The Course Project

Part 1:

The first part is to develop a Mapper and Reducer application to calculate the *range* (the difference between max and min values) of *sky ceiling height* (meters) for *each observation month* from NCDC records (note: 99999 indicates missing value, and [01459] indicate good quality value).

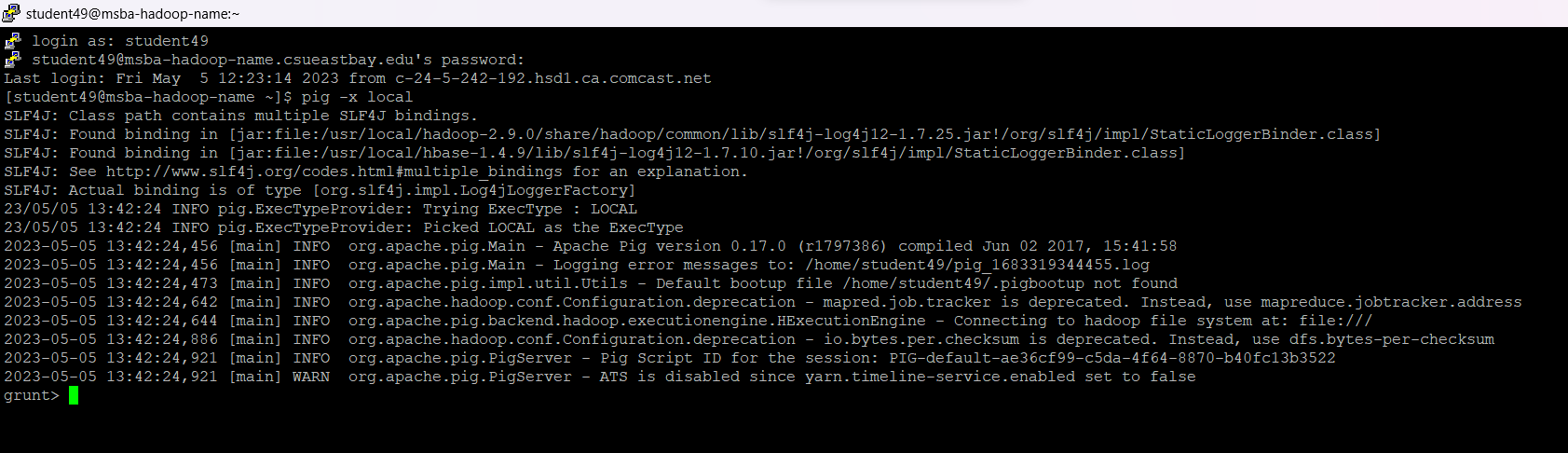
Part 2:

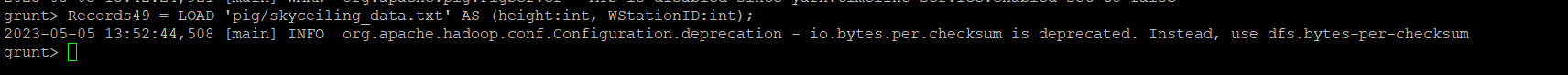
The second part is to develop a python application that can be implemented in PySpark to calculate the *average visibility distance* (meters) for *each USAF weather station ID* from NCDC records (note: 999999 indicates missing value, and [01459] indicate good quality value).

Part 3:

The third part is to load the text file into Pig and get the range of sky ceiling height for each USAF weather station ID.

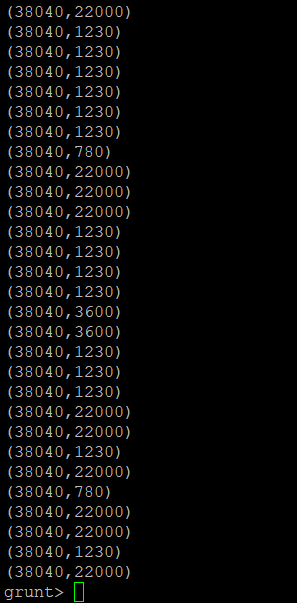
pig –x local



Records49 = LOAD 'pig/skyceiling\_data.txt' AS (height:int, WStationID:int); 

dump Records49;

