**Documentation of Sort - Map reduce**

-Prepared by Vignesh.R

Since we use twelve reducers by design each reducer index is simply (month -1) where month takes values from 1 through 12. This is the key component that ensures that each month goes to a separate Reducer in the proper order. That is, records for January go to Reducer 0, records for February go to Reducer 1, and so on until records for December go to Reducer 11.

Multiple Reducers are used only when you run the job in psuedo-cluster or full cluster mode. Running this job locally in your single node cluster causes only a single Reducer to be used, which results in a single output file with its data sorted by month. In the local mode the Partitioner is bypassed and all the Mapper output records go to the single reducer.

**Creating .jar file for Sort and exporting it:-**

**Step 1:**

Open Eclipse and Click on File > New > Java Project.

**Step 2:**

Give the name ‘Sort’ as project name and click ‘Finish’.

**Step 3:**

Right click on ‘Sort’ project and select ‘Properties’. Click ‘Java Build Path’ and switch to Libraries tab and click on ‘Add external JARs’.

**Step 4:**

Select all the JAR files in usr >> lib >> hadoop directory to add them.

**Step 5:**

Again add all jar files in usr >> lib >> hadoop >> client directory and press OK.

**Step 6:**

**Right click on src, New >>Class.**

**Step 7:**

Enter the project name as ‘SortAscMonthDescWeekMRJob’ and click ‘Finish’.

**Step 8:**

Open browser and copy and paste the Java Source code of Sort program from the link given. The packages are automatically generated by the Eclipse.

**Website link:** <https://github.com/Apress/pro-apache-hadoop/blob/master/prohadoop/src/main/java/org/apress/prohadoop/c6/SortAscMonthDescWeekMRJob.java>

**Website link:**<https://github.com/Apress/pro-apache-hadoop/blob/master/prohadoop/src/main/java/org/apress/prohadoop/utils/AirlineDataUtils.java>

**Website link:**<https://github.com/Apress/pro-apache-hadoop/blob/master/prohadoop/src/main/java/org/apress/prohadoop/c6/MonthDoWOnlyWritable.java>

**Website link:**<https://github.com/Apress/pro-apache-hadoop/blob/master/prohadoop/src/main/java/org/apress/prohadoop/c6/MonthDoWWritable.java>

**Website link:**<https://github.com/Apress/pro-apache-hadoop/blob/master/prohadoop/src/main/java/org/apress/prohadoop/c6/DelaysWritable.java>

**Step 9:**

Right click on the **SortAscMonthDescWeekMRJob** project and select Export >> Java >> JAR file. Then click on ‘Next’.

**Step 10:**

Name the JAR file and click ‘Finish’.

