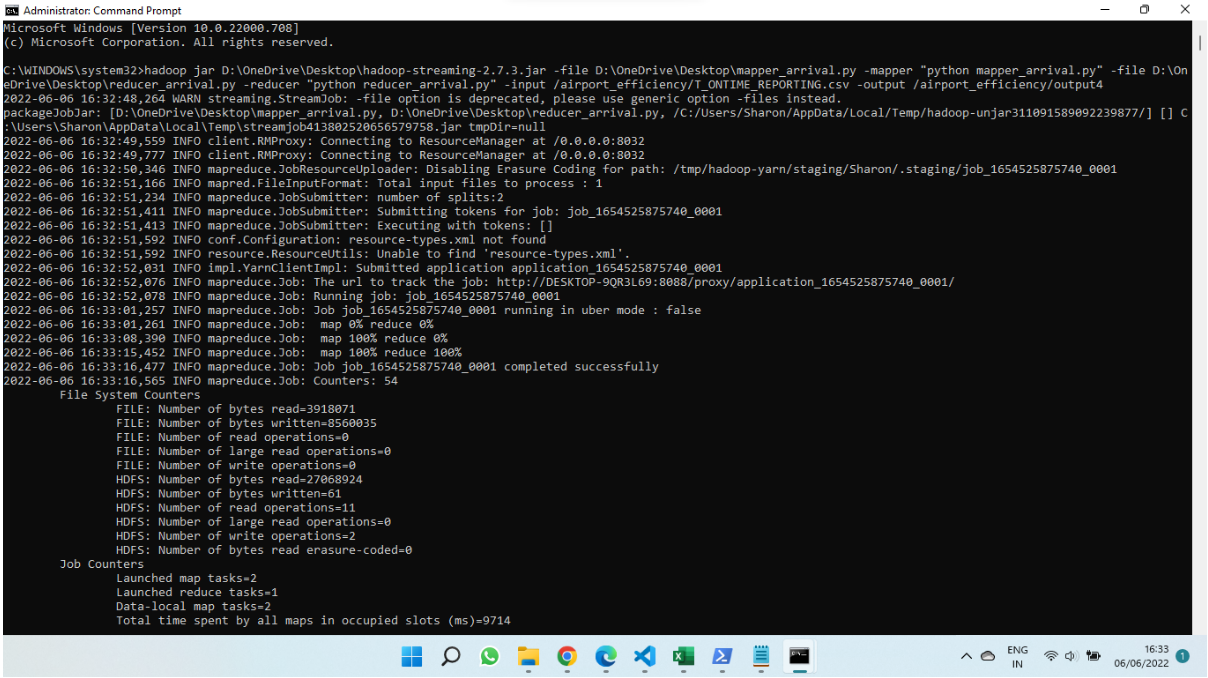
```
ABE 20.930481283422246      -11.0       794.0
ABI  26.74074074074074      -11.0       263.0
ABQ         8.635311143270622       -18.0       911.0
ABR 37.45       -13.0       1259.0
ABY 10.135802469135802      -21.0       291.0
ACT 13.744897959183673      -17.0       202.0
ACV 11.0 -17.0       578.0
ACY 8.086153846153847       -21.0       670.0
```

2. Computing a ranking list that contains top 10 airports by their average Arrival delay.

- For the mapper function each line is read from the csv file and split into tokens.
- Attributes Destination and Arrival Delay is read and printed to the reducer.
- The reducer takes this input and checks whether Arrival delay column has a value.
- Dictionary arrival_delay stores destination as the key and delay as the value. This ensures unique airports.
- The key of the dictionary arrival_delay is iterated and average values are found.
- Finally, the average values are sorted in reverse order and the top 10 airports are printed along with their ranks.

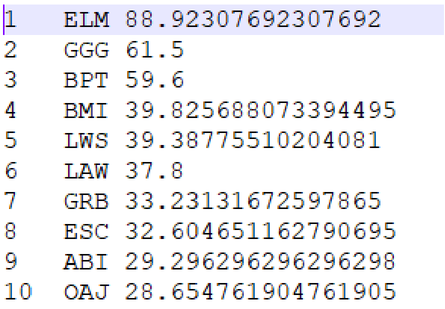


```python
for dep in arrival_delay.keys():
    ave_dep = sum(arrival_delay[dep])*1.0 / len(arrival_delay[dep])
    avg_delay.append((dep,ave_dep))

final_list=[]
final_list=sorted(avg_delay, key=itemgetter(1),reverse=True)
#Reducer
for rank,delay in enumerate(final_list):
    if rank<10:
        print ('%s\t%s\t%s'% (rank+1,delay[0],delay[1]))
```

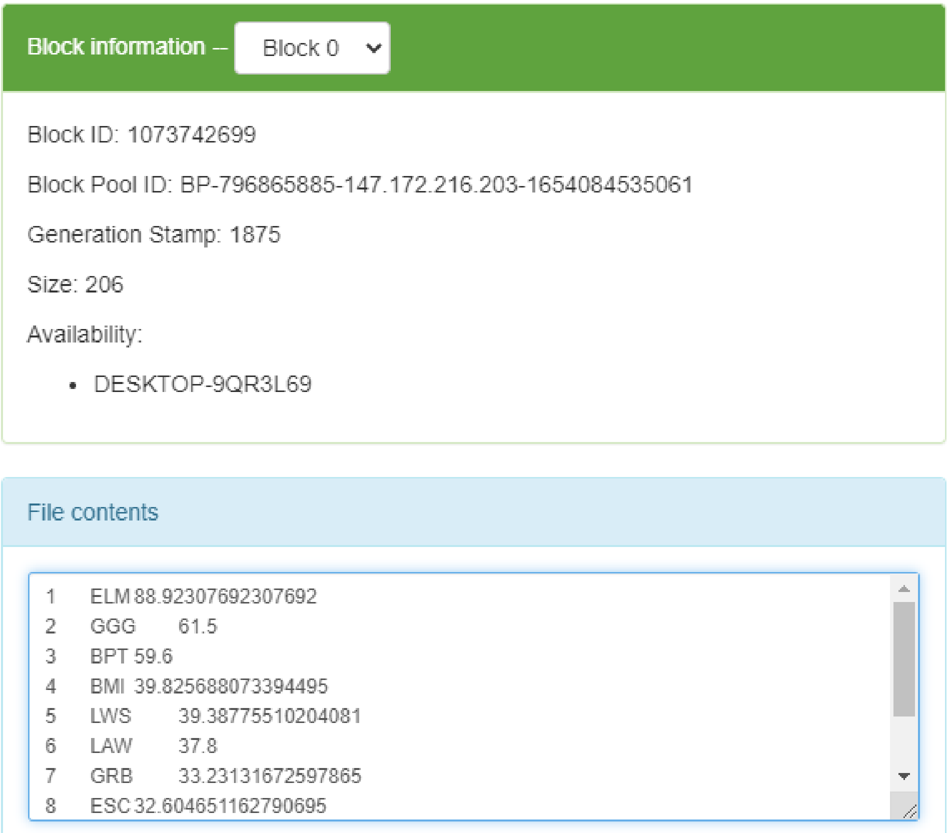


```
Microsoft Windows [Version 10.0.22000.708]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\mapper_arrival.py -mapper "python mapper_arrival.py" -file D:\On
eDrive\Desktop\reducer_arrival.py -reducer "python reducer_arrival.py" -input /airport_efficiency/T_ONTIME_REPORTING.csv -output /airport_efficiency/output4
2022-06-06 16:32:48,264 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [D:\OneDrive\Desktop\mapper_arrival.py, D:\OneDrive\Desktop\reducer_arrival.py, /C:/Users/Sharon/AppData/Local/Temp/hadoop-unjar311091589092239877/] [] C
:\Users\Sharon\AppData\Local\Temp\streamjob413802520656579758.jar tmpDir=null
2022-06-06 16:32:49,559 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-06 16:32:49,777 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-06 16:32:50,346 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654525875740_0001
2022-06-06 16:32:51,166 INFO mapred.FileInputFormat: Total input files to process : 1
2022-06-06 16:32:51,234 INFO mapreduce.JobSubmitter: number of splits:2
2022-06-06 16:32:51,411 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654525875740_0001
2022-06-06 16:32:51,413 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-06 16:32:51,592 INFO conf.Configuration: resource-types.xml not found
2022-06-06 16:32:51,592 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-06-06 16:32:52,031 INFO impl.YarnClientImpl: Submitted application application_1654525875740_0001
2022-06-06 16:32:52,076 INFO mapreduce.Job: The url to track the job: http://DESKTOP-9QR3L69:8088/proxy/application_1654525875740_0001/
2022-06-06 16:32:52,078 INFO mapreduce.Job: Running job: job_1654525875740_0001
2022-06-06 16:33:01,257 INFO mapreduce.Job: Job job_1654525875740_0001 running in uber mode : false
2022-06-06 16:33:01,261 INFO mapreduce.Job:  map 0% reduce 0%
2022-06-06 16:33:08,390 INFO mapreduce.Job:  map 100% reduce 0%
2022-06-06 16:33:15,452 INFO mapreduce.Job:  map 100% reduce 100%
2022-06-06 16:33:16,477 INFO mapreduce.Job: Job job_1654525875740_0001 completed successfully
2022-06-06 16:33:16,565 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=3918071
                FILE: Number of bytes written=8560035
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=27068924
                HDFS: Number of bytes written=61
                HDFS: Number of read operations=11
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=9714
```

