Management of Big Data and Tools – DS8003 – Fall 2016

MidTerm

NAJLIS, BERNARDO - Student Number #500744793

Open-book, Open-notes, Open-internet

Dataset:

1. We are going to use the Dataset : **midterm_data.zip** [Download Dataset from our D2L website under MidTerm Exam]
2. Unzip the midterm_data.zip.
3. The zip contains files: **u.data** and **u.item.less** and **u.join**
4. **u.data** -- The dataset has 100000 ratings by 943 users on 1682 movies. The file has **4 tab** ("\t") separated columns. The first column is the **user id,** the second column is the **movie id,** the third column is the **rating,** and the fourth column is a **timestamp**.
5. **u.item** - Information about the items (movies); this is a tab separated file with 3 columns. The first column is **movie id**, the second column is**movie name**, and the third column is **release date**.
6. **u.join -** has data from u.data and u.item combined. The first column column has "A" or "B". **"A" denotes u.data and "B" denotes u.item.less.**You can use it for the Map-Reduce job for the question number 2.
7. Copy **u.data and u.item.less and u.join** to the virtual machine (**Filezilla**)
8. Copy **u.data and u.item.less and u.join** from virtual machine into HDFS (**hadoop fs -put**)
9. Create **one table for u.data and one table for u.item.less** in Hive

Submission:

1. Submit the hive commands, Map-reduce python programs, sample results (described below), and screenshot(s) to show that queries or jobs executed.

2. Submit using Assessment -> Assignments-> MidTerm

3. Submit Text (Word) document or PDF file. Paste your code and any explanations into that file. You can additionally also submit your python codes.

4. Table creation and loading statements do not have to be included in the submission

Question: Solve each question using Hive Query and Map-Reduce (Total of 25)

Note: You **DO NOT** have to de-duplicate the results. The same movie with same rating can appear multiple times in the results.

1. Find all rows (or lines) with rating greater than 3 and output the movie id and rating. [Hive - 5 points; Map-Reduce - 5 points]

- **Sample Result For Submission:** Submit all lines/rows corresponding to ID 998

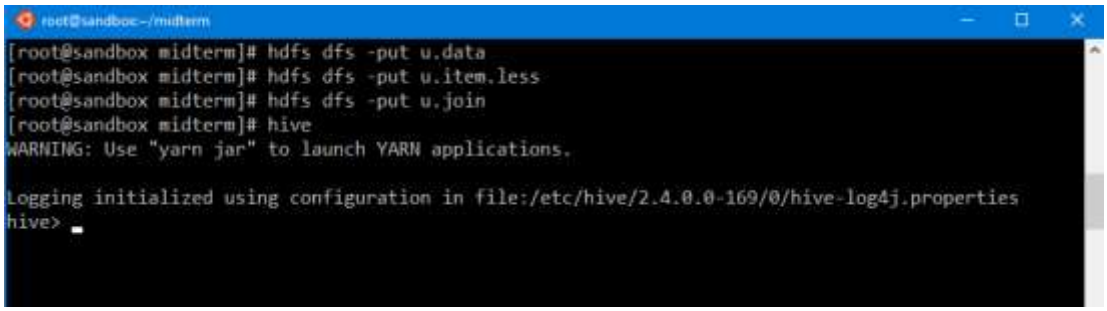
2. Find all rows (or lines) with rating greater than 3 and output the movie name and rating. [Hive - 7.5 points; Map-Reduce - 7.5 points]

- **Sample Result for Submission:** Submit all lines/rows corresponding to Movie Name "Cabin Boy (1994)"

Resolution

0. Environment setup
    a. Files copied into HDFS and invoke hive.

```
hdfs dfs -ls
hdfs dfs -put u.data
hdfs dfs -put u.item.less
hdfs dfs -put u.join
hive
```



    b. Create database and tables in hive.

```
CREATE DATABASE midterm;
USE midterm;
CREATE TABLE midterm.data (
      user_id STRING,
      movie_id STRING,
      rating INT,
      ts TIMESTAMP
) ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t';
LOAD DATA INPATH '/user/root/u.data'
OVERWRITE INTO TABLE midterm.data;
SELECT * FROM midterm.data LIMIT 5;
CREATE TABLE midterm.item (
      movie_id INT,
      movie_name STRING,
      release_date STRING
) ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t';
LOAD DATA INPATH '/user/root/u.item.less'
OVERWRITE INTO TABLE midterm.item;
SELECT * FROM midterm.item LIMIT 5;
```

