



---

# BIG DATA ANALYTICS

## ASSIGNMENT-01

---

### Hadoop Installation/setup and Designing Additional Queries



Submitted by:

Name: Khuram Shahzad

Roll No: p218742

Subject: Big Data Analytics (MS DS)

Submitted to:

Dr. Omar Usman Khan

JUNE 27, 2022

FAST - NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES - PESHAWAR CAMPUS  
Jamrud Road 160 Industrial Estate Road, Peshawar - PK

# BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

## Table of Contents

0	Task -0: Hadoop step Up and Installation in Google could:	5
0.1	ssh key generating.	5
0.2	Login with Instance.	5
0.3	Update .....	6
0.4	Install open jdk .....	6
0.5	Creating new user for Hadoop .....	6
0.6	Download Hadoop .....	7
0.7	setting configuration .....	7
0.8	setting hdfs configuration .....	8
0.9	starting services .....	8
1	Task -01: Creating Your Directory Space .....	9
2	Task -02: Understanding the System .....	10
2.1	How many data nodes are part of the Hadoop topology?	10
2.2	What are the IP addresses of these datanodes? .....	10
2.3	What is the configured and present capacity of the HDFS? .....	10
2.4	What is the default file replication count? .....	10
2.5	Task 02 steps: .....	10
2.5.1	Step a .....	10
2.5.2	Step b .....	11
2.5.3	Step c .....	11
3	Task -03: Getting sample data .....	12
3.1	How to Upload file in Google Instances .....	12
3.1.1	Step1: In instances Click on SSH .....	12
3.1.2	This screen will be shown .....	12
3.1.3	Click on upload icon .....	12
3.1.4	We can see that file uploaded successfully .....	13
3.1.5	Upload file from Instances to Hadoop user directory. ....	13
3.1.6	Now run hdfs fsck command to see detail about file .....	13
3.1.7	What is the default block size (in Mb) of the airline_data.csv file? .....	14
3.1.8	Is there any missing replicas for the file airline_data.csv?.....	14
3.1.9	What command will you use to change this block size to 6 Mb (remember to convert into bytes) .....	14
3.1.10	How many blocks are used by airline_data.csv after changing block size in Question 2?.....	15

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

3.1.11	How many missing replicas are there for file airline_data.csv after block change?	15
3.1.12	Why are there missing replicas?.....	16
4	Task -04: Setting up First Map Reduce Job .....	17
4.1	Mapper Code: .....	17
4.2	Reducer Code: .....	17
4.3	Write and run locally.....	17
4.4	Uploaded files VM instances and its user.....	18
4.5	To run Job in Hadoop: .....	18
4.6	What was the <key,value> pair used in this query?.....	19
4.7	How many mapper threads were used?.....	19
4.8	How many mapper threads were used?.....	19
4.9	What was the time spent by all mapper threads?.....	19
4.10	What was the time spent by all reducer threads? .....	19
4.11	What is the file name in which your output is located?.....	19
4.12	Variation 1:.....	20
4.13	Variation 2:.....	20
4.14	Variation 3:.....	20
5	Task -05: Designing Additional Queries .....	21
5.1	Task 5a: Present the Total Flights per Year as a percentage .....	22
	<key, value>=.....	22
	< key_year, value_flights > .....	22
	Output Folder:.....	22
5.1.1	Command used for this task .....	22
5.1.2	Mapper Code: .....	22
5.1.3	Reducer Code: .....	23
5.2	Task 5B: Present the Total Flights per Year as a percentage .....	24
	<key, value>=.....	24
	<month, value_flights > .....	24
	Output Folder:.....	24
5.2.1	Mapper Code: .....	24
5.2.2	Reducer Code: .....	24
5.2.3	Command: .....	24
5.3	Task 5c: which airline carrier has flown the most flights over the 10 year period? ..	25
	<key, value>=.....	25

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

<Airline, value_flights > .....	25
Output Folder:.....	25
5.3.1    Commands used for this task: .....	25
5.3.2    Mapper Code: .....	25
5.3.3    Reducer Code: .....	25
5.4    Task 5d: Which Airport has been the most busiest over the 10 year?.....	26
<key, value>=.....	26
<airport, flights> .....	26
Output Folder:.....	26
5.4.1    Command used in this task: .....	26
5.4.2    Mapper Code: .....	26
5.4.3    Reducer Code: .....	26
5.5    Task 5e: Which Airport has the Largest Flights to Cancellation Ratio? .....	27
<key, value>=.....	27
<key_airport,value_flights, value_cancel> .....	27
Output Folder:.....	27
5.5.1    Mapper Code: .....	27
5.5.2    Reducer Code: .....	27
5.6    Task 5f: Find the Total Amount of Delay Minutes Grouped by Airline .....	29
<key, value>=.....	29
<airline, arr_delay> .....	29
Output Folder:.....	29
.....	29
5.6.1    Mapper Code: .....	29
5.6.2    Reducer Code: .....	30
5.7    Task 5g: Find the Airport with most Canceled Flights in 2016. ....	31
<key, value>=.....	31
<key_airport, arr_cancelled> .....	31
Output Folder:.....	31
5.7.1    Mapper Code: .....	31
5.7.2    Reducer Code: .....	31
5.8    Task 5h: Find the average delay time for an airport that is the most busiest of all other airports. ....	32
<key, value> .....	32
<key_airport, value_flights, value_delay> .....	32

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

Output Folder:.....	32
5.8.1    Mapper Code .....	32
5.8.2    Reducer Code: .....	32
5.9    TASK 5J: WHAT IS THE PROBABILITY THAT A FLIGHT WILL BE CANCELLED DUE TO BAD WEATHER AT THE MOST BUSIEST AIRPORT OF ALL OTHER AIRPORTS? .....	33
5.9.1    Output Folder:.....	33
5.10    Task 5k: Find out the airport that have highest security delay than of all other Airports?.....	33
Output Folder:.....	33
5.10.1    Mapper code: .....	34
5.10.2    Reducer Code: .....	34

## 0 TASK -0: HADOOP STEP UP AND INSTALLATION IN GOOGLE CLOUD:

## 0.1 SSH KEY GENERATING.

## 0.2 LOGIN WITH INSTANCE

(anjum@192) - [~]  
\$ ssh 34.125.110.186  
The authenticity of host '34.125.110.186 (34.125.110.186)' can't be established.  
ED25519 key fingerprint is SHA256:9kS+/pVRIfFuUsfOvfumqde18jMpc3srixQgobkxBXU.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '34.125.110.186' (ED25519) to the list of known hosts.  
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.13.0-1024-gcp x86\_64)  
[Ubuntu Campus] Fwd: Holidays for Eid-Ul-Fitr 2022 Forwarded message From:  
\* Documentation: <https://help.ubuntu.com>  
\* Management: <https://landscape.canonical.com>  
\* Support: <https://ubuntu.com/advantage>  
Management BSC 4B Class Forwarded message  
System information as of Fri Apr 29 06:42:38 UTC 2022 in ICT Day broadcast - part 1 - View online  
System load: 0.11 Processes: 109  
Usage of /: 18.0% of 9.52GB Users logged in: 0  
Memory usage: 5% IPv4 address for ens4: 10.182.0.2  
Swap usage: 0% Fwd: Pandaship 2022 - foodpanda Summer Internship Program Forwarded message  
1 update can be applied immediately. To learn more about updates, view online Hikuram, Gaming is competitive – especially  
To see these additional updates run: apt list --upgradable  
Google Cloud Platfo. [Update] Changes to the Google Cloud Platform Third-Party Subprocessor  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
The programs included with the Ubuntu system are free software; be purchased  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/\*copyright. Forwarded message From: Faheem K

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

### 0.3 UPDATE

```
anjum@instance-1:~$ sudo apt update
Hit:1 http://us-west4.gce.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-west4.gce.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]- part 1 - View online Join our live broadcast
Get:4 http://us-west4.gce.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:5 http://us-west4.gce.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:6 http://us-west4.gce.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB] - View broadcast! - View
Get:7 http://us-west4.gce.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
[...]
```

### 0.4 INSTALL OPEN JDK

```
10 packages can be upgraded. Run 'apt list --upgradable' to see them.
anjum@instance-1:~$ sudo apt install openjdk-8-jdk
Reading package lists ... Done [Fwd: Holidays for Eid-ul-Fitr 2022] -> Forwarded message -> From: pwr hr <pwr...
Building dependency tree
Reading state information ... Done [Fundamentals of Management BSC 4B Class] -> Forwarded message ->
The following packages were automatically installed and are no longer required:
libatasmart4 libblockdev-fs2 libblockdev-loop2 libblockdev-part-err2 libblockdev-part2
libblockdev-swap2 libblockdev-utils2 libblockdev2 libmm-glib0 libnuma1 libparted-fs-resize0
libudisks2-0 usb-modeswitch usb-modeswitch-data
Use 'sudo apt autoremove' to remove them. [REGISTER] Women Rock-IT special edition for Girls in ICT Day broadcas
The following additional packages will be installed:
adwaita-icon-theme at-spi2-core ca-certificates-java fontconfig fontconfig-config
```

