**WordCount**

**Implementation—**

* Part1.java contains the source code of Map and Reduce Class.
* StopWord.java has the list of stop words to be removed during map phase.
* MiscUtils.java has the comparator to sort the HashMap by values.
* FileCopy.java is to copy file containing the Top20 words to HDFS.

**How to run –**

Hadoop jar WordCount.jar <PackageName.ClassName> <InputDirectory> <OutputDirectory>

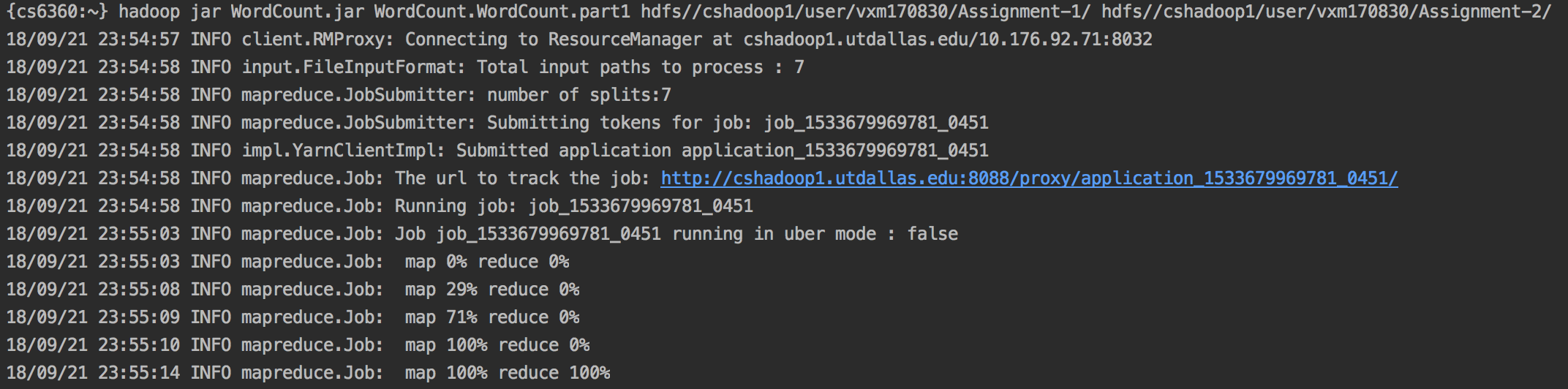
**Example --**

hadoop jar WordCount.jar WordProblem.part1 hdfs//cshadoop1/user/vxm170830/Assignment-1/ hdfs//cshadoop1/user/vxm170830/part1/

**Deliverables –**

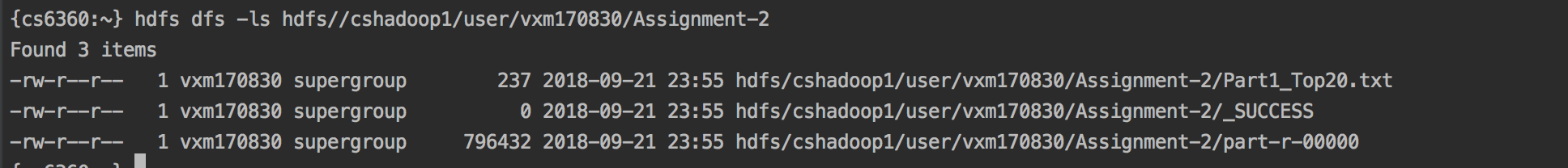
* WordCount.java
* StopWord.java
* FileCopy.java
* MiscUtils.java

**Screenshots—**

****

**Output—**

* Part-r-00000 contains the total count for each key.
* Part1\_Top20.txt contains the top 20 most frequent used words.

****

**WordCount-TopN**

**Implementation --**

* WordCount-TopN.java contains the source code of Map and Reduce Class.
* MiscUtils.java has the comparator to sort the HashMap by values.
* FileCopy.java is to copy file containing the Top20 words to HDFS.

**How to run –**

Hadoop jar WordCount-TopN.jar <PackageName.ClassName> <InputDirectory> <OutputDirectory>

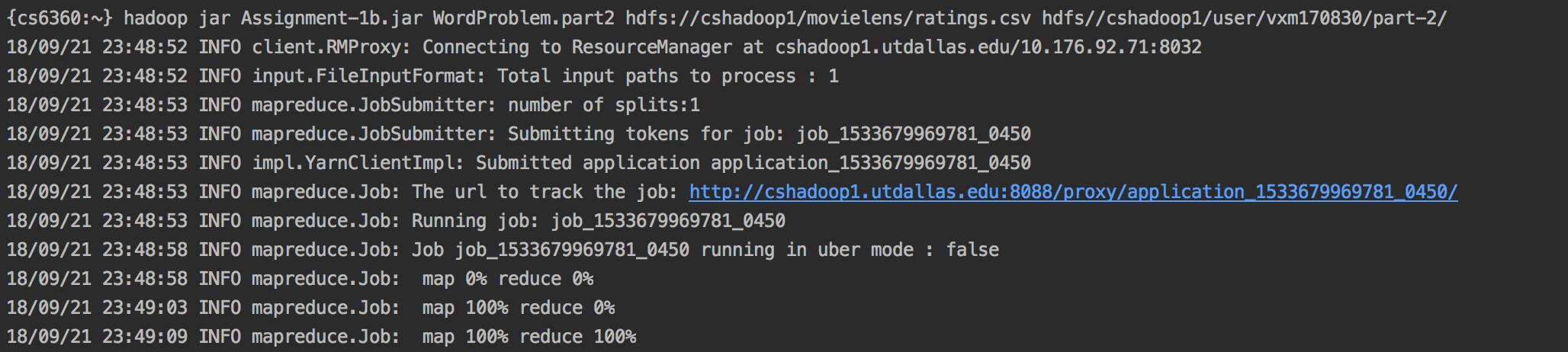
**Example --**

hadoop jar WordCount-TopN.jar WordProblem.part2 hdfs://cshadoop1/movielens/ratings.csv hdfs//cshadoop1/user/vxm170830/part-2/

**Deliverables –**

* WordCount-TopN.java
* FileCopy.java
* MiscUtiils.java

**Screenshots—**

****

**Output—**

* Part-r-00000 contains the total count for each key.
* Part2\_Top20.txt contains the top 20 movies with highest average rating.

