Kajol Tanesh Shah
A20496724
Fall 2022

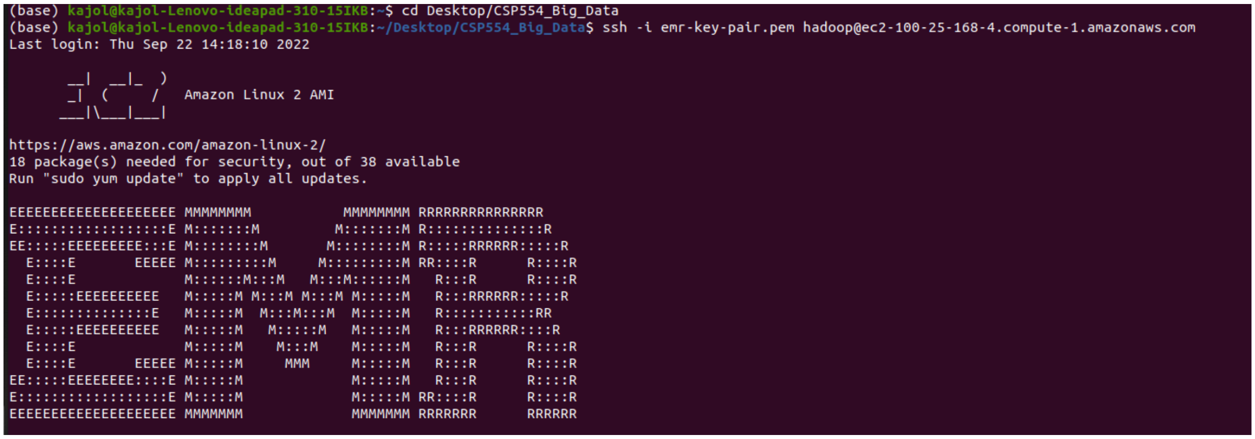
CSP554—Big Data Technologies

Assignment #3 (Modules 03a & 03b, 15 points)

5) Install the mrjob library on your EMR master node.

a) ssh to the master node (/home/hadoop) as you did in assignment #2

Ans.



```
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ cd Desktop/CSP554_Big_Data
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~/Desktop/CSP554_Big_Data$ ssh -i emr-key-pair.pem hadoop@ec2-100-25-168-4.compute-1.amazonaws.com
Last login: Thu Sep 22 14:18:10 2022

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
18 package(s) needed for security, out of 38 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR
```

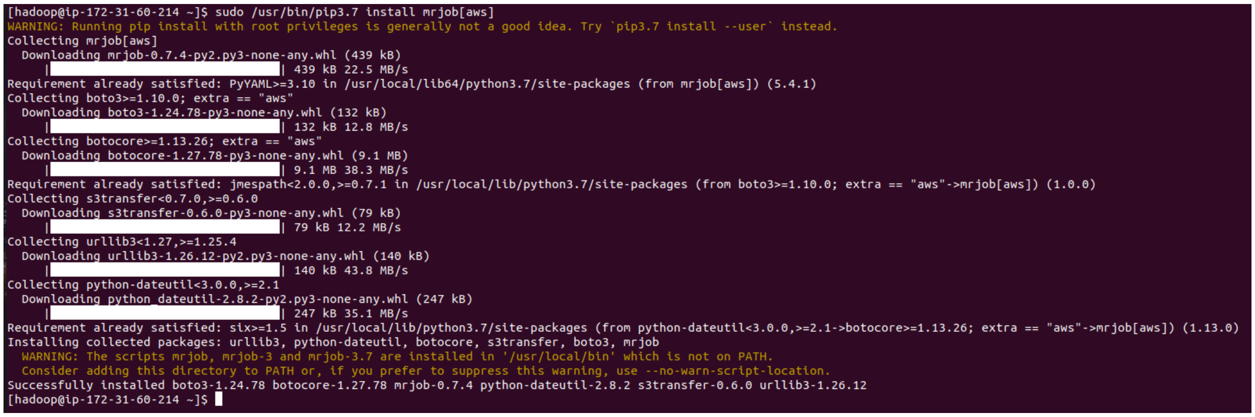
b) Enter the following (note if the first command does not work, try the second)

sudo /usr/bin/pip3.7 install mrjob[aws]

or try:

sudo /usr/bin/pip3 install mrjob[aws]

Ans.



```
[hadoop@ip-172-31-60-214 ~]$ sudo /usr/bin/pip3.7 install mrjob[aws]
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3.7 install --user` instead.
Collecting mrjob[aws]
  Downloading mrjob-0.7.4-py2.py3-none-any.whl (439 kB)
     |████████████████████████████████| 439 kB 22.5 MB/s
Requirement already satisfied: PyYAML>=3.10 in /usr/local/lib64/python3.7/site-packages (from mrjob[aws]) (5.4.1)
Collecting boto3>=1.10.0; extra == "aws"
  Downloading boto3-1.24.78-py3-none-any.whl (132 kB)
     |████████████████████████████████| 132 kB 12.8 MB/s
Collecting botocore>=1.13.26; extra == "aws"
  Downloading botocore-1.27.78-py3-none-any.whl (9.1 MB)
     |████████████████████████████████| 9.1 MB 38.3 MB/s
Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /usr/local/lib/python3.7/site-packages (from boto3>=1.10.0; extra == "aws"->mrjob[aws]) (1.0.0)
Collecting s3transfer<0.7.0,>=0.6.0
  Downloading s3transfer-0.6.0-py3-none-any.whl (79 kB)
     |████████████████████████████████| 79 kB 12.2 MB/s
Collecting urllib3<1.27,>=1.25.4
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
     |████████████████████████████████| 140 kB 43.8 MB/s
Collecting python-dateutil<3.0.0,>=2.1
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
     |████████████████████████████████| 247 kB 35.1 MB/s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-packages (from python-dateutil<3.0.0,>=2.1->botocore>=1.13.26; extra == "aws"->mrjob[aws]) (1.13.0)
Installing collected packages: urllib3, python-dateutil, botocore, s3transfer, boto3, mrjob
  WARNING: The scripts mrjob, mrjob-3 and mrjob-3.7 are installed in '/usr/local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed boto3-1.24.78 botocore-1.27.78 mrjob-0.7.4 python-dateutil-2.8.2 s3transfer-0.6.0 urllib3-1.26.12
[hadoop@ip-172-31-60-214 ~]$ █
```

6) Next you will set up to execute the provided WordCount.py map reduce program found in the "Assignments" section of the Blackboard. This is the exact same program we saw in class.

