

HW Assignment 1 - CS267-Fall 2023

Hadoop/MapReduce Hands-on Exercises

Name : Sanket Kulkarni

Assignment Report Content:

Sr. No	Task Name	Page No.
1	Setup	2
2	SnapShot of Setup	6
2	Exercise - 1	7
3	Question - 1	8
4	Exercise - 2	9
5	Question - 2	10
6	Exercise - 3	12
7	Question - 3	15
8	Question - 4	16
9	Exercise - 4	17
10	Question - 5	21
11	Outcomes	27
12	References	27

Setup:

Steps	Errors Encountered
1. Instead of going with virtual box or vmware, I chose to dual boot my machine with ubuntu. I booted a usb with ubuntu OS and then installed ubuntu as well.	While installing the new booted OS, the boot enable flag of my BIOS was disabled, so I had to debug and enable this.
2. Installed JRE with version 11 and added the environment vars	N/A
3. Installed JDK	Added the path to env var so that it is accessible from hadoop env
4. Installed Hadoop and added its path to ./bashrc	N/A
5. I referred to the 7th point of setup on ubuntu given in the assignment. I tried setting up a multi node cluster as well, similar to one we have in production. While doing so, I changed Hadoop-env.xml, core-site.xml, hdfs-site.xml , yarn-site.xml	I did not place the right java home path initially, it was pointing to some other location. After debugging changed the file and jar file was able to run.
6. I created one hadoop user on my machine and now the directories of hadoop are completely isolated from the other code.	N/A
7. Installation of Python	N/A
8. Installation of Eclipse and setting up the project repo from where we will be exporting the jar files	Jar files needed in import of mapper and reducer were explicitly copied from hadoop commons to code repository.

Primary commands used in the setup:

```
# Download packages and install Java and SSH
$ sudo apt-get update
$ sudo apt-get install openjdk-11-jdk
$ sudo apt-get install openssh-server

# Update JAVA path in your bashrc file
$ sudo vi ~/.bashrc OR $ sudo gedit ~/.bashrc
  export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
$ source ~/.bashrc
```

Creating hadoop :

```
sudo adduser -ingroup hadoop hduser
```

Generate the SSH Key:

```
ssh-keygen -t rsa -P ""
```

enable SSH access:

```
sudo cat /home/hduser/.ssh/id_rsa.pub >> /home/hduser/.ssh/authorized_keys
```

```
$ sudo vi /usr/local/hadoop/conf/core-site.xml
```

To update below config files, using above command opening each file, one after the other

Open the mapred-site.xml using the vi editor -

```
$ sudo vi /usr/local/hadoop/conf/mapred-site.xml
```

To configure master:

```
$ sudo vi /usr/local/hadoop/conf/masters
```

To change the configs of HDFS-SITE.xml

```
sudo vi /usr/local/hadoop/conf/hdfs-site.xml
```

Setup: Config files screenshots

Conf/*-site.xml

```
<configuration>

<property>

<name>hadoop.tmp.dir</name>

<value>/home/snagpal/Desktop/hadoopdata</value>

<description></description>

</property>

<property> <

<name>fs.defaultFS</name>

<value>hdfs://snagpal:54310</value>

<!-- <description>The Name of tegh default file system</description>-->

</property>

</configuration>
```

Conf/* mapred-site.xml

2. Conf/* mapred-site.xml

The next file to be configured is mapred-site.xml

```
<property>

<name>mapred.job.tracker</name>

<value>localhost:54311</value><!--localhost denotes the hostname-- > <description>The
host and port that the MapReduce job tracker runs at. If "local", then jobs are run in-
process as a single map and reduce task.

</description>

</property>
```

Conf/* hdfs-site.xml

```
<property>

<name>dfs.replication</name>

<value>1</value>

<description>Default block replication. The actual number of replications can be
specified when the file is created. The default is used if replication is not specified
in create time.

</description>

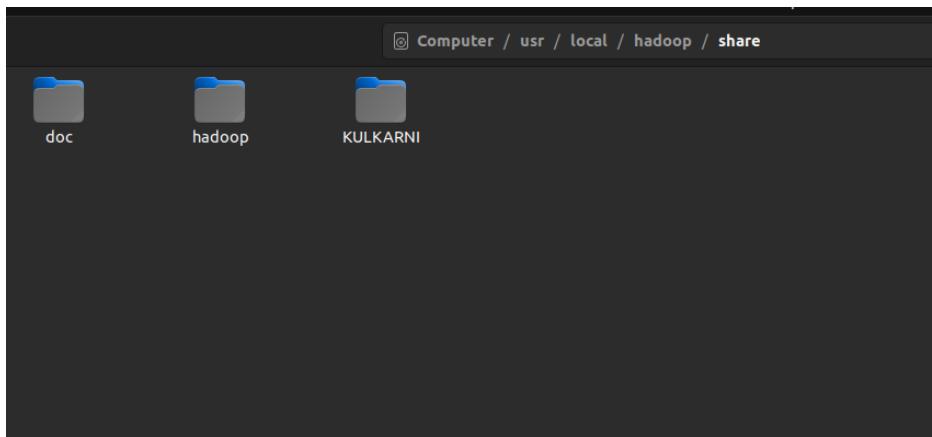
</property>
```

Snapshots:

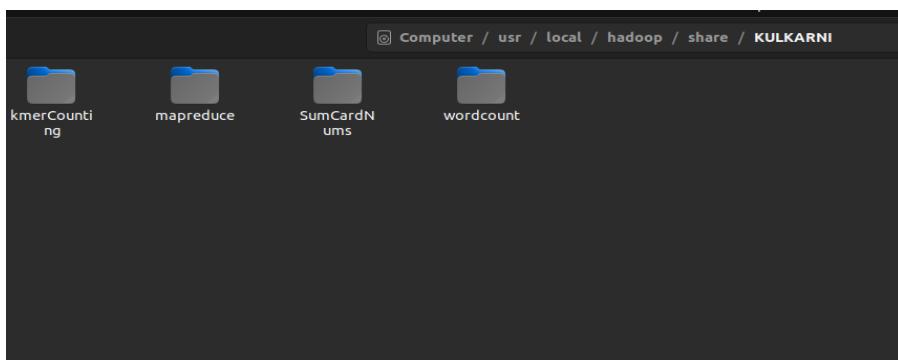
A Snapshot from .bashrc file in ubuntu, which showcases the Hadoop environment vars

```
export HADOOP_HOME=/usr/local/hadoop  
  
export HADOOP_INSTALL=$HADOOP_HOME  
  
export HADOOP_MAPRED_HOME=$HADOOP_HOME  
  
export HADOOP_COMMON_HOME=$HADOOP_HOME  
  
export HADOOP_HDFS_HOME=$HADOOP_HOME  
  
export YARN_HOME=$HADOOP_HOME  
  
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native  
  
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin  
  
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
```

Folder Structure Snapshot:



Snapshot of LASTNAME folder created with all code jars and files inside



Exercise 1:

To run exercise one

- Created the KULKARNI (My last name) folder in the share folder inside hadoop.
 - Copied mapreduce examples inside this folder
 - Ran the pi value code using map reduce example.
 - As asked, ran with 5 mapper and 5 reducer combination in the first go
 - Achieved pi value as ~ 3.68

```
hadoop@kulkarni-OptiPlex-5090:~$ hadoop jar /usr/local/hadoop/share/KULKARNI/mapreduce/hadoop-mapreduce-examples-2.8.1.jar pi 5 5
Number of Maps = 5
Samples per Map = 5
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstanc
ANCE()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Wrote Input for Map #0
Wrote Input for Map #1
Wrote Input for Map #2
Wrote Input for Map #3
Wrote Input for Map #4
Starting Job
23/09/10 20:07:00 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
23/09/10 20:07:00 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
23/09/10 20:07:01 INFO input.FileInputFormat: Total input files to process : 5
23/09/10 20:07:01 INFO mapreduce.Job: 2014-09-23 20:07:01 JobID: job_local9208806876_0001
23/09/10 20:07:01 INFO mapreduce.Job: 2014-09-23 20:07:01 User: hadoop
23/09/10 20:07:01 INFO mapreduce.Job: 2014-09-23 20:07:01 JobUrl: http://localhost:8080/
23/09/10 20:07:01 INFO mapreduce.Job: 2014-09-23 20:07:01 Job: job_local9208806876_0001
23/09/10 20:07:01 INFO mapred.LocalJobRunner: OutputCommitter set in config null
23/09/10 20:07:01 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/10 20:07:01 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 20:07:01 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
23/09/10 20:07:01 INFO mapred.LocalJobRunner: Waiting for map tasks
23/09/10 20:07:01 INFO mapred.LocalJobRunner: Starting task; attempt_local9208806876_0001_m_000000_0
23/09/10 20:07:01 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/10 20:07:01 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 20:07:01 INFO mapred.Task: Using NewSecondaryCalculatorForPreserve : []
23/09/10 20:07:01 INFO mapred.Task: 2014-09-23 20:07:01 Task ID: attempt_local9208806876_0001_m_000000_0
23/09/10 20:07:01 INFO mapred.MapTask: (EQUATOR) 0 kvlt 26214396(104857568)
23/09/10 20:07:01 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
23/09/10 20:07:01 INFO mapred.MapTask: soft limit at 83886080
23/09/10 20:07:01 INFO mapred.MapTask: bufSize = 0; bufvfd = 104857600
23/09/10 20:07:01 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
23/09/10 20:07:01 INFO mapred.MapTask: mapred.localJobBufferCollector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
23/09/10 20:07:01 INFO mapred.LocalJobRunner:
23/09/10 20:07:01 INFO mapred.MapTask: Starting flush of map output
23/09/10 20:07:01 INFO mapred.MapTask: Spilling map output
23/09/10 20:07:01 INFO mapred.MapTask: bufstart = 0; bufend = 18; bufvfd = 104857600
23/09/10 20:07:01 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26214392(104857568); length = 5/6553600
23/09/10 20:07:01 INFO mapred.Task: Task attempt_local9208806876_0001_m_000000_0 is done. And is in the process of committing
23/09/10 20:07:02 INFO mapred.LocalJobRunner: map
23/09/10 20:07:02 INFO mapred.Task: attempt_local9208806876_0001_m_000000_0 is done.
23/09/10 20:07:02 INFO mapred.LocalJobRunner: Finishing task attempt_local9208806876_0001_m_000000_0
23/09/10 20:07:02 INFO mapred.LocalJobRunner: Starting task; attempt_local9208806876_0001_m_000001_0
23/09/10 20:07:02 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/10 20:07:02 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
```

Output image for pi 5, 5:

As asked in the second task,

- Ran with 10 mapper and 10 reducer combination in the second go
 - Achieved pi value as ~ 3.20

```

cp: target './KULKARNI/' is not a directory
hadoop@sanket-Lenovo:/usr/local/hadoop/share/hadoop$ cd ..
hadoop@sanket-Lenovo:/usr/local/hadoop/share$ cp -r hadoop/mapreduce ./KULKARNI/
hadoop@sanket-Lenovo:/usr/local/hadoop/share$ cd KULKARNI/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI$ ls
mapred.jar
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI$ hadoop jar /usr/local/hadoop/share/KULKARNI/mapreduce/hadoop-mapreduce-examples-2.8.1.jar pi 10 10
Number of Maps = 10
Samples per Map = 10
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInst
ance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Wrote input for Map #3
Wrote input for Map #4
Wrote input for Map #5
Wrote input for Map #6
Wrote input for Map #7
Wrote input for Map #8
Wrote input for Map #9
Starting Job
23/09/10 19:16:47 INFO Configuration.deprecation: session_id is deprecated. Instead, use dfs.metrics.session-id
23/09/10 19:16:47 INFO Configuration.deprecation: mapred.local.dir is deprecated. Instead, use dfs.local.dir
23/09/10 19:16:47 INFO InputFormat: Total input files to process : 10
23/09/10 19:16:47 INFO mapreduce.JobSubmitter: number of splits:10
23/09/10 19:16:48 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local330743653_0001
23/09/10 19:16:48 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
23/09/10 19:16:48 INFO mapreduce.Job: Running job: job_local330743653_0001
23/09/10 19:16:48 INFO mapred.LocalJobRunner: OutputCommitter set in configuration.
23/09/10 19:16:48 INFO OutputCommitter: File Output Committer algorithm version is 1
23/09/10 19:16:48 INFO OutputCommitter: Fileoutputcommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 19:16:48 INFO mapred.LocalJobRunner: OutputCommitter: org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
23/09/10 19:16:48 INFO mapred.LocalJobRunner: Waiting for map tasks
23/09/10 19:16:48 INFO mapred.LocalJobRunner: Starting task: attempt_local330743653_0001_m_000000
23/09/10 19:16:48 INFO mapred.LocalJobRunner: Fileoutputcommitter algorithm version is 1
23/09/10 19:16:48 INFO mapred.LocalJobRunner: Fileoutputcommitter cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 19:16:48 INFO mapred.Task: Using ResourceCalculatorProcessTree : []
23/09/10 19:16:48 INFO mapred.MapTask: Processing split: file:/usr/local/hadoop/share/KULKARNI/QuasiMonteCarlo_169439866054_468189241/.ln/part4:0+118
23/09/10 19:16:49 INFO mapred.MapTask: (EQUATOR) 0 kv1 26214396(10485764)
23/09/10 19:16:49 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100

```

Output Image for pi 10 10:

```

task_local330743653_0001_r_000000
23/09/10 19:16:51 INFO mapred.LocalJobRunner: reduce > reduce
23/09/10 19:16:51 INFO mapred.Task: Task attempt_local330743653_0001_r_000000_0 done.
23/09/10 19:16:51 INFO mapred.LocalJobRunner: Finishing task: attempt_local330743653_0001_r_000000_0
23/09/10 19:16:51 INFO mapred.LocalJobRunner: reduce task executor complete.
23/09/10 19:16:51 INFO mapreduce.Job: map 100% reduce 100%
23/09/10 19:16:51 INFO mapreduce.Job: Job job_local330743653_0001 completed successfully
23/09/10 19:16:51 INFO mapreduce.Counters: 
  File System Counters
    FILE: Number of bytes read=3413926
    FILE: Number of bytes written=6902795
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
  Map-Reduce Framework
    Map input records=10
    Map output records=20
    Map output bytes=180
    Map output materialized bytes=280
    Input split bytes=1536
    Combine input records=0
    Combine output records=0
    Reduce input groups=2
    Reduce shuffle bytes=280
    Reduce input records=20
    Redundant Map output records=0
    Spilled Records=46
    Shuffled Maps =10
    Failed Shuffles=0
    Merged Map outputs=10
    GC time elapsed (ms)=77
    Total committed heap usage (bytes)=2595225600
  Shuffle Errors
    BAD_ID=0
    CONNECTION=0
    IO_ERROR=0
    WRONG_LENGTH=0
    NETWORK_ERROR=0
    WRONG_REDUCE=0
  File Input Format Counters
    Bytes Read=1300
  File Output Format Counters
    Bytes Written=109
Job Finished in 3 seconds
Estimated value of PI is 3.20000000000000000000000000000000
hadoop@sanket-Lenovo: /usr/local/hadoop/share/KULKARNI$ 

```

Question 1: What is the result of pi value of pi for each case? What is the difference?

Answer:

With the combination of 5 mapper and 5 reducer

- PI value : 3.6800000000000000000000000000000

With the combination of 10 mapper and 10 reducer

- PI Value: 3.2000000000000000000000000000000

Observation: When we keep the mapper and reducer combination lower, the approx estimation is more error redundant. As the number of mappers and reducers increased, the PI value was estimated and then combined more accurately.

Exercise 2:

The next program to test is the Hadoop word count program. This example reads text files and counts how often words occur.

```
hadoop@sanket-Lenovo:~/Desktop/Java/hadoop$ hadoop jar /usr/local/hadoop/share/KULKARNI/mapreduce/hadoop-mapreduce-examples-2.8.1.jar wordcount /usr/local/hadoop/share/KULKARNI/wordcou
nt/input /usr/local/hadoop/share/KULKARNI/wordcount/output
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInst
ance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: All illegal access operations will be denied in a future release
23/09/10 22:07:58 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session.id
23/09/10 22:07:58 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
23/09/10 22:07:59 INFO InputFileInputFormat: Total input files in process : 1
23/09/10 22:07:59 INFO mapreduce.JobSubmissionHandler: Number of splits:1
23/09/10 22:07:59 INFO mapreduce.JobSubmissionHandler: Submitting job for job: job_local1808203099_0001
23/09/10 22:07:59 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
23/09/10 22:07:59 INFO mapreduce.Job: Running job: job_local1808203099_0001
23/09/10 22:07:59 INFO mapred.LocalJobRunner: OutputCommitter set in config null
23/09/10 22:07:59 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/10 22:07:59 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 22:07:59 INFO mapred.LocalJobRunner: Waiting for map tasks
23/09/10 22:07:59 INFO mapred.LocalJobRunner: Starting task: attempt_local1808203099_0001_m_000000_0
23/09/10 22:07:59 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/10 22:07:59 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/10 22:07:59 INFO mapred.Task: Using ResourceCalculatorProcessTree : [ ]
23/09/10 22:07:59 INFO mapred.MapTask: Using Mapper spilt file:///usr/local/hadoop/share/KULKARNI/wordcount/input/bibleverses.nopunc:0+9068074
23/09/10 22:07:59 INFO mapred.MapTask: EQUATOR: 1048576000
23/09/10 22:07:59 INFO mapred.MapTask: mapstart = 0; bufvoid = 104857600
23/09/10 22:07:59 INFO mapred.MapTask: kvstart = 20214396; length = 6553600
23/09/10 22:08:00 INFO mapred.MapTask: Map input collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
23/09/10 22:08:00 INFO mapreduce.Job: map_local1808203099_0001 running in uber mode : false
23/09/10 22:08:00 INFO mapreduce.Job: map 0% reduce 0%
23/09/10 22:08:00 INFO mapred.LocalJobRunner: 
23/09/10 22:08:00 INFO mapred.MapTask: Starting flush of map output
23/09/10 22:08:00 INFO mapred.MapTask: Spilling map output
23/09/10 22:08:00 INFO mapred.MapTask: bufend = 15919397; bufvoid = 104857600
23/09/10 22:08:00 INFO mapred.MapTask: kvstart = 20214396(104857584); kvend = 19727208(77168832); length = 6937189/6553600
23/09/10 22:08:02 INFO mapred.MapTask: Flintstart spill 0
23/09/10 22:08:02 INFO mapred.Task: Taskattempt_local1808203099_0001_m_000000_0 is done. And is in the process of committing
23/09/10 22:08:02 INFO mapred.Task: map
23/09/10 22:08:02 INFO mapred.Task: Task 'attempt_local1808203099_0001_m_000000_0' done.
23/09/10 22:08:02 INFO mapred.LocalJobRunner: Finishing task: attempt_local1808203099_0001_m_000000_0
23/09/10 22:08:02 INFO mapred.LocalJobRunner: map had executed complete
```

```
Reduce input records=41788
Reduce output records=41788
Spilled Records=83576
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=76
Total committed heap usage (bytes)=308281344
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=9068074
File Output Format Counters
  Bytes Written=450684
hadoop@sanket-Lenovo:~/usr/local/hadoop/share/KULKARNI/wordcount/input$ hadoop fs -get /usr/local/hadoop/share/KULKARNI/wordcount/output output
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInst
ance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use -illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
hadoop@sanket-Lenovo:~/usr/local/hadoop/share/KULKARNI/wordcount/input$ cd
hadoop@sanket-Lenovo:~$ ls
auth.key auth_key.pub core-site.xml hadoop-2.8.1.tar.gz hadoop-env.sh hdfs-site.xml snap
hadoop@sanket-Lenovo:~$ cd /usr/local/hadoop/share/KULKARNI/wordcount/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/wordcount$ ls
Input  output
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/wordcount$ cd output/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/wordcount$
```

```

Activities Terminal Sep 10 23:07
hadoop@sanket-Lenovo: /usr/local/hadoop/share/KULKARNI/wordcount/output
the 93739
and 79182
of 53121
to 33929
i 30240
that 24407
in 24350
a 23504
my 17312
he 17087
for 16941
you 16603
is 16529
his 15822
not 15440
with 14263
be 14186
it 14141
shall 13557
me 12123
him 12069
thou 11249
lord 11182
but 10571
they 9998
have 9990
all 9728
this 9680
as 9532
unto 9458

```

Understandings:

- Each mapper takes an input as a line and breaks it into an array of words. It then pushes the key, val pair of 1 as the count in the intermediate space.
- Reducer then picks these values and combines them. It stores the output in the output directory mentioned in the terminal path.

