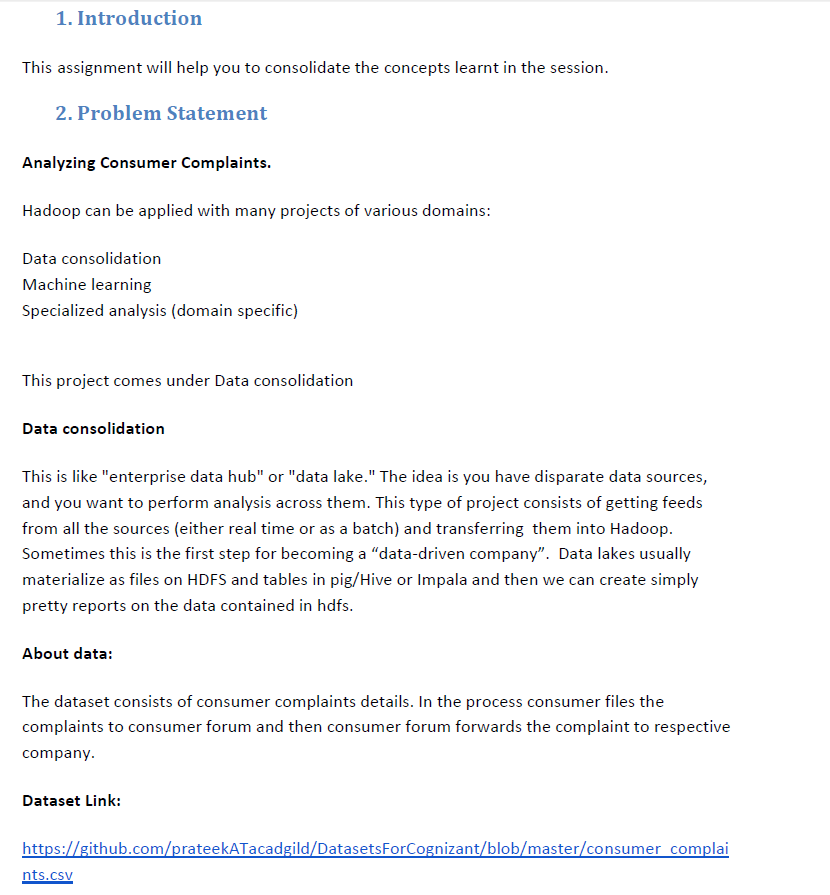
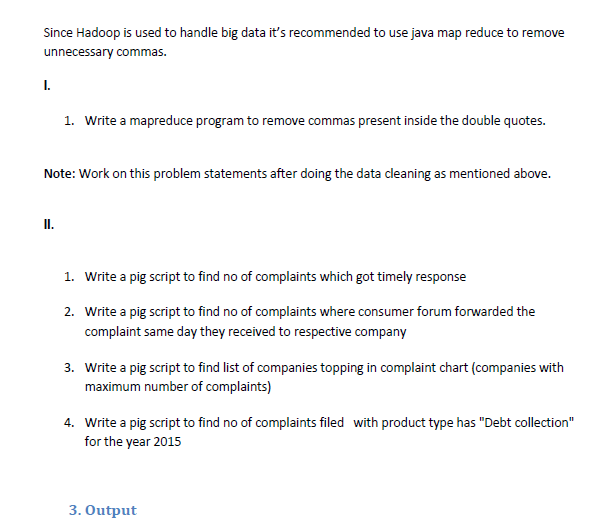
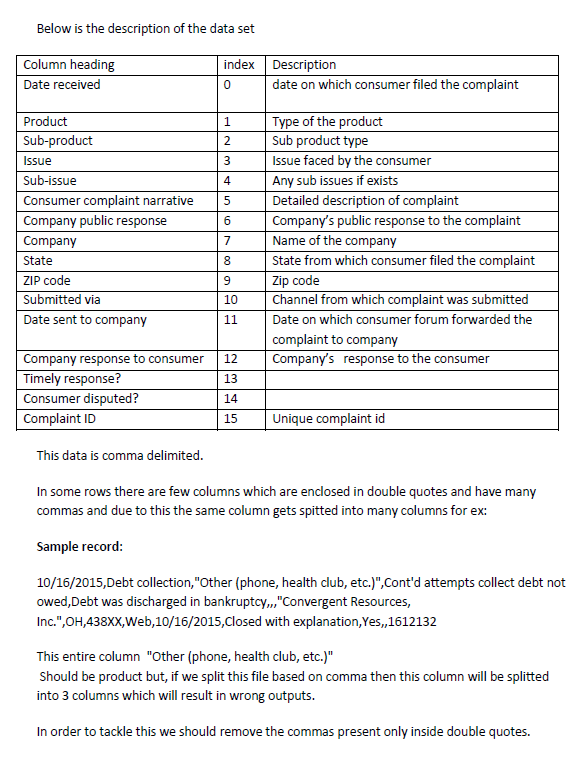
Project 1.2