Step 1:

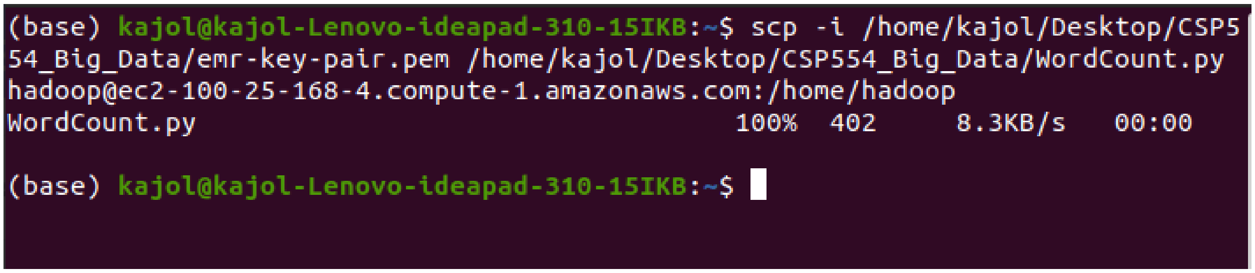
Download the two files "w.data" and "WordCount.py" to your PC or Mac. They are part of the documents included with the assignment.

Step 2:

Now open another terminal window (but don't use it to ssh to the master node). This will allow you to access files on your PC or MAC to upload them to the Hadoop master node.

From this terminal window use the secure copy (scp) program to move the WordCount.py file to the /home/hadoop directory of the master node.

Ans.



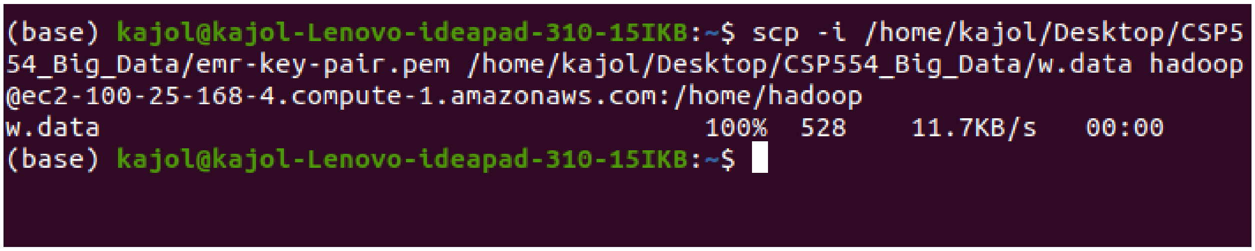
```
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ scp -i /home/kajol/Desktop/CSP5
54_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/WordCount.py
hadoop@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
WordCount.py                                     100%  402     8.3KB/s   00:00

(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ 
```

Step 3:

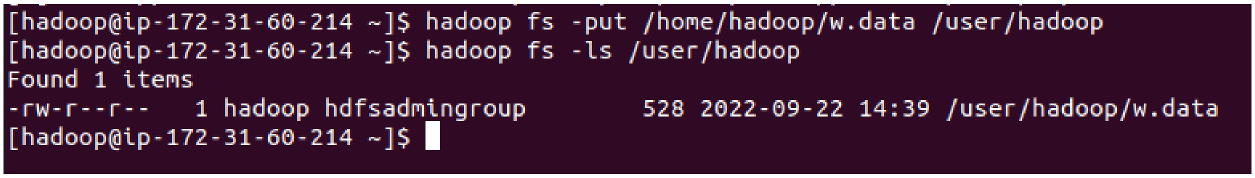
Do the same for the assignment file w.data. That is move it to the directory /home/Hadoop on the Hadoop master node Linux file system.

In this case copy the file from the Linux "/home/hadoop" directory to the Hadoop file system (HDFS), say to the directory "/user/hadoop"

Ans.



```
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ scp -i /home/kajol/Desktop/CSP5
54_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/w.data hadoop
@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
w.data                                         100%  528    11.7KB/s   00:00
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ 
```



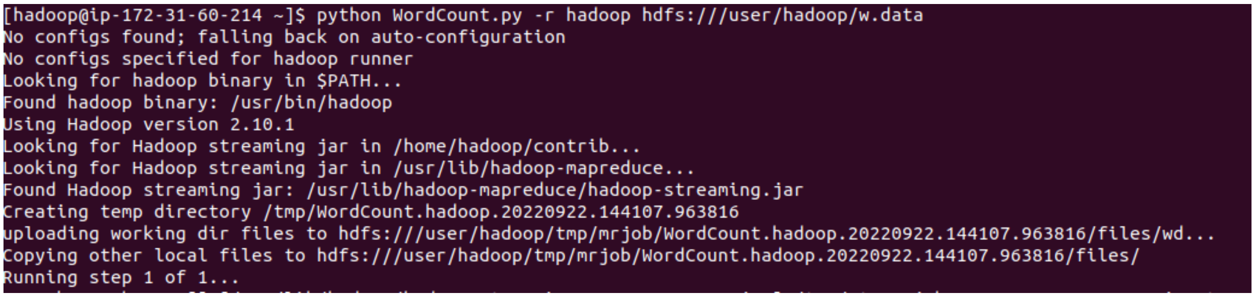
```
[hadoop@ip-172-31-60-214 ~]$ hadoop fs -put /home/hadoop/w.data /user/hadoop
[hadoop@ip-172-31-60-214 ~]$ hadoop fs -ls /user/hadoop
Found 1 items
-rw-r--r--   1 hadoop hdfsadmingroup        528 2022-09-22 14:39 /user/hadoop/w.data
[hadoop@ip-172-31-60-214 ~]$ 
```

Step 4:

Now execute the following

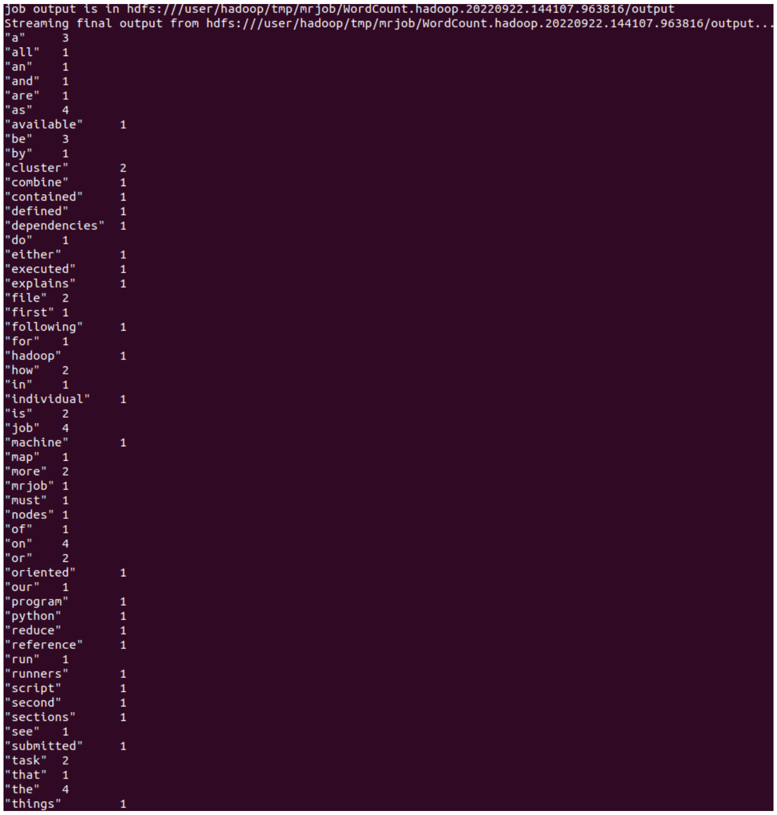
python WordCount.py -r hadoop hdfs:///user/hadoop/w.data

Ans.



```
[hadoop@ip-172-31-60-214 ~]$ python WordCount.py -r hadoop hdfs:///user/hadoop/w.data
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
Found Hadoop streaming jar: /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
Creating temp directory /tmp/WordCount.hadoop.20220922.144107.963816
uploading working dir files to hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.144107.963816/files/wd...
Copying other local files to hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.144107.963816/files/
Running step 1 of 1...
```

Output



```
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.144107.963816/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.144107.963816/output...
"a"     3
"all"   1
"an"    1
"and"   1
"are"   1
"as"    4
"available"     1
"be"    3
"by"    1
"cluster"       2
"combine"       1
"contained"     1
"defined"       1
"dependencies"  1
"do"    1
"either"        1
"executed"      1
"explains"      1
"file"  2
"first" 1
"following"     1
"for"   1
"hadoop"        1
"how"   2
"in"    1
"individual"    1
"is"    2
"job"   4
"machine"       1
"map"   1
"more"  2
"mrjob" 1
"must"  1
"nodes" 1
"of"    1
"on"    4
"or"    2
"oriented"      1
"our"   1
"program"       1
"python"        1
"reduce"        1
"reference"     1
"run"   1
"runners"       1
"script"        1
"second"        1
"sections"      1
"see"   1
"submitted"     1
"task"  2
"that"  1
"the"   4
"things"        1
```

5) Now slightly modify the WordCount.py program. Call the new program WordCount2.py.

Instead of counting how many words there are in the input documents (w.data), modify the program to count how many words begin with the small letters a-n and how many begin with anything else.

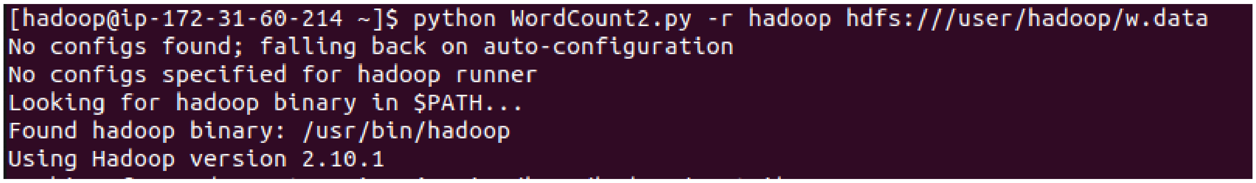
The output file should look something like

a_to_n, 12

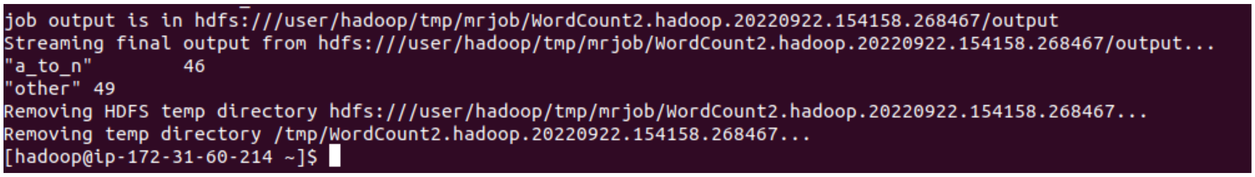
other, 21

Now execute the program and see what happens.

Ans.



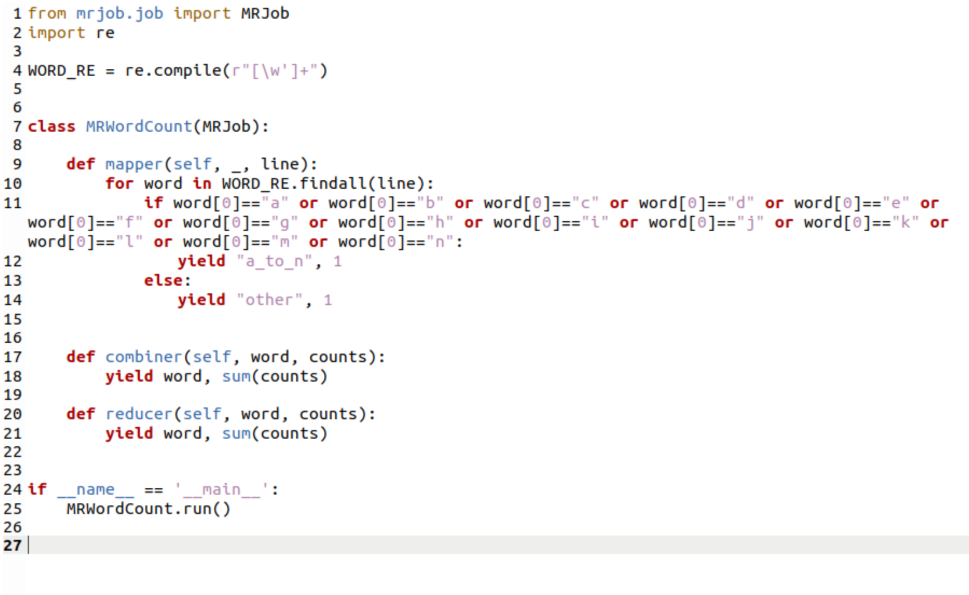
```
[hadoop@ip-172-31-60-214 ~]$ python WordCount2.py -r hadoop hdfs:///user/hadoop/w.data
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
```



```
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467/output...
"a_to_n"        46
"other" 49
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467...
Removing temp directory /tmp/WordCount2.hadoop.20220922.154158.268467...
[hadoop@ip-172-31-60-214 ~]$ 
```