Output: Ranking of Airports



```
1    ELM 88.92307692307692
2    GGG 61.5
3    BPT 59.6
4    BMI 39.825688073394495
5    LWS 39.38775510204081
6    LAW 37.8
7    GRB 33.23131672597865
8    ESC 32.604651162790695
9    ABI 29.296296296296298
10   OAJ 28.654761904761905
```



Block information – Block 0

Block ID: 1073742699

Block Pool ID: BP-796865885-147.172.216.203-1654084535061

Generation Stamp: 1875

Size: 206

Availability:

- DESKTOP-9QR3L69

File contents

```
ELM 88.92307692307692
GGG     61.5
BPT 59.6
BMI  39.825688073394495
LWS     39.38775510204081
LAW     37.8
GRB     33.23131672597865
ESC 32.604651162790695
```

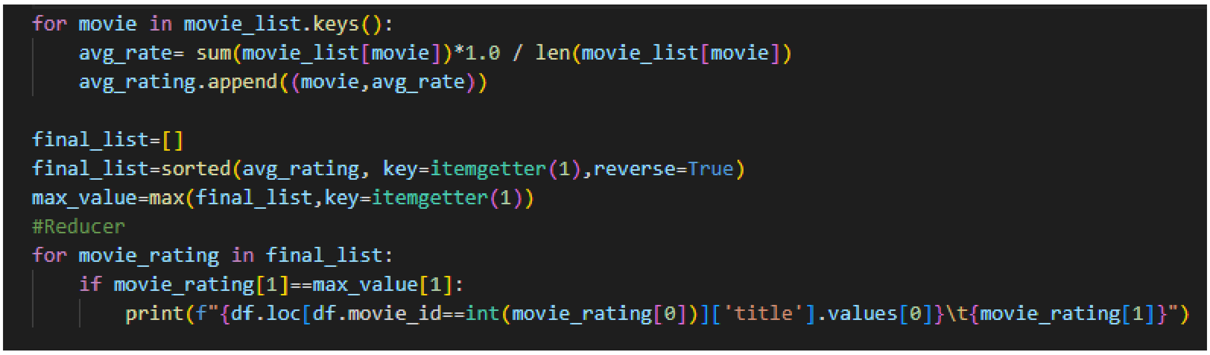
Exercise 3: Analysis of Movie dataset using Map and Reduce

ratings.dat - UserID::MovieID::Rating::Timestamp

movies.dat - MovieID::Title::Genres

1. Find the movie title which has the maximum average rating?

- For the mapper function each line is read from the ratings.dat file and split into user_id,movie_id,rating,timestamp. Movie_Id and rating is printed to the reducer.
- The reducer takes this input and checks whether each movie has a rating
- Dictionary movie_list stores movie as the key and rating as the value. This ensures unique movies.
- The dictionary movie_list is iterated based on key: “movies” and average values are found.
- Finally, the average values are sorted in reverse order and movie with max average rating is found out.
- The title of the movie is found from the “movies.dat” file and movies with max avg rating is printed.

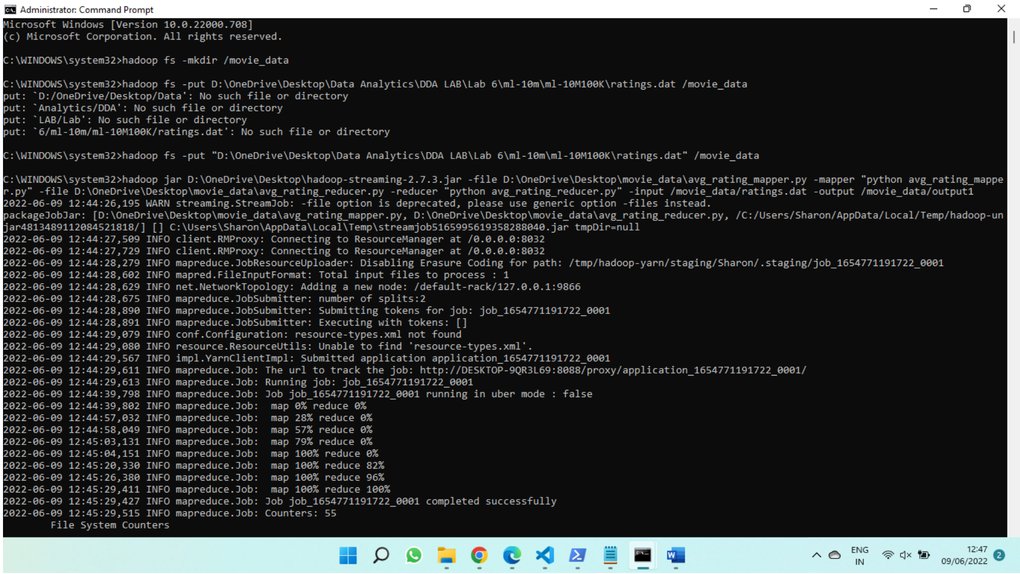


```
for movie in movie_list.keys():
    avg_rate= sum(movie_list[movie])*1.0 / len(movie_list[movie])
    avg_rating.append((movie,avg_rate))

final_list=[]
final_list=sorted(avg_rating, key=itemgetter(1),reverse=True)
max_value=max(final_list,key=itemgetter(1))
#Reducer
for movie_rating in final_list:
    if movie_rating[1]==max_value[1]:
        print(f"{df.loc[df.movie_id==int(movie_rating[0])]['title'].values[0]}\t{movie_rating[1]}")
```