QUESTION

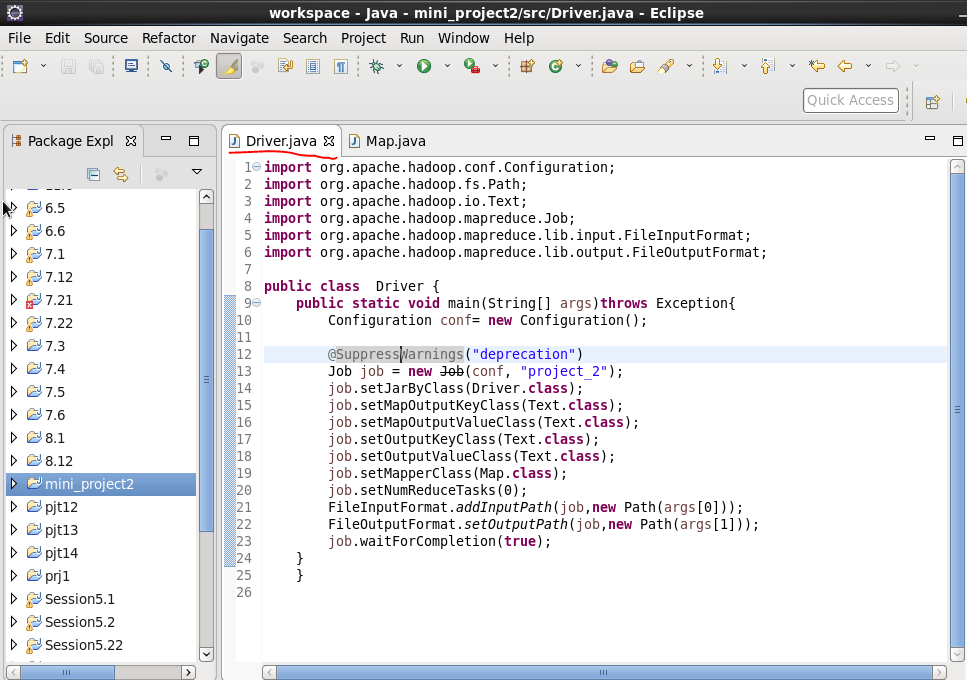
**Consumer complaint Analysis**





1. **MAP REDUCE TO REMOVE ALL COMMAS WITHIN DOUBLE QUOTES**
2. **DRIVER CLASS**

We need to Import the necessary jar file then we will declare the Configuration object for the job. Then we will Declaring the Job for the first task and will set jar by class. Then we will set map Output Key value pairs , reducer Output key value pairs and Mapper class .As it is a map only job so set number of reduce tasks as “0”. Then we will set the input path from which the data will be processed and also the output path and will set job wait for completion as true as this will wait for our job completion.