### 0.5 CREATING NEW USER FOR HADOOP

```
anjum@instance-1:~$ sudo adduser hdoop
Adding user `hdoop' ...
Adding new group `hdoop' (1003) ...
Adding new user `hdoop' (1002) with group `hdoop' ...
Creating home directory `/home/hdoop' ...
Copying files from `/etc/skel' ...
New password: [NUCES PWR Campus] -> Fwd: Holidays for Eid-ul-Fitr 2022 -> Forwarded message -> From: pwr...
Retype new password:
passwd: password updated successfully
Changing the user information for hdoop
Enter the new value, or press ENTER for the default
  Cisco Full Name []:
    Room Number []:
  Cisco Work Phone []:
  Cisco Home Phone []:
  Cisco Other []:
Is the information correct? [Y/n] [Pandaship 2022 - Foodpanda Summer Internship Program] -> F
anjum@instance-1:~$ su - hdoop
Password: [NUCES PWR Campus] -> Fwd: [NUCES PWR Campus] -> Forwarded message -> From: Faheem
hdoop@instance-1:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
-bash: /home/hdoop/.ssh/authorized_keys: No such file or directory
hdoop@instance-1:~$ ssh-keygen
[NUCES PWR Campus] -> Changes to the Google Cloud Platform Third-Party Subprocess
Command 'ssh-keygen' not found, did you mean:
  command 'ssh-keygen' from deb openssh-client (1:8.2p1-4ubuntu0.4)
  NUCES PWR Campus -> Fwd: Recommend Books for the Library to be purchased -> Forwarded message -> From: Faheem
Try: apt install <deb name>
[NUCES PWR Campus] -> Fwd: Defaulter List S2 -> Forwarded message -> From: Faheem
hdoop@instance-1:~$ ssh-keygen
```

# BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

## 0.6 DOWNLOAD HADOOP

```
hadoop@instance-1:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
Created directory '/home/hadoop/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:2tBzDRT5B0NRH7JoVivBdhHTZ+qTeEJwDjmAQladNc hadoop@instance-1
The key's randomart image is: wu. Holidays for Eid-ul-Fitr 2022 — Forwarded message — From: pwr hr <pw
+---[RSA 3072]---+
|oo=+. . . ==**o|
|.o * o. E +o+=oB|
|. o o . . + ... +**o|
| Cisco io+oob+.g|ic.
| S++ . o |
| Cisco Networking. c.
| . o. |
| . o |
| NUCES PWR Campus |
+---[SHA256]---+
hadoop@instance-1:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hadoop@instance-1:~$ chmod 0600 ~/.ssh/authorized_keys
hadoop@instance-1:~$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.2.3/hadoop-3.2.3.tar.g
z Google Cloud Platform [Update] Changes to the Google Cloud Platform Third-Party Subprocessors list - we'-- 2022-04-29 06:47:37-- https://dlcdn.apache.org/hadoop/common/hadoop-3.2.3/hadoop-3.2.3.tar.g
z Resolving dlcdn.apache.org (dlcdn.apache.org) ... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443 ... connected.
HTTP request sent, awaiting response ... 200 OK [No license available for the Library to be purchased] — Forwarded message
Length: 492241961 (469M) [application/x-gzip]
Saving to: 'hadoop-3.2.3.tar.gz'
```

## 0.7 SETTING CONFIGURATION

```
hadoop@instance-1:~$ tar xzf hadoop-3.2.3.tar.gz
hadoop@instance-1:~$ nano ~/.bashrc
hadoop@instance-1:~$ readlink -f /usr/bin/javac
/usr/lib/jvm/java-8-openjdk-amd64/bin/javac
hadoop@instance-1:~$ nano ~/hadoop-3.2.3/etc/hadoop/hadoop-env.sh
hadoop@instance-1:~$ nano ~/hadoop-3.2.3/etc/hadoop/core-site.xml
hadoop@instance-1:~$ nano ~/hadoop-3.2.3/etc/hadoop/hdfs-site.xml
hadoop@instance-1:~$ nano ~/hadoop-3.2.3/etc/hadoop/mapred-site.xml
hadoop@instance-1:~$ nano ~/hadoop-3.2.3/etc/hadoop/yarn-site.xml
hadoop@instance-1:~$ hdfs namenode -format
```

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

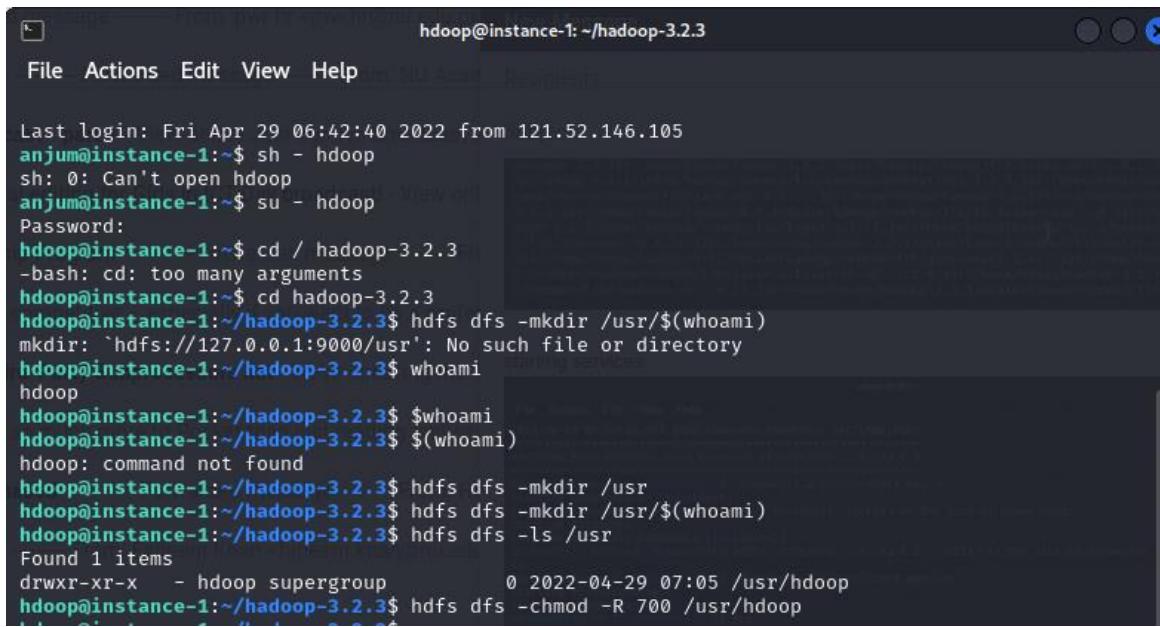
### 0.8 SETTING HDFS CONFIGURATION

```
hadoop@instance-1:~$ cd hadoop-3.2.3/
hadoop@instance-1:~/hadoop-3.2.3$ hdfs namenode -format
WARNING: /home/hadoop/hadoop-3.2.3/logs does not exist. Creating.
2022-04-29 06:56:27,199 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG: host = instance-1/10.182.0.2
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.2.3
STARTUP_MSG: classpath = /home/hadoop/hadoop-3.2.3/etc/hadoop:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/commons-collections-3.2.2.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/pwr.hr@kerb-client-1.0.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/animal-sniffer-annotations-1.17.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/curator-client-2.13.0.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jackson-core-asl-1.9.13.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jetty-io-9.4.40.v20210413.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jersey-servlet-1.19.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/commons-compress-1.21.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/error_prone_annotations-2.2.0.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/netty-3.10.6.Final.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jetty-webapp-9.4.40.v20210413.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/token-provider-1.0.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/commons-io-2.8.0.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/nimbus-jose-jwt-9.8.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/kerb-util-1.0.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/commons-codec-1.11.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/paranamer-2.3.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/kerby-pkix-1.0.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/hadoop-annotations-3.2.3.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/re2j-1.1.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/failureaccess-1.0.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/log4j-1.2.17.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/zookeeper-3.4.14.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/json-smart-2.4.7.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/javax.activation-api-1.2.0.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jackson-xc-1.9.13.jar:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/jackson-ann
```

### 0.9 STARTING SERVICES

```
anum@192:~$ 
File Actions Edit View Help 
2022-04-29 06:56:28,953 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at instance-1/10.182.0.2
*****/ 
hadoop@instance-1:~/hadoop-3.2.3$ ~/hadoop-3.2.3/sbin/start-dfs.sh
Starting namenodes on [localhost]
localhost: Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Starting datanodes
Starting secondary namenodes [instance-1]
instance-1: Warning: Permanently added 'instance-1,10.182.0.2' (ECDSA) to the list of known hosts.
hadoop@instance-1:~/hadoop-3.2.3$ ~/hadoop-3.2.3/sbin/start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hadoop@instance-1:~/hadoop-3.2.3$ jps
6944 Jps Networking Ac. Fwd: Fundamentals of Management BSC 4B Class — Forwarded message — From: NU Acad
5905 NameNode [LAST CHANCE to REGISTER] Women Rock-IT special edition for Girls in ICT Day broadcast! - View onl
6293 SecondaryNameNode 6055 DataNode 6633 NodeManager 6474 ResourceManagers
hadoop@instance-1:~/hadoop-3.2.3$ ls -lh
total 216K
-rw-rw-r-- 1 hdoop hdoop 148K Mar 10 05:39 LICENSE.txt
-rw-rw-r-- 1 hdoop hdoop 22K Mar 10 05:39 NOTICE.txt
-rw-rw-r-- 1 hdoop hdoop 1.4K Mar 10 05:39 README.txt
drwxr-xr-x 2 hdoop hdoop 4.0K Mar 20 01:58 bin
drwxrwxr-x 3 hdoop hdoop 4.0K Apr 29 06:56 dfsdata
drwxr-xr-x 3 hdoop hdoop 4.0K Mar 20 01:20 etc
drwxr-xr-x 2 hdoop hdoop 4.0K Mar 20 01:58 include
drwxr-xr-x 3 hdoop hdoop 4.0K Mar 20 01:58 lib
drwxr-xr-x 4 hdoop hdoop 4.0K Mar 20 01:58 libexec
drwxrwxr-x 3 hdoop hdoop 4.0K Apr 29 06:57 logs
drwxr-xr-x 3 hdoop hdoop 4.0K Mar 20 01:20 sbin
```

## 1 TASK -01: CREATING YOUR DIRECTORY SPACE



```
hadoop@instance-1: ~/hadoop-3.2.3
File Actions Edit View Help

Last login: Fri Apr 29 06:42:40 2022 from 121.52.146.105
anjum@instance-1:~$ sh - hdoop
sh: 0: Can't open hdoop
anjum@instance-1:~$ su - hdoop
Password:
hadoop@instance-1:~$ cd / hadoop-3.2.3
-bash: cd: too many arguments
hadoop@instance-1:~$ cd hadoop-3.2.3
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -mkdir /usr/$(whoami)
mkdir: `hdfs://127.0.0.1:9000/usr': No such file or directory
hadoop@instance-1:~/hadoop-3.2.3$ whoami
hadoop
hadoop@instance-1:~/hadoop-3.2.3$ $whoami
hadoop@instance-1:~/hadoop-3.2.3$ $(whoami)
hadoop: command not found
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -mkdir /usr
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -mkdir /usr/$(whoami)
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -ls /usr
Found 1 items
drwxr-xr-x  - hadoop supergroup          0 2022-04-29 07:05 /usr/hadoop
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -chmod -R 700 /usr/hadoop
```

## 2 TASK -02: UNDERSTANDING THE SYSTEM

It's to be noted that we have defined a function for each step and we have made the generalized code.