Command used to run:

```

hadoop jar
/usr/local/hadoop/share/KULKARNI/mapreduce/hadoop-mapreduce-examples-2.8.1.jar
wordcount /usr/local/hadoop/share/KULKARNI/wordcount/input
/usr/local/hadoop/share/KULKARNI/wordcount/output

```

Question 2: What are the top 10 most frequently used words in the corpus?

the	93739
and	79182
of	53121
to	33929
i	30240
that	24407
in	24350

a	23504
my	17312
he	17087

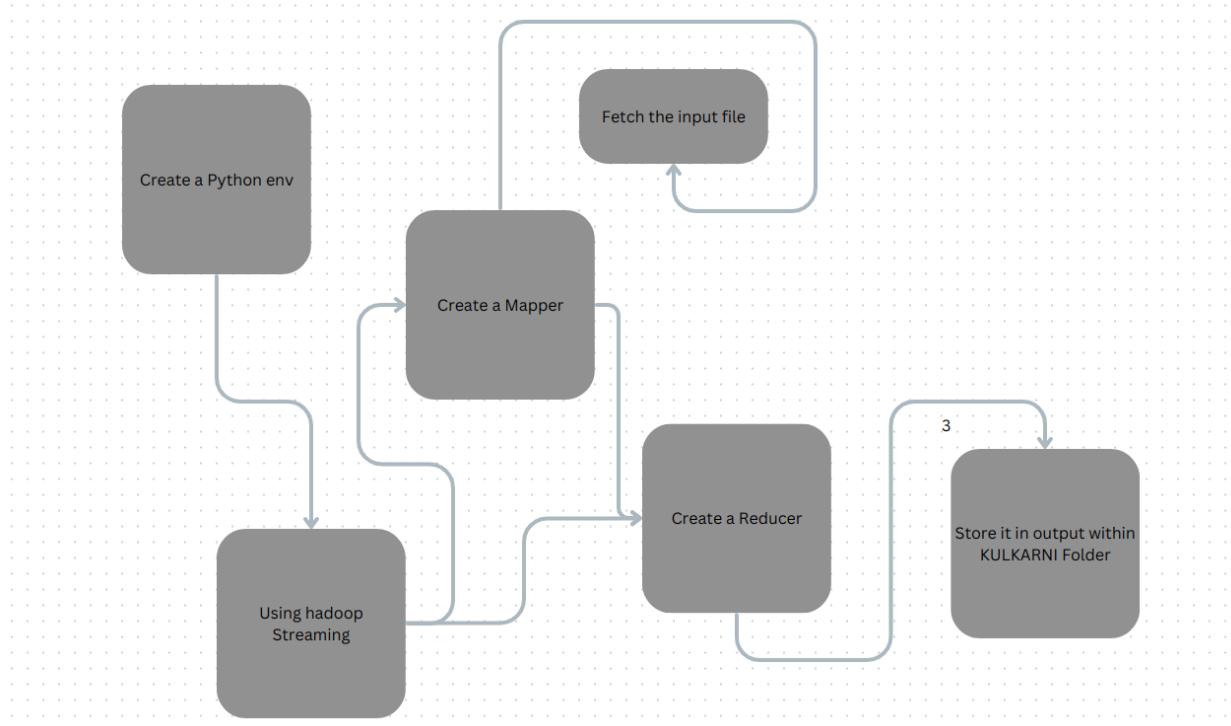
To get the most sorted one's in the file in descending order, I used below command of HDFS:

```
$ hadoop fs -cat part-000* | sort -k2,2nr > sorted_output_simple.txt
```

Exercise 3: Hadoop Kmer Counting

Approach:

- I have used python to write the mapper and reducer of this problem
- The purpose was to understand how hadoop streaming is used to run the mapper and reducers written in python



Sample Input 1: **ACACACAGT** to test whether the code is running properly or not. I stored this in a text file and then fed it as input to the reducer.

Name of the input file given : kmer_count_input.txt
Name of the output file generated : kmer_count_output
Time taken : few seconds

Command Used:

```
hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-* .jar \
-files kmer_count_mapper.py,kmer_count_reducer.py -mapper "kmer_count_mapper.py 9" \
-reducer kmer_count_reducer.py \
-input kmer_count_input.txt -output kmer_count_output
```

Homework 1: Sanket Kulkarni

```

hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming*.jar -files kmer_count_mapper.py,kmer_count_reducer.py -mapper "kmer_count_mapper.py 3" -reducer kmer_count_reducer.py -input kmer_count_input.txt -output kmer_count_output_simple
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
23/09/20 01:39:03 INFO jvm.JvmMetrics: Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
23/09/20 01:39:03 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId= -
23/09/20 01:39:03 INFO mapred.FileInputFormat: Total input files to process : 1
23/09/20 01:39:03 INFO mapreduce.JobSubmitter: number of splits:1
23/09/20 01:39:03 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local152041820_0001
23/09/20 01:39:03 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_mapper.py as file:/tmp/hadoop-hadoop/mapred/local/1695199143745/kmer_count_mapper.py
23/09/20 01:39:03 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_reducer.py as file:/tmp/hadoop-hadoop/mapred/local/1695199143746/kmer_count_reducer.py
23/09/20 01:39:04 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
23/09/20 01:39:04 INFO mapreduce.Job: Running job: job_local152041820_0001
23/09/20 01:39:04 INFO mapred.LocalJobRunner: OutputCommitter set in config null
23/09/20 01:39:04 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapred.FileOutputCommitter
23/09/20 01:39:04 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/20 01:39:04 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/20 01:39:04 INFO mapred.LocalJobRunner: Waiting for map tasks
23/09/20 01:39:04 INFO mapred.LocalJobRunner: Starting task: attempt_local152041820_0001_m_000000_0
23/09/20 01:39:04 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/20 01:39:04 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/20 01:39:04 INFO mapred.Task: Using ResourceCalculatorProcessTree : []
23/09/20 01:39:04 INFO mapred.MapTask: Processing split: file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_input.txt:0+10
23/09/20 01:39:04 INFO mapred.MapTask: numReduceTasks=1
23/09/20 01:39:04 INFO mapred.MapTask: (EQUATOR) 0 kvl 26214396(104857584)
23/09/20 01:39:04 INFO mapred.MapTask: mapred.task.to.sort.mb: 100
23/09/20 01:39:04 INFO mapred.MapTask: soft limit at 83866960
23/09/20 01:39:04 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
23/09/20 01:39:04 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
23/09/20 01:39:04 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
23/09/20 01:39:04 INFO streaming.PipeMapRed: PipeMapRed exec [/usr/local/hadoop/share/KULKARNI/kmerCounting/./kmer_count_mapper.py, 3]
23/09/20 01:39:04 INFO Configuration.deprecation: mapred.work.out.dir is deprecated. Instead, use mapreduce.task.outdir

23/09/20 01:39:05 INFO MapReduceJob: Job: Job job_local152041820_0001 completed successfully
23/09/20 01:39:05 INFO mapreduce.Job: Job: Counters: 30
  File System Counters
    FILE: Number of bytes read=270872
    FILE: Number of bytes written=925714
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
  Map-Reduce Framework
    Map input records=1
    Map output records=7
    Map output bytes=42
    Map output materialized bytes=62
    Input split bytes=123
    Combine input records=0
    Combine output records=0
    Reduce input groups=4
    Reduce shuffle bytes=62
    Reduce input records=7
    Reduce output records=4
    Spilled Records=14
    Shuffled Maps =1
    Failed Shuffles=0
    Merged Map outputs=1
    GC time elapsed (ms)=11
    Total committed heap usage (bytes)=471859200
  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=10
  File Output Format Counters
    Bytes Written=36
23/09/20 01:39:05 INFO streaming.StreamJob: Output directory: kmer_count_output_simple

```

Output:

ACA	3
CAC	2
AGT	1
CAG	1

Question 3: What are the top 10 most frequently occurring 9-mers in E coli?

Code Snippet:

```

file Edit Selection View Go Run Terminal Help
EXPLORER kmer_count_reducer.py kmer_count_mapper.py
KULKARNI
> city_to_conference
> conf_to_city
> conference_all
> kmerCounting
> hg19_kmer_count_output
> hg19_kmer_count_small_output
> kmer_count_output_129
> kmer_count_output_ecoli
> kmer_count_output_hg19
> kmer_count_output_simple
> kmer_count_output_small_hg19
> new_kmer_count_output
> kmer_count_mapper.py.swp
> kmer_count_mapper.py.swp
> mapper_kmer_counting.py.swp
> ecoli.fa
> hg19_small.fa
> hg19.fa
> kmer_count_input.txt
kmer_count_mapper.py
kmer_count_reducer.py
sorted_output.txt
mapreduce
> SunCardNums
> webCrawlers
> wordcount
> conference_data.csv
    #!/usr/bin/env python3
    import sys
    # Get the k-mer size from the command-line argument
    if len(sys.argv) != 2:
        print("Usage: kmer_mapper.py <k>")
        sys.exit(1)
    k = int(sys.argv[1])
    # Read input from STDIN (standard input)
    input_data = ""
    for line in sys.stdin:
        if line.startswith('>') or line.startswith('<'):
            continue
        line = line.strip() # Remove leading/trailing whitespaces
        line = line.replace(" ", "") # Remove spaces in the input
        line = ''.join([e for e in line if e.isalnum()]) # Remove non-alphanumeric characters
        input_data += line.upper() # Convert to uppercase and append to the input_data
    # Process the entire input as a single line
    for i in range(len(input_data) - k + 1):
        kmer = input_data[i:i+k] # Extract k-mer from the sequence
        print(f'{kmer}\t1') # Output k-mer and count 1

```

Changes done after comments:

- Added leading and trailing space check with a check for in between space
- Removed alphanumerics
- If line starts with ‘>’ ignored that line as it is a genome
- Converted whole input to uppercase

Name of the input file given : ecoli.fa

Name of the output file generated : kmer_count_output_ecoli

Time taken : few minutes

Command Used:

```

hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-*jar \
-files kmer_count_mapper.py,kmer_count_reducer.py -mapper "kmer_count_mapper.py 9"
-reducer kmer_count_reducer.py \
-input ecoli.fa -output kmer_count_output_ecoli