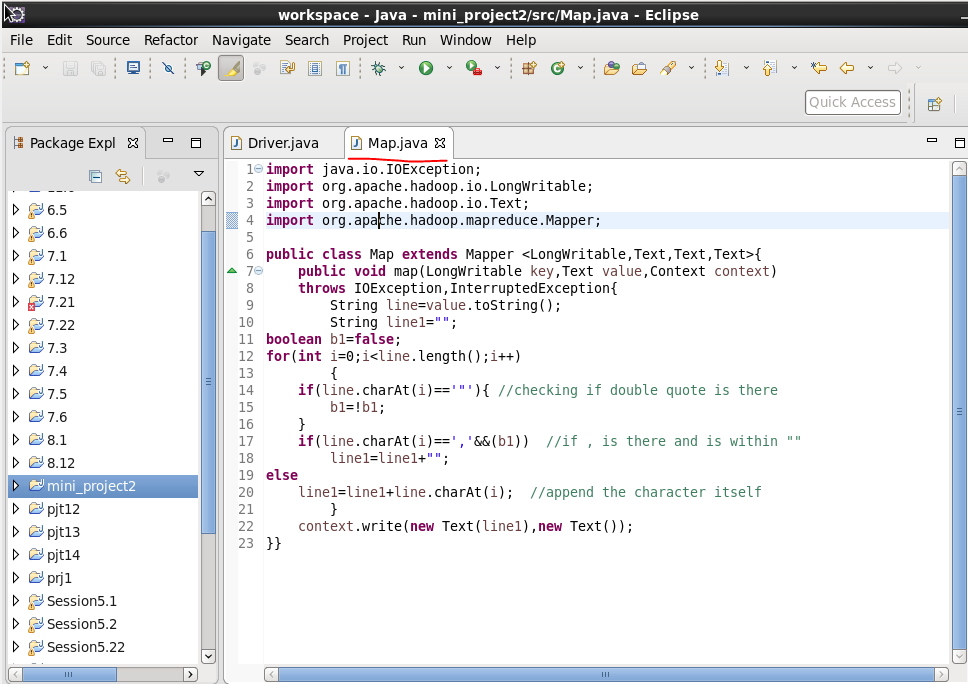


1. **MAPPER CLASS**

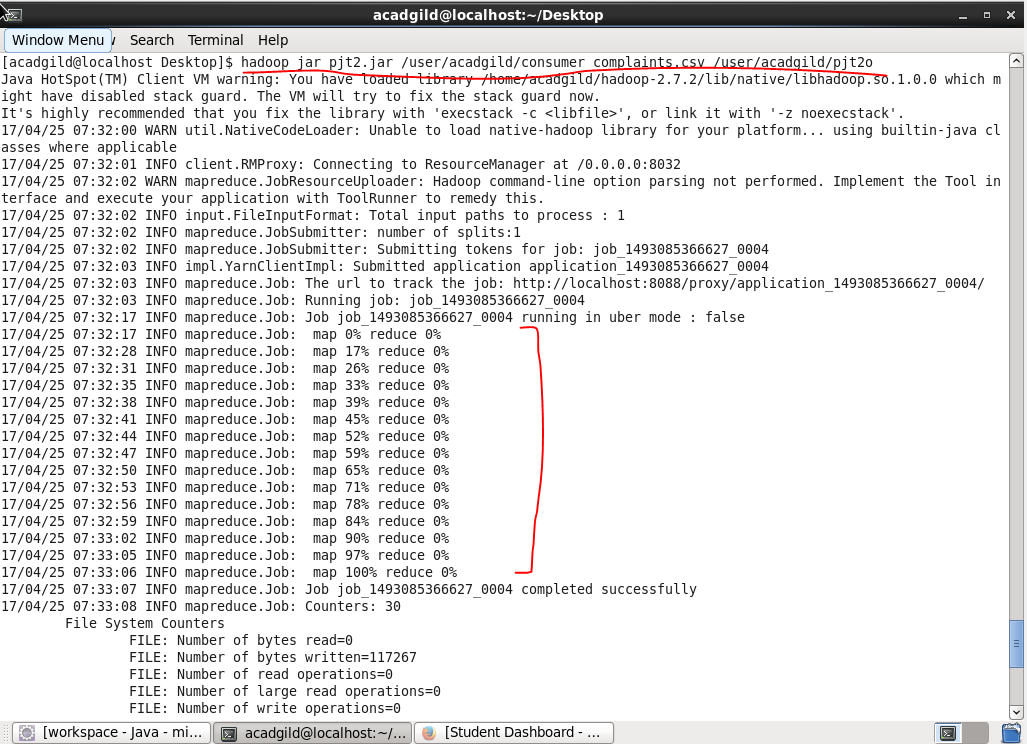
* **WE WILL SET A BOOLEAN AS FALSE AND IT WILL REVERSES ITS STATE WHENEVER THERE IS A DOUBLE QUOTE**
* **WHENEVER IT ENCOUNTERS A COMMA AND IS WITHIN DOULE QUOTES IT REPLACES THE COMMA AND IT STORE OR ELSE THE CHARACTER IS WRITTEN WITH THE HELP OF If CONDITION Description**