### 2.1 HOW MANY DATA NODES ARE PART OF THE HADOOP TOPOLOGY?

- Number of data-nodes: 1
- Live data nodes (1):

### 2.2 WHAT ARE THE IP ADDRESSES OF THESE DATANODES?

- Number of data-nodes: 1
- Name: 127.0.0.1:9866 (localhost)

### 2.3 WHAT IS THE CONFIGURED AND PRESENT CAPACITY OF THE HDFS?

- Configured Capacity: 10222829568 (9.52 GB)
- Present Capacity: 4910075904 (4.57 GB)

### 2.4 WHAT IS THE DEFAULT FILE REPLICATION COUNT?

- Default replication factor: 1

### 2.5 TASK 02 STEPS:

#### 2.5.1 Step a

```
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfs -ls /usr
Found 1 items
drwxr-xr-x - hadoop supergroup          0 2022-04-29 07:05 /usr/hadoop
hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfsadmin -printTopology
Rack: /default-rack
127.0.0.1:9866 (localhost) as read/write/execute permissions

hadoop@instance-1:~/hadoop-3.2.3$ hdfs dfsadmin -report
Configured Capacity: 10222829568 (9.52 GB)
Present Capacity: 4910075904 (4.57 GB)
DFS Remaining: 4910047232 (4.57 GB)
DFS Used: 28672 (28 KB)
DFS Used%: 0.00%
Replicated Blocks:
Under replicated blocks: 0
Blocks with corrupt replicas: 0
Missing blocks: 0
Missing blocks (with replication factor 1): 0
Low redundancy blocks with highest priority to recover: 0
Pending deletion blocks: 0
Erasure Coded Block Groups:
Low redundancy block groups: 0
Block groups with corrupt internal blocks: 0
Missing block groups: 0
Low redundancy blocks with highest priority to recover: 0
Pending deletion blocks: 0

Live datanodes (1):
Name: 127.0.0.1:9866 (localhost)
Hostname: instance-1.us-west4-b.c.bigdataanalytics-348706.internal
Decommission Status : Normal
Configured Capacity: 10222829568 (9.52 GB)
DFS Used: 28672 (28 KB)
Non DFS Used: 5295976448 (4.93 GB)
DFS Remaining: 4910047232 (4.57 GB)
DFS Used%: 0.00%
DFS Remaining%: 48.03%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Fri May 13 09:38:05 UTC 2022
Last Block Report: Fri May 13 08:31:53 UTC 2022
Num of Blocks: 0
```

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

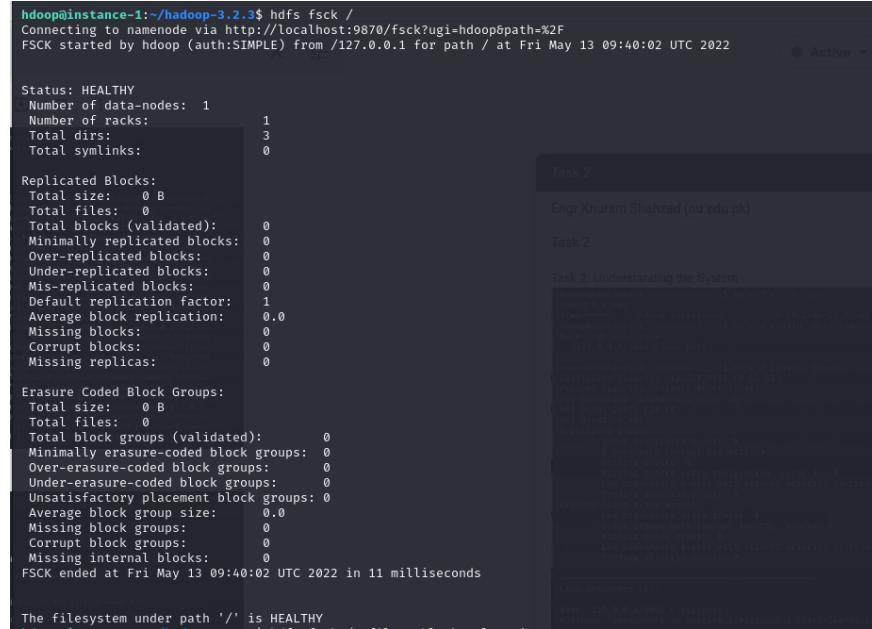
### 2.5.2 Step b

```
hadoop@instance-1:~/hadoop-3.2.3$ hdfs fsck /
Connecting to namenode via http://localhost:9870/fsck?ugi=hadoop&path=%2F
FSCK started by hdoop (auth:SIMPLE) from /127.0.0.1 for path / at Fri May 13 09:40:02 UTC 2022
Status: HEALTHY
Number of data-nodes: 1
Number of racks: 1
Total dirs: 3
Total symlinks: 0

Replicated Blocks:
Total size: 0 B
Total files: 0
Total blocks (validated): 0
Minimally replicated blocks: 0
Over-replicated blocks: 0
Under-replicated blocks: 0
Mis-replicated blocks: 0
Default replication factor: 1
Average block replication: 0.0
Missing blocks: 0
Corrupt blocks: 0
Missing replicas: 0

Erasure Coded Block Groups:
Total size: 0 B
Total files: 0
Total block groups (validated): 0
Minimally erasure-coded block groups: 0
Over-erasure-coded block groups: 0
Under-erasure-coded block groups: 0
Unsatisfactory placement block groups: 0
Average block group size: 0.0
Missing block groups: 0
Corrupt block groups: 0
Missing internal blocks: 0
FSCK ended at Fri May 13 09:40:02 UTC 2022 in 11 milliseconds

The filesystem under path '/' is HEALTHY
```



### 2.5.3 Step c

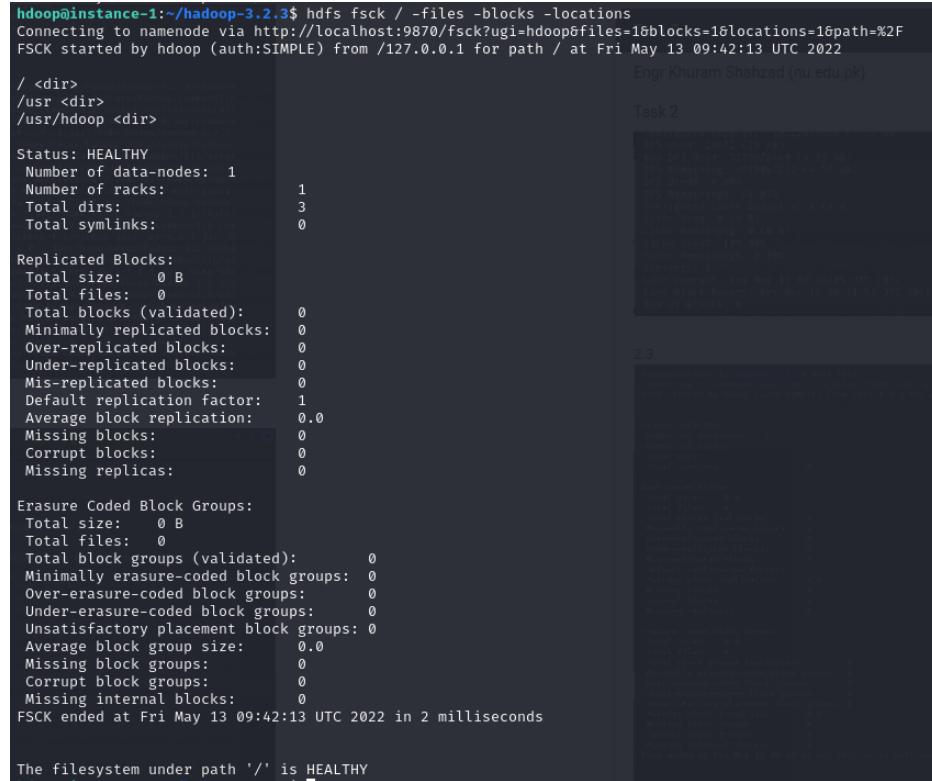
```
hadoop@instance-1:~/hadoop-3.2.3$ hdfs fsck / -files -blocks -locations
Connecting to namenode via http://localhost:9870/fsck?ugi=hadoop&files=1&blocks=1&locations=1&path=%2F
FSCK started by hdoop (auth:SIMPLE) from /127.0.0.1 for path / at Fri May 13 09:42:13 UTC 2022
/ <dir>
/usr <dir>
/usr/hadoop <dir>

Status: HEALTHY
Number of data-nodes: 1
Number of racks: 1
Total dirs: 3
Total symlinks: 0

Replicated Blocks:
Total size: 0 B
Total files: 0
Total blocks (validated): 0
Minimally replicated blocks: 0
Over-replicated blocks: 0
Under-replicated blocks: 0
Mis-replicated blocks: 0
Default replication factor: 1
Average block replication: 0.0
Missing blocks: 0
Corrupt blocks: 0
Missing replicas: 0

Erasure Coded Block Groups:
Total size: 0 B
Total files: 0
Total block groups (validated): 0
Minimally erasure-coded block groups: 0
Over-erasure-coded block groups: 0
Under-erasure-coded block groups: 0
Unsatisfactory placement block groups: 0
Average block group size: 0.0
Missing block groups: 0
Corrupt block groups: 0
Missing internal blocks: 0
FSCK ended at Fri May 13 09:42:13 UTC 2022 in 2 milliseconds

The filesystem under path '/' is HEALTHY
```



## 3 TASK -03: GETTING SAMPLE DATA

### 3.1 How to Upload File in Google Instances

#### 3.1.1 Step1: In instances Click on SSH

The screenshot shows the Google Cloud Platform's VM Instances page. A single instance named "instance-1" is listed. The "SSH" column for this instance has a dropdown menu open, with the option "Open terminal" highlighted. Other options in the menu include "Copy address", "Edit", and "Delete". The page also includes a sidebar for selecting an instance, permissions, and labels.