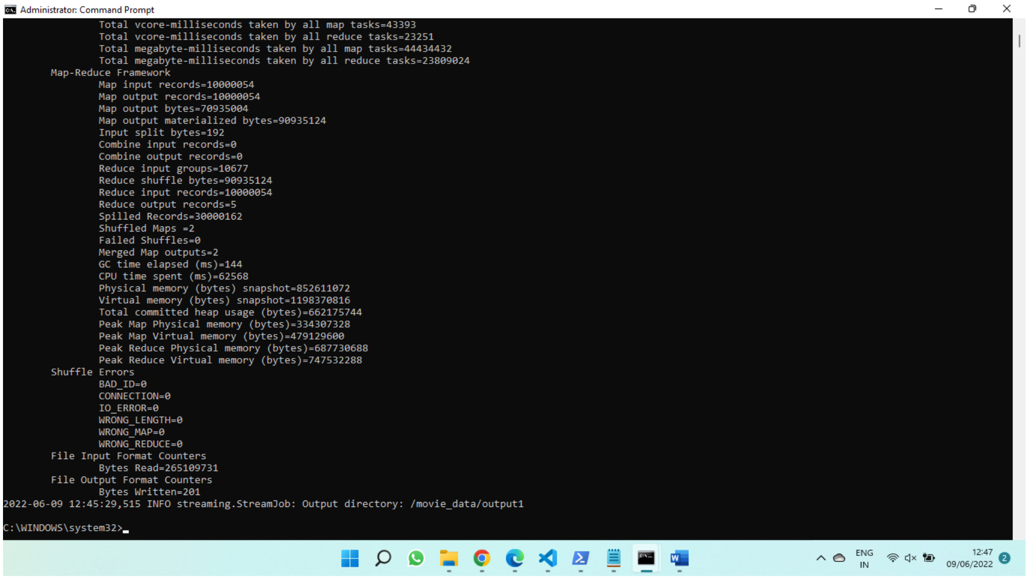
Commands Used:

hadoop fs -put D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\ml-10m\ml-10M100K\ratings.dat /movie_data

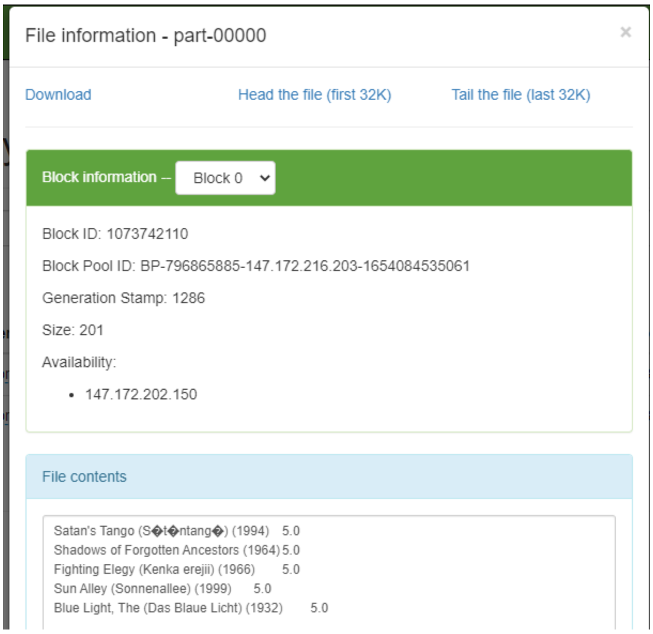
hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\avg_rating_mapper.py -mapper "python avg_rating_mapper.py" -file D:\OneDrive\Desktop\movie_data\avg_rating_reducer.py -reducer "python avg_rating_reducer.py" -input /movie_data/ratings.dat -output /movie_data/output1



```
Microsoft Windows [Version 10.0.22000.708]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>hadoop fs -mkdir /movie_data

C:\WINDOWS\system32>hadoop fs -put D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\ml-10m\ml-10M100K\ratings.dat /movie_data
put: `D:/OneDrive/Desktop/Data': No such file or directory
put: `Analytics/DDA': No such file or directory
put: `LAB/Lab': No such file or directory
put: `6/ml-10m/ml-10M100K/ratings.dat': No such file or directory

C:\WINDOWS\system32>hadoop fs -put "D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\ml-10m\ml-10M100K\ratings.dat" /movie_data

C:\WINDOWS\system32>hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\avg_rating_mapper.py -mapper "python avg_rating_mappe
r.py" -file D:\OneDrive\Desktop\movie_data\avg_rating_reducer.py -reducer "python avg_rating_reducer.py" -input /movie_data/ratings.dat -output /movie_data/output1
2022-06-09 12:44:26,195 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [D:\OneDrive\Desktop\movie_data\avg_rating_mapper.py, D:\OneDrive\Desktop\movie_data\avg_rating_reducer.py, /C:/Users/Sharon/AppData/Local/Temp/hadoop-un
jar4813489112084521818/] [] C:\Users\Sharon\AppData\Local\Temp\streamjob5165995619358288040.jar tmpDir=null
2022-06-09 12:44:27,509 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 12:44:27,729 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 12:44:28,279 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654771191722_0001
2022-06-09 12:44:28,602 INFO mapred.FileInputFormat: Total input files to process : 1
2022-06-09 12:44:28,629 INFO net.NetworkTopology: Adding a new node: /default-rack/127.0.0.1:9866
2022-06-09 12:44:28,675 INFO mapreduce.JobSubmitter: number of splits:2
2022-06-09 12:44:28,890 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654771191722_0001
2022-06-09 12:44:28,891 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-09 12:44:29,079 INFO conf.Configuration: resource-types.xml not found
2022-06-09 12:44:29,080 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-06-09 12:44:29,567 INFO impl.YarnClientImpl: Submitted application application_1654771191722_0001
2022-06-09 12:44:29,611 INFO mapreduce.Job: The url to track the job: http://DESKTOP-9QR3L69:8088/proxy/application_1654771191722_0001/
2022-06-09 12:44:29,613 INFO mapreduce.Job: Running job: job_1654771191722_0001
2022-06-09 12:44:39,798 INFO mapreduce.Job: Job job_1654771191722_0001 running in uber mode : false
2022-06-09 12:44:39,802 INFO mapreduce.Job:  map 0% reduce 0%
2022-06-09 12:44:57,032 INFO mapreduce.Job:  map 28% reduce 0%
2022-06-09 12:44:58,049 INFO mapreduce.Job:  map 57% reduce 0%
2022-06-09 12:45:03,131 INFO mapreduce.Job:  map 79% reduce 0%
2022-06-09 12:45:04,151 INFO mapreduce.Job:  map 100% reduce 0%
2022-06-09 12:45:20,330 INFO mapreduce.Job:  map 100% reduce 82%
2022-06-09 12:45:26,380 INFO mapreduce.Job:  map 100% reduce 96%
2022-06-09 12:45:29,411 INFO mapreduce.Job:  map 100% reduce 100%
2022-06-09 12:45:29,427 INFO mapreduce.Job: Job job_1654771191722_0001 completed successfully
2022-06-09 12:45:29,515 INFO mapreduce.Job: Counters: 55
        File System Counters
```



```
		Total vcore-milliseconds taken by all map tasks=43393
		Total vcore-milliseconds taken by all reduce tasks=23251
		Total megabyte-milliseconds taken by all map tasks=44434432
		Total megabyte-milliseconds taken by all reduce tasks=23809024
	Map-Reduce Framework
		Map input records=10000054
		Map output records=10000054
		Map output bytes=70935004
		Map output materialized bytes=90935124
		Input split bytes=192
		Combine input records=0
		Combine output records=0
		Reduce input groups=10677
		Reduce shuffle bytes=90935124
		Reduce input records=10000054
		Reduce output records=5
		Spilled Records=30000162
		Shuffled Maps =2
		Failed Shuffles=0
		Merged Map outputs=2
		GC time elapsed (ms)=144
		CPU time spent (ms)=62568
		Physical memory (bytes) snapshot=852611072
		Virtual memory (bytes) snapshot=1198370816
		Total committed heap usage (bytes)=662175744
		Peak Map Physical memory (bytes)=334307328
		Peak Map Virtual memory (bytes)=479129600
		Peak Reduce Physical memory (bytes)=687730688
		Peak Reduce Virtual memory (bytes)=747532288
	Shuffle Errors
		BAD_ID=0
		CONNECTION=0
		IO_ERROR=0
		WRONG_LENGTH=0
		WRONG_MAP=0
		WRONG_REDUCE=0
	File Input Format Counters
		Bytes Read=265109731
	File Output Format Counters
		Bytes Written=201
2022-06-09 12:45:29,515 INFO streaming.StreamJob: Output directory: /movie_data/output1

C:\WINDOWS\system32>_
```