**Steps :**

* Importing the necessary jars for map reduce
* Overriding the map method of mapper class for our computation
* Storing the entire value of mapper in the string “line”
* Defining a string line1 where the new records after removing commas within quotes will be concatenated
* Setting a boolean b1 as False
* running a for loop starting from starting index to last index of string line
* WHENEVER the char at that index is double quotes(“), the Boolean b1 gets reversed
* If the char at the index is comma and boolean is true indicating that within double quotes the comma is replaced
* characters get appended with the new string line1.

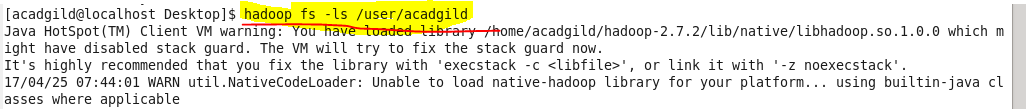


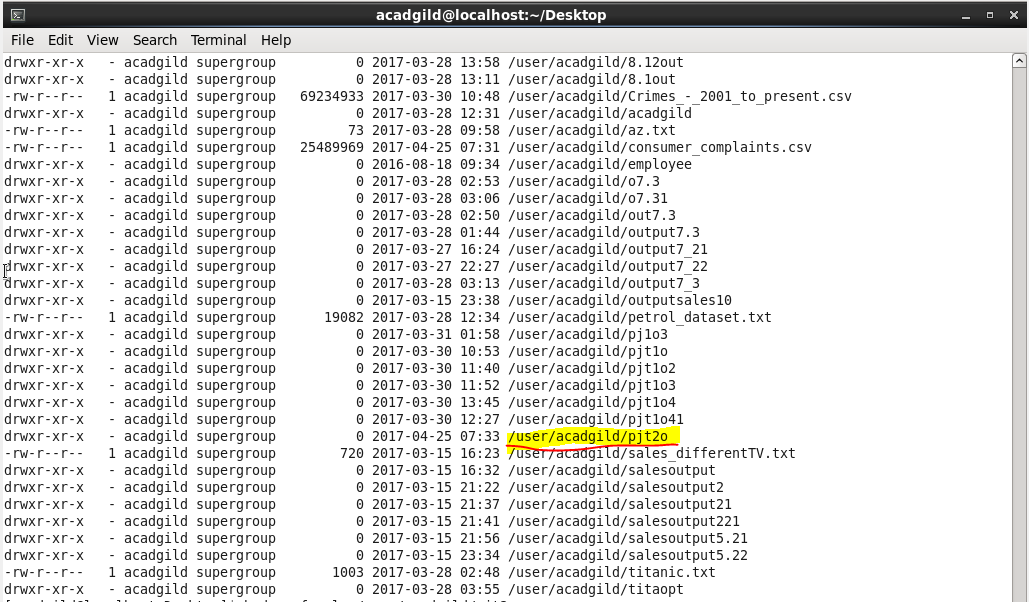
**RUNNING JAR**

****

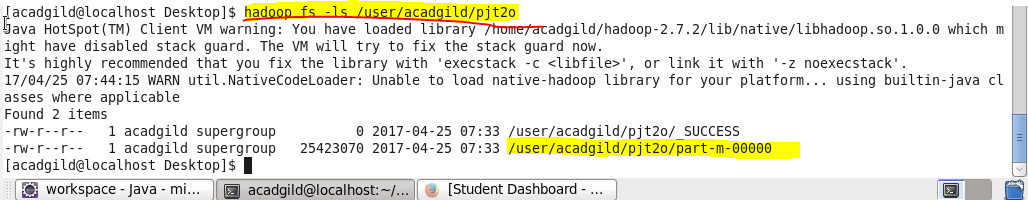
**Output**

**Listing the output file**

****

****

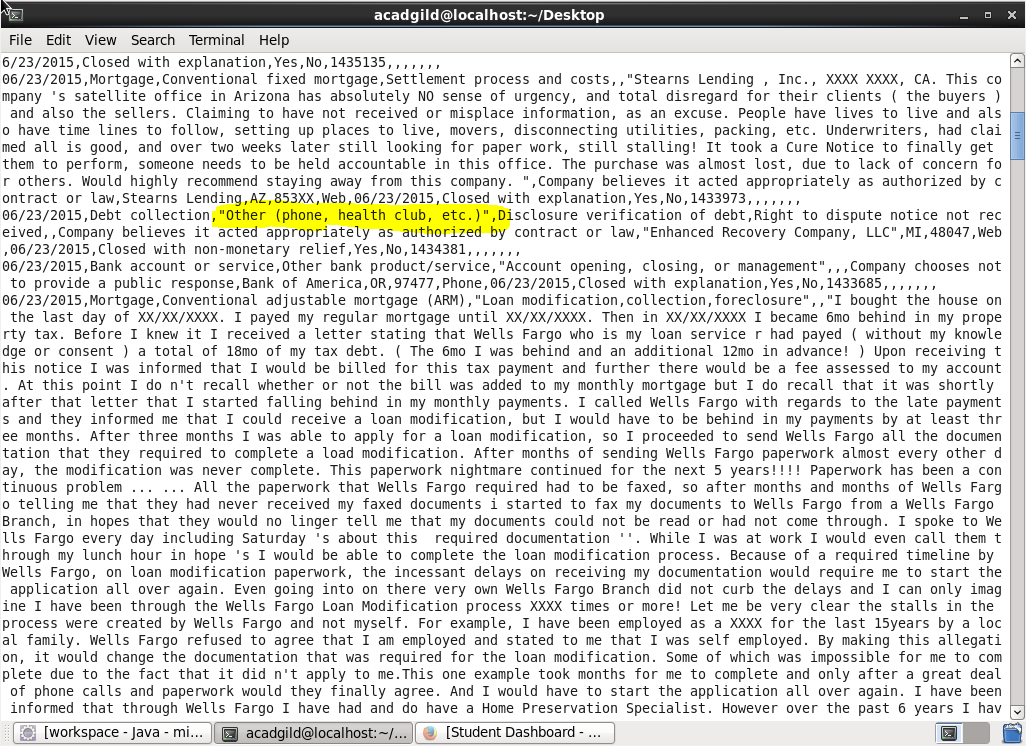
**Checking for our output file after mapreduce :**