#### 3.1.2 This screen will be shown

```
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.13.0-1024-gcp x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Tue May 17 05:40:16 UTC 2022

 System load: 0.12      Processes:          123
 Usage of /: 52.1% of 9.52GB  Users logged in:   1
 Memory usage: 42%        IPv4 address for ens4: 10.182.0.2
 Swap usage: 0%

 * Super-optimized for small spaces - read how we shrank the memory
 footprint of MicroK8s to make it the smallest full K8s around.

 https://ubuntu.com/blog/microk8s-memory-optimisation

 18 updates can be applied immediately.
 To see these additional updates run: apt list --upgradable

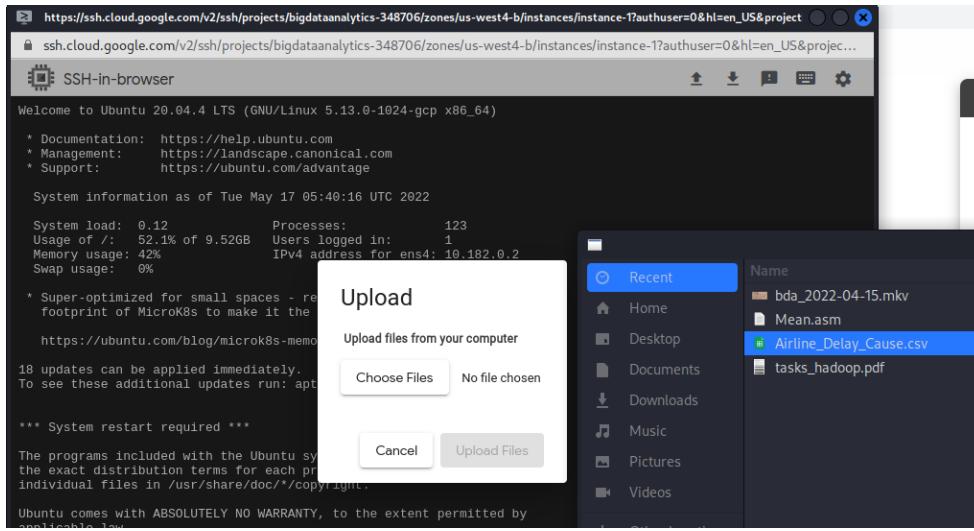
 *** System restart required ***

 The programs included with the Ubuntu system are free software;
 the exact distribution terms for each program are described in the
 individual files in /usr/share/doc/*copyright.

 Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
 applicable law.
```

#### 3.1.3 Click on upload icon

# BIG DATA ANALYTICS ASSIGNMENT (Hadoop)



### 3.1.4 We can see that file uploaded successfully

```
itskhuramshahzad@instance-1:~$ ls /  
bin dev home lib32 libx32 media opt root sbin srv var  
boot etc lib lib64 lost+found mnt proc run snap sys usr  
itskhuramshahzad@instance-1:  
itskhuramshahzad@instance-1:~$ ls /home/  
anjum hdoop itskhuramshahzad ubuntu  
itskhuramshahzad@instance-1:~$ ls /home/anjum/  
itskhuramshahzad@instance-1:~$ ls /itskhuramshahzad/  
ls: cannot access '/itskhuramshahzad': No such file or directory  
itskhuramshahzad@instance-1:~$ ls /home/itskhuramshahzad/  
Airline_Delay_Cause.csv  
itskhuramshahzad@instance-1:~$ cls
```

Now came to local system in Hadoop

### 3.1.5 Upload file from Instances to Hadoop user directory.

```
hadoop@instance-1:~$ hdfs dfs -put /home/itskhuramshahzad/Airline_Delay_Cause.csv /usr/hadoop/Airline_Delay_Cause.csv  
hadoop@instance-1:~$ hdfs dfs -ls /usr/hadoop  
Found 1 items  
-rw-r--r-- 1 hadoop supergroup 277233 2022-05-17 05:47 /usr/hadoop/Airline_Delay_Cause.csv
```

### 3.1.6 Now run hdfs fsck command to see detail about file

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

```
hadoop@instance-1:~$ hdfs fsck /usr/hadoop/Airline_Delay_Cause.csv
Connecting to namenode via http://localhost:9870/fsck?ugi=hadoop&path=%2Fusr%2Fhadoop%2FAirline_Delay_Cause.csv
FSCK started by hadoop (auth:SIMPLE) from /127.0.0.1 for path /usr/hadoop/Airline_Delay_Cause.csv at Tue May 17 06:08:21 UTC 2022

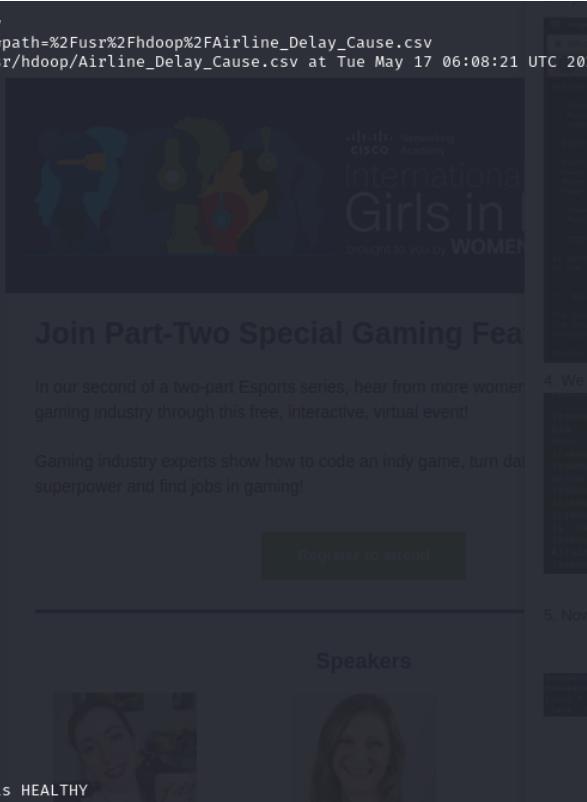
Status: HEALTHY
Number of data-nodes: 1
Number of racks: 1
Total dirs: 0
Total symlinks: 0

Replicated Blocks:
Total size: 277233 B
Total files: 1
Total blocks (validated): 1 (avg. block size 277233 B)
Minimally replicated blocks: 1 (100.0 %)
Over-replicated blocks: 0 (0.0 %)
Under-replicated blocks: 0 (0.0 %)
Mis-replicated blocks: 0 (0.0 %)
Default replication factor: 1
Average block replication: 1.0
Missing blocks: 0
Corrupt blocks: 0
Missing replicas: 0 (0.0 %)

Erasure Coded Block Groups:
Total size: 0 B
Total files: 0
Total block groups (validated): 0
Minimally erasure-coded block groups: 0
Over-erasure-coded block groups: 0
Under-erasure-coded block groups: 0
Unsatisfactory placement block groups: 0
Average block group size: 0.0
Missing block groups: 0
Corrupt block groups: 0
Missing internal blocks: 0

FSCK ended at Tue May 17 06:08:22 UTC 2022 in 13 milliseconds

The filesystem under path '/usr/hadoop/Airline_Delay_Cause.csv' is HEALTHY
hadoop@instance-1:~$
```



# Questions

**3.1.7 What is the default block size (in Mb) of the airline\_data.csv file?**

- 277233 Bytes.