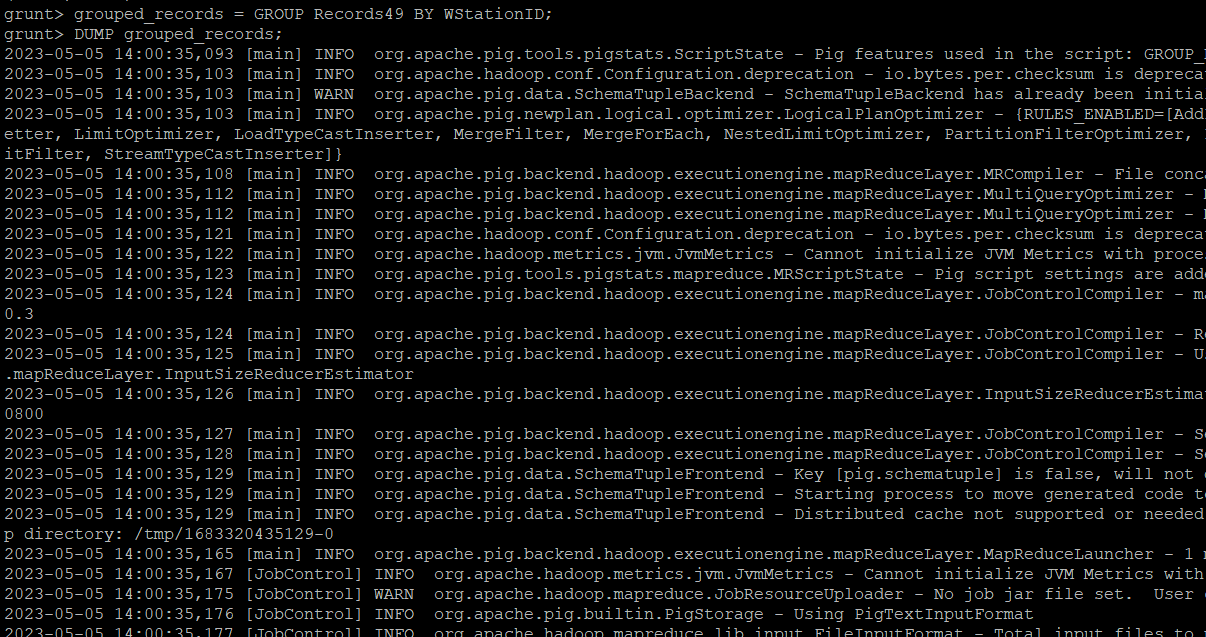


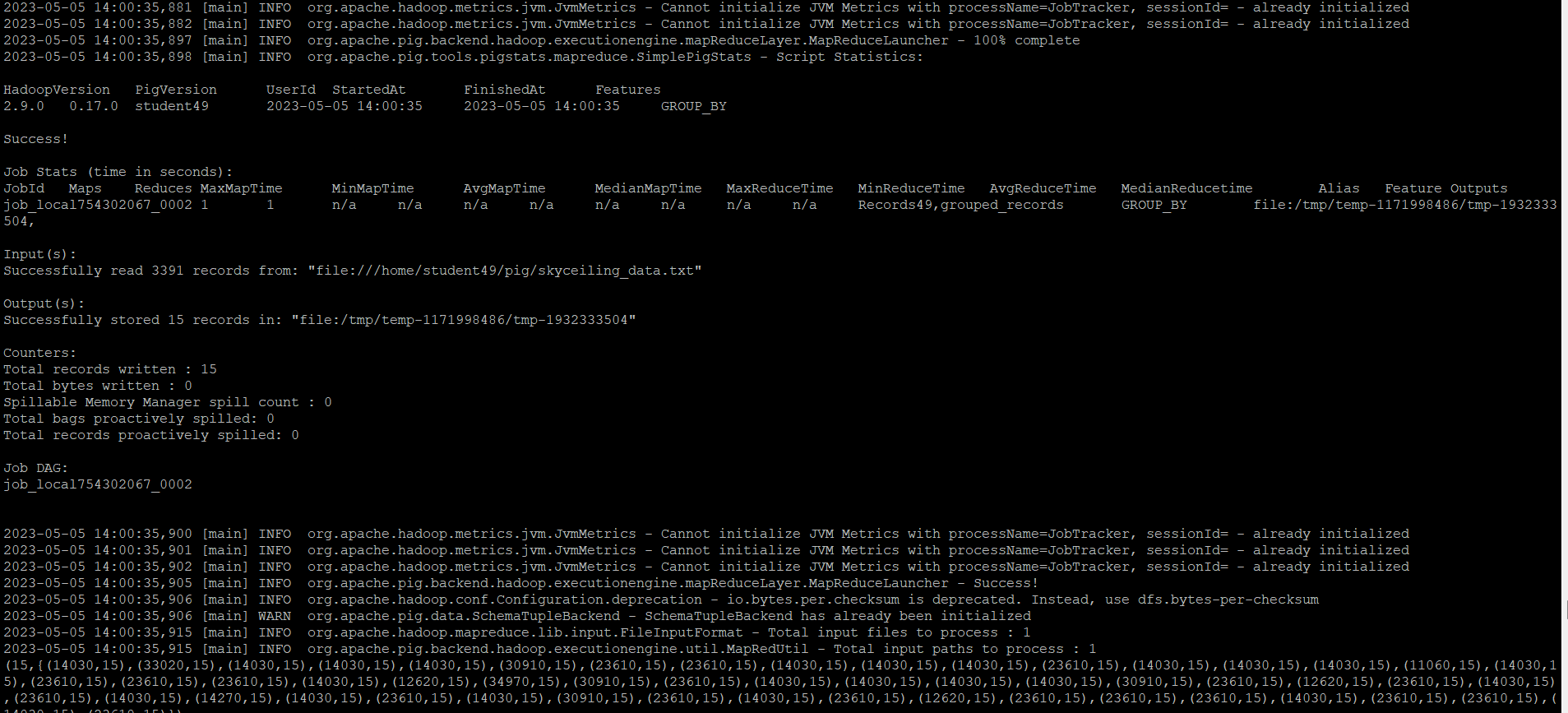