****

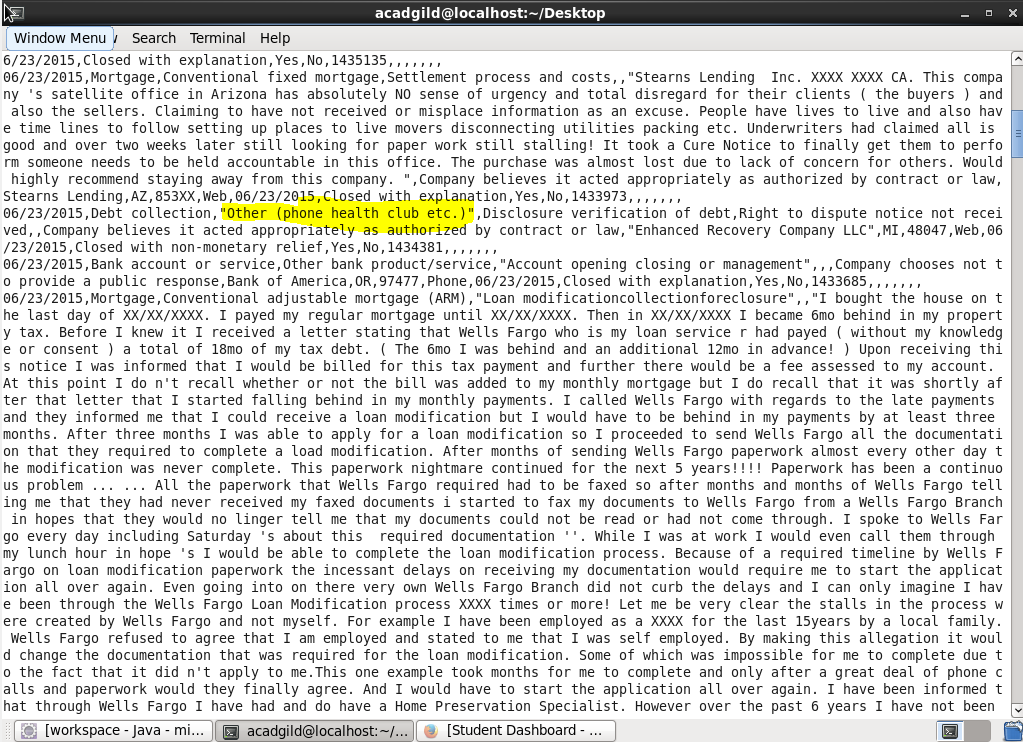
**To compare the output with the input file lets check for the input first:**

**On Hadoop fs –cat /user/acadgild/consumer\_complaints.csv**

**Which is our input csv file –**

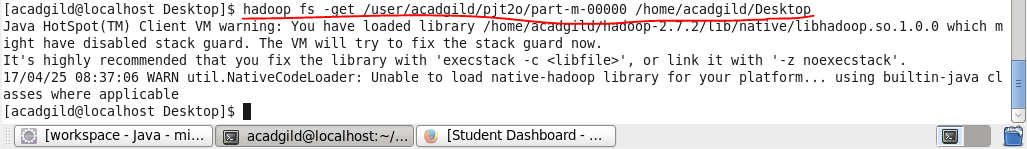
****

**Now lets go to the output file after map reduce :**

****

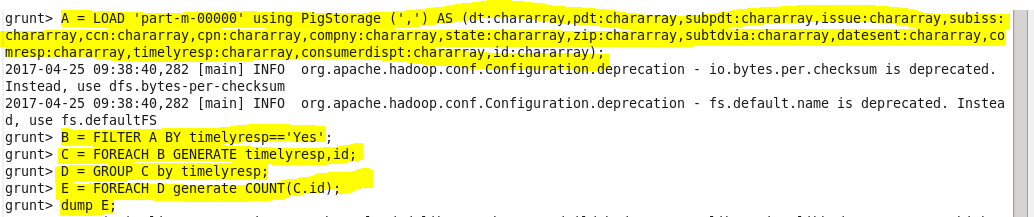
**AS FROM THE FIGURE WE CAN SEE THAT THE COMMA WITHIN DOUBLE QUOTES ARE REPLACED AS HIGHLIGHTED**