**3.1.8 Is there any missing replicas for the file airline\_data.csv?**

- NO

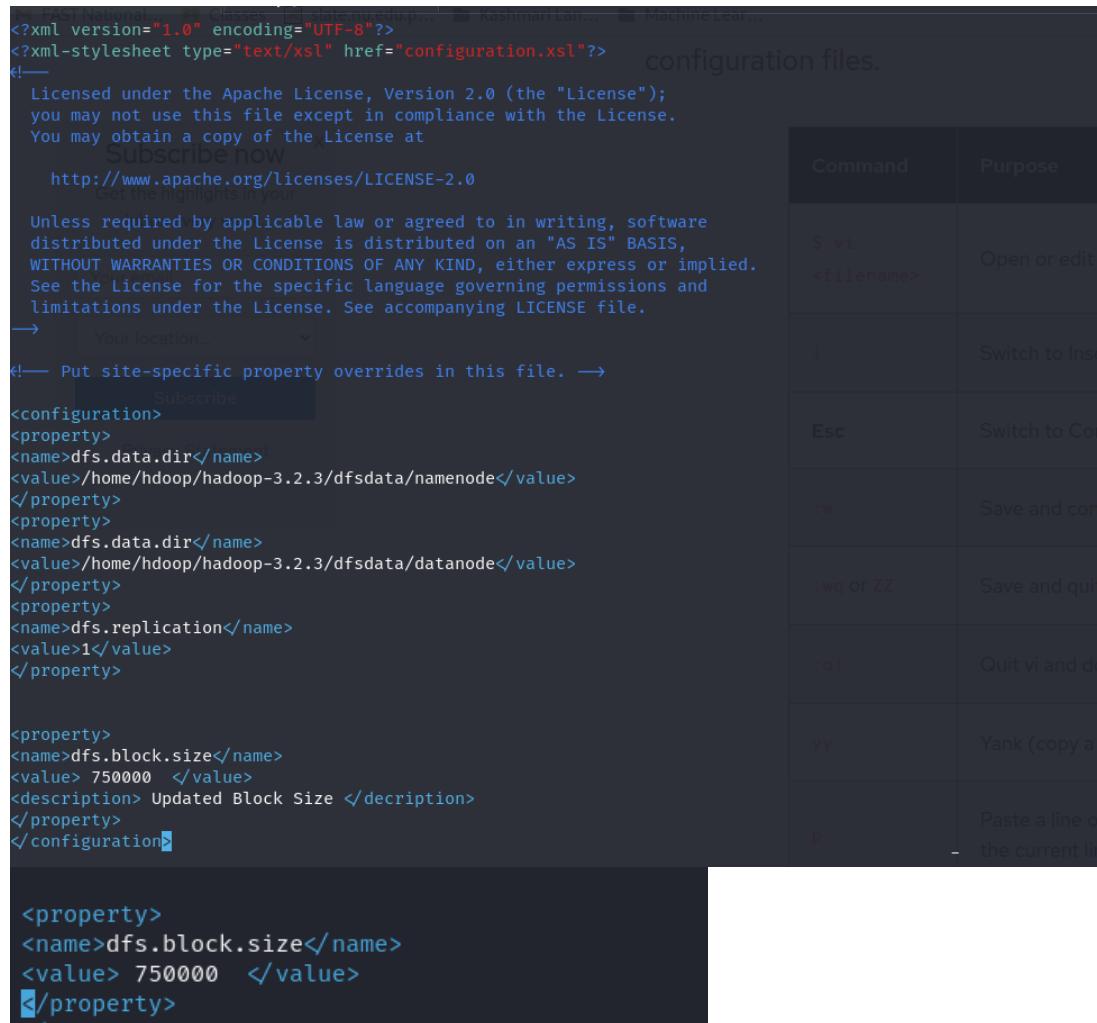
**3.1.9 What command will you use to change this block size to 6 Mb (remember to convert into bytes)**

- To change for single file.

```
hadoop@instance-1:~$ hdfs dfs -D dfs.blocksize=6291456 -put /home/itskhuramshahzad/Airline_Delay_Cause.csv /usr/hadoop/Airline_Delay_Cause3.csv
hadoop@instance-1:~$ hadoop fs -stat %o /usr/hadoop/Airline_Delay_Cause3.csv
6291456                                1 bytes = 9.547e+10 Megabytes    10 bytes = 9.5467e+10 Megabytes   2500 bytes = 0.0024 Megabytes
hadoop@instance-1:~$ hdfs getconf -confKey dfs.blocksize
6291456
hadoop@instance-1:~$
```

- hdfs dfs -D dfs.blocksize=6291456 -put /home/itskhuramshahzad/airline\_data.csv /usr/hadoop/airline\_data.csv
- To change in configuration  
vim ~/hadoop-3.2.3/etc/hadoop/hdfs-site.xml

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)



The terminal window shows the Apache License 2.0 and a configuration file for HDFS. The configuration file contains the following properties:

```
<configuration>
<property>
<name>dfs.data.dir</name>
<value>/home/hadoop/hadoop-3.2.3/dfsdata/namenode</value>
</property>
<property>
<name>dfs.data.dir</name>
<value>/home/hadoop/hadoop-3.2.3/dfsdata/datanode</value>
</property>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>

<property>
<name>dfs.block.size</name>
<value> 750000 </value>
<description> Updated Block Size </description>
</property>
</configuration>
```

At the bottom of the configuration file, there is a highlighted section:

```
<property>
<name>dfs.block.size</name>
<value> 750000 </value>
</property>
```

### 3.1.10 How many blocks are used by airline\_data.csv after changing block size in Question 2?

```
hadoop@instance-1:~$ hdfs getconf -confKey dfs.blocksize
134217728
hadoop@instance-1:~$ hadoop fs -stat %o /usr/hadoop/Airline_Delay_Cause2.csv
6291456
hadoop@instance-1:~$
```

5 blocks are used.

### 3.1.11

How many missing replicas are there for file airline\_data.csv after block change?

Zero.

```
Mis-replicated blocks: 0 (0.0 %)
Default replication factor: 1
Average block replication: 1.0
Missing blocks: 0
Corrupt blocks: 0
Missing replicas: 0 (0.0 %)
```

Erasures Coded Block Groups:

### 3.1.12

#### Why are there missing replicas?

Missing blocks can happen when all replicas of a block in the file are corrupted or all replicas go missing

## 4 TASK -04: SETTING UP FIRST MAP REDUCE JOB

---

### 4.1 MAPPER CODE:

```
reducer.py
1#!/usr/bin/python
2import sys
3for line in sys.stdin:
4    data = line.strip().split(",");
5    key= data[0]
6    value= 1
7    print("{0}\t{1}".format(key, value))
8
```

### 4.2 REDUCER CODE:

```
1#!/usr/bin/python
2import sys
3total =0
4oldkey= None
5
6for line in sys.stdin:
7    data = line.strip().split("\t")
8    thiskey= data[0]
9    value = data[1]
10   if thiskey!= oldkey and oldkey!= None:
11       print("{0}\t{1}".format(oldkey, total))
12       oldkey= thiskey
13       total=0
14   oldkey= thiskey
15   total+= float(value)
16if oldkey != None:
17    print("{0}\t{1}".format(oldkey, total))
18
```

### 4.3 WRITE AND RUN LOCALLY



```
(anum@192)-[~]
$ cat airline_data.csv | ./mapper.py | sort | ./reducer.py
2010 17575.0
2011 15585.0
2012 14387.0
2013 16089.0
2014 13980.0
2015 13528.0
2016 12217.0
2017 12518.0
2018 20231.0
2019 20946.0
year 1.0
```

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

### 4.4 UPLOADED FILES VM INSTANCES AND ITS USER.

```
hadoop@instance-1:~$ hdfs dfs -put /home/itskhuramshahzad/mapper.py /usr/hadoop/mapper.py
hadoop@instance-1:~$ hdfs dfs -put /home/itskhuramshahzad/reducer.py /usr/hadoop/reducer.py
hadoop@instance-1:~$
```

### 4.5 TO RUN JOB IN HADOOP:

```
hadoop@instance-1:~$ hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -file /home/itskhuramshahzad/mapperX.py -file /home/itskhuramshahzad/reducerX.py -mapper mapperX.py -reducer ./reducerX.py -input /usr/hadoop/airline_data.csv -output /usr/hadoop/_output14
2022-05-27 16:07:02,735 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [/home/itskhuramshahzad/mapperX.py, /home/itskhuramshahzad/reducerX.py, /tmp/hadoop-unjar458475931461339751/] [] /tmp/streamjob640
6172208663873126.jar tmpDir=null
2022-05-27 16:07:03,970 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:5353
2022-05-27 16:07:04,222 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:5353
2022-05-27 16:07:04,561 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hadoop/.staging/job_165366094282_0005
6094282_0005
2022-05-27 16:07:04,913 INFO mapred.FileInputFormat: Total input files to process : 1
2022-05-27 16:07:04,995 INFO mapreduce.JobSubmitter: number of splits:5
2022-05-27 16:07:05,199 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_165366094282_0005
2022-05-27 16:07:05,201 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-05-27 16:07:05,441 INFO conf.Configuration: resource-types.xml not found
2022-05-27 16:07:05,441 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-05-27 16:07:05,532 INFO impl.YarnClientImpl: Submitted application application_165366094282_0005
2022-05-27 16:07:05,571 INFO mapreduce.Job: The url to track the job: http://instance-1:5349/proxy/application_165366094282_0005/
2022-05-27 16:07:05,571 INFO mapreduce.Job: Job job_165366094282_0005 running in uber mode : false
2022-05-27 16:07:12,779 INFO mapreduce.Job: map 0% reduce 0%
2022-05-27 16:07:31,028 INFO mapreduce.Job: map 20% reduce 0%
2022-05-27 16:07:32,081 INFO mapreduce.Job: map 60% reduce 0%
2022-05-27 16:07:33,081 INFO mapreduce.Job: map 100% reduce 0%
2022-05-27 16:07:38,131 INFO mapreduce.Job: map 100% reduce 0%
2022-05-27 16:07:39,146 INFO mapreduce.Job: Job job_165366094282_0005 completed successfully
2022-05-27 16:07:39,265 INFO mapreduce.Job: Counters : 55
File System Counters
FILE: Number of bytes read=1413519
FILE: Number of bytes written=4266269
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=26661848
HDFS: Number of bytes written=139
HDFS: Number of read operations=20
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
HDFS: Number of bytes read erasure-coded=0
```

The terminal window shows the command used to run the Hadoop streaming job, the log output including file counts and sizes, and the final success message. The Microsoft Word document lists various metrics such as map and reduce tasks, total time spent, and memory usage.

