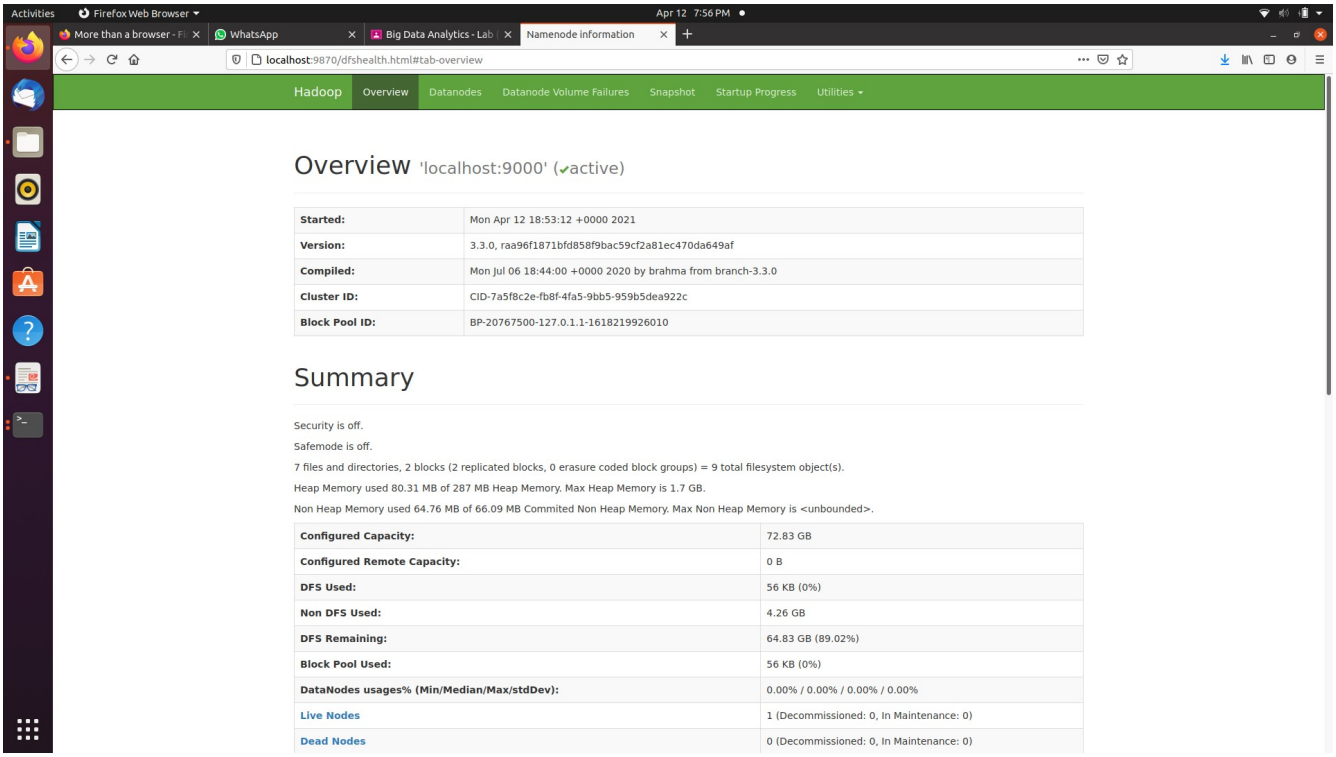
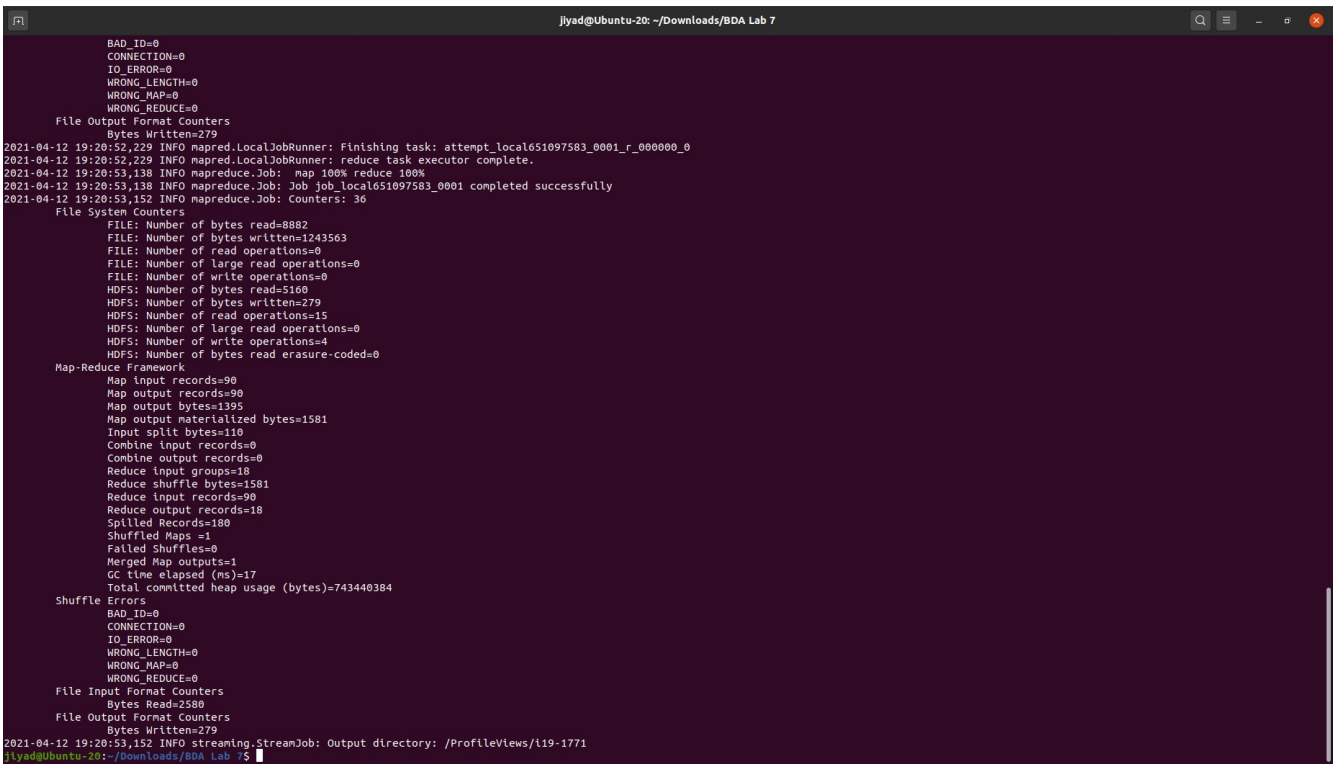
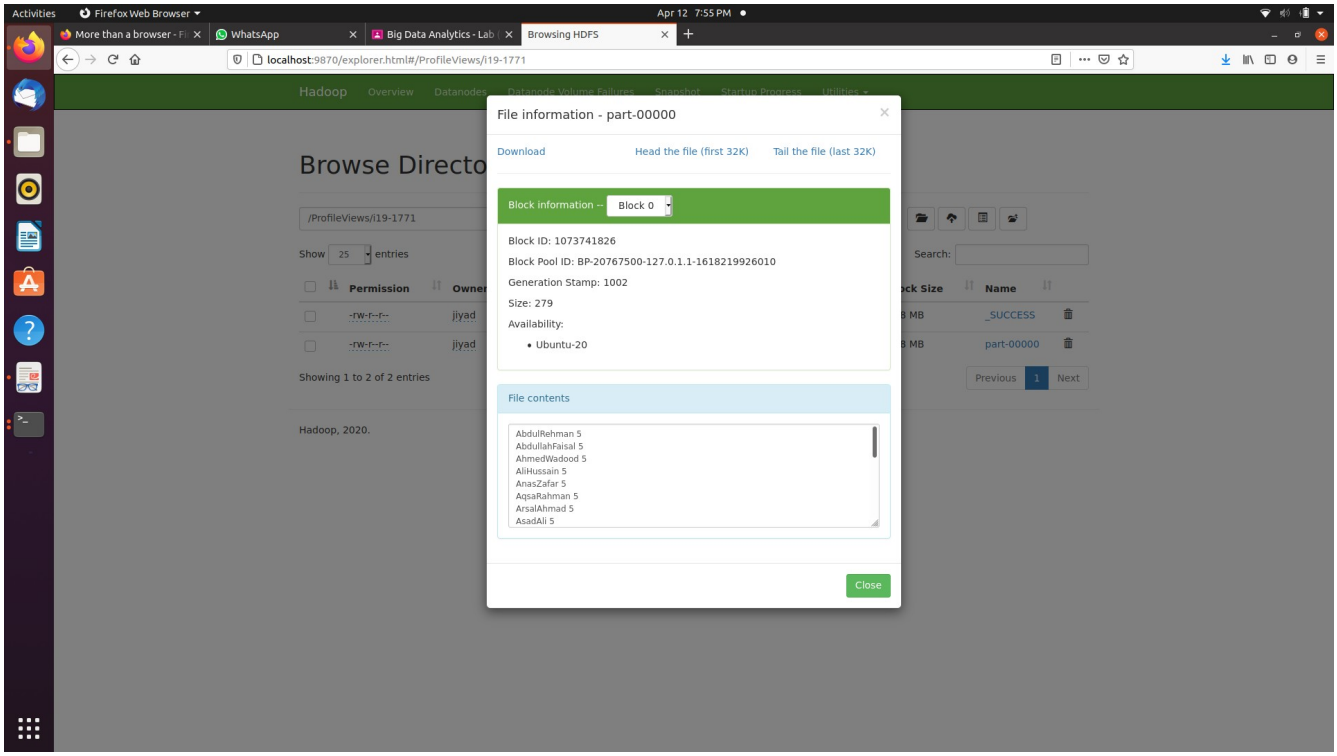

Activities
Firefox Web Browser
Apr 12 7:56 PM
More than a browser - Fi
WhatsApp
Big Data Analytics - Lab
Namenode information
localhost:9870/dfshealth.html#tab-overview
Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities
Overview 'localhost:9000' (active)
Started: Mon Apr 12 18:53:12 +0000 2021
Version: 3.3.0, raa96f1871bfd858f9bac59cf2a81ec470da649af
Compiled: Mon Jul 06 18:44:00 +0000 2020 by brahma from branch-3.3.0
Cluster ID: CID-7a5f8c2e-fb8f-4fa5-9bb5-959b5dea922c
Block Pool ID: BP-20767500-127.0.1.1-1618219926010
Summary
Security is off.