NOW GOING TO PIG WE WIL **USE GET COMMAND TO COPY TO LOCAL FOR DOING PIG**



**PIG COMMANDS**

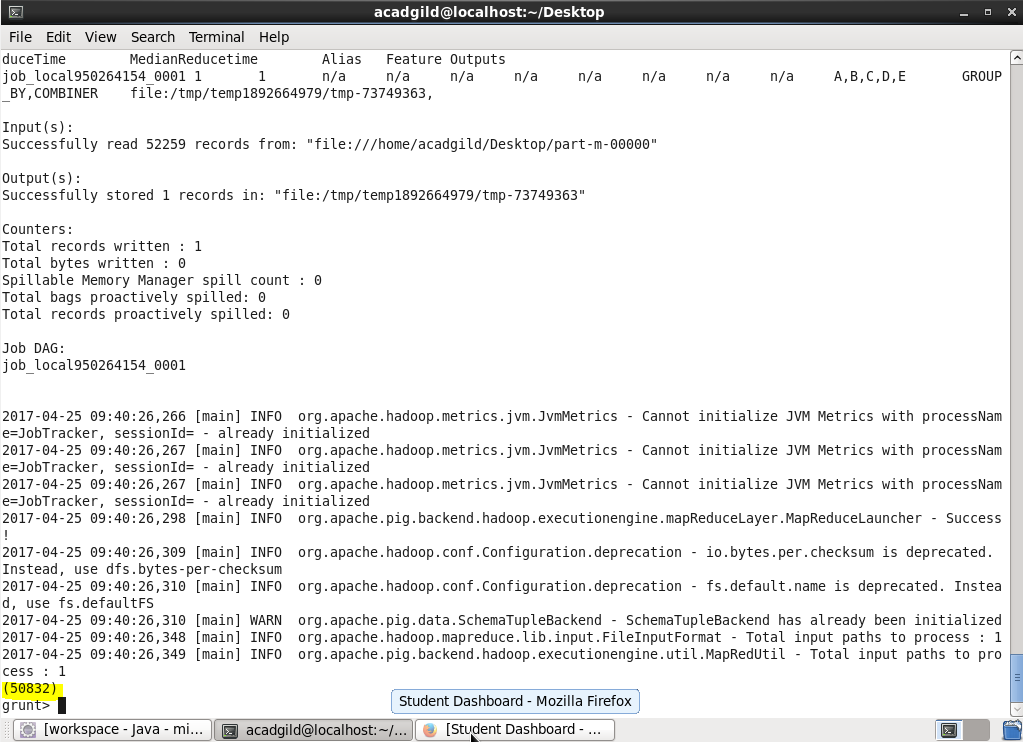
1. **Write a pig script to find no of complaints which got timely response**

****

**LOGIC:**

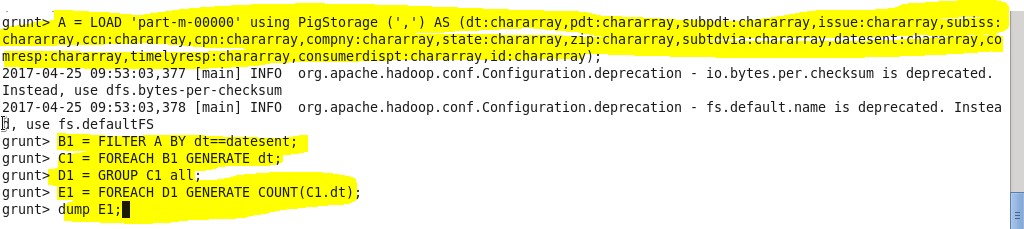
* **loading our input data**
* **filter complaints where timely response is true**
* **group by timely response and make a count of id**
* **Display the count using dump.**

**OUTPUT :**

****

1. **Write a pig script to find no of complaints where consumer forum forwarded the complaint same day they received to respective company**

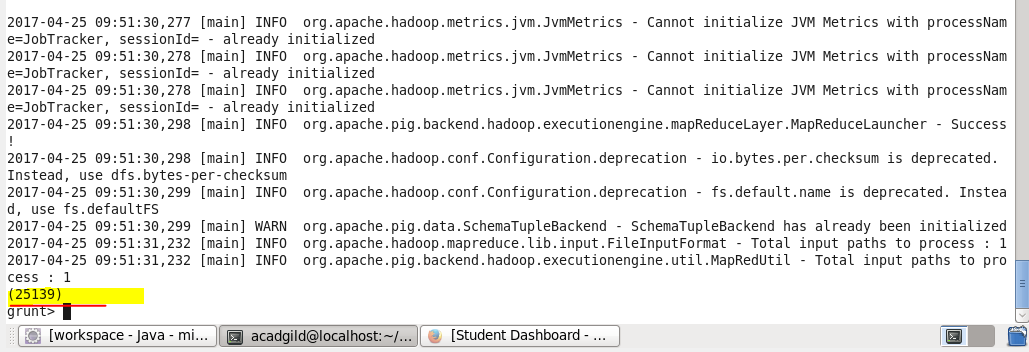
**PIG COMMAND**

****

**LOGIC:**

* **loading cleared data**
* **filter where date received =date sent to company**
* **making COUNT AFTER GROUPING BY ALL**
* **display the total count using dump.**