Launched map tasks=5  
Launched reduce tasks=1  
Data-local map tasks=5  
Total time spent by all maps in occupied slots (ms)=86216  
Total time spent by all reduces in occupied slots (ms)=4100  
Total time spent by all map tasks (ms)=86216  
Total time spent by all reduce tasks (ms)=4100  
Total vcore-milliseconds taken by all map tasks=86216  
Total vcore-milliseconds taken by all reduce tasks=4100  
Total megabyte-milliseconds taken by all map tasks=88285184  
Total megabyte-milliseconds taken by all reduce tasks=4198400

Map-Reduce Framework

- Map input records=157057
- Map output records=157057
- Map output bytes=1099399
- Map output materialized bytes=1413543
- Input split bytes=500
- Combine input records=0
- Combine output records=0
- Reduce input groups=11
- Reduce shuffle bytes=1413543
- Reduce input records=157057
- Reduce output records=11
- Spilled Records=314114
- Shuffled Maps = 5
- Failed Shuffles=0
- Merged Map outputs=5
- GC time elapsed (ms)=1468
- CPU time spent (ms)=8450
- Physical memory (bytes) snapshot=1554722816
- Virtual memory (bytes) snapshot=15194402816
- Total committed heap usage (bytes)=1188036608
- Peak Map Physical memory (bytes)=277102592
- Peak Map Virtual memory (bytes)=2534268928
- Peak Reduce Physical memory (bytes)=179793920
- Peak Reduce Virtual memory (bytes)=2535227392

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=26661348

File Output Format Counters

- Bytes Written=139

22-05-27 16:07:39,271 INFO streaming.StreamJob: Output directory: /usr/hadoop/\_output14

- hadoop jar \$HADOOP\_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -file /home/itskhuramshahzad/mapperX.py -file /home/itskhuramshahzad/reducerX.py -mapper mapperX.py -reducer ./reducerX.py -input /usr/hadoop/airline\_data.csv -output /usr/hadoop/\_output14  
or
- hadoop jar \$HADOOP\_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -files /home/itskhuramshahzad/mapperX.py -file /home/itskhuramshahzad/reducerX.py -mapper mapperX.py -reducer ./reducerX.py -input /usr/hadoop/airline\_data.csv -output /usr/hadoop/\_output14

# Questions

## 4.6 WHAT WAS THE <KEY,VALUE> PAIR USED IN THIS QUERY?

Key was the Year

And value?

## 4.7 HOW MANY MAPPER THREADS WERE USED?

- Launched map tasks=5

## 4.8 HOW MANY MAPPER THREADS WERE USED?

- Launched reduce tasks=1

## 4.9 WHAT WAS THE TIME SPENT BY ALL MAPPER THREADS?

- Total time spent by all maps in occupied slots (ms)=86216
- Total time spent by all map tasks (ms)=86216
- Total vcore-milliseconds taken by all map tasks=86216
- Total megabyte-milliseconds taken by all map tasks=88285184

## 4.10 WHAT WAS THE TIME SPENT BY ALL REDUCER THREADS?

- Total time spent by all reduces in occupied slots (ms)=4100
- Total time spent by all reduce tasks (ms)=4100
- Total vcore-milliseconds taken by all reduce tasks=4100
- Total megabyte-milliseconds taken by all reduce tasks=4198400

## 4.11 WHAT IS THE FILE NAME IN WHICH YOUR OUTPUT IS LOCATED?

- Output directory: /usr/hadoop/\_output14

#### 4.12 VARIATION 1:

# of Reducer Tasks	airline_data.csv block size variation (Mb)				
	2 MB	4 MB	8 MB	16 MB	Default
2	514893	476332	59484	16296	18438
4	249571	132127	60183	17220	18104
8	250578	139141	57556	16322	18742
16	261119	136873	57444	16482	18706

- hdfs dfs -D mapred.reduce.tasks=8 -put /home/itskhuramshahzad/airline\_data.csv /usr/hadoop/airline\_dataD8.csv
- hdfs dfs -D dfs.blocksize=8388608 -D mapred.reduce.tasks=2 -put /home/itskhuramshahzad/airline\_data.csv /usr/hadoop/airline\_data24.csv

#### 4.13 VARIATION 2:

# of Mapper Tasks	airline_data.csv block size variation (Mb)				
	2 MB	4 MB	8 MB	16 MB	Default
2	2097152b	4194304b	8388608b	16777216	
4	251854s	136445	57065	16065s	17839
8	249292s	129286	57049	16484s	17700
16	259378s	136540	56928	16171s	18515
	250639s	138447	59177	16046s	18238

#### 4.14 VARIATION 3:

# of Mapper Tasks	# of Reducer Tasks			
	2	4	8	16
2	17218s	16662s	16820s	15892s
4	16049s	15582s	17067s	17236s
8	16371s	16160s	16018s	16728s
16	16097s	16579s	17044s	15997s

- hdfs dfs -D dfs.blocksize=16777216 -D mapred.map.tasks=2 -D mapred.reduce.tasks=2 -put /home/itskhuramshahzad/airline\_data.csv /usr/hadoop/airline\_data1622.csv

## 5 TASK -05: DESIGNING ADDITIONAL QUERIES

- Note: we have used best result input file for: no of mapper , no of reducer and block size.
- Command upload:

```
hdfs dfs -D dfs.blocksize=16777216 -D mapred.map.tasks=4 -D mapred.reduce.tasks=2 -put
/home/itskhuramshahzad/airline_data.csv /usr/hadoop/airline_data_best_results.csv
```

```
hadoop@instance-1:~$ hdfs dfs -ls /usr/hadoop/years_counts_output
Found 2 items
-rw-r--r-- 1 hdoop supergroup          0 2022-06-11 06:23 /usr/hadoop/years_counts_output/_SUCCESS
-rw-r--r-- 1 hdoop supergroup      139 2022-06-11 06:23 /usr/hadoop/years_counts_output/part-00000
hadoop@instance-1:~$ hdfs dfs -cat /usr/hadoop/years_counts_output/part-00000
2010    17575.0
2011    15585.0
2012    14387.0
2013    16089.0
2014    13980.0
2015    13528.0
2016    12217.0
2017    12518.0
2018    20231.0
2019    20946.0
year    1.0
<Key, Value> Pair used: <>
hadoop@instance-1:~$
```

so we have used this file for all 5 tasks.

## 5.1 TASK 5A: PRESENT THE TOTAL FLIGHTS PER YEAR AS A PERCENTAGE

<key, value>=	< key_year, value_flights >
Output Folder:	/usr/hadoop/Task_5a/output3/
Sample Output	
<pre>2022-06-21 04:23:48,831 INFO streaming.StreamJob: Output directory: /usr/hadoop/Task_5a/output3 hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5a/output3/part-00000 Year No: 2010    Total flight of year : 6450117.0      Total flight Percentage per Year: 10 % Year No: 2011    Total flight of year : 6085281.0      Total flight Percentage per Year: 9 % Year No: 2012    Total flight of year : 6096762.0      Total flight Percentage per Year: 9 % Year No: 2013    Total flight of year : 6369482.0      Total flight Percentage per Year: 10 % Year No: 2014    Total flight of year : 5819811.0      Total flight Percentage per Year: 9 % Year No: 2015    Total flight of year : 5819079.0      Total flight Percentage per Year: 9 % Year No: 2016    Total flight of year : 5617658.0      Total flight Percentage per Year: 8 % Year No: 2017    Total flight of year : 5674621.0      Total flight Percentage per Year: 9 % Year No: 2018    Total flight of year : 7213446.0      Total flight Percentage per Year: 11 % Year No: 2019    Total flight of year : 7422037.0      Total flight Percentage per Year: 11 % hadoop@instance-1:~\$</pre>	

### 5.1.1 Command used for this task

- hadoop@instance-1:~\$ hdfs dfs -put /home/itskhuramshahzad/5a\_mapperX.py /usr/hadoop/Task\_5a/5a\_mapperX.py
- hadoop@instance-1:~\$ hdfs dfs -put /home/itskhuramshahzad/5a\_reducerX.py /usr/hadoop/Task\_5a/5a\_reducerX.py
- hadoop jar \$HADOOP\_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -file /home/itskhuramshahzad/5a\_mapperX.py -file /home/itskhuramshahzad/5a\_reducerX.py -mapper 5a\_mapperX.py -reducer ./5a\_reducerX.py -input /usr/hadoop/airline\_data\_best\_results.csv -output /usr/hadoop/Task\_5a/output

### 5.1.2 Mapper Code:

```
1#!/usr/bin/python3
2import sys
3import csv
4for line in sys.stdin:
5    data = line.strip().split(",");
6    key_year= data[0]
7    if data[5].strip().isdigit():
8        value_flights= float (data[8].strip())
9    else:
10        value_flights=0
11    print("{0}\t{1}".format(key_year,value_flights))
12
```

### 5.1.3 Reducer Code:

```
5a_mapperX.py

1 #!/usr/bin/python3
2 import sys
3 import csv
4 total_year =0
5 total_flights =0
6 oldkey= None
7 f = open("data.txt", "w")
8 for line in sys.stdin:
9     data = line.strip().split("\t")
10    thiskey= data[0]
11    value_flights = data[1]
12    if thiskey!= oldkey and oldkey!= None:
13        with open('data.txt', 'a') as file:
14            file.write(str(oldkey))
15            file.write("\t")
16            file.write(str(total_flights))
17            file.write("\n")
18        oldkey= thiskey
19        total_flights=0
20    oldkey= thiskey
21    total_flights+= float(value_flights)
22 totalflights=0
23 with open('data.txt', 'r') as file:
24     csv_reader = csv.reader(file, delimiter='\t')
25     for data in csv_reader:
26         totalflights+= float (data[1])
27 with open('data.txt', 'r') as file:
28     csv_reader = csv.reader(file, delimiter='\t')
29     for data in csv_reader:
30         key_year= data[0]
31         flightcount= float (data[1])
32         percentage= int ((flightcount/totalflights)*100)
33         print("Year No: {0}\t Total flight of year : {1}\t Total flight Percentage per Year: {2} %"
34             .format(key_year, flightcount, percentage))
35
```