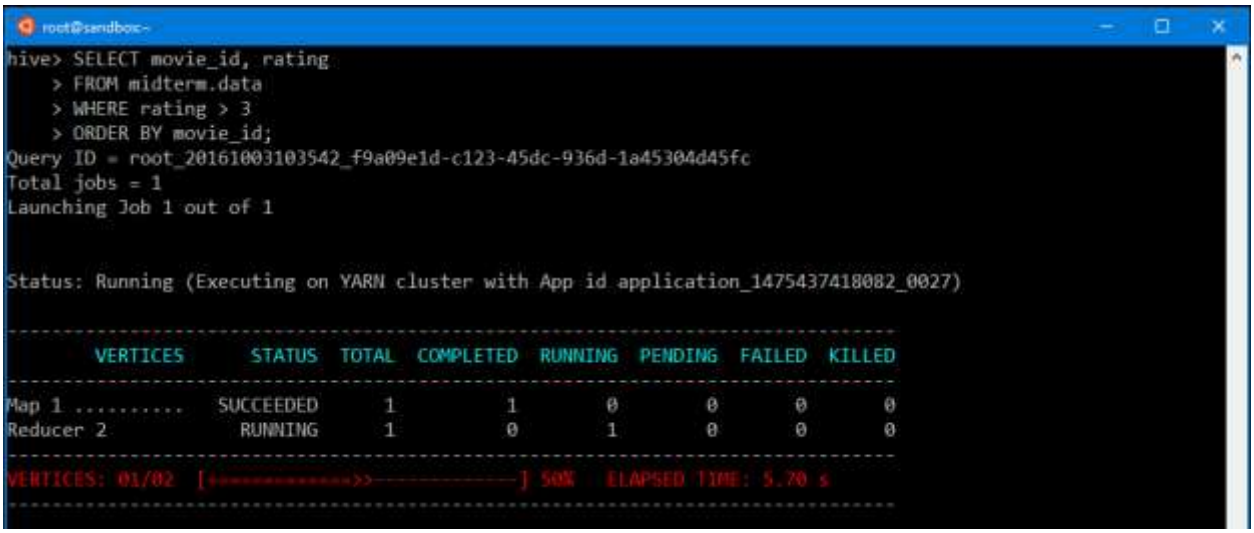
```
hive> CREATE DATABASE midterm;
OK
Time taken: 0.669 seconds
hive> USE midterm;
OK
Time taken: 0.494 seconds
hive> CREATE TABLE midterm.data (
    > user_id STRING,
    > movie_id STRING,
    > rating INT,
    > ts TIMESTAMP
    > ) ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t';
OK
Time taken: 1.272 seconds
hive> LOAD DATA INPATH '/user/root/u.data'
    > OVERWRITE INTO TABLE midterm.data;
Loading data to table midterm.data
chgrp: changing ownership of 'hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/midterm.db/data/u.data'
: User does not belong to hdfs
Table midterm.data stats: [numFiles=1, totalSize=1979173]
OK
Time taken: 1.684 seconds
hive> SELECT * FROM midterm.data LIMIT 5;
OK
196     242     3       NULL
186     302     3       NULL
22      377     1       NULL
244     51      2       NULL
166     346     1       NULL
Time taken: 0.683 seconds, Fetched: 5 row(s)
hive>
```