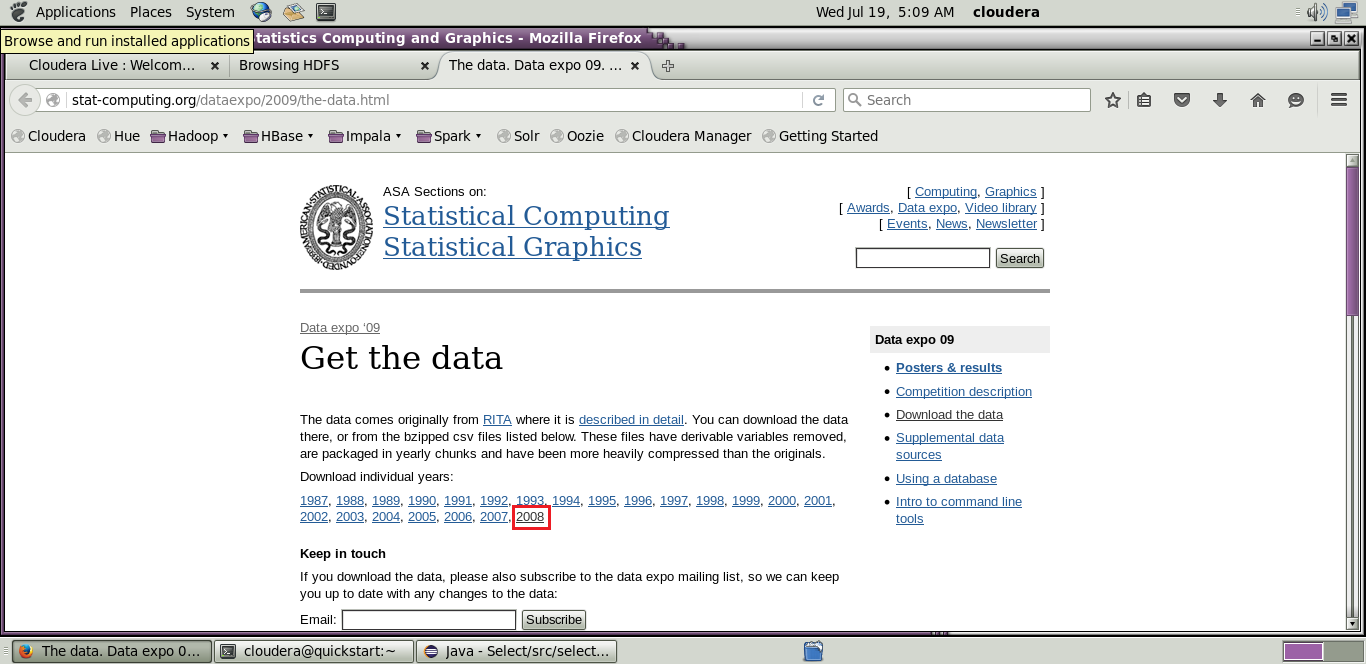
**Creating a file for Mapreduce job to work on:**

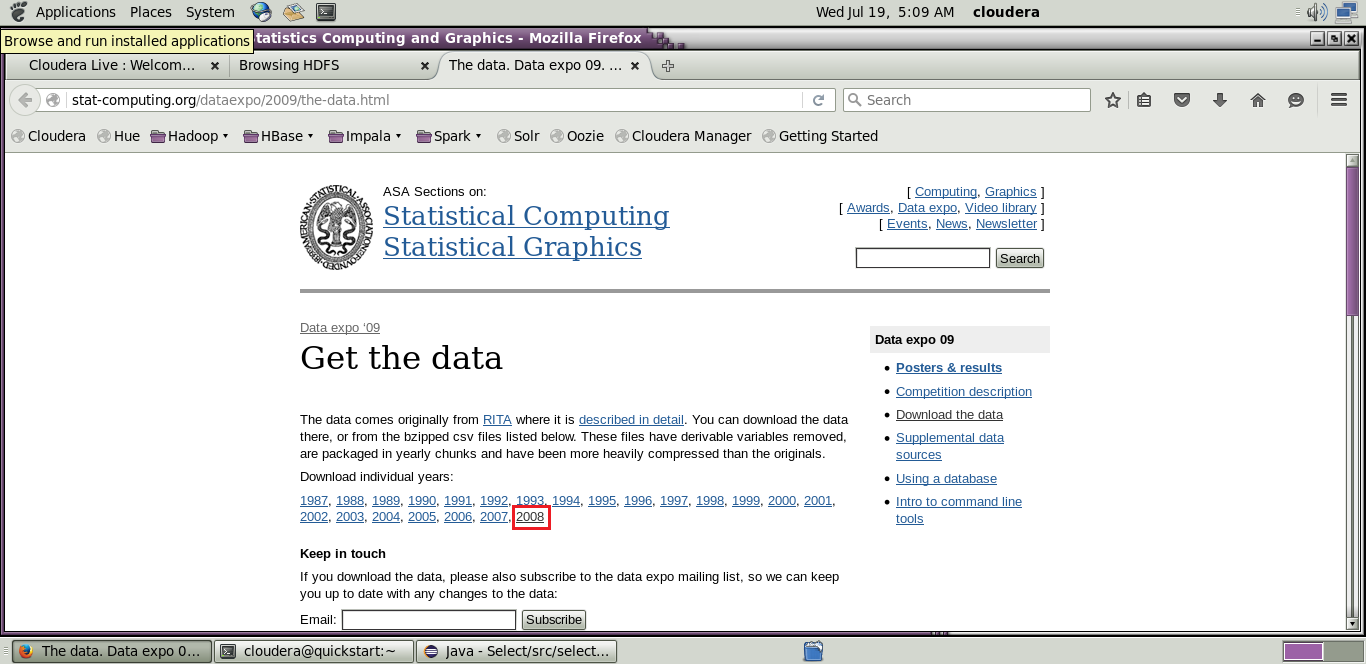
**Step 11:**Make a new Directory using the following command.