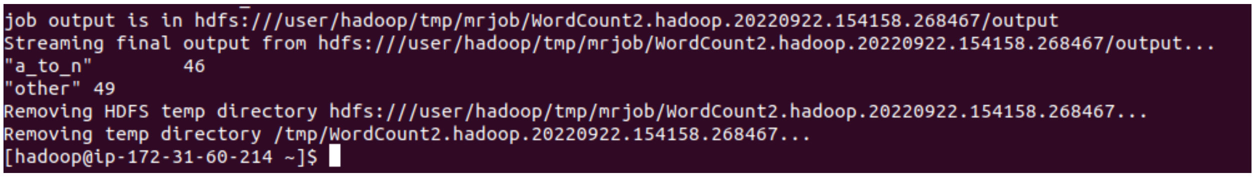
6) (5 points) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.\

Ans. Modified code:-



```python
from mrjob.job import MRJob
import re

WORD_RE = re.compile(r"[\w']+")


class MRWordCount(MRJob):

    def mapper(self, _, line):
        for word in WORD_RE.findall(line):
            if word[0]=="a" or word[0]=="b" or word[0]=="c" or word[0]=="d" or word[0]=="e" or word[0]=="f" or word[0]=="g" or word[0]=="h" or word[0]=="i" or word[0]=="j" or word[0]=="k" or word[0]=="l" or word[0]=="m" or word[0]=="n":
                yield "a_to_n", 1
            else:
                yield "other", 1


    def combiner(self, word, counts):
        yield word, sum(counts)

    def reducer(self, word, counts):
        yield word, sum(counts)


if __name__ == '__main__':
    MRWordCount.run()


```

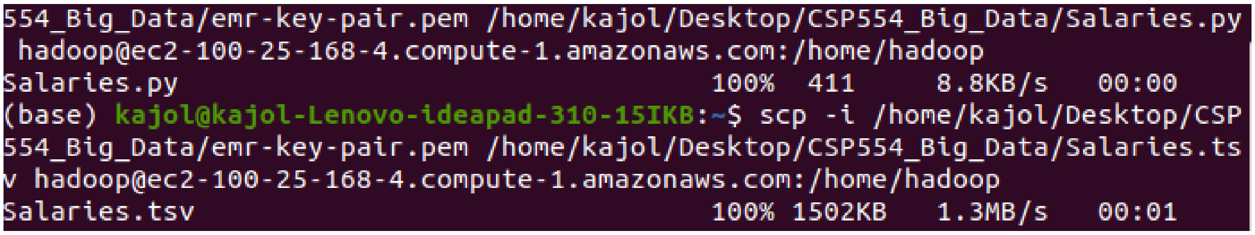
output for WordCount2.py



```
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467/output...
"a_to_n"        46
"other" 49
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.154158.268467...
Removing temp directory /tmp/WordCount2.hadoop.20220922.154158.268467...
[hadoop@ip-172-31-60-214 ~]$ 
```

7) Now do the same as the above for the files Salaries.py and Salaries.tsv. The ".tsv" file holds department and salary information for Baltimore municipal workers. Have a look at Salaries.py for the layout of the ".tsv" file and how to read it in to our map reduce program.

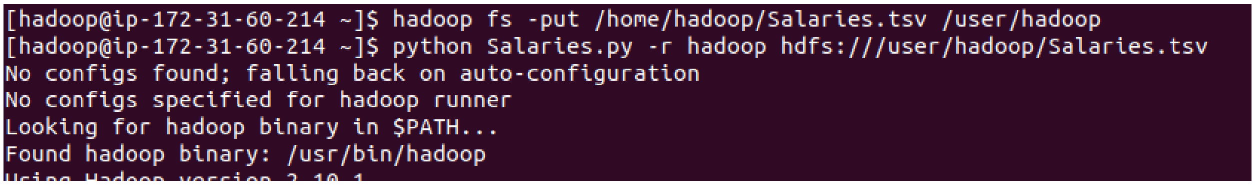
Ans.



```
554_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/Salaries.py
 hadoop@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
Salaries.py                                   100%  411      8.8KB/s   00:00
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ scp -i /home/kajol/Desktop/CSP
554_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/Salaries.ts
v hadoop@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
Salaries.tsv                                  100% 1502KB    1.3MB/s   00:01
```