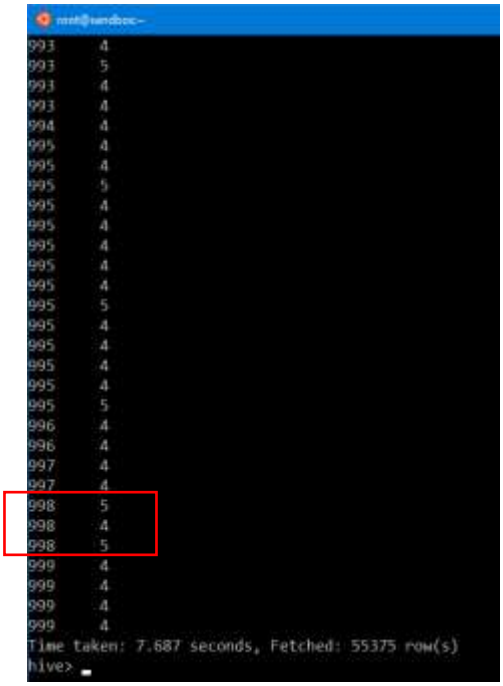
```
hive> CREATE TABLE midterm.item (
    > movie_id INT,
    > movie_name STRING,
    > release_date STRING
    > ) ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t';
OK
Time taken: 1.251 seconds
hive> LOAD DATA INPATH '/user/root/u.item.less'
    > OVERWRITE INTO TABLE midterm.item;
Loading data to table midterm.item
chgrp: changing ownership of 'hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/midterm.db/item/u.item.
less': User does not belong to hdfs
Table midterm.item stats: [numFiles=1, totalSize=68616]
OK
Time taken: 1.612 seconds
hive> SELECT * FROM midterm.item LIMIT 5;
OK
1       Toy Story (1995)        01-Jan-1995
2       GoldenEye (1995)        01-Jan-1995
3       Four Rooms (1995)       01-Jan-1995
4       Get Shorty (1995)       01-Jan-1995
5       Copycat (1995)  01-Jan-1995
Time taken: 0.744 seconds, Fetched: 5 row(s)
hive>
```

1. **Hive HQL Query code**

```
SELECT movie_id, rating
FROM midterm.data
WHERE rating > 3
ORDER BY movie_id;
```

<u>Note</u>: ORDER BY is not required to solve this and was added just to produce the sample output requested.





Mapper Python Code:

```
#!/usr/bin/env python

import sys

for line in sys.stdin:
      line = line.strip()
      movie = line.split('\t')
      if len(movie) == 4:
            try:
                  print "%s\t%s" % (movie[2], movie[1])
            except:
                  continue
```

Reducer Python Code:

```
#!/usr/bin/env python

import sys

for line in sys.stdin:
      line = line.strip()
      movie = line.split('\t')
      if len(movie) == 2:
            try:
                  rating = int(movie[0])
                  movieid = int(movie[1])
            except ValueError:
                  continue
            if rating > 3:
                  print ("%i\t%i") % (movie_id, rating)
```

Execution in Hadoop:

```
hadoop jar /usr/hdp/2.4.0.0-169/hadoop-mapreduce/hadoop-
streaming-2.7.1.2.4.0.0-169.jar -file ./midterm_item1_mapper.py -
mapper midterm_item1_mapper.py -file ./midterm_item1_reducer.py -
reducer midterm_item1_reducer.py -input /user/root/u.data -output
/user/root/midterm_item1.out
```