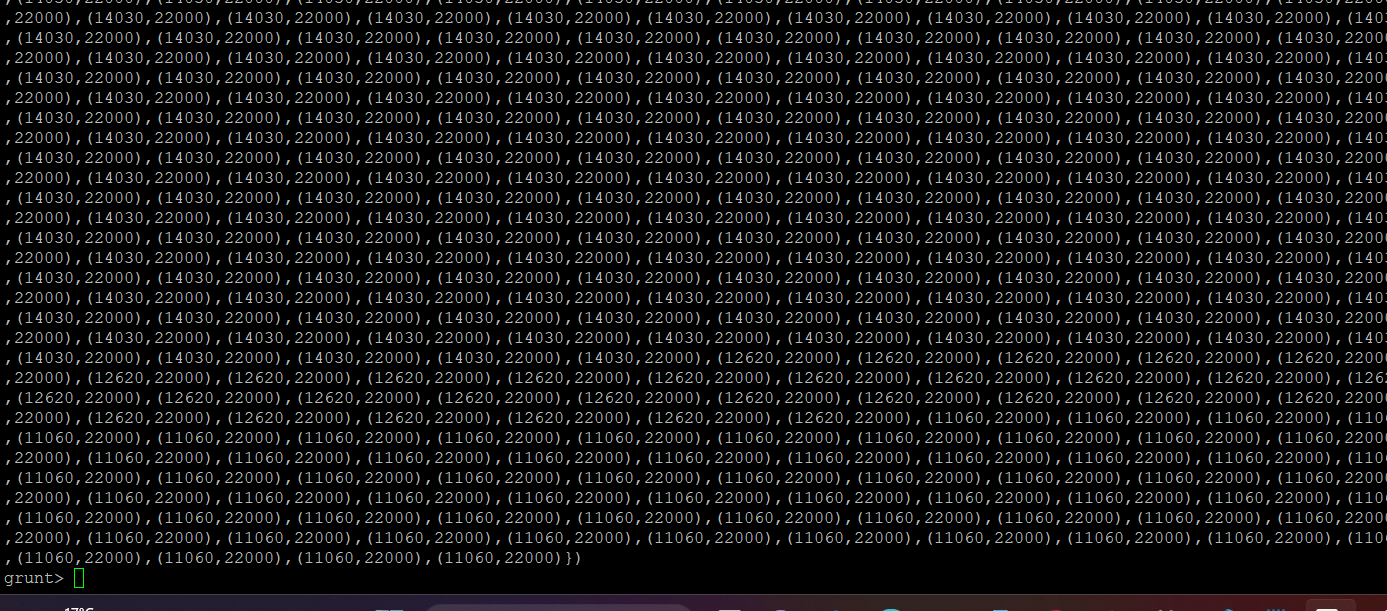
grouped\_records = GROUP Records49 BY WStationID;



DUMP grouped\_records;