```

```

hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI$ cd ..
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI$ cd kmerCounting/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ ls
hg19.fa          hg19-small.kmer_count_output_ecoli_new      kmer_count_mapper.py      kmer_count_output_ecoli      kmer_count_output_simple      kmer_count_reducer.py      sorted_output.txt
hg19.small.kmer_count_output_ecoli      kmer_count_input.txt      kmer_count_mapper.py      kmer_count_output_hg19      kmer_count_output_small     kmer_count_reducer.py      sorted_output.txt
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-.jar -files kmer_count_mapper.py,kmer_count_reducer.py -mapper "python3 kmer_count_mapper.py 9"-reducer "python3 kmer_count_reducer.py" -input ecoli.fa -output kmer_count_output_ecoli_new
ARNING: An illegal reflective access operation has occurred
ARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.config.getInst
ARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
ARNING: Use -illegal-access=warn to enable warnings of further illegal reflective access operations
ARNING: All illegal access operations will be denied in a future release
3/10/05 01:41:05 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
3/10/05 01:41:05 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
3/10/05 01:41:05 INFO mapred.FileInputFormat: Total input files to process : 1
3/10/05 01:41:05 INFO mapred.JobSubmitter: number of splits:1
3/10/05 01:41:05 INFO mapred.JobSubmitter: Submitting Tokens for job: job_local1481835142_0001
3/10/05 01:41:05 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_mapper.py as file:/tmp/hadoop-hadoop/mapred/local/1696495261746/kmer_count_mapper.py
3/10/05 01:41:05 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_reducer.py as file:/tmp/hadoop-hadoop/mapred/local/1696495261747/kmer_count_reducer.py
3/10/05 01:41:05 INFO mapred.Job: The url to track the job: http://localhost:8088/
3/10/05 01:41:02 INFO mapred.Job: Running job: job_local1481835142_0001
3/10/05 01:41:02 INFO mapred.LocalJobRunner: OutputCommitter set in config null
3/10/05 01:41:02 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapred.FileOutputCommitter
3/10/05 01:41:02 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:41:02 INFO output.FileOutputCommitter: File Output Committer skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:41:02 INFO mapred.LocalJobRunner: Waiting for map tasks
3/10/05 01:41:02 INFO mapred.LocalJobRunner: Starting task: attempt_local1481835142_0001_m_000000_0
3/10/05 01:41:02 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:41:02 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:41:02 INFO mapred.Task: Using ReducerCalculatorTree: []
3/10/05 01:41:02 INFO mapred.Task: Processing split: file:/usr/local/hadoop/share/KULKARNI/kmerCounting/ecoli.fa:0+4786040
3/10/05 01:41:02 INFO mapred.MapTask: mapreduce.task.attemptId: 0
3/10/05 01:41:02 INFO mapred.MapTask: mapreduce.task.attemptId: 0
3/10/05 01:41:02 INFO mapred.MapTask: mapreduce.task.to.sort.mb: 100
3/10/05 01:41:02 INFO mapred.MapTask: soft limit at 83866880
3/10/05 01:41:02 INFO mapred.MapTask: bufferStart = 0; bufVolOff = 104857600
3/10/05 01:41:02 INFO mapred.MapTask: collectorClass = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
3/10/05 01:41:02 INFO streaming.PipeMapRed: PipeMapRed exec [/usr/bin/python3, kmer_count_mapper.py, 9]
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.work.dir is deprecated. Instead, use mapreduce.task.dir
3/10/05 01:41:02 INFO Configuration.deprecation: map.input.start is deprecated. Instead, use mapreduce.map.input.start
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.task.is.map is deprecated. Instead, use mapreduce.task.ismap
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.task.isattempted is deprecated. Instead, use mapreduce.task.attemptId
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.local.dir is deprecated. Instead, use mapreduce.cluster.local.dir
3/10/05 01:41:02 INFO Configuration.deprecation: map.input.file is deprecated. Instead, use mapreduce.map.input.file
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skipprecords
3/10/05 01:41:02 INFO Configuration.deprecation: map.input.length is deprecated. Instead, use mapreduce.map.input.length
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.job.id is deprecated. Instead, use mapreduce.job.id
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.task.partition is deprecated. Instead, use mapreduce.job.user.name
3/10/05 01:41:02 INFO Configuration.deprecation: mapred.task.partition is deprecated. Instead, use mapreduce.task.partition
3/10/05 01:41:02 INFO streaming.PipeMapRed: R/W=s/1/B in=NA [rec/s] out=NA [rec/s]
3/10/05 01:41:02 INFO streaming.PipeMapRed: R/W=s/10/0 in=NA [rec/s] out=NA [rec/s]

```

Output:

To get the most sorted one's in the file in descending order, I used below command of HDFS:

```
$ hadoop fs -cat part-000* | sort -k2,2nr > sorted_output_simple.txt
```

New output

sorted_output_simple.txt	
/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_ecoli_new	Save
1 CAGGCCAG 294	
2 CGACGGCA 290	
3 GCGCTGGG 273	
4 CTGGCGCTG 253	
5 CGCTGGGG 249	
6 TTGGCGCTG 249	
7 CCCAGCAG 244	
8 CGCCAGCG 244	
9 GCGACGGC 241	
10 CGCCAGCA 231	
11 CGCTGGGC 231	
12 GGGCTGGC 230	
13 CGCCAGCG 225	
14 CAAGCCCTG 224	

Question 4: Run the above mapper and reducer with input file of hg-19

Name of the input file given : hg19.fa

Name of the output file generated : kmer_count_output_hg19

Time taken : about 4 hours

Command Used:

```

hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-.jar \
-files kmer_count_mapper.py,kmer_count_reducer.py -mapper "kmer_count_mapper.py 9" \
-reducer kmer_count_reducer.py \
-input hg19.fa -output kmer_count_output_hg19

```

Mapper Code Content:

1. Take the argument from command line regarding what should be the value of 'k' in the kmer
2. For every sentence in the file, remove whitespaces and trailing spaces. This will help us ensure text wrapping.
3. Create a subset of size k and then store it in intermediate space with count as 1
4. The storing will be done by adding tabs between the string and count

Reducer Code Content:

1. Split the input lines based on tabs that we added initially in the mapper
2. Convert the count which was stored earlier to integer value
3. If kmer already exist, increment the count
4. Else, add the kmer as a new entry

Changes done after comments:

- Added leading and trailing space check with a check for in between space
- Removed alphanumerics
- If line starts with '>' ignored that line as it is a genome
- Converted whole input to uppercase

New Code Snippet:

```

EXPLORER ... ↵ kmer_count_mapper.py ✘
KULKARNI
> city_to_conference
> conf_to_city
> conference_all
< kmerCounting
> hg19_kmer_count_output
> hg19_kmer_count_small_output
> kmer_count_output_129
> kmer_count_output_ecoli
> kmer_count_output_ecoli_new
> kmer_count_output_hg19
> kmer_count_output_hg19f_new
> kmer_count_output_simple
> kmer_count_output_small_hg19
< new_kmer_count_output
  => _SUCCESS
  => _SUCCESS.crc
  => part-00000.crc
  => part-00000
  => sorted_output_simple.txt
  => .kmer_count_mapper.py.swp
  => .kmer_count_mapper.py.swp
  => .mapper_kmer_counting.py.swp
  => ecoli.fa
  => hg19_small.fa
  => hg19.fa
  => kmer_count_input.txt
  => kmer_count_mapper.py
  => kmer_count_reducer.py
  => sorted_output.txt
> mapreduce
> SumCardNums
> webCrawler
> wordcount
< conference_data.tsv

```

```

kmerCounting > kmer_count_mapper.py
1   #!/usr/bin/env python3
2
3   import sys
4
5   # Get the k-mer size from the command-line argument
6   if len(sys.argv) != 2:
7       print("Usage: kmer_mapper.py <k>")
8       sys.exit(1)
9
10  k = int(sys.argv[1])
11
12  # Read input from STDIN (standard input)
13  input_data = ""
14  for line in sys.stdin:
15      if line.startswith('>') or line.startswith('<'):
16          continue
17      line = line.strip() # Remove leading/trailing whitespaces
18      line = line.replace(" ", "") # Remove spaces in the input
19      line = ''.join([e for e in line if e.isalnum()]) # Remove non-alphanumeric characters
20      input_data += line.upper() # Convert to uppercase and append to the input_data
21
22  # Process the entire input as a single line
23  for i in range(len(input_data) - k + 1):
24      kmer = input_data[i:i+k] # Extract k-mer from the sequence
25      print(f"{kmer}\t1") # Output k-mer and count 1
26
27

```

Screenshot of hg-19 on Small Input:

Homework 1: Sanket Kulkarni

```
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_ecall_new
3/10/05 00:10:29 INFO mapreduce.Job: Job job_local4493253233_0001 failed with state FAILED due to: NA
3/10/05 00:10:29 INFO mapreduce.Job: Counters: 0
3/10/05 00:10:29 ERROR streaming.StreamJob: Job is not successful!
streaming Command Failed!
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ rm -rf kmer_count_output_small hg19/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming*.jar -files kmer_count_mapper.py,kmer_count_reducer.py -mapper "python3 kmer_count_mapper.py -reducer "python3 kmer_count_reducer.py"-input hg19_small.fa -output kmer_count_output_small.hg19
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstnace()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: All illegal access operations will be denied in a future release
3/10/05 00:11:09 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
3/10/05 00:11:09 INFO jvm.JvmMetrics: Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
3/10/05 00:11:09 INFO mapred.FileInputFormat: Total input files to process : 1
3/10/05 00:11:09 INFO mapreduce.Job: Input splits:
3/10/05 00:11:09 INFO mapreduce.Job: Submitter Token for job: job_local4493257_0001
3/10/05 00:11:09 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_mapper.py as file:/tmp/hadoop-mapred/local/1696489869541/kmer_count_mapper.py
3/10/05 00:11:09 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_reducer.py as file:/tmp/hadoop-mapred/local/1696489869542/kmer_count_reducer.py
3/10/05 00:11:09 INFO mapred.JobClient: The url to track the job: http://localhost:8080/
3/10/05 00:11:09 INFO mapred.LocalJobRunner: OutputCommitter set in config null
3/10/05 00:11:09 INFO mapreduce.Job: Running job: job_local4493257_0001
3/10/05 00:11:09 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapred.FileOutputCommitter
3/10/05 00:11:09 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 00:11:09 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 00:11:09 INFO output.FileOutputCommitter: Waiting for map tasks
3/10/05 00:11:09 INFO mapred.LocalJobRunner: map attempt local4493257_0001_m_000000_0
3/10/05 00:11:09 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 00:11:09 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 00:11:09 INFO mapred.Task: Using ResourceCalculatorTree : []
3/10/05 00:11:09 INFO mapred.MapTask: Processing split: file:/usr/local/hadoop/share/KULKARNI/kmerCounting/hg19_small.fa@0+33554432
3/10/05 00:11:09 INFO mapred.MapTask: numReduceTasks=1
3/10/05 00:11:09 INFO mapred.MapTask: [EQUATOR] mapattempt: 26214396104857594
3/10/05 00:11:10 INFO mapred.MapTask: mapreduce.task.io.sort.mb= 100
3/10/05 00:11:10 INFO mapred.MapTask: soft limit at 83886080
3/10/05 00:11:10 INFO mapred.MapTask: buffstart = 0; bufvolts = 104857600
3/10/05 00:11:10 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
3/10/05 00:11:10 INFO mapred.MapTask: Map output collected class org.apache.hadoop.mapred.MapTask$MapOutputBuffer
3/10/05 00:11:10 INFO mapred.MapTask: Map output collected records: 15353803983, kmer count mapper.py, 3
3/10/05 00:11:10 INFO configuration.deprecation: mapred.workDir is deprecated. Instead, use mapreduce.task.output.dir
3/10/05 00:11:10 INFO configuration.deprecation: mapred.input.start is deprecated. Instead, use mapreduce.map.input.start
3/10/05 00:11:10 INFO configuration.deprecation: mapred.task.is.map is deprecated. Instead, use mapreduce.map.ismap
3/10/05 00:11:10 INFO configuration.deprecation: mapred.task.id is deprecated. Instead, use mapreduce.task.attemptId
3/10/05 00:11:10 INFO configuration.deprecation: mapred.job.id is deprecated. Instead, use mapreduce.task.id
3/10/05 00:11:10 INFO configuration.deprecation: mapred.localDir is deprecated. Instead, use mapreduce.cluster.local.dir
3/10/05 00:11:10 INFO configuration.deprecation: mapred.localDirs is deprecated. Instead, use mapreduce.cluster.local.dirs
3/10/05 00:11:10 INFO configuration.deprecation: map_input_file is deprecated. Instead, use mapreduce.map.input.file
3/10/05 00:11:10 INFO configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skiprecords
3/10/05 00:11:10 INFO configuration.deprecation: mapred.input.length is deprecated. Instead, use mapreduce.map.input.length
3/10/05 00:11:10 INFO configuration.deprecation: mapred.job.id is deprecated. Instead, use mapreduce.job.job.id
3/10/05 00:11:10 INFO configuration.deprecation: user.name is deprecated. Instead, use mapreduce.job.user.name
3/10/05 00:11:10 INFO configuration.deprecation: mapred.task.partition: Saved output of task attempt_local4493257_0001_r_000000 to file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_small_hg19/_temporary/0/task_local4493257_0001_r_000000
3/10/05 00:11:10 INFO streaming.PipeMapRed: R/W/S=5/0/0 [rec/s] outNA [rec/s]
3/10/05 00:11:10 INFO streaming.PipeMapRed:
```