## 5.2 TASK 5B: PRESENT THE TOTAL FLIGHTS PER YEAR AS A PERCENTAGE

<key, value>=	<month, value_flights >
Output Folder:	/usr/hadoop/Task_5b/output/
Sample Output	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5b/output2/part-00000 The busiest Month is : 7      Total flight of this month : 5592214 hadoop@instance-1:~\$</pre>	

### 5.2.1 Mapper Code:

```
1#!/usr/bin/python3
2import sys
3f = open("data.txt", "w")
4for line in sys.stdin:
5    data = line.strip().split(",")
6    key_month= data[1]
7    if data[5].strip().isdigit():
8        value_flights= float (data[5].strip())
9    else:
10        value_flights=0
11    print("{0}\t{1}".format(key_month,value_flights))
12
```

### 5.2.2 Reducer Code:

```
1#!/usr/bin/python3
2import sys
3Dict = {1: 0, 2: 0, 3: 0, 4: 0, 5: 0, 6: 0, 7: 0, 8: 0, 9: 0, 10: 0, 11: 0, 12: 0}
4for line in sys.stdin:
5    data = line.strip().split("\t")
6    thiskey= data[0]
7    flights = data[1]
8    if thiskey.strip().isdigit():
9        val= int(Dict[int (thiskey)])
10       val+= float (flights)
11       Dict.update({int (thiskey): int (val)})
12busiest_month= 1
13busiest_month_flights =Dict[1];
14for x in Dict:
15    if int (Dict[x]) > busiest_month_flights:
16        busiest_month=x
17        busiest_month_flights= Dict[x]
18
19print("\nThe busiest Month is : {0}\tTotal flight of this month : {1}"
20.format(busiest_month, busiest_month_flights))
21
```

### 5.2.3 Command:

- hdfs dfs -put /home/itskhuramshahzad/5b\_reducerX.py /usr/hadoop/Task\_5b/5b\_reducerX.py
- hdfs dfs -put /home/itskhuramshahzad/5b\_reducerX.py /usr/hadoop/Task\_5b/5b\_reducerX.py
- hadoop jar \$HADOOP\_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -file /home/itskhuramshahzad/5b\_mapperX.py -file /home/itskhuramshahzad/5b\_reducerX.py -mapper 5b\_mapperX.py -reducer ./reducerX.py -iducerX.py -mapper 5b\_mapperX.py -reducer ./5b\_reducerX.py -input /usr/hadoop/airline\_data\_best\_results.csv -output /usr/hadoop/Task\_5b/output1
- hdfs dfs -cat /usr/hadoop/Task\_5b/output2/part-00000

### 5.3 TASK 5C: WHICH AIRLINE CARRIER HAS FLOWN THE MOST FLIGHTS OVER THE 10 YEAR PERIOD?

<key, value>=	<Airline, value_flights >
Output Folder:	/usr/hadoop/Task_5c/output/
Sample Output	
<pre>2022-06-22 18:58:10,396 INFO streaming.StreamJob: Output directory: /usr/hadoop/Task_5c/output1 hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5c/output1/part-00000  Airline Carrier : WN      has flown most flights over the 10 year period are: 12333317.0 hadoop@instance-1:~\$</pre>	

#### 5.3.1 Commands used for this task:

- hdfs dfs -mkdir /usr/hadoop/Task\_5c
- hdfs dfs -put /home/itskhuramshahzad/5c\_reducerX.py /usr/hadoop/Task\_5c/5c\_reducerX.py
- hdfs dfs -put /home/itskhuramshahzad/5c\_mapperX.py /usr/hadoop/Task\_5c/5c\_mapperX.py

#### 5.3.2 Mapper Code:

```
1#!/usr/bin/python3
2import sys
3import csv
4for line in sys.stdin:
5    data = line.strip().split(",")
6    key_airline= data[2]
7    if data[5].strip().isdigit():
8        value_flights= float (data[5].strip())
9    else:
10        value_flights=0
11    print("{0}\t{1}".format(key_airline,value_flights))
```

#### 5.3.3 Reducer Code:

```
1#!/usr/bin/python3
2import sys
3Dict = {}
4for line in sys.stdin:
5    data = line.strip().split("\t")
6    thiskey= data[0]
7    carrier = data[1]
8    if thiskey in Dict.keys():
9        val= float(Dict[thiskey])
10       val+= float (carrier)
11       Dict.update({thiskey: float (val)})
12    else:
13        Dict[thiskey]= float (carrier)
14busiest_carrier= ""
15busiest_carrier_flights =0;
16flag= True
17for x in Dict:
18    if flag:
19        busiest_carrier= x
20        busiest_carrier_flights= usiest_carrier_flights= Dict[x]
21        flag= False
22    elif float (Dict[x]) > busiest_carrier_flights:
23        busiest_carrier=x
24        busiest_carrier_flights= Dict[x]
25print("\nAirline Carrier : {0}\thas flown most flights over the 10 year period are: {1}"
26 .format(busiest_carrier, busiest_carrier_flights))
```

## 5.4 TASK 5D: WHICH AIRPORT HAS BEEN THE MOST BUSIEST OVER THE 10 YEAR?

<b>&lt;key, value&gt;=</b>	<b>&lt;airport, flights&gt;</b>
<b>Output Folder:</b>	/usr/hadoop/Task_5d/output/
<b>Sample Output</b>	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5d/output/part-00000 The Airport: ATL has been the most busiest over the 10 year period , flights: 3881775 hadoop@instance-1:~\$</pre>	

### 5.4.1 Command used in this task:

- hdfs dfs -mkdir /usr/hadoop/Task\_5d
- hdfs dfs -put /home/itskhuramshahzad/5d\_reducerX.py /usr/hadoop/Task\_5d/5d\_reducerX.py
- hdfs dfs -put /home/itskhuramshahzad/5d\_mapperX.py /usr/hadoop/Task\_5d/5d\_mapperX.py
- hadoop jar \$HADOOP\_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar -file /home/itskhuramshahzad/5d\_mapperX.py -file /home/itskhuramshahzad/5d\_reducerX.py -mapper 5d\_mapperX.py -reducer ./5d\_reducerX.py -input /usr/hadoop/airline\_data\_best\_results.csv -output /usr/hadoop/Task\_5d/output

### 5.4.2 Mapper Code:

```
1#!/usr/bin/python3
2import sys
3import csv
4for line in sys.stdin:
5    data = line.strip().split(",")
6    key_airport= data[4]
7    if data[5].strip().isdigit():
8        value_flights= int (data[5].strip())
9    else:
10        value_flights=0
11    value_percentage=0
12    print("{0}\t{1}".format(key_airport,value_flights))
13
14|
```

### 5.4.3 Reducer Code:

```
1#!/usr/bin/python3
2import sys
3Dict = {}
4for line in sys.stdin:
5    data = line.strip().split("\t")
6    thiskey= data[0]
7    flights = data[1]
8    if thiskey in Dict.keys():
9        val= int(Dict[thiskey])
10       val+= int (flights)
11       Dict.update({thiskey: int (val)})
12    else:
13        Dict[thiskey]= int (flights)
14busiest_airport= ""
15busiest_airport_flights =0;
16flag= True
17for x in Dict:
18    if flag:
19        busiest_airport= x
20        busiest_airport_flights= Dict[x]
21        flag= False
22    elif int (Dict[x]) > busiest_airport_flights:
23        busiest_airport=x
24        busiest_airport_flights= Dict[x]
25print("\n The Airport: {0} has been the most busiest over the 10 year period , flights: {1}"
26.format(busiest_airport, busiest_airport_flights))
```

## 5.5 TASK 5E: WHICH AIRPORT HAS THE LARGEST FLIGHTS TO CANCELLATION RATIO?

<key, value>=	<key_airport,value_flights, value_cancel>
Output Folder:	/usr/hadoop/Task_5e/output/
Sample Output	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5e/output/part-00000 The Airport: ORD has the Largest Flights to Cancellation Ratio [ 3028013 : 88288.0 ] = 34.296993928959765</pre>	

### 5.5.1 Mapper Code:

```
5e_mapperX.py
```

```
1#!/usr/bin/python3
2import sys
3value_flights=0
4value_cancel=0
5
6for line in sys.stdin:
7    data = line.strip().split(",");
8    key_airport= data[4]
9    if data[5].strip().isdigit() and data[15].strip().isdigit():
10        value_flights= int (data[5].strip())
11        value_cancel= int (data[15].strip())
12    else:
13        value_flights=0
14        value_cancel=0
15    print ("{}\\t{}\\t{}".format(key_airport,value_flights, value_cancel))
16
```

### 5.5.2 Reducer Code:

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

```
5e_mapperX.py          5e_reducerX.py
```

```
1 #!/usr/bin/python3
2 import sys
3 Dict = {}
4 for line in sys.stdin:
5     data = line.strip().split("\t")
6     thiskey= data[0]
7     flights = data[1]
8     cancellation = data[2]
9     if thiskey in Dict.keys():
10         val= Dict[thiskey]
11         val_1 = val['0'];
12         val_2 = val['1'];
13         val_1+= int (flights)
14         val_2+= float (cancellation)
15         new_entry = {"0": val_1, "1": val_2}
16         Dict[thiskey].update(new_entry)
17         #print("airport: {0}\t{1}\t{2}".format(thiskey, val_1, val_2))
18     else:
19         val_1= int (flights)
20         val_2= float (cancellation)
21         new_entry = {"0": val_1, "1": val_2}
22         Dict[thiskey]= new_entry
23
24         val_1= int (flights)
25         val_2= float (cancellation)
26         new_entry = {"0": val_1, "1": val_2}
27         Dict[thiskey]= new_entry
28
29 airport= ""
30 airport_flights =0;
31 airport_cancellation=0
32 flag= True
33 for x in Dict:
34     val= Dict[x]
35     flights= val['0']
36     cancellation= val['1']
37     if flag:
38         airport= x
39         airport_flights= val['0']
40         airport_cancellation= val['1']
41         flag= False
42     elif float (cancellation) > airport_cancellation:
43         val= Dict[x]
44         airport= x
45         airport_flights= val['0']
46         airport_cancellation= val['1']
47 print("\n The Airport: {0} has the Largest Flights to Cancellation Ratio [ {1} : {2} ] = {3}" I
48 .format(airport, airport_flights, airport_cancellation,airport_flights/airport_cancellation ))
```