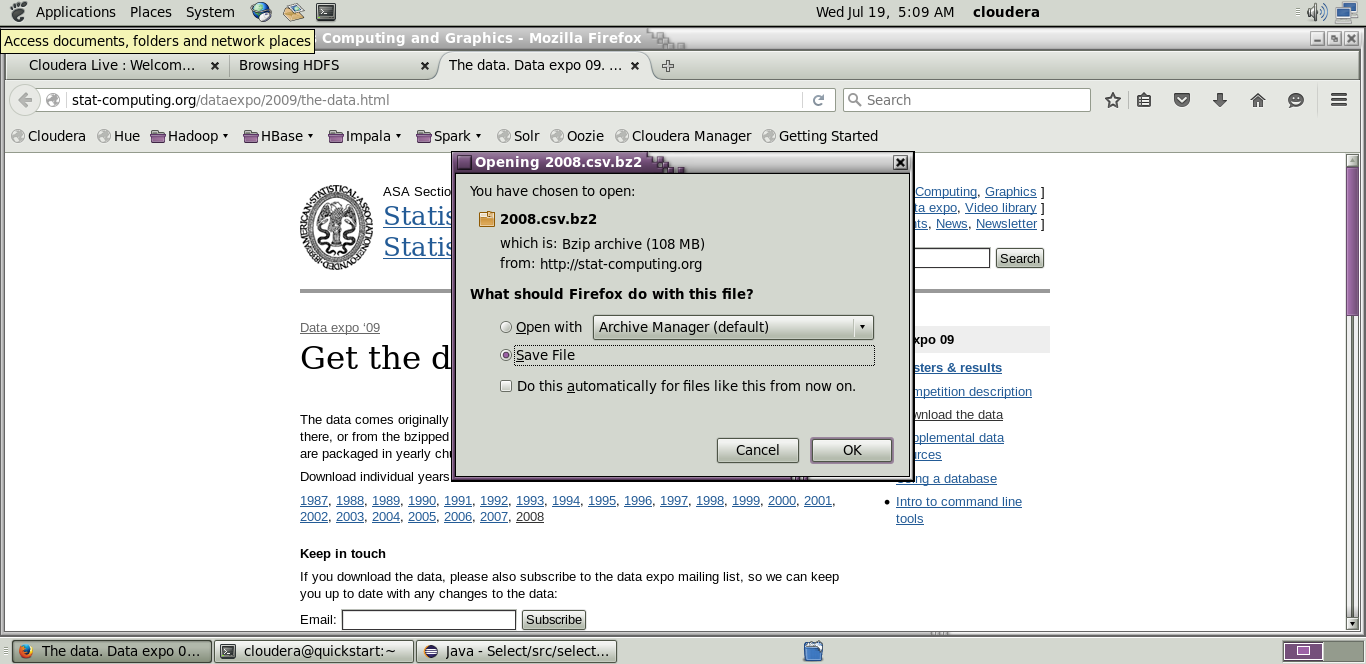
**Code:hadoop fs -mkdir /airline**

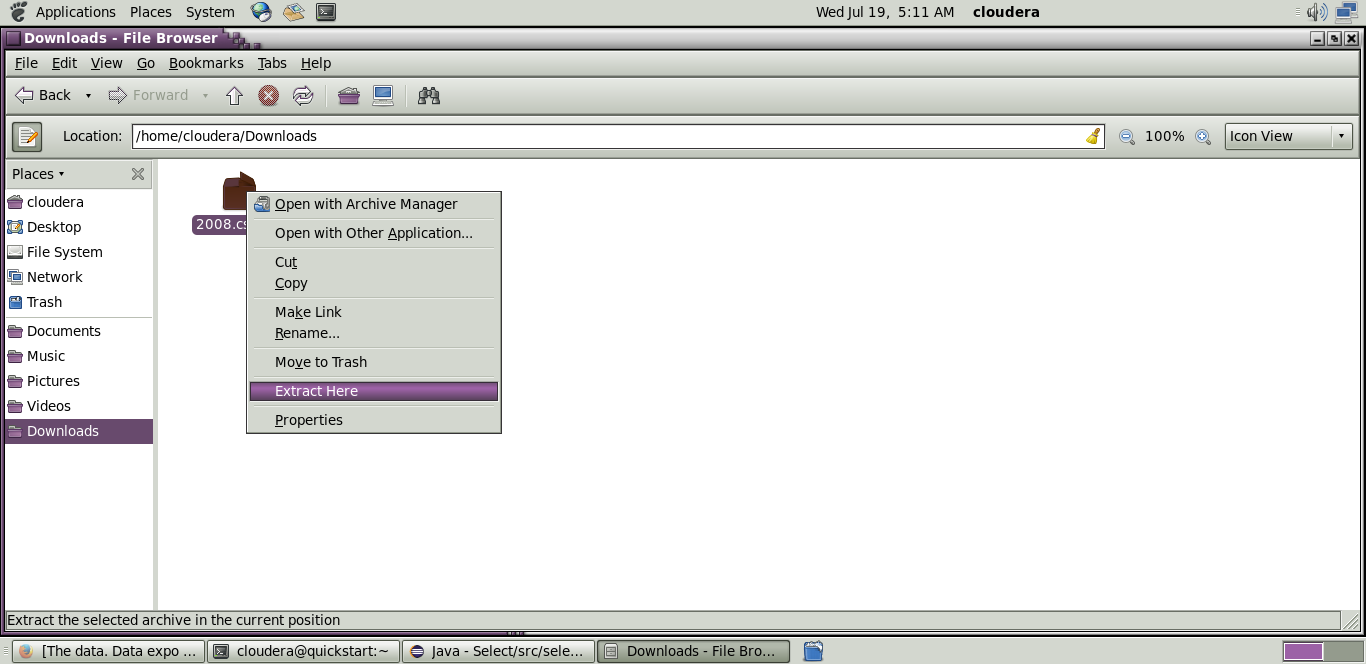
**Step 12:**Use the following link to download the csv file.