Completion of hg-19 smaller

```
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_ecall_new
3/10/05 00:19:45 INFO streaming.PipeMapRed: R/W/S=128200000/261806/0 in:1017460-128200000/126 [rec/s] out:2077=261806/126 [rec/s]
3/10/05 00:19:45 INFO streaming.PipeMapRed: MRErrorThread done
3/10/05 00:19:45 INFO streaming.PipeMapRed: mapredfinisched
3/10/05 00:19:45 INFO mapred.Task: TaskAttempt local4493257_0001_r_000000_0 is done. And is in the process of committing
3/10/05 00:19:45 INFO mapred.Task: TaskAttempt local4493257_0001_r_000000_0 is allowed to commit now
3/10/05 00:19:45 INFO output.FileOutputCommitter: Saved output of task attempt_local4493257_0001_r_000000_0 to file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_small_hg19/_temporary/0/task_local4493257_0001_r_000000
3/10/05 00:19:45 INFO mapred.LocalJobRunner: Records R/W=125482859/25774 > reduce
3/10/05 00:19:45 INFO mapred.Task: Task attempt_local4493257_0001_r_000000_0 done.
3/10/05 00:19:45 INFO mapred.LocalJobRunner: Finishing task: attempt_local4493257_0001_r_000000_0
3/10/05 00:19:45 INFO mapred.LocalJobRunner: Job mapred.mapoutput.localjobrunnerexecutor complete.
3/10/05 00:19:46 INFO mapreduce.Job: map 100% reduce 100%
3/10/05 00:19:46 INFO mapreduce.Job: Job job_local44493257_0001 completed successfully
3/10/05 00:19:46 INFO mapreduce.Job: Counters: 30
  File System Counters
    FILE: Number of bytes read=145338976
    FILE: Number of bytes written=15538083983
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
  Map-Reduce Framework
    Map Input records=2638004
    Map Input bytes=128204018
    Map output bytes=1539528216
    Map output materialized bytes=1796116276
    Input split bytes=464
    Combine input records=0
    Combine output records=0
    Reduce input groups=1796116276
    Reduce output bytes=1796116276
    Reduce input records=128294018
    Reduce output records=262198
    Spilled Records=405322556
    Shuffled Maps =4
    Failed Maps=0
    Merged Map outputs=4
    GC time elapsed (ms)=421
    Total committed heap usage (bytes)=1603272704
  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=13150510
    File Outputs=3693703
    Bytes Written=3693701
3/10/05 00:19:46 INFO streaming.StreamJob: Output directory: kmer_count_output_small_hg19
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ cd kmer_count_output_small_hg19/
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ hadoop fs -cat part-000* | sort -k2,2nr > sorted_output_simple.txt
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstnace()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
```

Output of hg-19 smaller:

sorted_output_simple.txt	
<i>/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_output_small_hg19</i>	
1	AAAAAAA
2	TTTTTTTT
3	TGTGTGTG
4	GTGTGTG
5	ACACACAC
6	CACACACAC
7	AGAGAGAGA
8	GAGAGAGAG
9	TCTCTCT
10	GGGGGGGG
11	CTCTCTCTC
12	TTTTGTTT
13	CAGCAGCAG
14	GAGGAGAG
15	TAAAAAAA
16	ATATATAT
17	ACACACAC
18	GGAGGAGGA
19	CCAGGCTGG
20	CCAGCCTGG

Screenshot of hg-19

```
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/kmerCounting$ hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming*.jar -files kmer_count_mapper.py,kmer_count_reducer.py -mapper "kmer_count_mapper.py 9" -reducer kmer_count_reducer.py -input hg19.fa -output kmer_count_output_hg19
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstances()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: All illegal reflective access operations will be denied in a future release
WARNING: All illegal reflective access operations will be denied in a future release
23/09/20 01:30:06 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
23/09/20 01:30:06 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
23/09/20 01:30:06 INFO jvm.JvmMetrics: Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
23/09/20 01:30:07 INFO mapred.FileInputFormat: Total input files to process : 1
23/09/20 01:30:07 INFO mapreduce.JobSubmitter: number of splits:96
23/09/20 01:30:07 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1482311563_0001
23/09/20 01:30:07 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_mapper.py as file:/tmp/hadoop-hadoop/mapred/local/1695198607643/kmer_count_mapper.py
23/09/20 01:30:07 INFO mapred.LocalDistributedCacheManager: Localized file:/usr/local/hadoop/share/KULKARNI/kmerCounting/kmer_count_reducer.py as file:/tmp/hadoop-hadoop/mapred/local/1695198607644/kmer_count_reducer.py
23/09/20 01:30:07 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
23/09/20 01:30:07 INFO mapreduce.Job: Running job: job_local1482311563_0001
23/09/20 01:30:07 INFO mapred.LocalJobRunner: OutputCommitter set in config null
23/09/20 01:30:08 INFO mapred.LocalJobRunner: OutputCommitter org.apache.hadoop.mapred.FileOutputCommitter
23/09/20 01:30:08 INFO mapred.LocalJobRunner: OutputCommitter: File Output Committer Algorithm version is 1
23/09/20 01:30:08 INFO mapred.LocalJobRunner: OutputCommitter: File Output Committer temporary folders under output directory:false, ignore cleanup failures: false
23/09/20 01:30:08 INFO mapred.LocalJobRunner: Waiting for map tasks
23/09/20 01:30:08 INFO mapred.LocalJobRunner: Starting task: attempt_local1482311563_0001_m_000000_0
23/09/20 01:30:08 INFO mapred.FileOutputCommitter: File Output Committer Algorithm version is 1
23/09/20 01:30:08 INFO mapred.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
23/09/20 01:30:08 INFO mapred.Task: Using ResourceCalculatorProcessTree : []
23/09/20 01:30:08 INFO mapred.MapTask: Processing split: /file:/usr/local/hadoop/share/KULKARNI/kmerCounting/hg19.fa:+0:33554432
23/09/20 01:30:08 INFO mapred.MapTask: numReduceTasks: 1
23/09/20 01:30:08 INFO mapred.MapTask: (EQUATOR) 0 kv1:26214396(104857584)
23/09/20 01:30:08 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
23/09/20 01:30:08 INFO mapred.MapTask: soft limit at 83886080
23/09/20 01:30:08 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
23/09/20 01:30:08 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
23/09/20 01:30:08 INFO mapred.MapTask: Map output collector class: org.apache.hadoop.mapred.MapTask$MapOutputBuffer
23/09/20 01:30:08 INFO stream伪PipeMapRed: PipeMapRed over [/usr/local/hadoop/share/KULKARNI/kmerCounting/,/kmer_count_mapper.py, 9]
23/09/20 01:30:08 INFO configuration.deprecation: mapred.work.output.dir is deprecated. Instead, use mapreduce.task.output.dtr
23/09/20 01:30:08 INFO configuration.deprecation: map.input.start is deprecated. Instead, use mapreduce.map.input.start
```

New Output for hg-19:

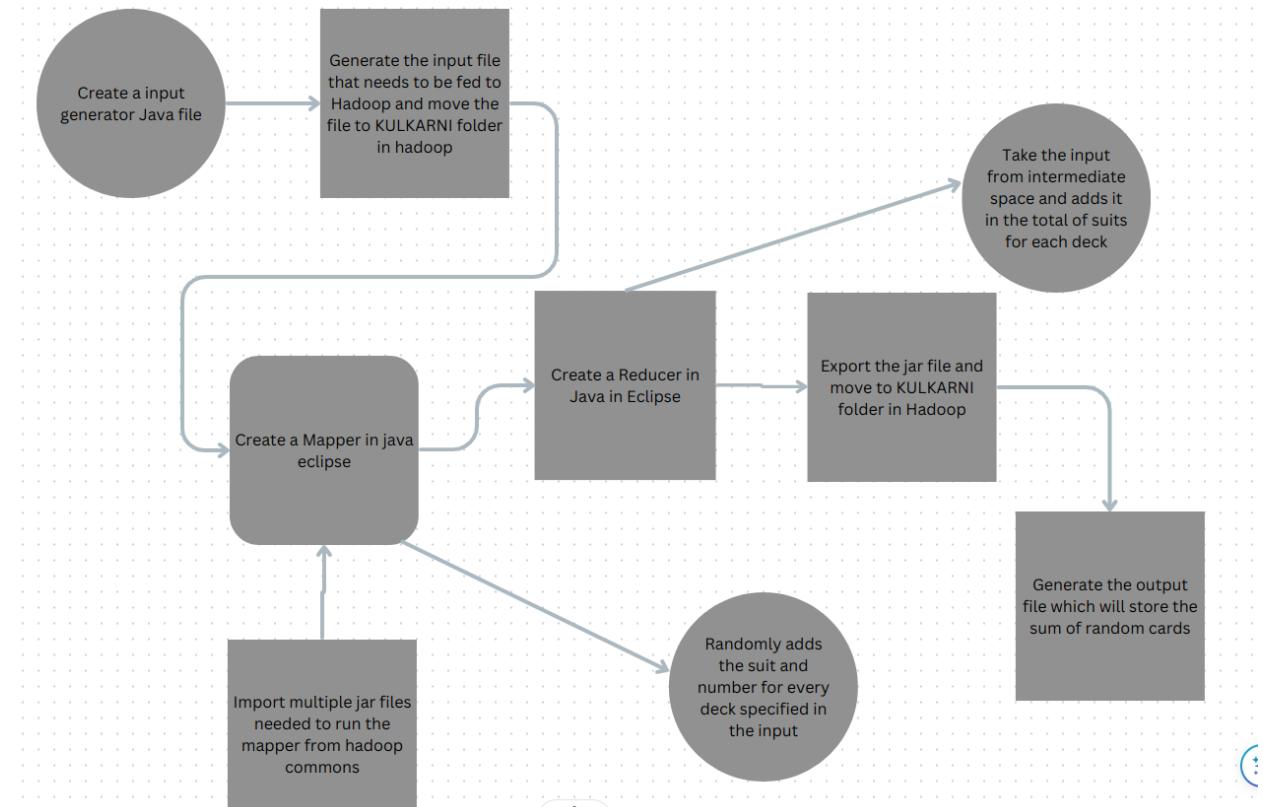
2	TTTTTTTT	3685757
3	AAAAAAA	3668151
4	TGTGTGTG	719557
5	ACACACAC	715531
6	GTGTGTG	655708
7	CACACACAC	651755
8	TATATATAT	633728
9	ATATATATA	633062
10	CCAGGCTGG	529211
11	CCAGCCTGG	525109