File information - part-00000

Block ID: 1073742110

Block Pool ID: BP-796865885-147.172.216.203-1654084535061

Generation Stamp: 1286

Size: 201

Availability:

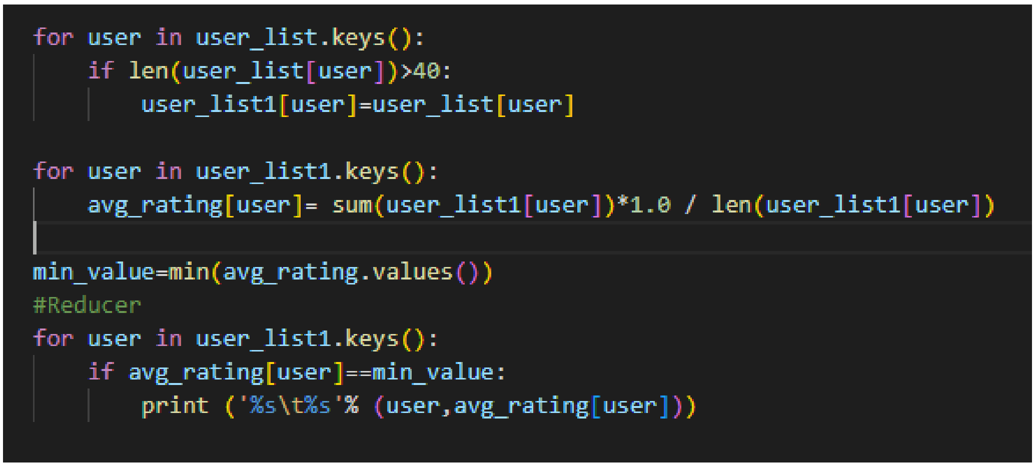
- 147.172.202.150

File contents

```
Satan's Tango (S�t�ntang�) (1994)	5.0
Shadows of Forgotten Ancestors (1964)	5.0
Fighting Elegy (Kenka erejii) (1966)	5.0
Sun Alley (Sonnenallee) (1999)	5.0
Blue Light, The (Das Blaue Licht) (1932)	5.0
```

2. Find the user who has assign lowest average rating among all the users who rated more than 40 times?

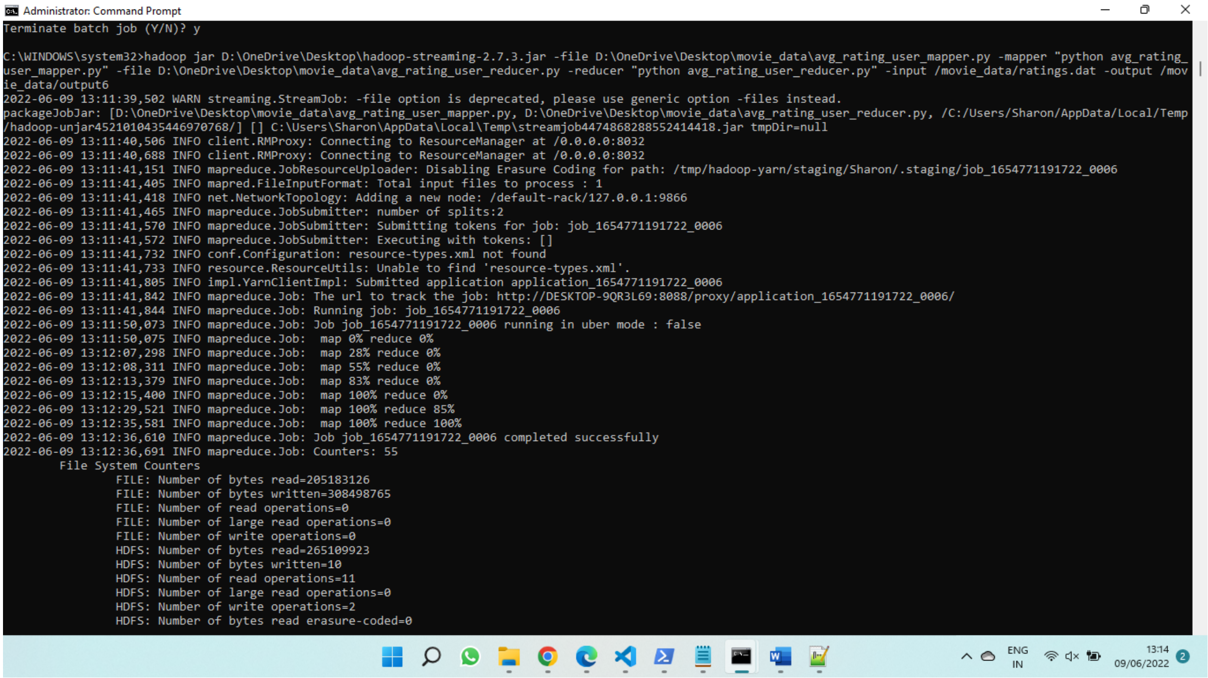
- For the mapper function each line is read from the ratings.dat file and split into user_id,movie_id,rating,timestamp. user_id and rating is printed to the reducer.
- The reducer takes this input and checks whether each user has a rating
- Dictionary user_list stores users as the key and rating as the value. This ensures unique users.
- The dictionary user_list is iterated based on key: “user” and users who have rated more than 40 times are found.
- Then, the average values are found from user_list1 and min value is computed.
- Finally, user who has assign lowest average rating is printed as the output



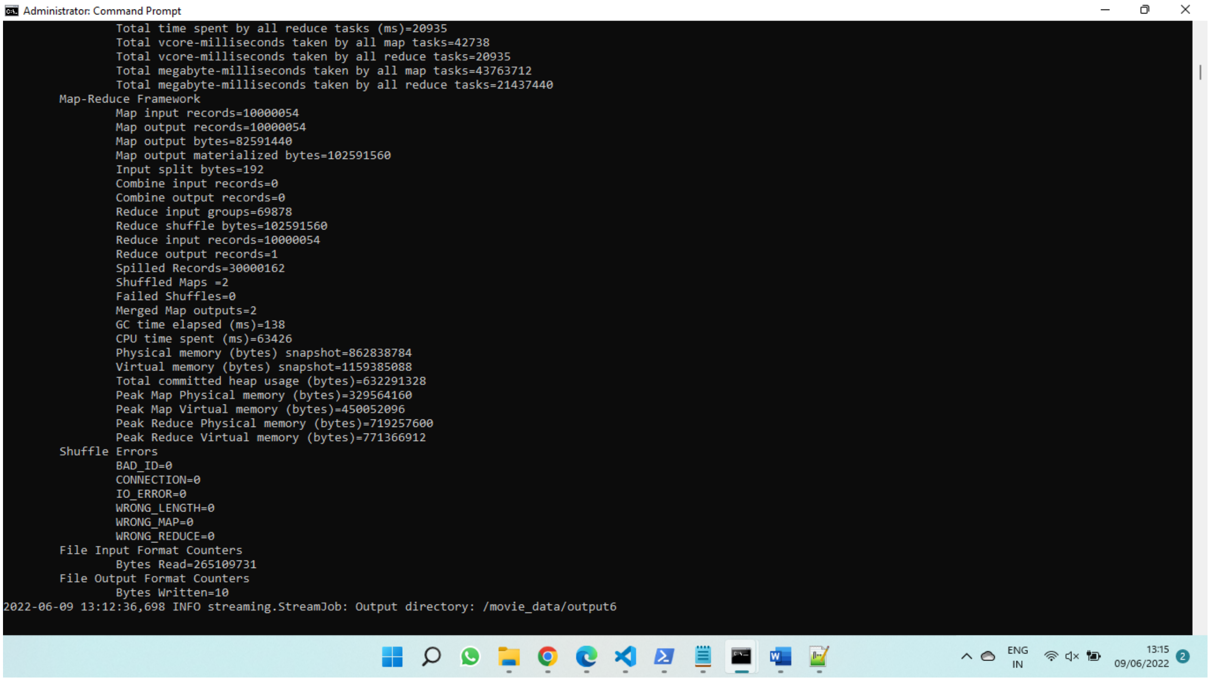
```python
for user in user_list.keys():
    if len(user_list[user])>40:
        user_list1[user]=user_list[user]

for user in user_list1.keys():
    avg_rating[user]= sum(user_list1[user])*1.0 / len(user_list1[user])

min_value=min(avg_rating.values())
#Reducer
for user in user_list1.keys():
    if avg_rating[user]==min_value:
        print ('%s\t%s'% (user,avg_rating[user]))
```