root@sandbox:~/midterm

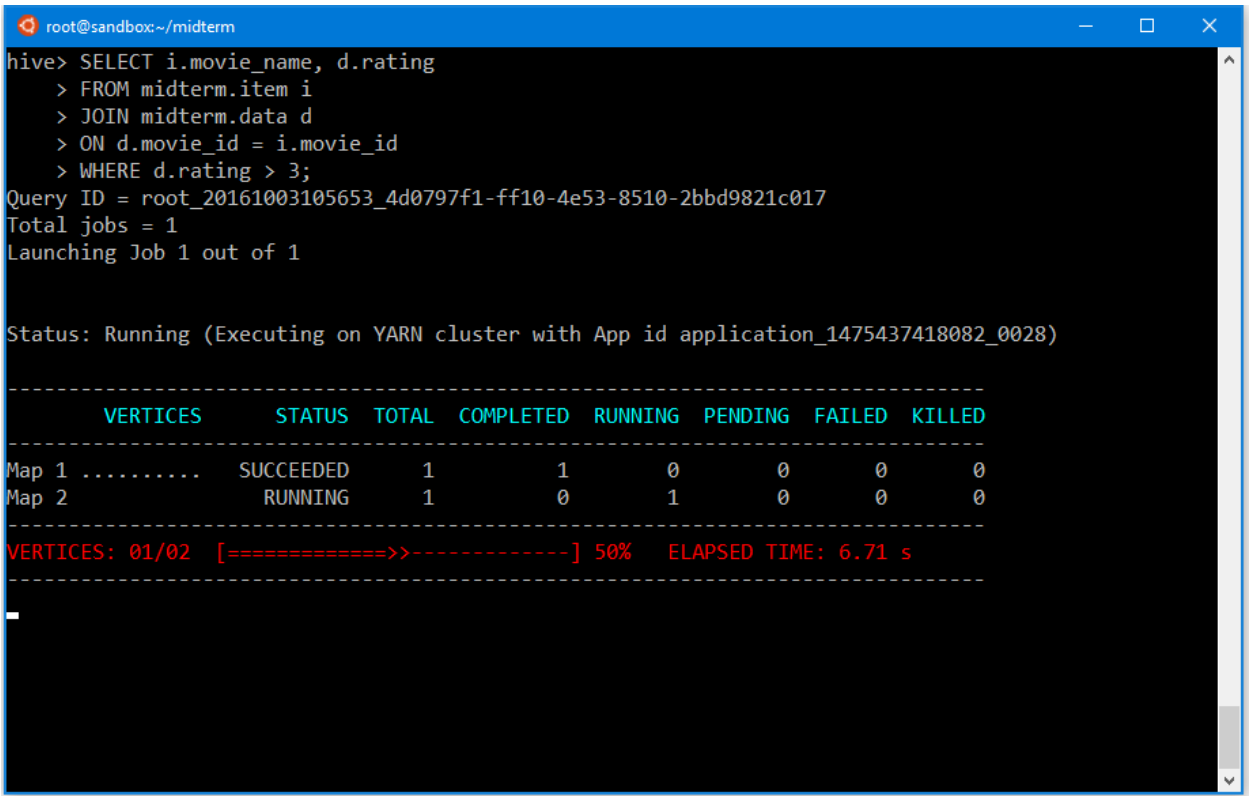
```
bnajlis@XPS15:~$ ssh -lroot -p2222 localhost
[root@sandbox midterm]# hadoop jar /usr/hdp/2.4.0.0-169/hadoop-mapreduce/hadoop-streaming-2.7.1
.2.4.0.0-169.jar -file ./midterm_item1_mapper.py -mapper midterm_item1_mapper.py -file ./midter
m_item1_reducer.py -reducer midterm_item1_reducer.py -input /user/root/u.data -output /user/roo
t/midterm_item1.out
WARNING: Use "yarn jar" to launch YARN applications.
16/10/03 10:26:14 WARN streaming.StreamJob: -file option is deprecated, please use generic opti
on -files instead.
packageJobJar: [./midterm_item1_mapper.py, ./midterm_item1_reducer.py] [/usr/hdp/2.4.0.0-169/ha
doop-mapreduce/hadoop-streaming-2.7.1.2.4.0.0-169.jar] /tmp/streamjob4513577705917767829.jar tm
pDir=null
16/10/03 10:26:16 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.horton
works.com:8188/ws/v1/timeline/
16/10/03 10:26:16 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com
/10.0.2.15:8050
16/10/03 10:26:17 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.horton
works.com:8188/ws/v1/timeline/
16/10/03 10:26:17 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com
/10.0.2.15:8050
16/10/03 10:26:18 INFO mapred.FileInputFormat: Total input paths to process : 1
16/10/03 10:26:18 INFO mapreduce.JobSubmitter: number of splits:2
16/10/03 10:26:18 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1475437418082_002
6
16/10/03 10:26:18 INFO impl.YarnClientImpl: Submitted application application_1475437418082_002
6
16/10/03 10:26:18 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:
8088/proxy/application_1475437418082_0026/
16/10/03 10:26:18 INFO mapreduce.Job: Running job: job_1475437418082_0026
16/10/03 10:26:26 INFO mapreduce.Job: Job job_1475437418082_0026 running in uber mode : false
16/10/03 10:26:26 INFO mapreduce.Job:  map 0% reduce 0%
16/10/03 10:26:33 INFO mapreduce.Job:  map 100% reduce 0%
16/10/03 10:26:39 INFO mapreduce.Job:  map 100% reduce 100%
16/10/03 10:26:40 INFO mapreduce.Job: Job job_1475437418082_0026 completed successfully
16/10/03 10:26:40 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=791421
                FILE: Number of bytes written=1990472
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=2038371
                HDFS: Number of bytes written=324817
                HDFS: Number of read operations=9
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=8058
```