Total file looks like :

2	TTTTTTTT	3685757
3	AAAAAAA	3668151
4	TGTGTGTG	719557
5	ACACACACA	715531
6	GTGTGTGTG	655708
7	CACACACAC	651755
8	TATATATAT	633728
9	ATATATATA	633062
10	CCAGGCTGG	529211
11	CCAGGCTGG	525109
12	CTGGGATTAA	466365
13	TAATCCCCAG	465841
14	GATTACAGG	464772
15	CCTGTAATC	464113
16	GGATTACAG	464002
17	CTGTAATCC	463017
18	CTCAGCCTC	461167
19	GAGGGCTGAG	459817
20	GCTGGGATT	455382
21	AATCCCAGC	454950
22	GGGATTACA	454308
23	TGTAATCCC	453943
24	TCAGCCTCC	451096
25	GGAGGCTGA	450156
26	GTAATCCCA	449030
27	TGGGATTAC	449014
28	CCTCAGCCT	446611
29	AGGCTGAGG	444230
30	GCCTCCCAA	429840
31	TTGGGAGGC	427616
32	CTTTTTTT	403102
33	AAAAAAAAG	399876
34	TTTTTATT	395116
35	AAAATAAAA	392797
36	TTTTTTTG	384581
37	CAAAAAAAA	382277
38	CCCAGCTG	375397
39	TTTTCTTT	375307
40	ATTTTTTT	374903
41	TTTGATTT	374715
42	AAATACAAA	373791
43	AAAAAAAT	373645
44	AAAAGAAA	372483
45	CAGCTGGG	372483
46	TTTCTTTT	365231
47	TGTATTTT	365183

Exercise 4: Hadoop Playing Cards Counting

Approach:



Source Code Snips:

Generation of cards randomly: Considered for face card as well now

New logic:

- For all 100 decks, loop through
- Then we generate 52 cards and give it to the next for loop which will remove 17 random elements from the card deck
- This random generated elements that are removed will contain numeric as well as face cards in them
- So we have a total of 3500 cards containing both faces as well as numeric cards.

```

File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X CardCountDriver.java CountNumReducer.java CountNumMapper.java RandomCardDeckGenerator.java
> MapperReducer RandomCardDeckGenerator
import java.io.*;
import java.util.*;
public class RandomCardDeckGenerator {
    public static void main(String[] args) {
        /* This function generates card deck of given size 100/1000, then shuffles them and puts them in an input file*/
        Random random = new Random();
        List<String> total_card_list = new ArrayList<>();
        String[] weight = {"2", "3", "4", "5", "6", "7", "8", "9", "10", "Jack", "Queen", "King", "Ace"};
        String[] type_of_cardsList = {"Clubs", "Diamonds", "Hearts", "Spades"};
        int maxCardDecks = 100;
        // We can change this value as per the inputs in HW. This can be made argument based as well

        for (int deck = 0; deck < maxCardDecks; deck++) {
            List<String> cardsList = new ArrayList<>();
            for (String type : type_of_cardsList) {
                for (String value : weight) {
                    cardsList.add("Deck Number " + deck + " " + type + " " + value);
                }
            }
            for (int i = 0; i < 17; i++) {
                int ranCard = random.nextInt(cardsList.size());
                cardsList.remove(ranCard);
            }
            total_card_list.addAll(cardsList);
        }
        Collections.shuffle(total_card_list); // Shuffle the card decks based on types and weights
        try (BufferedWriter writer = new BufferedWriter(new FileWriter("rando_input_1000.txt"))) {
            for (String card : total_card_list) {
                writer.write(card + "\n");
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
    
```

Problems Javadoc Declaration Console X

<terminated> RandomCardDeckGenerator [Java Application] /home/sanket/p2/pool/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.linux.x86_64.17.0.8.v20230831-1047/jre/bin/java (Oct 5, 2023, 1:16:26 AM - 1:16:26 AM)

Driver code to set mapper and reducer:

```

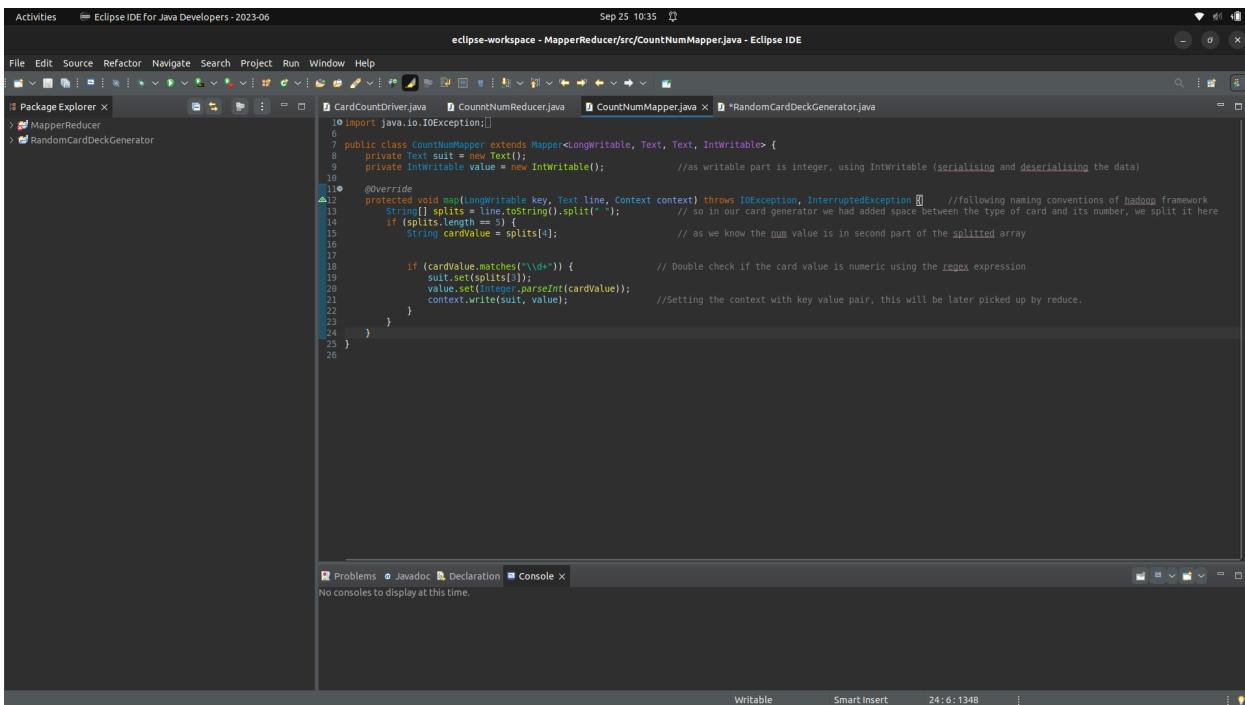
Activities Eclipse IDE for Java Developers - 2023-06 Sep 25 10:35
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X CardCountDriver.java CountNumReducer.java CountNumMapper.java RandomCardDeckGenerator.java
> MapperReducer RandomCardDeckGenerator
import org.apache.hadoop.conf.Configuration;
public class CardCountDriver {
    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
        Job job = job.newInstance(conf, "Card Counting started"); //creates a new mapreduce job with the specified configuration
        job.setJarByClass(CardCountDriver.class); // setting the driver class
        job.setMapperClass(CountNumMapper.class); // setting the mapper
        job.setReducerClass(CountNumReducer.class); // setting the reducer
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);
        job.setInputFormatClass(TextInputFormat.class);
        job.setOutputFormatClass(TextOutputFormat.class); //setting input and output files formats
        TextInputFormat.addInputPath(job, new Path(args[0])); // getting the input path specified while writing jar command
        TextOutputFormat.setOutputPath(job, new Path(args[1])); // getting the output path specified while writing jar command
        System.exit(job.waitForCompletion(true) ? 0 : 1); // once job completed it will return and exit
    }
}
    
```

Problems Javadoc Declaration Console X

No consoles to display at this time.