Commands Used:

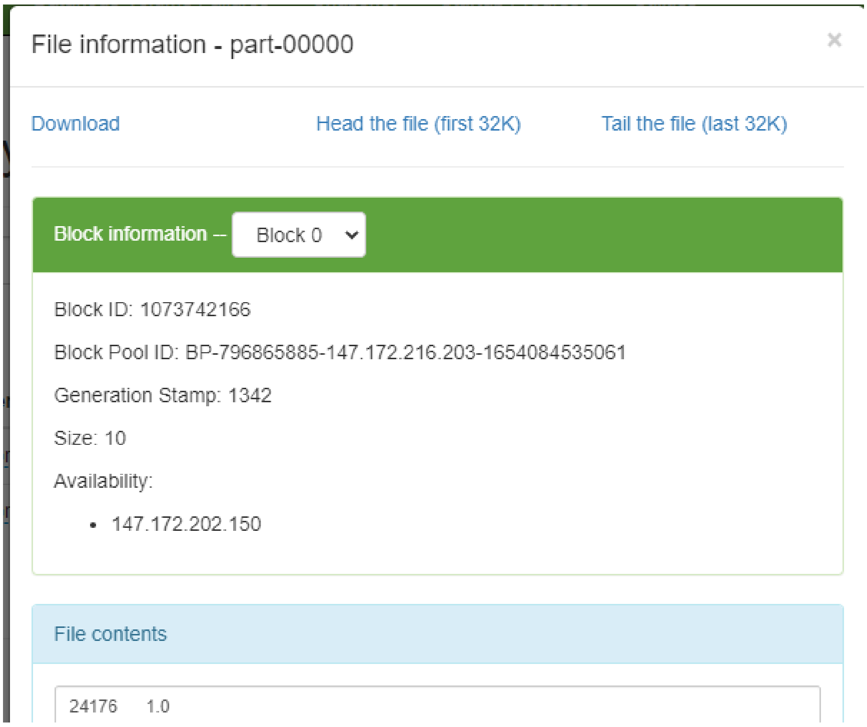
hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\avg_rating_user_mapper.py -mapper "python avg_rating_user_mapper.py" -file D:\OneDrive\Desktop\movie_data\avg_rating_user_reducer.py -reducer "python avg_rating_user_reducer.py" -input /movie_data/ratings.dat -output /movie_data/output2



```
Terminate batch job (Y/N)? y

C:\WINDOWS\system32>hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\avg_rating_user_mapper.py -mapper "python avg_rating_
user_mapper.py" -file D:\OneDrive\Desktop\movie_data\avg_rating_user_reducer.py -reducer "python avg_rating_user_reducer.py" -input /movie_data/ratings.dat -output /mov
ie_data/output6
2022-06-09 13:11:39,502 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [D:\OneDrive\Desktop\movie_data\avg_rating_user_mapper.py, D:\OneDrive\Desktop\movie_data\avg_rating_user_reducer.py, /C:/Users/Sharon/AppData/Local/Temp
/hadoop-unjar4521010435446970768/] [] C:\Users\Sharon\AppData\Local\Temp\streamjob4474868288552414418.jar tmpDir=null
2022-06-09 13:11:40,506 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 13:11:40,688 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 13:11:41,151 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654771191722_0006
2022-06-09 13:11:41,405 INFO mapred.FileInputFormat: Total input files to process : 1
2022-06-09 13:11:41,418 INFO net.NetworkTopology: Adding a new node: /default-rack/127.0.0.1:9866
2022-06-09 13:11:41,465 INFO mapreduce.JobSubmitter: number of splits:2
2022-06-09 13:11:41,570 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654771191722_0006
2022-06-09 13:11:41,572 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-09 13:11:41,732 INFO conf.Configuration: resource-types.xml not found
2022-06-09 13:11:41,733 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-06-09 13:11:41,805 INFO impl.YarnClientImpl: Submitted application application_1654771191722_0006
2022-06-09 13:11:41,842 INFO mapreduce.Job: The url to track the job: http://DESKTOP-9QR3L69:8088/proxy/application_1654771191722_0006/
2022-06-09 13:11:41,844 INFO mapreduce.Job: Running job: job_1654771191722_0006
2022-06-09 13:11:50,073 INFO mapreduce.Job: Job job_1654771191722_0006 running in uber mode : false
2022-06-09 13:11:50,075 INFO mapreduce.Job:  map 0% reduce 0%
2022-06-09 13:12:07,298 INFO mapreduce.Job:  map 28% reduce 0%
2022-06-09 13:12:08,311 INFO mapreduce.Job:  map 55% reduce 0%
2022-06-09 13:12:13,379 INFO mapreduce.Job:  map 83% reduce 0%
2022-06-09 13:12:15,400 INFO mapreduce.Job:  map 100% reduce 0%
2022-06-09 13:12:29,521 INFO mapreduce.Job:  map 100% reduce 85%
2022-06-09 13:12:35,581 INFO mapreduce.Job:  map 100% reduce 100%
2022-06-09 13:12:36,610 INFO mapreduce.Job: Job job_1654771191722_0006 completed successfully
2022-06-09 13:12:36,691 INFO mapreduce.Job: Counters: 55
        File System Counters
                FILE: Number of bytes read=205183126
                FILE: Number of bytes written=308498765
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=265109923
                HDFS: Number of bytes written=10
                HDFS: Number of read operations=11
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
```



```
                Total time spent by all reduce tasks (ms)=20935
                Total vcore-milliseconds taken by all map tasks=42738
                Total vcore-milliseconds taken by all reduce tasks=20935
                Total megabyte-milliseconds taken by all map tasks=43763712
                Total megabyte-milliseconds taken by all reduce tasks=21437440
        Map-Reduce Framework
                Map input records=10000054
                Map output records=10000054
                Map output bytes=82591440
                Map output materialized bytes=102591560
                Input split bytes=192
                Combine input records=0
                Combine output records=0
                Reduce input groups=69878
                Reduce shuffle bytes=102591560
                Reduce input records=10000054
                Reduce output records=1
                Spilled Records=30000162
                Shuffled Maps =2
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=138
                CPU time spent (ms)=63426
                Physical memory (bytes) snapshot=862838784
                Virtual memory (bytes) snapshot=1159385088
                Total committed heap usage (bytes)=632291328
                Peak Map Physical memory (bytes)=329564160
                Peak Map Virtual memory (bytes)=450052096
                Peak Reduce Physical memory (bytes)=719257600
                Peak Reduce Virtual memory (bytes)=771366912
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=265109731
        File Output Format Counters
                Bytes Written=10
2022-06-09 13:12:36,698 INFO streaming.StreamJob: Output directory: /movie_data/output6
```