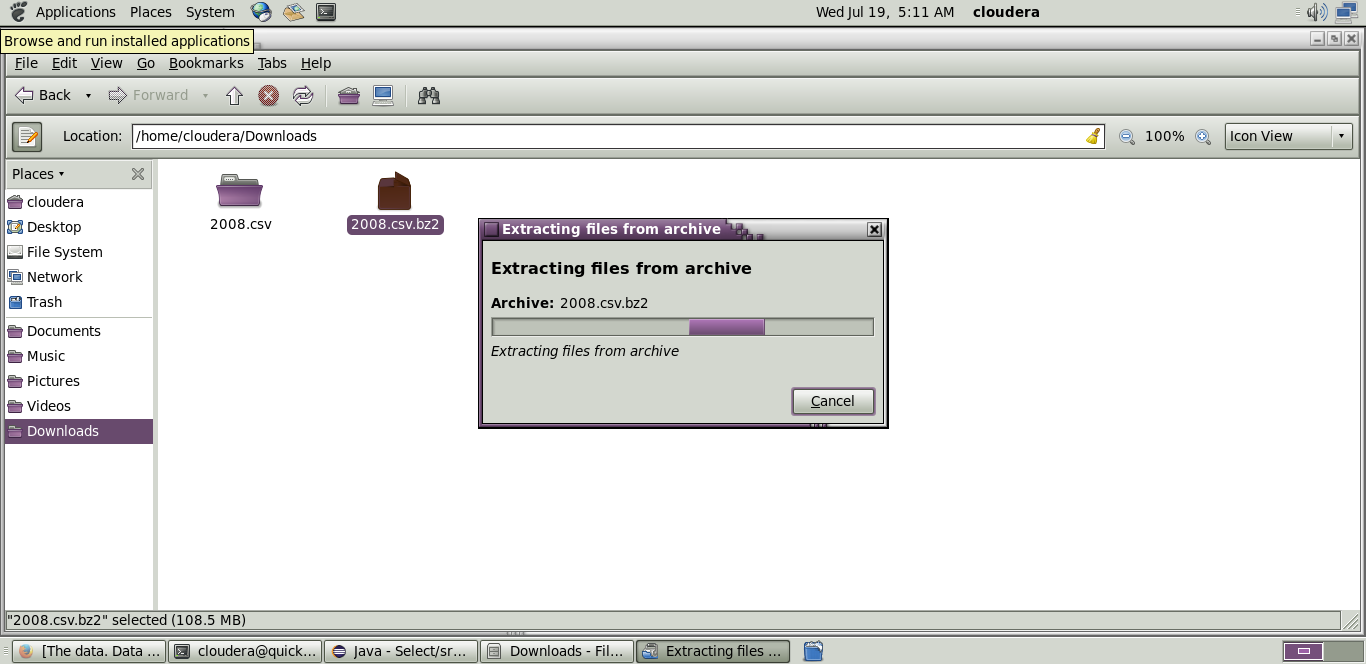
**Website link:**

[http://stat-computing.org/dataexpo/2009/the-data.html](http://stat-computing.org/dataexpo/2009/the-data.html )

[](http://stat-computing.org/dataexpo/2009/the-data.html )