Writable Smart Insert 8:1:320

Mapper Code:

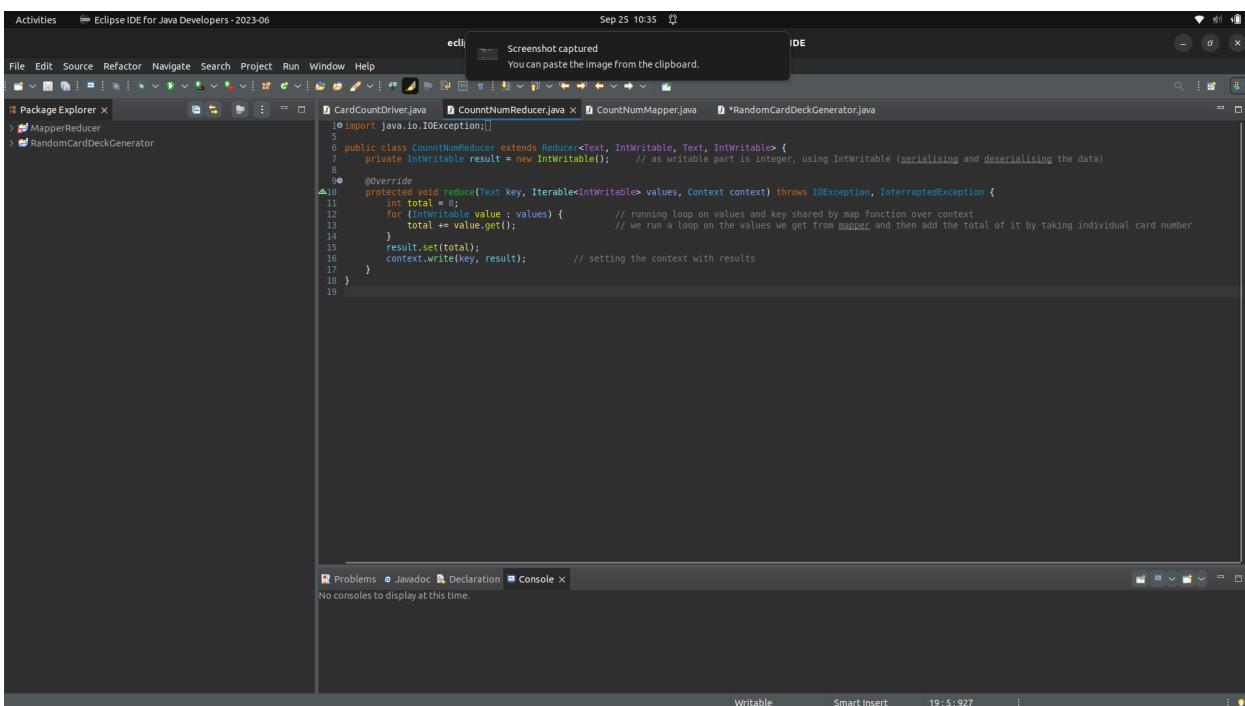


```

Activities  Eclipse IDE for Java Developers - 2023-06 Sep 25 10:35
eclipse-workspace - MapperReducer/src/CountNumMapper.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X CardCountDriver.java CountNumReducer.java CountNumMapper.java *RandomCardDeckGenerator.java
import java.io.IOException;
public class CountNumMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
    private Text suit = new Text();
    private IntWritable value = new IntWritable();
    @Override
    protected void map(LongWritable key, Text line, Context context) throws IOException, InterruptedException {
        String[] splits = line.toString().split(" ");
        if (splits.length == 3) {
            String cardValue = splits[2];
            if (cardValue.matches("\\d+")) {
                suit.set(splits[0]);
                value.set(Integer.parseInt(cardValue));
                context.write(suit, value);
            }
        }
    }
}

```

Reducer Code:



```

Activities  Eclipse IDE for Java Developers - 2023-06 Sep 25 10:35
eclipse-workspace - MapperReducer/src/CountNumReducer.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X CardCountDriver.java CountNumReducer.java CountNumMapper.java *RandomCardDeckGenerator.java
import java.io.IOException;
public class CountNumReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
    private IntWritable result = new IntWritable();
    @Override
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException, InterruptedException {
        int total = 0;
        for (IntWritable value : values) {
            total += value.get();
        }
        result.set(total);
        context.write(key, result);
    }
}

```

Steps:

1. Create a java file that will create our input file. This java code will fetch number of deck from the command line or predefined input
2. Move this file to hadoop → Kulkarni folder
3. Create a java project in Eclipse, the files should include driver, mapper and reducer class.
4. The driver will be responsible to set the mapper and reducer class as per the hadoop configs
5. In mapper, the code will do following things
 - a. Split the input in different array
 - b. Cardvalue will be stored in array[4]
 - c. Check the suit value at array[3]
 - d. Also check if the card is a face card or a numeric, my mapper was already doing this check so did not have to change anything in it. Just made the input change
6. In reducer, the code will make summations based on suits
 - a. Run a loop to iterate over the context values set
 - b. Add the sum with respect to the card suit type
 - c. Set the total and generate an output file
7. A sample input file will look like below: This contains face cards as well, which will be removed in mapper later

```
*hadoop_command* rando_input_1000.txt
1 Deck Number 76 Heart 6
2 Deck Number 440 Diamonds 3
3 Deck Number 507 Clubs Jack
4 Deck Number 796 Spades Ace
5 Deck Number 360 Spades King
6 Deck Number 704 Diamonds 2
7 Deck Number 625 Hearts Jack
8 Deck Number 625 Hearts 1
9 Deck Number 20 Heart Jack
10 Deck Number 541 Clubs Jack
11 Deck Number 656 Diamonds King
12 Deck Number 741 Clubs 6
13 Deck Number 951 Diamonds Ace
14 Deck Number 566 Hearts 7
15 Deck Number 637 Clubs 5
16 Deck Number 568 Clubs 2
17 Deck Number 207 Spades 4
18 Deck Number 163 Heart 8
19 Deck Number 341 Diamonds 6
20 Deck Number 0 Diamonds 4
21 Deck Number 70 Clubs 3
22 Deck Number 168 Heart Queen
23 Deck Number 168 Spades 6
24 Deck Number 26 Spades 3
25 Deck Number 508 Heart 7
26 Deck Number 488 Diamonds 7
27 Deck Number 62 Diamonds Jack
28 Deck Number 528 Heart King
29 Deck Number 955 Clubs King
30 Deck Number 408 Clubs 6
31 Deck Number 89 Clubs 8
32 Deck Number 448 Hearts 9
33 Deck Number 518 Clubs 8
34 Deck Number 304 Heart 2
35 Deck Number 69 Clubs 2
36 Deck Number 887 Clubs 10
37 Deck Number 210 Diamonds 3
38 Deck Number 448 Diamonds 2
39 Deck Number 967 Diamonds King
40 Deck Number 308 Diamonds 5
41 Deck Number 885 Spades 9
42 Deck Number 156 Heart Queen
43 Deck Number 877 Clubs 6
44 Deck Number 770 Diamonds 6
Deck Number 279 Clubs Jack
```

Here the Deck number is the deck that was randomly chosen, then we have suits and then card value.

8. There are 3500 lines for 100 decks and 35000 for 1000 decks

Question 5: Try 100 decks and 1000 decks respectively?

Command used :

For 100 Decks

```
hadoop jar /usr/local/hadoop/share/KULKARNI/mapreduce/random_card_addition.jar
CardCountDriver /usr/local/hadoop/share/KULKARNI/SumCardNums/rando_input_100.txt
/usr/local/hadoop/share/KULKARNI/SumCardNums/new_output_100
```

For 1000 Decks

```
hadoop jar /usr/local/hadoop/share/KULKARNI/mapreduce/random_card_addition.jar
CardCountDriver /usr/local/hadoop/share/KULKARNI/SumCardNums/rando_input_1000.txt
/usr/local/hadoop/share/KULKARNI/SumCardNums/new_output_1000
```

Screenshots for 100 decks on new input:

```
hadoop@sankey-Lenovo:/usr/local/hadoop/share/KULKARNI/SumCardNums$ rm -rf new_output_1000/
hadoop@sankey-Lenovo:/usr/local/hadoop/share/KULKARNI/SumCardNums$ hadoop jar /usr/local/hadoop/share/KULKARNI/SumCardNums/random_card_addition.jar CardCountDriver /usr/local/hadoop/share/KULKARNI/SumCardNums/rando_input_100
ARNNG: An illegal reflective access operation has occurred
ARNNG: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInst
ARNNG: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
ARNNG: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
ARNNG: All illegal access operations will be denied in a future release
3/10/05 01:23:49 INFO mapred.JobClient: Configuration deprecating JobTracker is deprecated. Instead, use dfs.metrics.session-id
3/10/05 01:23:49 INFO mapred.JobClient: Configuration deprecating Job processName=JobTracker, sessionId=
3/10/05 01:23:49 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
3/10/05 01:23:49 INFO mapreduce.JobSubmissionWriter: number of splits:1
3/10/05 01:23:49 INFO mapreduce.JobSubmissionWriter: Submitting tokens for job: job_local1450046861_0001
3/10/05 01:23:50 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
3/10/05 01:23:50 INFO mapreduce.Job: Number of splits: 1
3/10/05 01:23:50 INFO mapred.LocalJobRunner: OutputCommitter set in Config null
3/10/05 01:23:50 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:23:50 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:23:50 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
3/10/05 01:23:50 INFO mapred.LocalJobRunner: Waiting for map tasks
3/10/05 01:23:50 INFO mapred.MapTask: Starting task: attempt_local1450046861_0001_m_000000
3/10/05 01:23:50 INFO mapred.MapTask: bufstart = 0, bufvoid = 104857600
3/10/05 01:23:50 INFO mapred.MapTask: kvstart = 26214396(104857584); length = 6553600
3/10/05 01:23:50 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
3/10/05 01:23:50 INFO mapred.LocalJobRunner:
3/10/05 01:23:50 INFO mapred.MapTask: Starting flush of map output
3/10/05 01:23:50 INFO mapred.MapTask: Spilling map output
3/10/05 01:23:50 INFO mapred.MapTask: bufstart = 0, bufvoid = 20698; bufvoid = 104957600
3/10/05 01:23:50 INFO mapred.MapTask: Kvstart = 26214396(104857584); Kvend = 26204700(104818800); length = 9697/6553600
3/10/05 01:23:50 INFO mapred.MapTask: Finished spill 0
3/10/05 01:23:50 INFO mapred.Task: Task:attempt_local1450046861_0001_m_000000 is done. And is in the process of committing
3/10/05 01:23:50 INFO mapred.Task: Task:attempt_local1450046861_0001_m_000000 is committed!
3/10/05 01:23:50 INFO mapred.Task: attempt_local1450046861_0001_m_000000 is done.
3/10/05 01:23:50 INFO mapred.Task: attempt_local1450046861_0001_m_000000 is finishing task: attempt_local1450046861_0001_m_000000
3/10/05 01:23:50 INFO mapred.LocalJobRunner: Map task executor complete.
3/10/05 01:23:50 INFO mapred.LocalJobRunner: Waiting for reduce tasks
3/10/05 01:23:50 INFO mapred.LocalJobRunner: Starting task: attempt_local1450046861_0001_r_000000
3/10/05 01:23:50 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:23:50 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:23:50 INFO mapred.ReduceTask: Using ShuffleConsumerPlugin: org.apache.hadoop.mapreduce.task.reduce.Shuffle$3498fcb
3/10/05 01:23:50 INFO reduce.MergeManagerImpl: MergerManager: memoryLimit=291601280, mergeThreshold=173400320, mergeShuffleLimit=73400320, mergeThreshold=193776848, ioSortFactor=10, memToMemMergeOutputsThreshold=10
3/10/05 01:23:50 INFO reduce.EventFetcher: attempt local1450046861_0001_r_000000_0 Thread started: EventFetcher for fetching Map Completion Events
3/10/05 01:23:50 INFO reduce.InMemoryMapOutput: Read 31550 bytes from map-output for attempt_local1450046861_0001_r_000000_0
3/10/05 01:23:50 INFO reduce.MergeManagerImpl: closeInMemoryFile > map-output of size: 31550, InMemoryMapOutputs.size() -> 0, commitMemory -> 0, usedMemory ->31550
3/10/05 01:23:50 INFO reduce.EventFetcher: EventFetcher is interrupted. Returning
```

```

hadoop@sankey-Lenovo: /usr/local/hadoop/share/KULKARNI/SumCardNums/new_output_1000
3/10/05 01:23:50 INFO reduce.MergeManagerImpl: Merged 1 segments, 31590 bytes to disk to satisfy reduce memory limit
3/10/05 01:23:50 INFO reduce.MergeManagerImpl: Merging 0 segments, 31554 bytes from disk
3/10/05 01:23:50 INFO reduce.MergeManagerImpl: Merged 0 segments, 0 bytes from memory into reduce
3/10/05 01:23:50 INFO mapred.Merger: Merging 1 sorted segments
3/10/05 01:23:50 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of total size: 31542 bytes
3/10/05 01:23:50 INFO mapred.LocalJobRunner: 1 / 1 copied.
3/10/05 01:23:50 INFO Configuration.deprecation: mapred.skip.on is deprecated. Instead, use mapreduce.job.skiprecords
3/10/05 01:23:50 INFO mapred.Task: Task attempt_local1450046861_0001_r_000000_0 is done. And is in the process of committing
3/10/05 01:23:50 INFO mapred.Task: Task attempt_local1450046861_0001_r_000000_0 is allowed to commit now
3/10/05 01:23:50 INFO output.FileOutputCommitter: Saved output of task 'attempt_local1450046861_0001_r_000000_0' to file:/usr/local/hadoop/share/KULKARNI/SumCardNums/new_output_100/_temporary/0/task_local1450046861_0001_r_000000
3/10/05 01:23:50 INFO mapred.LocalJobRunner: reduce > reduce
3/10/05 01:23:50 INFO mapred.Task: attempt_local1450046861_0001_r_000000_0' done.
3/10/05 01:23:50 INFO mapred.LocalJobRunner: task attempt_local1450046861_0001_r_000000_0
3/10/05 01:23:50 INFO mapred.LocalJobRunner: reduce task executor complete
3/10/05 01:23:51 INFO mapreduce.Job: Job job_local1450046861_0001 running in uber mode : false
3/10/05 01:23:51 INFO mapreduce.Job: map 100% reduce 100%
3/10/05 01:23:51 INFO mapreduce.Job: Job job_local1450046861_0001 completed successfully
3/10/05 01:23:51 INFO mapreduce.Job: Counters:
      FILE: Number of bytes read=246246
      FILE: Number of bytes written=745550
      FILE: Number of read operations=0
      FILE: Number of large read operations=0
      FILE: Number of write operations=0
Map-Reduce Metrics:
      Map Input Records=3580
      Map output records=2425
      Map output bytes=26925
      Map output materialized bytes=31554
      Input split bytes=134
      Combine input records=0
      Combine output records=0
      Reduce Input groups=4
      Reduce shuffle bytes=31554
      Reduce input records=2425
      Reduce output records=4
      Split bytes=14850
      Shuffled Maps =1
      Failed Shuffles=0
      Merged Map outputs=1
      GC time elapsed (ms)=7
      Total committed heap usage (bytes)=265289728
Shuffle Metrics:
      BAD_ID=0
      CONNECTION=0
      IO_ERROR=0
      WRONG_LENGTH=0
      WRONG_MAP=0
      WRONG_PARTITION=0
      File Input Format Counters
        Bytes Read=87194
      File Output Format Counters
        Bytes Written=0
hadoop@sankey-Lenovo: /usr/local/hadoop/share/KULKARNI/SumCardNums$ hadoop jar /usr/local/hadoop/share/KULKARNI/SumCardNums/random_card_addition.jar CardCountDriver /usr/local/hadoop/share/KULKARNI/SumCardNums/fa