Output from bash command line:

```
root@sandbox:~/midterm
[root@sandbox midterm]# hdfs dfs -ls /user/root/midterm_item1.out
Found 2 items
-rw-r--r--   3 root root          0 2016-10-03 10:26 /user/root/midterm_item1.out/_SUCCESS
-rw-r--r--   3 root root     324817 2016-10-03 10:26 /user/root/midterm_item1.out/part-00000
[root@sandbox midterm]# hdfs dfs -cat /user/root/midterm_item1.out/part-00000 | grep 998
998     4
998     5
998     5
[root@sandbox midterm]#
```

2. **Hive HQL Query Code:**

```
SELECT i.movie_name, d.rating
FROM midterm.item i
      JOIN midterm.data d
      ON d.movie_id = i.movie_id
WHERE d.rating > 3;
```



Query Results:

```
root@sandbox:~/midterm
Basquiat (1996) 4
Star Wars (1977)        5
Citizen Kane (1941)     5
Welcome to the Dollhouse (1995) 5
Independence Day (ID4) (1996)   5
Love and Death on Long Island (1997)    4
Stealing Beauty (1996)  5
Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1963)     5
Star Trek: Generations (1994)   4
Seven Years in Tibet (1997)     4
Rising Sun (1993)       4
Nightmare Before Christmas, The (1993)  5
Vertigo (1958)  5
Bridge on the River Kwai, The (1957)    5
Black Sheep (1996)      4
Marvin's Room (1996)    4
Willy Wonka and the Chocolate Factory (1971)    4
Fear (1996)     5
Executive Decision (1996)       4
Piano, The (1993)       4
African Queen, The (1951)       4
Pete's Dragon (1977)    4
William Shakespeare's Romeo and Juliet (1996)   4
Anastasia (1997)        4
Back to the Future (1985)       5
Time taken: 8.658 seconds, Fetched: 55375 row(s)
hive>
```

Query Execution (modified to produce sample results for submission):

```
root@sandbox:~/midterm
hive> SELECT i.movie_name, d.rating
    > FROM midterm.data d
    > JOIN midterm.item i
    > ON d.movie_id = i.movie_id
    > WHERE d.rating > 3
    > AND i.movie_name = 'Cabin Boy (1994)';
Query ID = root_20161003110154_2c014233-a159-40fb-978c-82d97af2c5f5
Total jobs = 1
Launching Job 1 out of 1


Status: Running (Executing on YARN cluster with App id application_1475437418082_0028)

--------------------------------------------------------------------------------
        VERTICES      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ..........   SUCCEEDED      1          1        0        0       0       0
Map 2 ..........   SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 1.22 s
--------------------------------------------------------------------------------
OK
Cabin Boy (1994)        5
Cabin Boy (1994)        5
Cabin Boy (1994)        4
Time taken: 2.538 seconds, Fetched: 3 row(s)
hive>
```

Mapper Python Code:

```
#!/usr/bin/env python

import sys

for line in sys.stdin:
      line = line.strip()
      row = line.split('\t')

      # if movie rating row ...
      if row[0] == "A" and len(row) == 5:
            try:
                  # get rating value
                  rating = int(row[3])
            except ValueError:
                  continue

            # Only output movie ratings with rating > 3
            if rating > 3:
                  try:
                        # create composite key based on movie id
plus "_B"
                        # to ensure sort puts ratings AFTER movie
names

                        # print movie_id, rating
                        print "%s_B\t%s" % (row[2], row[3])
                  except:
                        continue

      # if movie name row...
      if row[0] == "B" and len(row) == 4:
            try:
                  # create composite key based on movie id plus
"_A"
                  # to ensure sort puts names BEFORE ratings

                  # print movie_id, movie_id, rating
                  print "%s_A\t%s" % (row[1], row[2])
            except:
                  continue
```