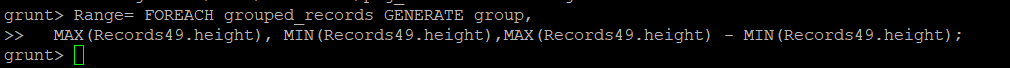




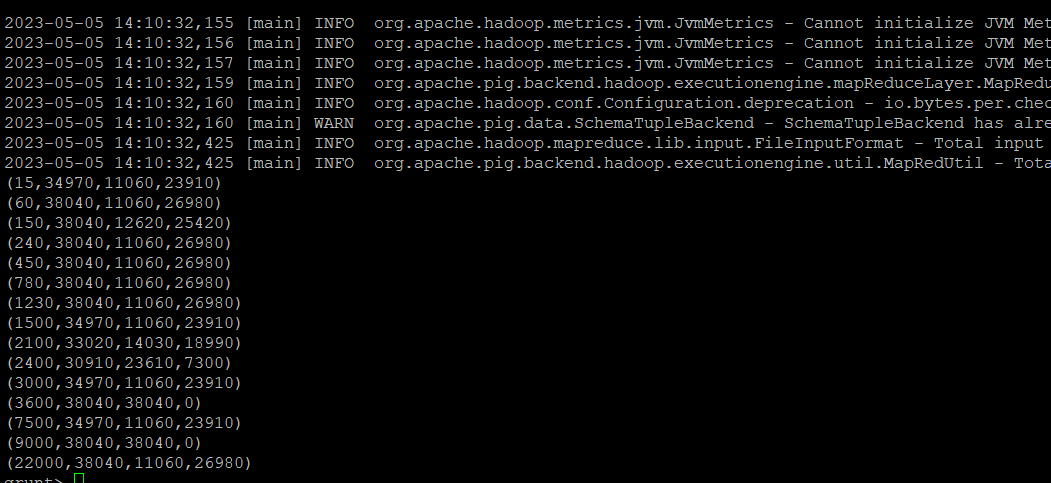


I’ve calculated the range along with minimum and maximum values in the below execution

Range= FOREACH grouped\_records GENERATE group,

MAX(Records49.height), MIN(Records49.height),MAX(Records49.height) - MIN(Records49.height); 

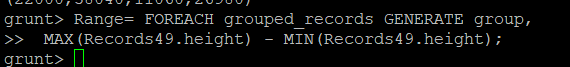
DUMP Range;



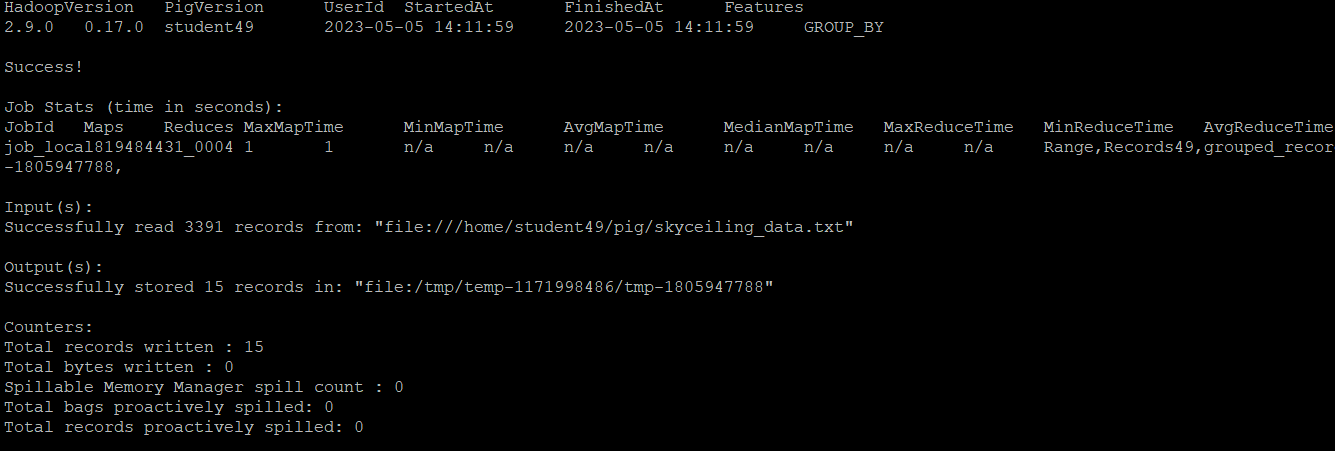
this is only for the range calculation.