Safemode is off.
7 files and directories, 2 blocks (2 replicated blocks, 0 erasure coded block groups) = 9 total filesystem object(s).
Heap Memory used 80.31 MB of 287 MB Heap Memory. Max Heap Memory is 1.7 GB.
Non Heap Memory used 64.76 MB of 66.09 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.
Configured Capacity: 72.83 GB
Configured Remote Capacity: 0 B
DFS Used: 56 KB (0%)
Non DFS Used: 4.26 GB
DFS Remaining: 64.83 GB (89.02%)
Block Pool Used: 56 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev): 0.00% / 0.00% / 0.00% / 0.00%
Live Nodes 1 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes 0 (Decommissioned: 0, In Maintenance: 0)


jiyad@Ubuntu-20: ~/Downloads/BDA Lab 7
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Output Format Counters
Bytes Written=279
2021-04-12 19:20:52,229 INFO mapred.LocalJobRunner: Finishing task: attempt_local651097583_0001_r_000000_0
2021-04-12 19:20:52,229 INFO mapred.LocalJobRunner: reduce task executor complete.
2021-04-12 19:20:53,138 INFO mapreduce.Job: map 100% reduce 100%
2021-04-12 19:20:53,138 INFO mapreduce.Job: Job job_local651097583_0001 completed successfully
2021-04-12 19:20:53,152 INFO mapreduce.Job: Counters: 36
File System Counters
FILE: Number of bytes read=8882
FILE: Number of bytes written=1243563
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=5160
HDFS: Number of bytes written=279
HDFS: Number of read operations=15
HDFS: Number of large read operations=0
HDFS: Number of write operations=4
HDFS: Number of bytes read erasure-coded=0
Map-Reduce Framework
Map input records=90
Map output records=90
Map output bytes=1395
Map output materialized bytes=1581
Input split bytes=110
Combine input records=0
Combine output records=0
Reduce input groups=18
Reduce shuffle bytes=1581
Reduce input records=90
Reduce output records=18
Spilled Records=180
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=17
Total committed heap usage (bytes)=743440384
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=2580
File Output Format Counters
Bytes Written=279
2021-04-12 19:20:53,152 INFO streaming.StreamJob: Output directory: /ProfileViews/i19-1771
jiyad@Ubuntu-20:~/Downloads/BDA Lab 7$



Mapper Code:

```
# input comes from STDIN (standard input)
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()
    # split the line into words
    words = line.split()
    y=[]
    k=words[-2]+words[-1]

    y.append(k)
    # increase counters
    for word in y:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e. the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        print (word, 1)
```

Reducer Code:

```
rom operator import itemgetter
import sys


current_word = None
current_count = 0
word = None


# input comes from STDIN
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()


    # parse the input we got from mapper.py
    word, count = line.split(' ')


    # convert count (currently a string) to int
    try:
        count = int(count)
    except ValueError:
        # count was not a number, so silently
        # ignore/discard this line
        continue


    # this IF-switch only works because Hadoop sorts map output
    # by key (here: word) before it is passed to the reducer
    if current_word == word:
        current_count += count
    else:
        if current_word:
            # write result to STDOUT
            print (current_word, current_count)
        current_count = count
        current_word = word


# do not forget to output the last word if needed!
if current_word == word:
    print (current_word, current_count)
```