```

Output file for 100 deck looks like:

		part-r-00000
1	clubs	3578
2	Diamonds	3737
3	Heart	3574
4	Spades	3689

b) Trying for 1000 decks

Screenshots for 1000 decks on new input

```

Bytes Written=60
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/SunCardNums$ hadoop jar /usr/local/hadoop/share/KULKARNI/SunCardNums/random_card_addition.jar CardCountDriver /usr/local/hadoop/share/KULKARNI/SunCardNums/rando_input_1000.txt /usr/local/hadoop/share/KULKARNI/SunCardNums/new_output_1000
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.8.1.jar) to method sun.security.krb5.Config.getInstanc
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
3/10/05 01:24:01 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
3/10/05 01:24:01 INFO mapred.JobTracker: Starting task attempt_local781573299_0001_m_000000
3/10/05 01:24:01 INFO mapred.FileOutputFormat: Total input files in process : 1
3/10/05 01:24:01 INFO mapreduce.JobSubmission: number of splits:1
3/10/05 01:24:02 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
3/10/05 01:24:02 INFO mapreduce.Job: Running job: job_local781573299_0001
3/10/05 01:24:02 INFO mapreduce.Job: Output Committer: null
3/10/05 01:24:02 INFO mapreduce.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:24:02 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:24:02 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
3/10/05 01:24:02 INFO mapred.LocalJobRunner: Waiting for map tasks
3/10/05 01:24:02 INFO mapred.LocalJobRunner: Starting task: attempt_local781573299_0001_m_000000_0
3/10/05 01:24:02 INFO mapreduce.Job: Task attempt_local781573299_0001_m_000000_0 is assigned to node: localhost/192.168.1.7:5431
3/10/05 01:24:02 INFO mapreduce.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:24:02 INFO mapred.Task: Using ResourceCalculatorForProcessTree : []
3/10/05 01:24:02 INFO mapred.MapTask: Processing split:file:/usr/local/hadoop/share/KULKARNI/SunCardNums/rando_input_1000.txt:+0+906094
3/10/05 01:24:02 INFO mapred.MapTask: (EQUATOR) 0 kv 26214396(104857584)
3/10/05 01:24:02 INFO mapred.MapTask: mapreduce.task.io.sort.MB: 100
3/10/05 01:24:02 INFO mapred.MapTask: mapreduce.task.io.sort.MB: 8388000
3/10/05 01:24:02 INFO mapred.MapTask: mapreduce.task.io.sort.MB: 104857600
3/10/05 01:24:02 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
3/10/05 01:24:02 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
3/10/05 01:24:02 INFO mapred.MapTask: LocalJobRunner:
3/10/05 01:24:02 INFO mapred.MapTask: Starting flush of map output
3/10/05 01:24:02 INFO mapred.MapTask: Splitting map output
3/10/05 01:24:02 INFO mapred.MapTask: bufvold = 265431; bufvold = 104857600
3/10/05 01:24:02 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend = 26117796(104471184); length = 96601/6553600
3/10/05 01:24:02 INFO mapred.MapTask: Finished spill 0
3/10/05 01:24:02 INFO mapred.Task: Task attempt_local781573299_0001_m_000000_0 is done. And is in the process of committing
3/10/05 01:24:02 INFO mapred.LocalJobRunner: map
3/10/05 01:24:02 INFO mapred.Task: Task attempt_local781573299_0001_m_000000_0 is done.
3/10/05 01:24:02 INFO mapred.LocalJobRunner: Job is finished task attempt_local781573299_0001_m_000000_0
3/10/05 01:24:02 INFO mapred.LocalJobRunner: map task executor complete.
3/10/05 01:24:02 INFO mapred.LocalJobRunner: Waiting for reduce tasks
3/10/05 01:24:02 INFO mapred.Task: Task attempt_local781573299_0001_r_000000_0
3/10/05 01:24:02 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 1
3/10/05 01:24:02 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
3/10/05 01:24:02 INFO mapred.Task: Using ResourceCalculatorForProcessTree : []
3/10/05 01:24:02 INFO mapred.ReduceTask: Using ShuffleConsumerPlugin: org.apache.hadoop.mapreduce.task.reduce.Shuffle@7ea76ad?
3/10/05 01:24:02 INFO reduce.MergerManagerImpl: MergerManager: memoryLimit=293601280, maxSingleShuffleUnitSize=3400320, mergeThreshold=193776848, ioSortFactor=10, memToMemMergeOutputsThreshold=10
3/10/05 01:24:02 INFO reduce.EventFetcher: attempt_local781573299_0001_r_000000_0 Thread started: EventFetcher for fetching Map Completion Events
3/10/05 01:24:02 INFO reduce.LocalFetcher: localFetcher#1 about to shuffle output of map attempt_local781573299_0001_r_000000_0 decompt: 313735 len: 313739 to MEMORY
3/10/05 01:24:02 INFO reduce.InMemoryMapOutput: Read 313735 bytes from map-output for attempt_local781573299_0001_r_000000_0
3/10/05 01:24:02 INFO reduce.MergeManagerImpl: closeInMemoryMapOutput-> map-output size: 313735, inMemoryMapoutputs.size() -> 1, commitMemory -> 0, usedMemory -> 313735
3/10/05 01:24:02 INFO reduce.EventFetcher: EventFetcher is interrupted.. Returning

hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/SunCardNums$ hadoop jar /usr/local/hadoop/share/KULKARNI/SunCardNums/random_card_addition.jar CardCountDriver /usr/local/hadoop/share/KULKARNI/SunCardNums/rando_input_1000.txt /usr/local/hadoop/share/KULKARNI/SunCardNums/new_output_1000
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/SunCardNums$ ls
new_output_1000  new_output_1000_output_100  output_1000  rando_input_1000.txt  rando_input_100.txt  random_card_addition.jar  ran_input_1000.txt  ran_input_100.txt
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/SunCardNums$ cd new_output_100
hadoop@sanket-Lenovo:/usr/local/hadoop/share/KULKARNI/SunCardNums$ sudo gedit part-r-00000

```

Output file for 1000 deck looks like:

```
Clubs 36495
2 Diamonds 35730
3 Heart 36298
4 Spades 36843
```

Outcomes:

Able to create a hadoop environment from scratch and setup its requirements alongside

Able to run the example jar files given in the hadoop folder by default

Able to run a python mapper and reducer using hadoop streaming and generate the output

Able to run a java mapper and reducer with driver class by creating its jar

References used:

https://www.3pillarglobal.com/insights/a-quick-set-up-guide-for-single-node-hadoop-clusters/?gclid=CjwKCAjwp_GJBhBmEiwALWBQk56GSPjAZAZPw7Qc5DqkqtQPuf_J1jiu54zNkXWsSQ7U89G6LBm4rBoC2lgQAvD_BwE

<https://www.projectpro.io/hadoop-tutorial/hadoop-multinode-cluster-setup>

<https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html>

<https://www.youtube.com/watch?v=wTkffAYsCBw>

<https://hadoop.apache.org/docs/stable/>