**Step 13:** Copy the downloaded file to the new directory created in HDFS.

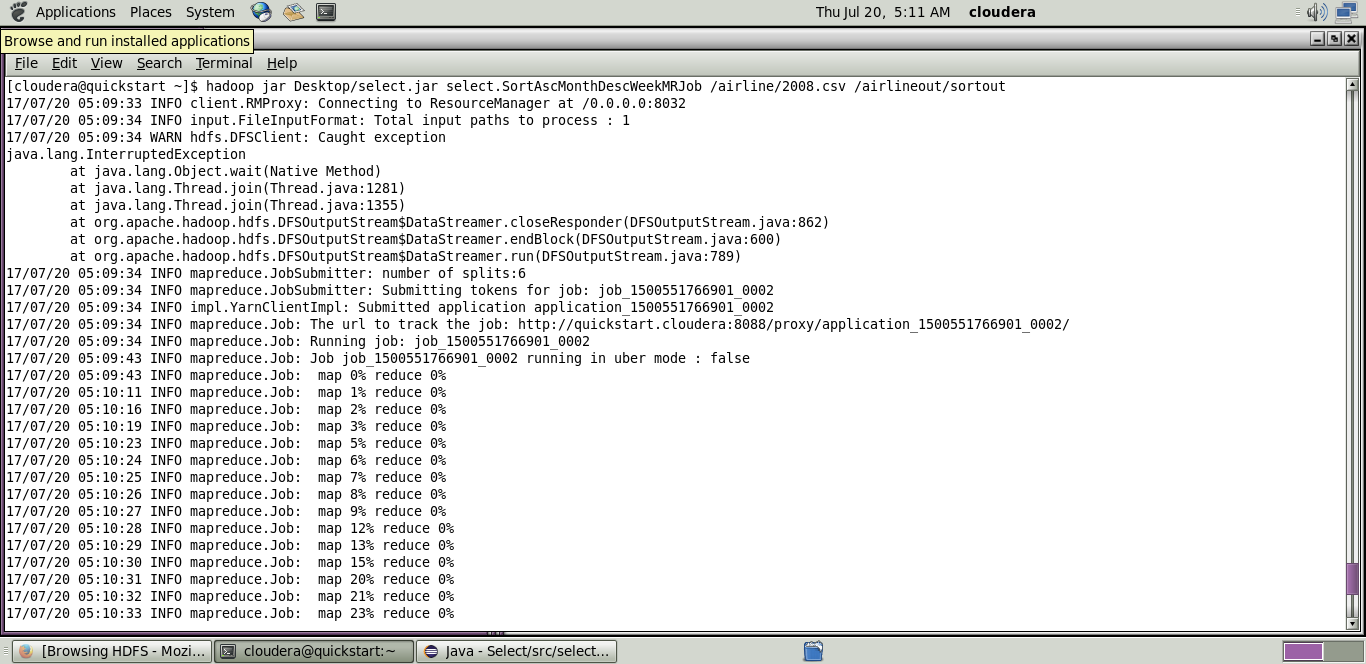
**Code:hadoop fs -put /home/cloudera/Downloads/2008.csv/airline**