Reducer Python Code:

```
#!/usr/bin/env python

import sys
```

```
current_movie_id = -1
movie_name = ""
for line in sys.stdin:
      line = line.strip()
      row = line.split('\t')
      if len(row) == 2:
            try:
                  # split composite key to recover movie id and row
type
                  info = row[0].split('_')
                  # movie_id is the first part of the composite key
                  movie_id = int(info[0])

            except ValueError:
                  continue

            #if row type is movie name
            if info[1] == 'A':
                  try:
                        #gets movie name
                        movie_name = row[1]
                  except ValueError:
                        continue

            # if row type is movie rating...
            if info[1] == 'B':
                  try:
                        # get rating
                        rating = int(row[1])
                        # print current movie name and rating
                        print "%s\t%i" % (movie_name, rating)
                  except ValueError:
                        continue
```

Execution in Hadoop:

```
hadoop jar /usr/hdp/2.4.0.0-169/hadoop-mapreduce/hadoop-
streaming-2.7.1.2.4.0.0-169.jar -file ./midterm_item2_mapper.py -
mapper midterm_item2_mapper.py -file ./midterm_item2_reducer.py -
reducer midterm_item2_reducer.py -input /user/root/u.join -output
/user/root/midterm_item2.out
```

```
root@sandbox:~/midterm
[root@sandbox midterm]# hadoop jar /usr/hdp/2.4.0.0-169/hadoop-mapreduce/hadoop-streaming-2.7.1.2.4
.0.0-169.jar -file ./midterm_item2_mapper.py -mapper midterm_item2_mapper.py -file ./midterm_item2_
reducer.py -reducer midterm_item2_reducer.py -input /user/root/u.join -output /user/root/midterm_it
em2.out
WARNING: Use "yarn jar" to launch YARN applications.
16/10/03 12:16:14 WARN streaming.StreamJob: -file option is deprecated, please use generic option -
files instead.
packageJobJar: [./midterm_item2_mapper.py, ./midterm_item2_reducer.py] [/usr/hdp/2.4.0.0-169/hadoop
-mapreduce/hadoop-streaming-2.7.1.2.4.0.0-169.jar] /tmp/streamjob6695191622309349387.jar tmpDir=nul
l
16/10/03 12:16:16 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonwork
s.com:8188/ws/v1/timeline/
16/10/03 12:16:16 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/10.
0.2.15:8050
16/10/03 12:16:17 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonwork
s.com:8188/ws/v1/timeline/
16/10/03 12:16:17 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/10.
0.2.15:8050
16/10/03 12:16:17 INFO mapred.FileInputFormat: Total input paths to process : 1
16/10/03 12:16:17 INFO mapreduce.JobSubmitter: number of splits:2
16/10/03 12:16:18 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1475437418082_0031
16/10/03 12:16:18 INFO impl.YarnClientImpl: Submitted application application_1475437418082_0031
16/10/03 12:16:18 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088
/proxy/application_1475437418082_0031/
16/10/03 12:16:18 INFO mapreduce.Job: Running job: job_1475437418082_0031
16/10/03 12:16:26 INFO mapreduce.Job: Job job_1475437418082_0031 running in uber mode : false
16/10/03 12:16:26 INFO mapreduce.Job:  map 0% reduce 0%
16/10/03 12:16:33 INFO mapreduce.Job:  map 100% reduce 0%
16/10/03 12:16:40 INFO mapreduce.Job:  map 100% reduce 100%
16/10/03 12:16:40 INFO mapreduce.Job: Job job_1475437418082_0031 completed successfully
16/10/03 12:16:41 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=822845
                FILE: Number of bytes written=2053320
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=2382433
                HDFS: Number of bytes written=743514
                HDFS: Number of read operations=9
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=8973
                Total time spent by all reduces in occupied slots (ms)=4604
                Total time spent by all map tasks (ms)=8973
                Total time spent by all reduce tasks (ms)=4604
```

Output from bash shell:

```
root@sandbox ~/midterm
[root@sandbox midterm]# hdfs dfs -ls /user/root/midterm_item2.out
Found 2 items
-rw-r--r--   3 root root          0 2016-10-03 15:02 /user/root/midterm_item2.out/_SUCCESS
-rw-r--r--   3 root root    1434752 2016-10-03 15:02 /user/root/midterm_item2.out/part-00000
[root@sandbox midterm]# hdfs dfs -cat /user/root/midterm_item2.out/part-00000 | grep Cabin
Cabin Boy (1994)        4
Cabin Boy (1994)        5
Cabin Boy (1994)        5
[root@sandbox midterm]#
```