## 5.6 TASK 5F: FIND THE TOTAL AMOUNT OF DELAY MINUTES GROUPED BY AIRLINE

<key, value>=	<airline, arr_delay>
Output Folder:	/usr/hadoop/Task_5f/output4/
Sample Output (Sorted in descending Order)	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5f/output4/part-00000 (1) The Airline: WN Total Amount of Delay Minutes Grouped are: 118025352 (2) The Airline: AA Total Amount of Delay Minutes Grouped are: 83920254 (3) The Airline: OO Total Amount of Delay Minutes Grouped are: 78760906 (4) The Airline: DL Total Amount of Delay Minutes Grouped are: 75761785 (5) The Airline: EV Total Amount of Delay Minutes Grouped are: 66347717 (6) The Airline: UA Total Amount of Delay Minutes Grouped are: 62923308 (7) The Airline: B6 Total Amount of Delay Minutes Grouped are: 40695644 (8) The Airline: MQ Total Amount of Delay Minutes Grouped are: 37970355 (9) The Airline: US Total Amount of Delay Minutes Grouped are: 17821781 (10) The Airline: F9 Total Amount of Delay Minutes Grouped are: 14194561</pre>	

### 5.6.1 Mapper Code:

```
GNU nano 4.8
#!/usr/bin/python3
import sys
value_flights=0
value_cancel=0
for line in sys.stdin:
    data = line.strip().split(",")
    key_airline= data[2]
    if data[17].strip().isdigit():
        arr_delay= int (data[17].strip())
        <key_airline, arr_delay>
    else:
        arr_delay=0
    print("{0}\t{1}".format(key_airline, arr_delay))
```

## BIG DATA ANALYTICS ASSIGNMENT (Hadoop)

### 5.6.2 Reducer Code:

```
GNU nano 4.8
#!/usr/bin/python3
import sys
import collections
Dict = {}
for line in sys.stdin:
    data = line.strip().split("\t")
    thiskey= data[0]
    arr_delay = data[1]
    val=0
    if thiskey in Dict.keys():
        val= Dict[thiskey]
        val+= int (arr_delay)
        Dict.update({thiskey: val})
    else:
        val+= int (arr_delay)
        Dict[thiskey]= val
airline= ""
airline_delay =0;
sorted_dict =sorted(Dict.items(), key=lambda x: x[1], reverse=True)
flag =0
for x in sorted_dict:
    if x!= "airport":
        airline= x[0]
        airline_delay= x[1]
        flag+=1
        print(" ({0}) The Airline: {1} Total Amount of Delay Minutes Grouped are: {2} "
              .format(flag, airline, airline_delay ))
        if flag==10:
            break
5f_reducerX.py
```

## 5.7 TASK 5G: FIND THE AIRPORT WITH MOST CANCELED FLIGHTS IN 2016.

<key, value>=	<key_airport, arr_cancelled>
Output Folder:	/usr/hadoop/Task_5g/output/
Sample Output	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5g/output/part-00000 /home/itskhuramshahzad/5g_reducerX.py -mapper 5g_mapperX.py -reducer /5g_reducerX.py -input /usr/hadoop/airline.data best_results.csv -output The Airport: WN with most Cancelled Flights in 2016 are: 15866.0 hadoop@instance-1:~\$</pre>	

### 5.7.1 Mapper Code:

```
1#!/usr/bin/python3
2import sys
3value_flights=0
4value_cancel=0
5for line in sys.stdin:
6    data = line.strip().split(",")
7    key_year= data[0]
8    key_airport= data[2]
9    if key_year == "2016":
10        if data[15].strip().isdigit():
11            arr_delay= float (data[15].strip())
12        else:
13            arr_delay=0
14        print("{0}\t{1}".format(key_airport,arr_cancelled))
15
16
```

### 5.7.2 Reducer Code:

```
1#!/usr/bin/python3
2import sys
3import collections
4Dict = {}
5for line in sys.stdin:
6    data = line.strip().split("\t")
7    thiskey= data[0]
8    cancellation = data[1]
9    if thiskey in Dict.keys():
10        val= Dict[thiskey]
11        val+= int (cancellation)
12        Dict.update({thiskey: val})
13    else:
14        val_1= int (cancellation)
15        Dict[thiskey]= val_1
16airline= ""
17airline_cancellation =0;
18sorted_dict =sorted(Dict.items(), key=lambda x: x[1], reverse=True)
19flag =0
20for x in sorted_dict:
21    if x!= "arr_cancelled":
22        airline= x[0]
23        airline_cancellation= x[1]
24        flag+=1
25        print("The Airport: {0} with most Cancelled Flights in 2016 are: {1} "
26        .format(airline, airline_cancellation ))
27        if flag==1:
28            break
```

## 5.8 TASK 5H: FIND THE AVERAGE DELAY TIME FOR AN AIRPORT THAT IS THE MOST BUSIEST OF ALL OTHER AIRPORTS.

<b>&lt;key, value&gt;</b>	<b>&lt;key_airport, value_flights, value_delay&gt;</b>
Output Folder:	/usr/hadoop/Task_5h/output/
<b>Sample Output</b>	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5h/output/part-00000 The Airport: ATL the most busiest of all other airports flights: 3881775, the average delay time: 5.692734664422602 hadoop@instance-1:~\$</pre>	

### 5.8.1 Mapper Code

```

1 #!/usr/bin/python3
2 import sys
3 value_flights=0
4 value_delay=0
5
6 for line in sys.stdin:
7     data = line.strip().split(" ")
8     key_airport= data[4]
9     if data[5].strip().isdigit() and data[17].strip().isdigit():
10         value_flights= int (data[5].strip())
11         value_delay= int (data[17].strip())
12     else:
13         value_flights=0
14         value_delay=0
15     print ("{} \t {} \t {}".format(key_airport,value_flights, value_delay))
16
17
18

```

### 5.8.2 Reducer Code:

```

1 #!/usr/bin/python3
2 import sys
3 Dict = {}
4 for line in sys.stdin:
5     data = line.strip().split("\t")
6     thiskey= data[0]
7     flights = data[1]
8     delay = data[2]
9     if thiskey in Dict.keys():
10         val= Dict[thiskey]
11         val_1 = val['0'];
12         val_2 = val['1'];
13         val_1+= int (flights)
14         val_2+= float (delay)
15         new_entry = {"0": val_1, "1": val_2}
16         Dict[thiskey].update(new_entry)
17         #print("airport: {} \t {} \t {}".format
18     else:
19         val_1= int (flights)
20         val_2= float (delay)
21         new_entry = {"0": val_1, "1": val_2}
22         Dict[thiskey]= new_entry
23 airport= ""
24 airport_flights =0;
25 airport_delay=0
26 total_delay=0
27
28 flag= True
29
```

## 5.9 TASK 5J: WHAT IS THE PROBABILITY THAT A FLIGHT WILL BE CANCELLED DUE TO BAD WEATHER AT THE MOST BUSIEST AIRPORT OF ALL OTHER AIRPORTS?

<key, value>	<key_airport,value_flights, value_bad_weather>
5.9.1 Output Folder:	/usr/hadoop/Task_5j/output/
<b>Sample Output</b>	
<pre>hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5j/output/part-00000 Most Busyest Airport: ATL has Probability that a Flight will be delayed due to Bad Weather is: 0.46171738444397215 hadoop@instance-1:~\$</pre>	

## 5.10 TASK 5K: FIND OUT THE AIRPORT THAT HAVE HIGHEST SECURITY DELAY THAN OF ALL OTHER AIRPORTS?

<key, value>	<key_airport,security_delay>
Output Folder:	/usr/hadoop/Task_5k/output/
<b>Sample Output</b>	
<pre>2022-06-23 04:59:21,599 INFO streaming.StreamJob: Output directory: /usr/hadoop/Task_5k/output hadoop@instance-1:~\$ hdfs dfs -cat /usr/hadoop/Task_5k/output/part-00000 The Airport: LAX has the highest security delay is : 46406.0 minuites hadoop@instance-1:~\$</pre>	

**5.10.1 Mapper code:**

```
Sk_reducerX.py
1#!/usr/bin/python3
2import sys
3for line in sys.stdin:
4    data = line.strip().split(",");
5    key_airport= data[4]
6    if data[21].strip().isdigit():
7        security_delay= int (data[21].strip())
8    else:
9        security_delay=0
10   print("{0}\t{1}".format(key_airport,security_delay))
11
12
13
```

**5.10.2 Reducer Code:**

```
Sk_reducerX.py
1#!/usr/bin/python3
2import sys
3Dict = {}
4for line in sys.stdin:
5    data = line.strip().split("\t")
6    thiskey= data[0]
7    security_delay = data[1]
8    if thiskey in Dict.keys():
9        val= Dict[thiskey]
10       val+= float (security_delay)
11       Dict.update({thiskey: val})
12    else:
13        val_1= float (security_delay)
14        Dict[thiskey]= val_1
15airport= ""
16security_delay=0
17flag= True
18for x in Dict:
19    val= Dict[x]
20    if flag:
21        airport= x
22        security_delay= val
23        flag= False
24    elif float (val) > security_delay:
25        val= Dict[x]
26        airport= x
27        security_delay= val
28print("\n The Airport: {0} has the highest security delay is : {1} minuite".format(airport, security_delay))
29
```