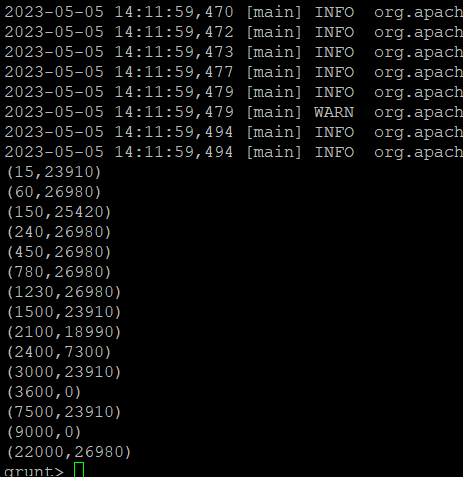
Range= FOREACH grouped\_records GENERATE group,

MAX(Records49.height) - MIN(Records49.height);



DUMP Range;

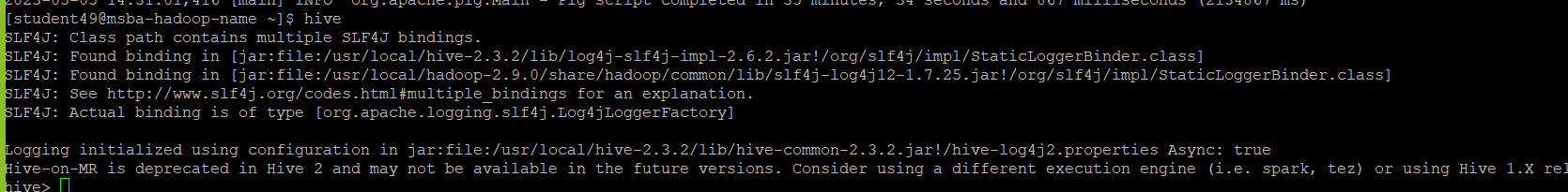




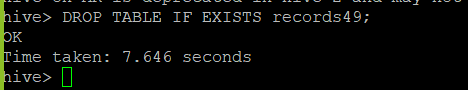
Part 4:

The fourth part is to load the text file into Hive and get the average sky ceiling height for each USAF weather station ID.

hive

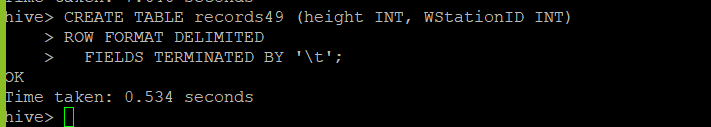


DROP TABLE IF EXISTS records49;

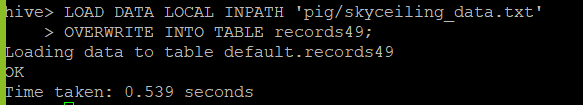


CREATE TABLE records49 (height INT, WStationID INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'; 

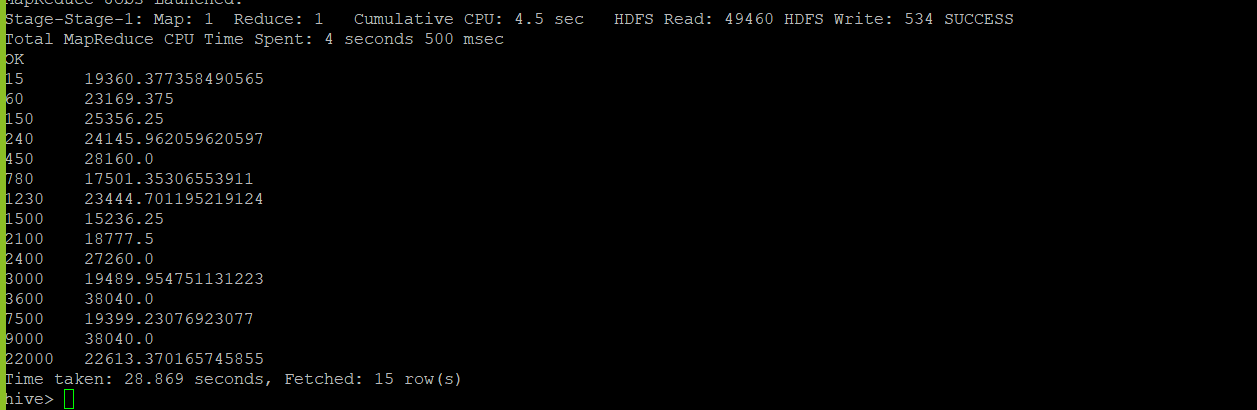
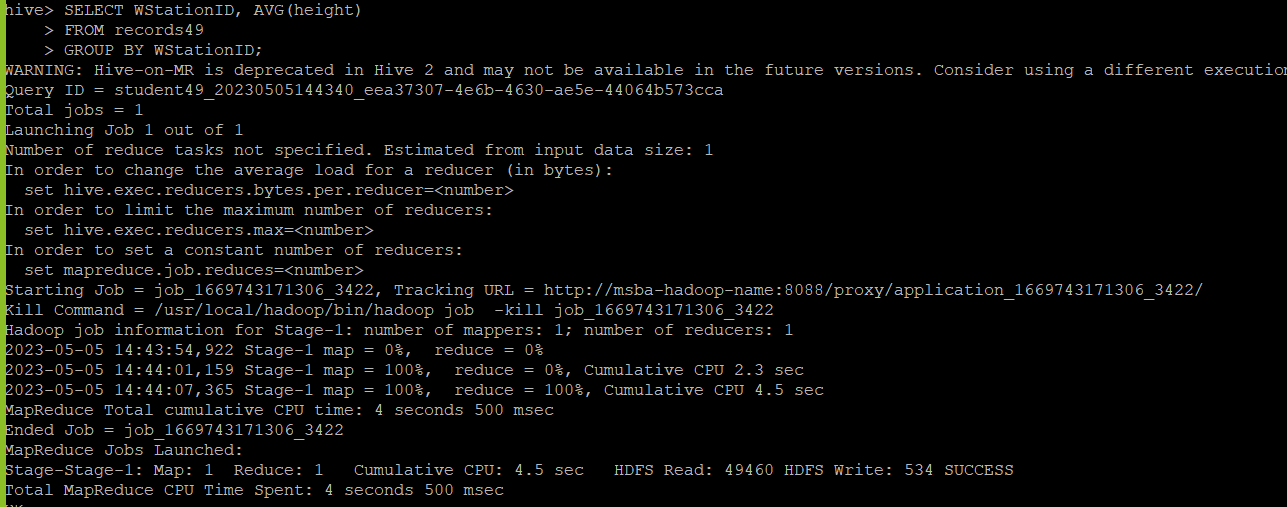
LOAD DATA LOCAL INPATH 'pig/skyceiling\_data.txt'