File information - part-00000

Block ID: 1073742166

Block Pool ID: BP-796865885-147.172.216.203-1654084535061

Generation Stamp: 1342

Size: 10

Availability:

- 147.172.202.150

File contents

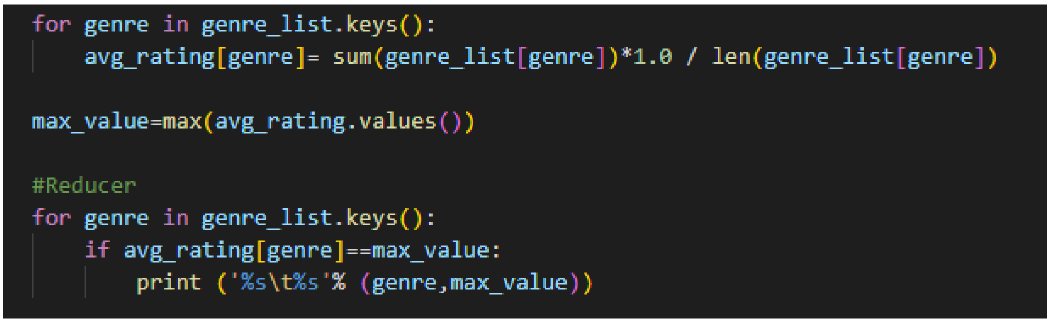
```
24176	1.0
```

User 24176 assigns the lowest average rating of 1 among all the users.

3. Find the highest average rated movie genre?

- As highest average rated genre is to be found out, preprocessing step needs to be carried out.
- In the preprocessing step, movies.dat and ratings.dat is combined and attributes “genre”,”rating” is selected from the dataframe. This is written to genre.csv file

- For the mapper function each line is read from the “genre.csv” file and split into genre,rating which is printed to the reducer.
- The reducer takes this input and checks whether each genre has a rating
- Dictionary genre_list stores genre as the key and rating as the value. This ensures unique genres.
- The dictionary genre_list is iterated based on key: “genre” and average values are found.
- Then, from the average values the maximum value is computed.
- Finally, genre which has highest average rating is given as the output

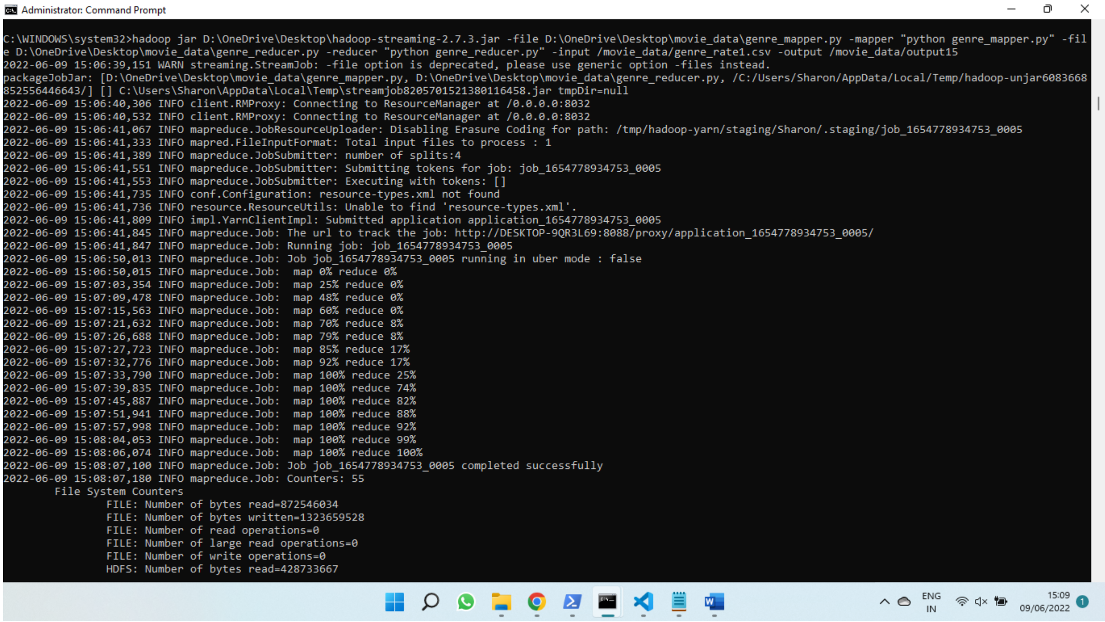


```
for genre in genre_list.keys():
    avg_rating[genre]= sum(genre_list[genre])*1.0 / len(genre_list[genre])

max_value=max(avg_rating.values())

#Reducer
for genre in genre_list.keys():
    if avg_rating[genre]==max_value:
        print ('%s\t%s'% (genre,max_value))
```

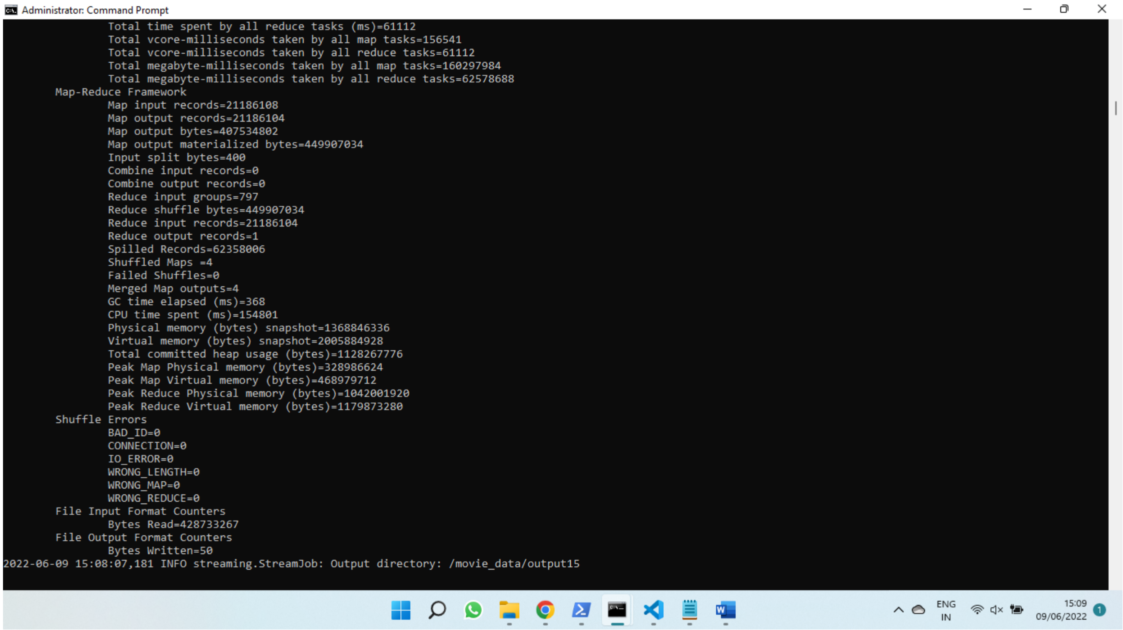
Commands Used:

hadoop fs -put "D:\OneDrive\Desktop\Data Analytics\DDA LAB\Lab 6\genre_rate1.csv" /movie_data