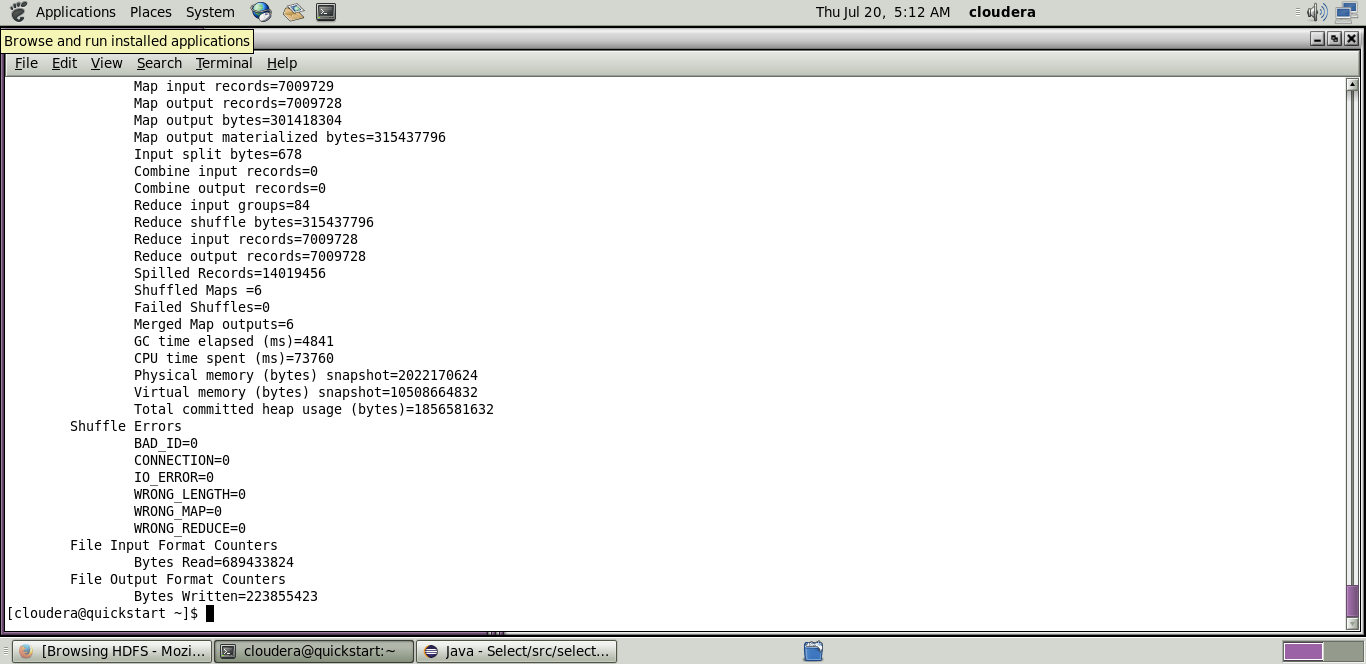
**Initializing mapreduce job:**

**Step 14:**

Initialize the mapreduce job by giving the following command and wait for sometime.

**Code:** hadoop jar Desktop/select.jar select.SortAscMonthDescWeekMRJob /airline/2008.csv /airlineout/sortout



Now wait for about 50-70 seconds while the mapreduce job is being performed for the data created earlier.

**Output mapreduce job:**

**Step 15:**

The output directory of the mapreduce program is listed using the following command.

**Code:** hadoop fs -ls/airlineout/sortout/\*

**Step 16:**

The final output of the mapreduce program is found using the following command.

**Code:**hadoop fs -cat /airlineout/sortout/\*