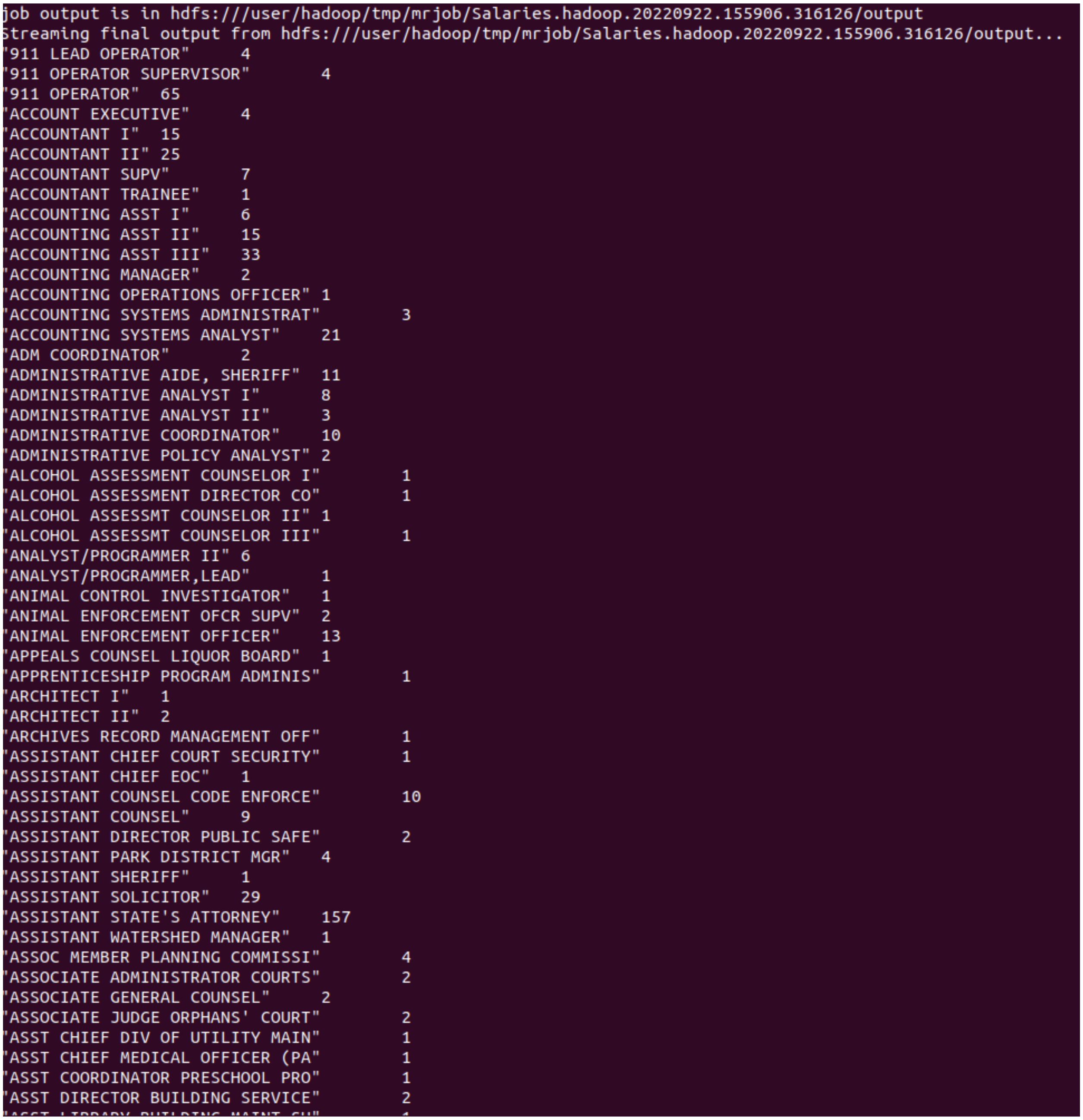
8) Execute the Salaries.py program to make sure it works. It should print out how many workers share each job title.

Ans.



```
[hadoop@ip-172-31-60-214 ~]$ hadoop fs -put /home/hadoop/Salaries.tsv /user/hadoop
[hadoop@ip-172-31-60-214 ~]$ python Salaries.py -r hadoop hdfs:///user/hadoop/Salaries.tsv
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
```

Output:-



```
job output is in hdfs:///user/hadoop/tmp/mrjob/Salaries.hadoop.20220922.155906.316126/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/Salaries.hadoop.20220922.155906.316126/output...
"911 LEAD OPERATOR"     4
"911 OPERATOR SUPERVISOR"       4
"911 OPERATOR"  65
"ACCOUNT EXECUTIVE"     4
"ACCOUNTANT I"  15
"ACCOUNTANT II" 25
"ACCOUNTANT SUPV"       7
"ACCOUNTANT TRAINEE"    1
"ACCOUNTING ASST I"     6
"ACCOUNTING ASST II"    15
"ACCOUNTING ASST III"   33
"ACCOUNTING MANAGER"    2
"ACCOUNTING OPERATIONS OFFICER" 1
"ACCOUNTING SYSTEMS ADMINISTRAT"        3
"ACCOUNTING SYSTEMS ANALYST"    21
"ADM COORDINATOR"       2
"ADMINISTRATIVE AIDE, SHERIFF"  11
"ADMINISTRATIVE ANALYST I"      8
"ADMINISTRATIVE ANALYST II"     3
"ADMINISTRATIVE COORDINATOR"    10
"ADMINISTRATIVE POLICY ANALYST" 2
"ALCOHOL ASSESSMENT COUNSELOR I"        1
"ALCOHOL ASSESSMENT DIRECTOR CO"        1
"ALCOHOL ASSESSMT COUNSELOR II" 1
"ALCOHOL ASSESSMT COUNSELOR III"        1
"ANALYST/PROGRAMMER II" 6
"ANALYST/PROGRAMMER,LEAD"       1
"ANIMAL CONTROL INVESTIGATOR"   1
"ANIMAL ENFORCEMENT OFCR SUPV"  2
"ANIMAL ENFORCEMENT OFFICER"    13
"APPEALS COUNSEL LIQUOR BOARD"  1
"APPRENTICESHIP PROGRAM ADMINIS"        1
"ARCHITECT I"   1
"ARCHITECT II"  2
"ARCHIVES RECORD MANAGEMENT OFF"        1
"ASSISTANT CHIEF COURT SECURITY"        1
"ASSISTANT CHIEF EOC"   1
"ASSISTANT COUNSEL CODE ENFORCE"        10
"ASSISTANT COUNSEL"     9
"ASSISTANT DIRECTOR PUBLIC SAFE"        2
"ASSISTANT PARK DISTRICT MGR"   4
"ASSISTANT SHERIFF"     1
"ASSISTANT SOLICITOR"   29
"ASSISTANT STATE'S ATTORNEY"    157
"ASSISTANT WATERSHED MANAGER"   1
"ASSOC MEMBER PLANNING COMMISSI"        4
"ASSOCIATE ADMINISTRATOR COURTS"        2
"ASSOCIATE GENERAL COUNSEL"     2
"ASSOCIATE JUDGE ORPHANS' COURT"        2
"ASST CHIEF DIV OF UTILITY MAIN"        1
"ASST CHIEF MEDICAL OFFICER (PA"        1
"ASST COORDINATOR PRESCHOOL PRO"        1
"ASST DIRECTOR BUILDING SERVICE"        2
"ASST LIBRARY BUILDING MAINT SU"        1
```