OVERWRITE INTO TABLE records49; 

SELECT WStationID, AVG(height)

FROM records49

GROUP BY WStationID;

You need to turn in:

**1)** Part 1:

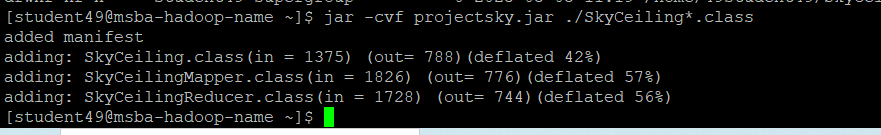
a. *if you are using JAVA to develop the Mapper and Reducer applications:* the three java files (mapper, reducer and main);

1. Creating the mapper, reducer and job(main) file
2. Compiling the files to create .class files

javac -classpath /home/student49/hadoop-common-2.6.1.jar:/home/student49/hadoop-mapreduce-client-core-2.6.1.jar:/home/student49/commons-cli-2.0.jar -d . SkyCeiling.java SkyCeilingMapper.java SkyCeilingReducer.java

1. And also creating the .jar file using the main class file

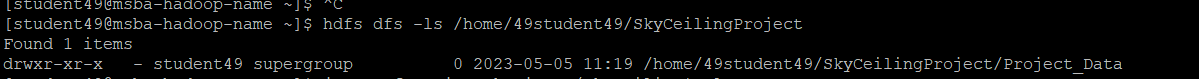
Jar -cvf SkyCeiling.jar ./SkyCeiling\*.class