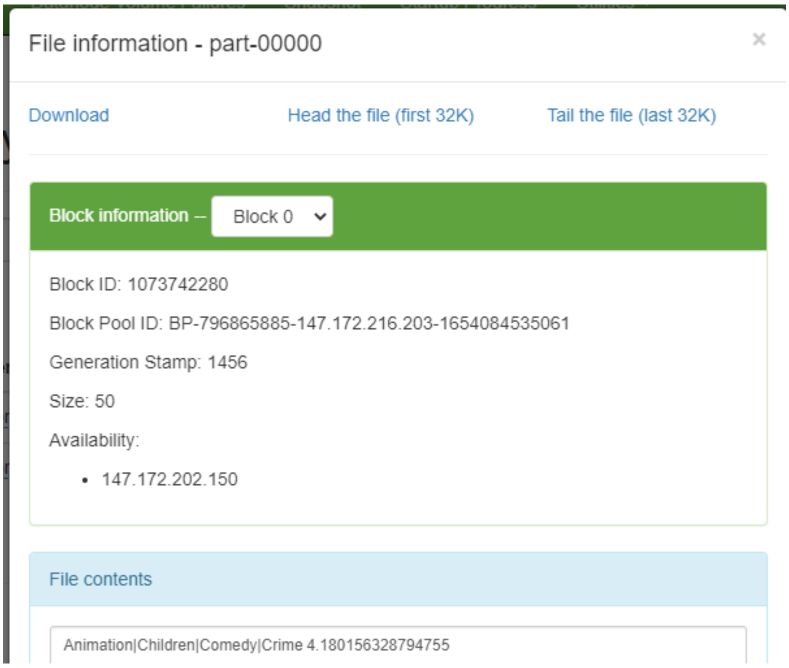
hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\genre_mapper.py -mapper "python genre_mapper.py" -file D:\OneDrive\Desktop\movie_data\genre_reducer.py -reducer "python genre_reducer.py" -input /movie_data/genre_rate1.csv -output /movie_data/output15



```
C:\WINDOWS\system32>hadoop jar D:\OneDrive\Desktop\hadoop-streaming-2.7.3.jar -file D:\OneDrive\Desktop\movie_data\genre_mapper.py -mapper "python genre_mapper.py" -fil
e D:\OneDrive\Desktop\movie_data\genre_reducer.py -reducer "python genre_reducer.py" -input /movie_data/genre_rate1.csv -output /movie_data/output15
2022-06-09 15:06:39,151 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [D:\OneDrive\Desktop\movie_data\genre_mapper.py, D:\OneDrive\Desktop\movie_data\genre_reducer.py, /C:/Users/Sharon/AppData/Local/Temp/hadoop-unjar6083668
852556446643/] [] C:\Users\Sharon\AppData\Local\Temp\streamjob8205701521380116458.jar tmpDir=null
2022-06-09 15:06:40,306 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 15:06:40,532 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2022-06-09 15:06:41,067 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Sharon/.staging/job_1654778934753_0005
2022-06-09 15:06:41,333 INFO mapred.FileInputFormat: Total input files to process : 1
2022-06-09 15:06:41,389 INFO mapreduce.JobSubmitter: number of splits:4
2022-06-09 15:06:41,551 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654778934753_0005
2022-06-09 15:06:41,553 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-06-09 15:06:41,735 INFO conf.Configuration: resource-types.xml not found
2022-06-09 15:06:41,736 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-06-09 15:06:41,809 INFO impl.YarnClientImpl: Submitted application application_1654778934753_0005
2022-06-09 15:06:41,845 INFO mapreduce.Job: The url to track the job: http://DESKTOP-9QR3L69:8088/proxy/application_1654778934753_0005/
2022-06-09 15:06:41,847 INFO mapreduce.Job: Running job: job_1654778934753_0005
2022-06-09 15:06:50,013 INFO mapreduce.Job: Job job_1654778934753_0005 running in uber mode : false
2022-06-09 15:06:50,015 INFO mapreduce.Job:  map 0% reduce 0%
2022-06-09 15:07:03,354 INFO mapreduce.Job:  map 25% reduce 0%
2022-06-09 15:07:09,478 INFO mapreduce.Job:  map 48% reduce 0%
2022-06-09 15:07:15,563 INFO mapreduce.Job:  map 60% reduce 0%
2022-06-09 15:07:21,632 INFO mapreduce.Job:  map 70% reduce 8%
2022-06-09 15:07:26,688 INFO mapreduce.Job:  map 79% reduce 8%
2022-06-09 15:07:27,723 INFO mapreduce.Job:  map 85% reduce 17%
2022-06-09 15:07:32,776 INFO mapreduce.Job:  map 92% reduce 17%
2022-06-09 15:07:33,790 INFO mapreduce.Job:  map 100% reduce 25%
2022-06-09 15:07:39,835 INFO mapreduce.Job:  map 100% reduce 74%
2022-06-09 15:07:45,887 INFO mapreduce.Job:  map 100% reduce 82%
2022-06-09 15:07:51,941 INFO mapreduce.Job:  map 100% reduce 88%
2022-06-09 15:07:57,998 INFO mapreduce.Job:  map 100% reduce 92%
2022-06-09 15:08:04,053 INFO mapreduce.Job:  map 100% reduce 99%
2022-06-09 15:08:06,074 INFO mapreduce.Job:  map 100% reduce 100%
2022-06-09 15:08:07,100 INFO mapreduce.Job: Job job_1654778934753_0005 completed successfully
2022-06-09 15:08:07,180 INFO mapreduce.Job: Counters: 55
        File System Counters
                FILE: Number of bytes read=872546034
                FILE: Number of bytes written=1323659528
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=428733667
```



```
                Total time spent by all reduce tasks (ms)=61112
                Total vcore-milliseconds taken by all map tasks=156541
                Total vcore-milliseconds taken by all reduce tasks=61112
                Total megabyte-milliseconds taken by all map tasks=160297984
                Total megabyte-milliseconds taken by all reduce tasks=62578688
        Map-Reduce Framework
                Map input records=21186108
                Map output records=21186104
                Map output bytes=407534802
                Map output materialized bytes=449907034
                Input split bytes=400
                Combine input records=0
                Combine output records=0
                Reduce input groups=797
                Reduce shuffle bytes=449907034
                Reduce input records=21186104
                Reduce output records=1
                Spilled Records=62358006
                Shuffled Maps =4
                Failed Shuffles=0
                Merged Map outputs=4
                GC time elapsed (ms)=368
                CPU time spent (ms)=154801
                Physical memory (bytes) snapshot=1368846336
                Virtual memory (bytes) snapshot=2005884928
                Total committed heap usage (bytes)=1128267776
                Peak Map Physical memory (bytes)=328986624
                Peak Map Virtual memory (bytes)=468979712
                Peak Reduce Physical memory (bytes)=1042001920
                Peak Reduce Virtual memory (bytes)=1179873280
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=428733267
        File Output Format Counters
                Bytes Written=50
2022-06-09 15:08:07,181 INFO streaming.StreamJob: Output directory: /movie_data/output15
```



File information - part-00000

Block ID: 1073742280

Block Pool ID: BP-796865885-147.172.216.203-1654084535061

Generation Stamp: 1456

Size: 50

Availability:

- 147.172.202.150

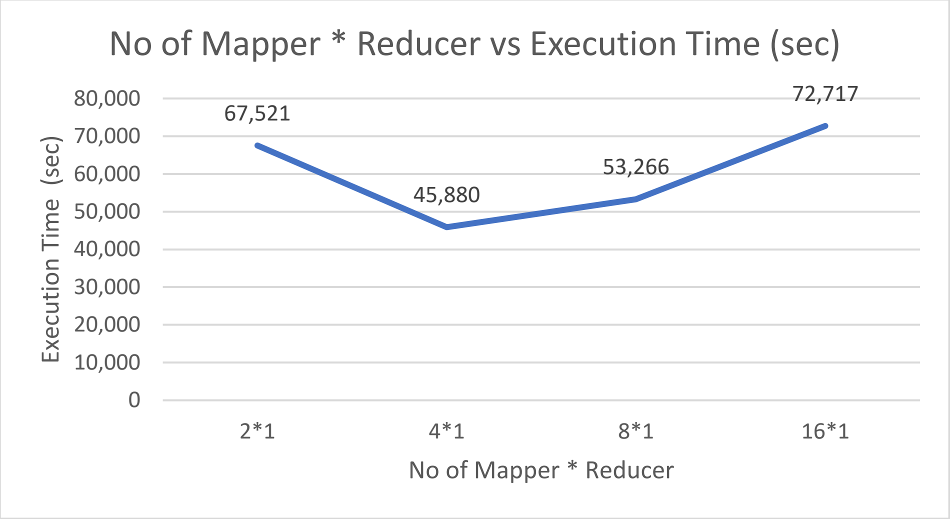
File contents

Animation|Children|Comedy|Crime 4.180156328794755

Task a: Performance Analysis

Mapper * Reducer =2*1=CPU time spent (ms)=67521
Mapper * Reducer =4*1= CPU time spent (ms)=45880
Mapper * Reducer =8*1= CPU time spent (ms)=53266
Mapper * Reducer =16*1=CPU time spent (ms)=72717

No of Mapper * Reducer	2*1	4*1	8*1	16*1
Time (sec)	67.521	45.880	53.266	72.717

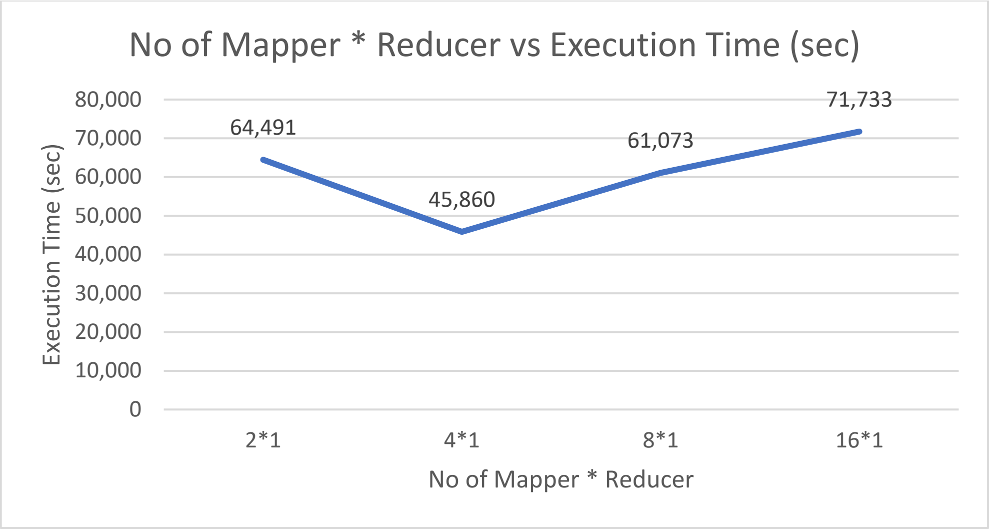


Performance Analysis: When the number of mapper * reducer is varied from 2*1 to 4*1, this tells us that as mappers are increased, number of splits in data gets increased. This leads to reduction of Execution time. It reduces from 67 sec to 45 sec. But as the number is increased more from 8 *1 to 16*1 , the time increases. This tells making more number of splits actually creates performance overhead, which lead to degradation of performance. Here time is indicated in sec and graph shows german numbering format

Task b: Performance Analysis

Mapper * Reducer =2*1=CPU time spent (ms)=64491
Mapper * Reducer =4*1=CPU time spent (ms)=45860
Mapper * Reducer =8*1=CPU time spent (ms)=61073
Mapper * Reducer =16*1=CPU time spent (ms)=71733

No of Mapper * Reducer	2*1	4*1	8*1	16*1
Time (sec)	64.491	45.860	61.073	71.733

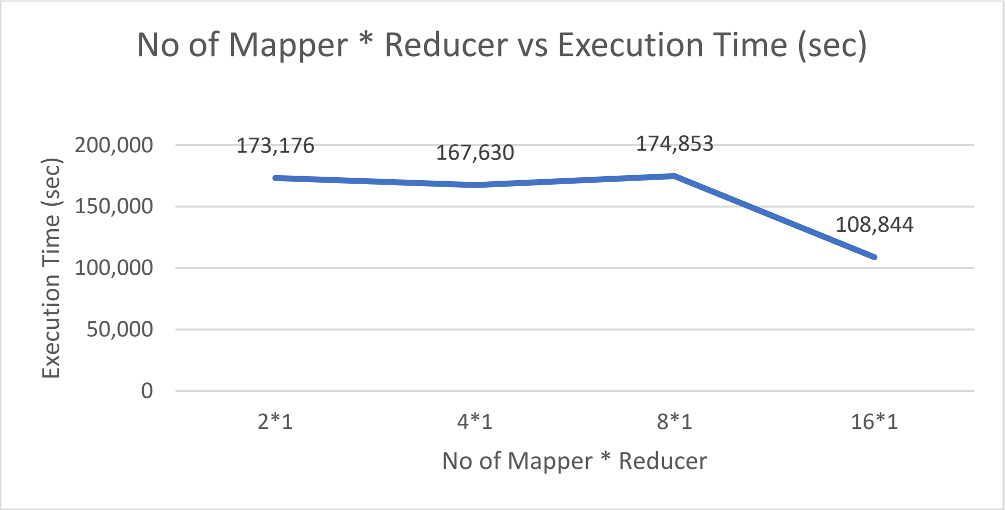


Performance Analysis: When the number of mapper * reducer is varied from 2*1 to 4*1, this tells us that as mappers are increased, number of splits in data gets increased. This leads to reduction of Execution time. But as the number is increased more from 8 *1 to 16*1 , the time increases. This tells making more number of splits actually creates performance overhead, which lead to degradation of performance.

Task c: Performance Analysis

Mapper * Reducer =2*1=CPU time spent (ms)=173176
Mapper * Reducer =4*1= CPU time spent (ms)=167630
Mapper * Reducer =8*1=CPU time spent (ms)=174853
Mapper * Reducer =16*1=CPU time spent (ms)=108844

No of Mapper * Reducer	**2*1**	**4*1**	**8*1**	**16*1**
Time (sec)	**173.176**	**167.630**	**174.853**	**108.844**



Performance Analysis: When the number of mapper * reducer is varied from 2*1 to 4*1, this tells us that as mappers are increased, number of splits in data gets increased. This leads to reduction of Execution time. As the number is increased more from 8 *1 to 16*1 , the time further decreases. As the dataset size is huge, having 16 mappers is advantageous. The performance increases with 16 mappers by a huge margin. Having a distributed architecture leads to speedup. Least time is with 16 mappers and 1 reducer which is 108 sec.

References:

https://www.geeksforgeeks.org/hadoop-streaming-using-python-word-count-problem/

https://stackoverflow.com/questions/47618867/how-to-run-python-mapreduce-in-hadoop-streaming

https://medium.com/@pedro.a.hdez.a/hadoop-3-2-2-installation-guide-for-windows-10-454f5b5c22d3#d3ca