9) Now modify the Salaries.py program. Call it Salaries2.py

Instead of counting the number of workers per department, change the program to provide the number of workers having High, Medium or Low annual salaries. This is defined as follows:

High	100,000.00 and above
Medium	50,000.00 to 99,999.99
Low	0.00 to 49,999.99

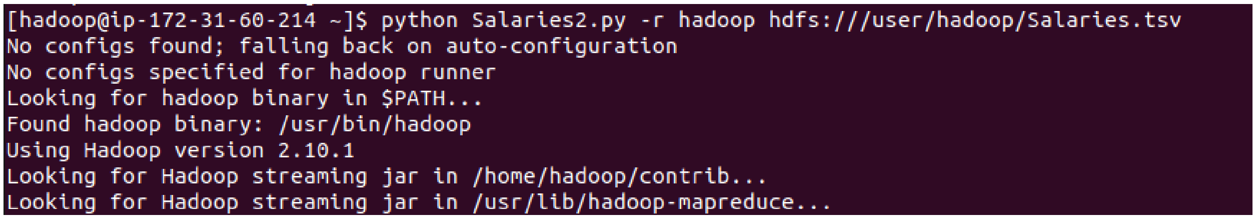
The output of the program should be something like the following (in any order):

High 20

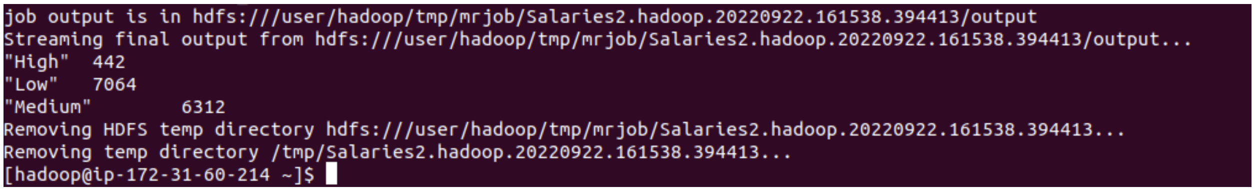
Medium 30

Low 10

Ans.



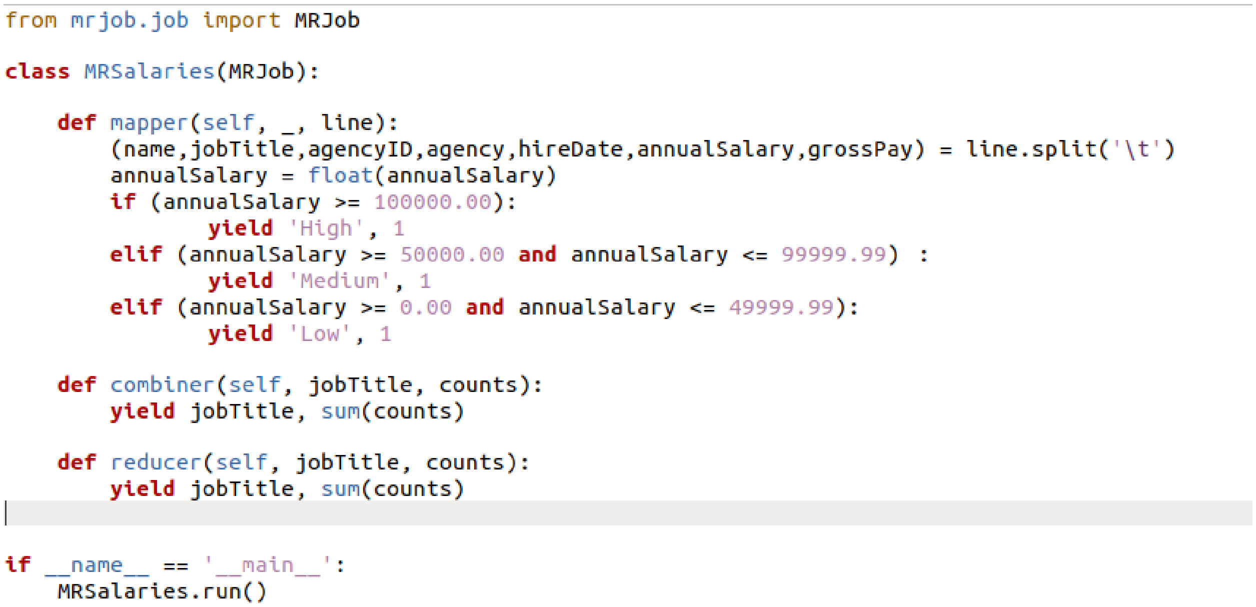
```
[hadoop@ip-172-31-60-214 ~]$ python Salaries2.py -r hadoop hdfs:///user/hadoop/Salaries.tsv
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
```



```
job output is in hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413/output...
"High"  442
"Low"   7064
"Medium"        6312
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413...
Removing temp directory /tmp/Salaries2.hadoop.20220922.161538.394413...
[hadoop@ip-172-31-60-214 ~]$
```

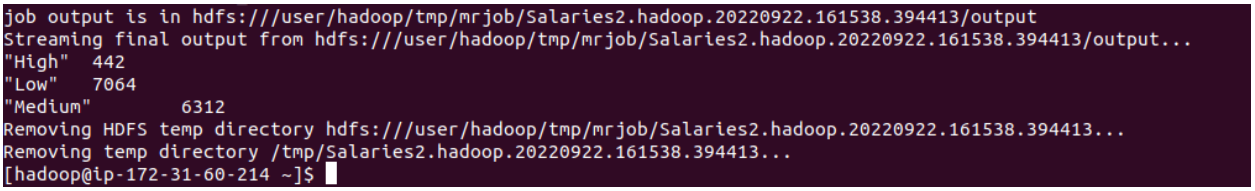
10) (5 points) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.