1. Create input directory:

hdfs dfs -mkdir /home/49student49/SkyCeilingProject

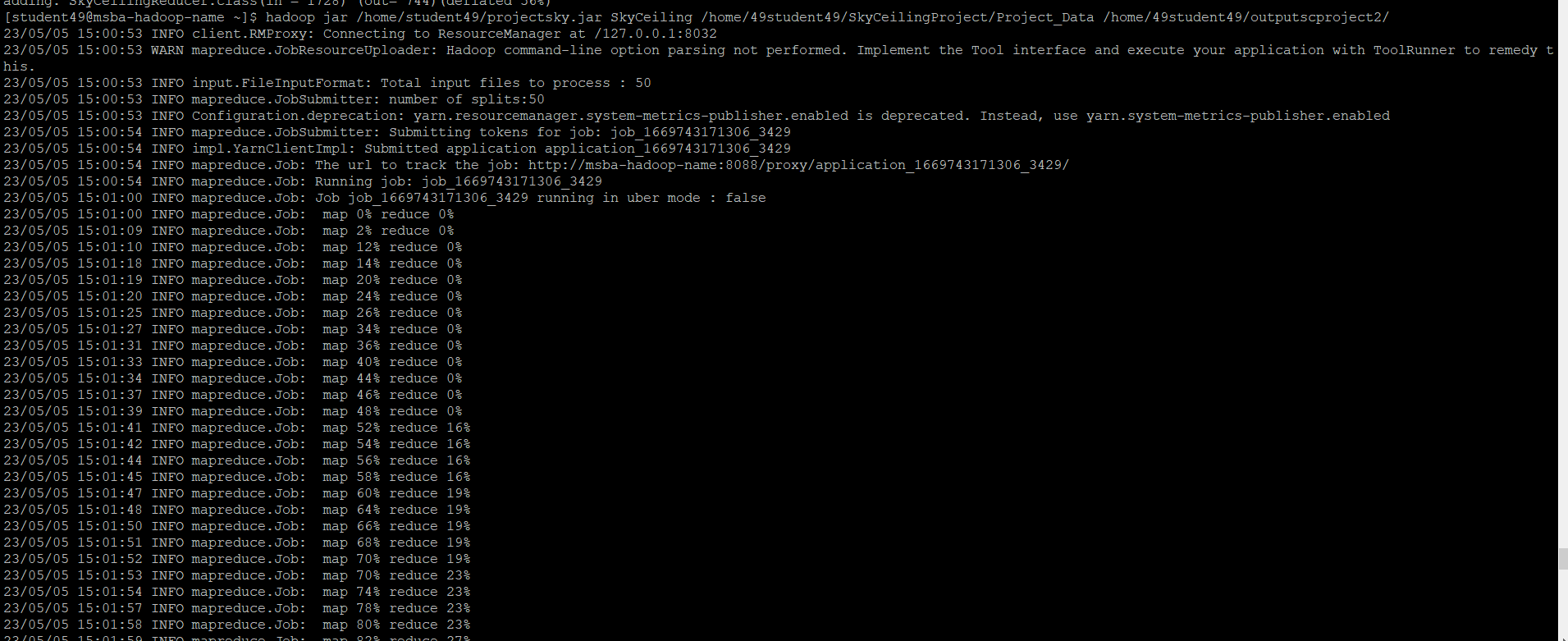
hdfs dfs -ls /home/49student49/SkyCeilingProject

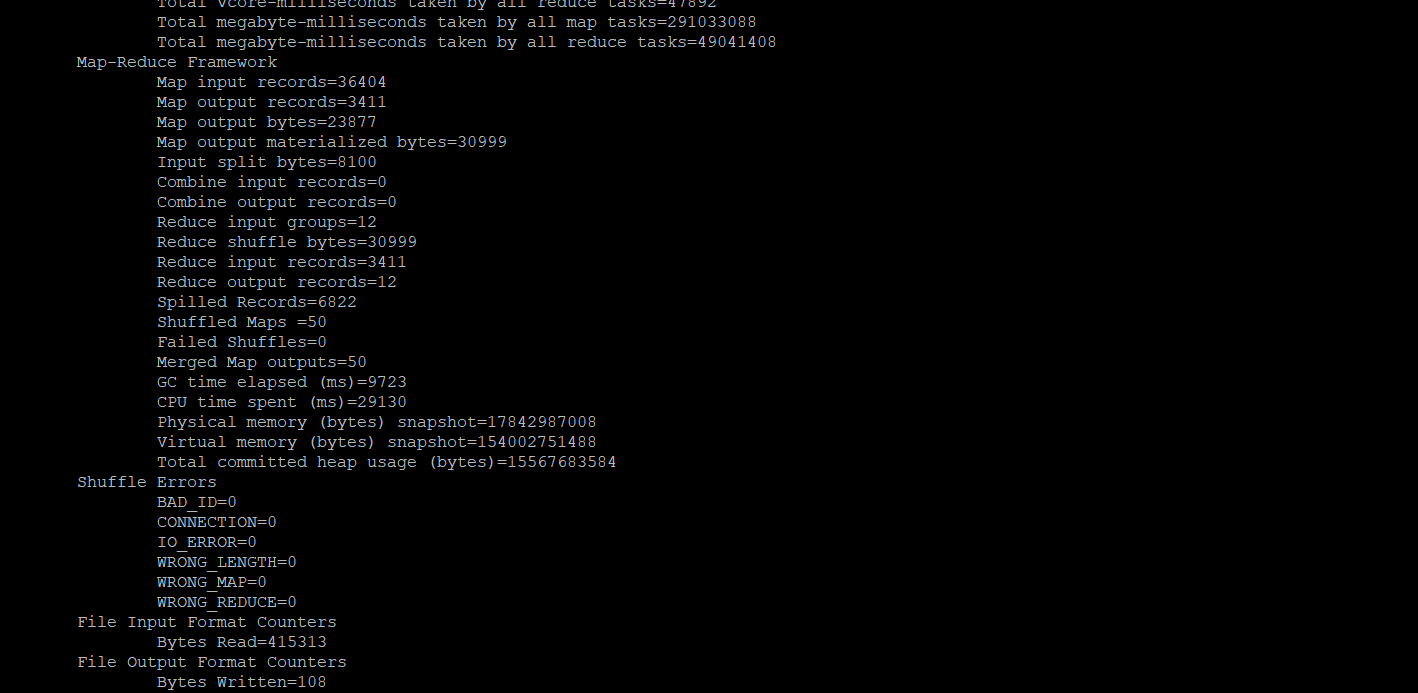


1. Copying from local to HDFS:

hdfs dfs -copyFromLocal /home/student49/Project\_Data /home/49student49/SkyCeilingProject/Project\_Data

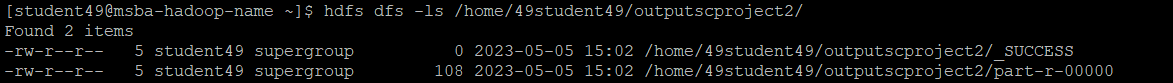
1. running the jar file:

hadoop jar /home/student49/projectsky.jar SkyCeiling /home/49student49/SkyCeilingProject/Project\_Data /home/49student49/outputscproject2/

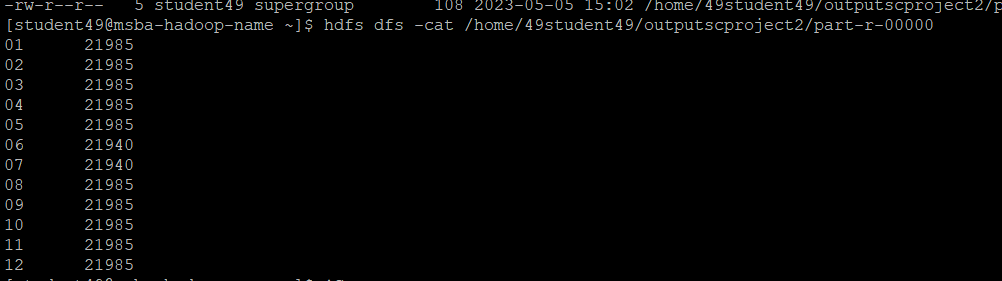


Output:

hdfs dfs -ls /home/49student49/outputscproject2/



hdfs dfs -cat /home/49student49/outputscproject2/part-r-00000



b. *if you are using Hadoop streaming jar and developing two python programs (mapper python file and reducer python file)*: the two python files (mapper and reducer);

c. *if you are using mrjob library and developing one python program with two functions:* the python file (with the mapper and reducer functions);

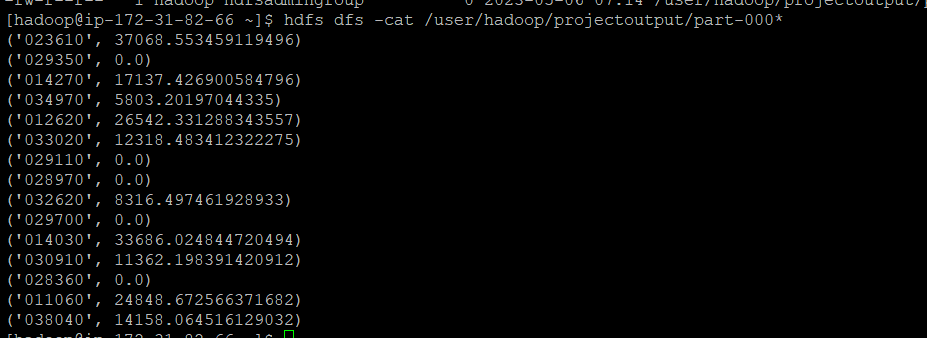
2) Part 2: the python program you developed;

hdfs dfs -mkdir /user/hadoop/input/

hdfs dfs -copyFromLocal Project\_Data /user/hadoop/input/

spark-submit --master local AverageDisplay.py

hdfs dfs -cat /user/hadoop/projectoutput/part-000\*



**3)** the commands from converting java files into a Jar file to running the Jar file in Hadoop, or the commands to execute the python files in Hadoop and in Spark;

**4)** the step by step commands and screenshots of solutions from all the parts;

The original dataset for this project is available on Blackboard.