Ans. Modified code:-



```
from mrjob.job import MRJob

class MRSalaries(MRJob):

    def mapper(self, _, line):
        (name,jobTitle,agencyID,agency,hireDate,annualSalary,grossPay) = line.split('\t')
        annualSalary = float(annualSalary)
        if (annualSalary >= 100000.00):
                yield 'High', 1
        elif (annualSalary >= 50000.00 and annualSalary <= 99999.99)  :
                yield 'Medium', 1
        elif (annualSalary >= 0.00 and annualSalary <= 49999.99):
                yield 'Low', 1

    def combiner(self, jobTitle, counts):
        yield jobTitle, sum(counts)

    def reducer(self, jobTitle, counts):
        yield jobTitle, sum(counts)


if __name__ == '__main__':
    MRSalaries.run()
```

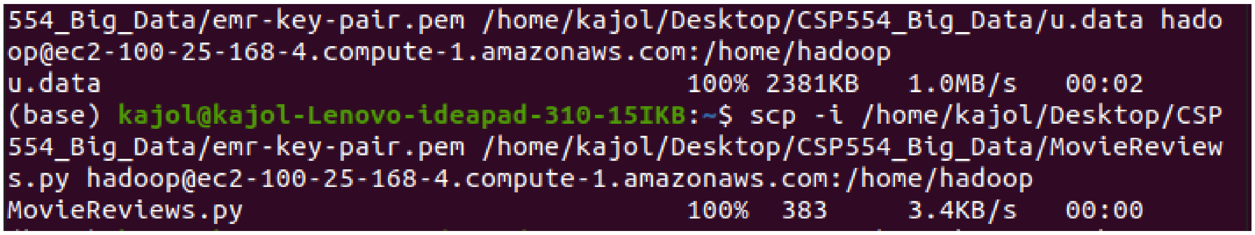
Output:-



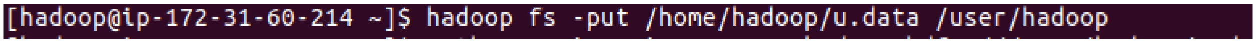
```
job output is in hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413/output...
"High"  442
"Low"   7064
"Medium"        6312
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.161538.394413...
Removing temp directory /tmp/Salaries2.hadoop.20220922.161538.394413...
[hadoop@ip-172-31-60-214 ~]$ 
```

11) Now copy the file u.data from the assignment to /user/hadoop. This is similar to the file used for some examples in Module 03b. **NOTE: unlike the slide deck examples, this version of u.data has fields separated by commas and not tabs.**

Ans.



```
554_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/u.data hado
op@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
u.data                                        100% 2381KB   1.0MB/s   00:02
(base) kajol@kajol-Lenovo-ideapad-310-15IKB:~$ scp -i /home/kajol/Desktop/CSP
554_Big_Data/emr-key-pair.pem /home/kajol/Desktop/CSP554_Big_Data/MovieReview
s.py hadoop@ec2-100-25-168-4.compute-1.amazonaws.com:/home/hadoop
MovieReviews.py                               100%  383     3.4KB/s   00:00
```



```
[hadoop@ip-172-31-60-214 ~]$ hadoop fs -put /home/hadoop/u.data /user/hadoop
```

12) (5 points) Review the slides 55-61 in lecture notes Module 3b. Now write a program to perform the task of outputting a count of the number of movies each user (identified via their user id) reviewed.

Output might look something like the following:

186: 2

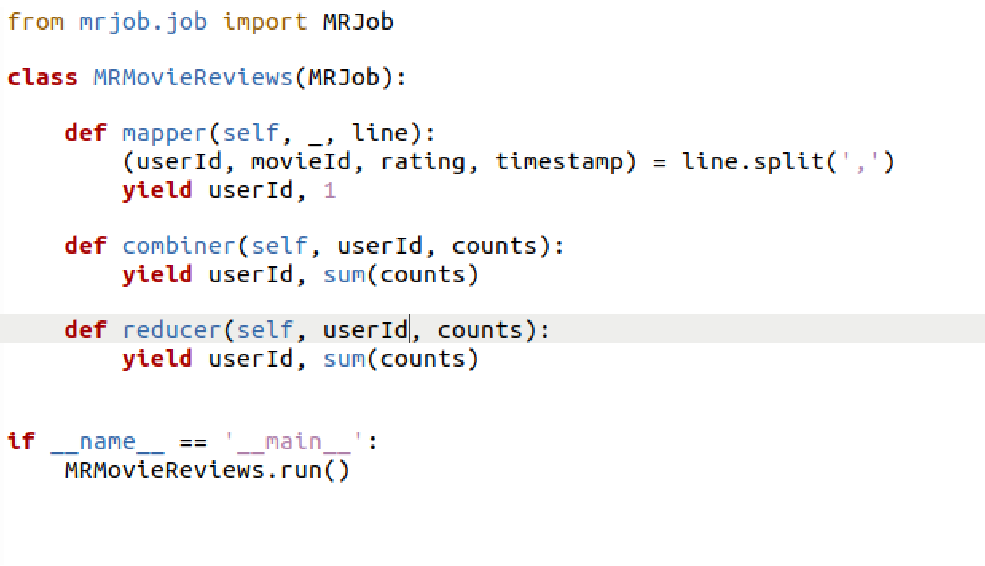
192: 2

112: 1

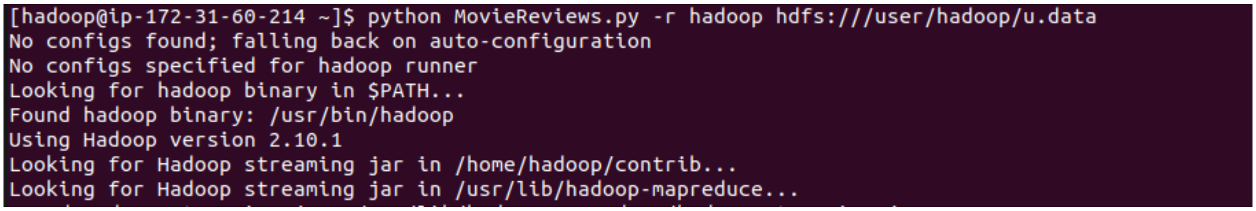
etc.

Submit a copy of this program and a screen shot of the results of the program's execution (only 10 lines or so of the result) as the output of your assignment.

Ans. Code:-

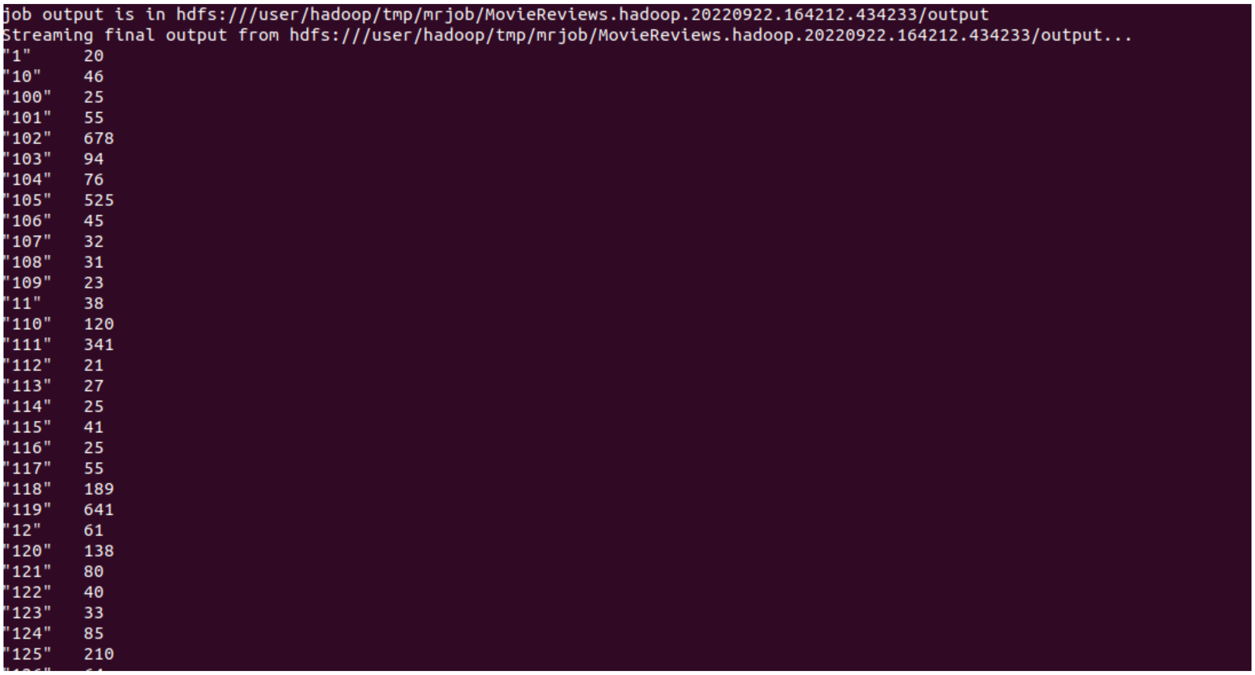


```
from mrjob.job import MRJob

class MRMovieReviews(MRJob):

    def mapper(self, _, line):
        (userId, movieId, rating, timestamp) = line.split(',')
        yield userId, 1

    def combiner(self, userId, counts):
        yield userId, sum(counts)

    def reducer(self, userId, counts):
        yield userId, sum(counts)


if __name__ == '__main__':
    MRMovieReviews.run()
```

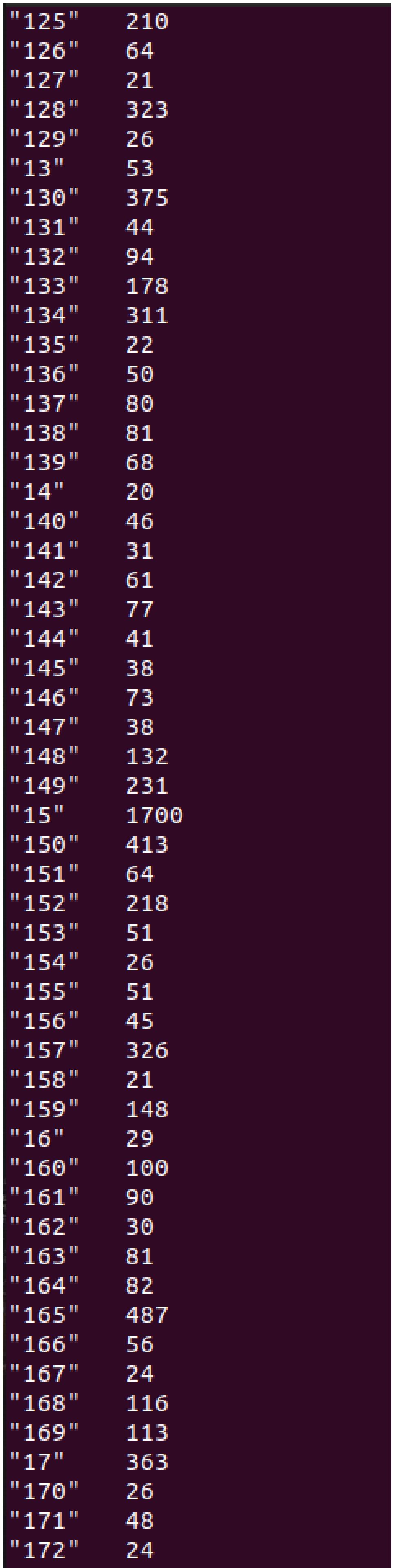


```
[hadoop@ip-172-31-60-214 ~]$ python MovieReviews.py -r hadoop hdfs:///user/hadoop/u.data
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
```

Output:-



```
job output is in hdfs:///user/hadoop/tmp/mrjob/MovieReviews.hadoop.20220922.164212.434233/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/MovieReviews.hadoop.20220922.164212.434233/output...
"1"     20
"10"    46
"100"   25
"101"   55
"102"   678
"103"   94
"104"   76
"105"   525
"106"   45
"107"   32
"108"   31
"109"   23
"11"    38
"110"   120
"111"   341
"112"   21
"113"   27
"114"   25
"115"   41
"116"   25
"117"   55
"118"   189
"119"   641
"12"    61
"120"   138
"121"   80
"122"   40
"123"   33
"124"   85
"125"   210
```



```
"125"   210
"126"   64
"127"   21
"128"   323
"129"   26
"13"    53
"130"   375
"131"   44
"132"   94
"133"   178
"134"   311
"135"   22
"136"   50
"137"   80
"138"   81
"139"   68
"14"    20
"140"   46
"141"   31
"142"   61
"143"   77
"144"   41
"145"   38
"146"   73
"147"   38
"148"   132
"149"   231
"15"    1700
"150"   413
"151"   64
"152"   218
"153"   51
"154"   26
"155"   51
"156"   45
"157"   326
"158"   21
"159"   148
"16"    29
"160"   100
"161"   90
"162"   30
"163"   81
"164"   82
"165"   487
"166"   56
"167"   24
"168"   116
"169"   113
"17"    363
"170"   26
"171"   48
"172"   24
```