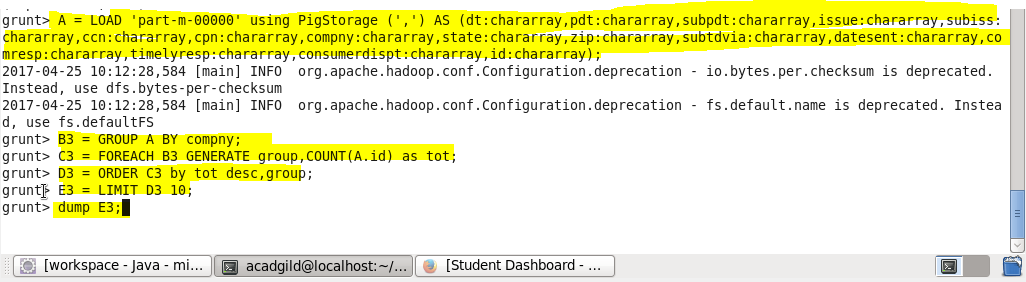
**OUTPUT**

****

1. **Write a pig script to find list of companies topping in complaint chart (companies with maximum number of complaints)**

**LETS MAKE A LIST OF TOP 10 COMPANIES**

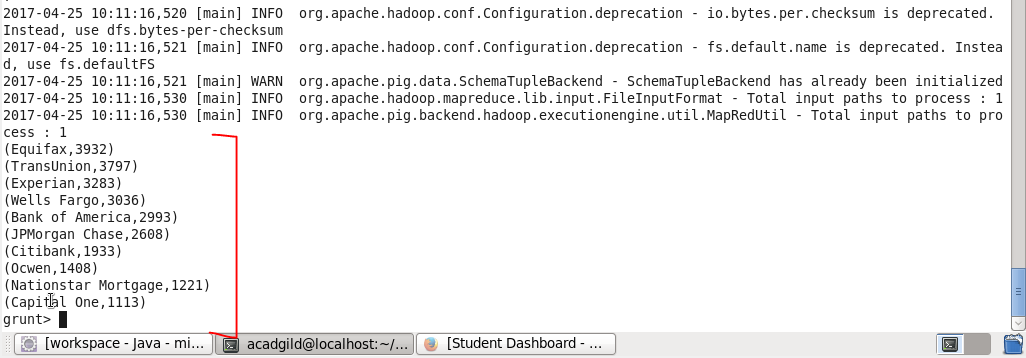
**PIG COMMAND**

****

**LOGIC :**

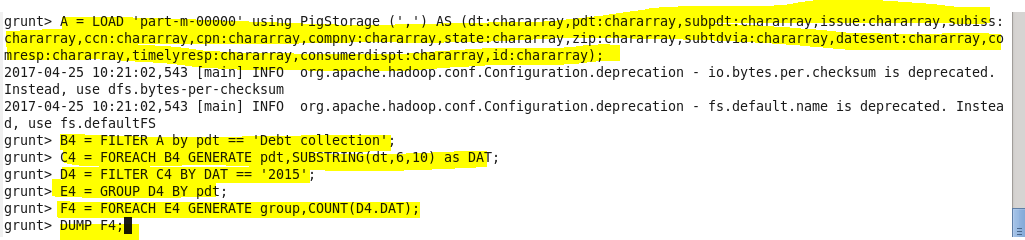
* **loading cleared data**
* **group by company name**
* **make a count of cid for Each company**
* **order by desc order of counts and limit by 10**
* **display the result using dump**

**output:**

****

1. **Write a pig script to find no of complaints filed with product type has "Debt collection" for the year 2015**

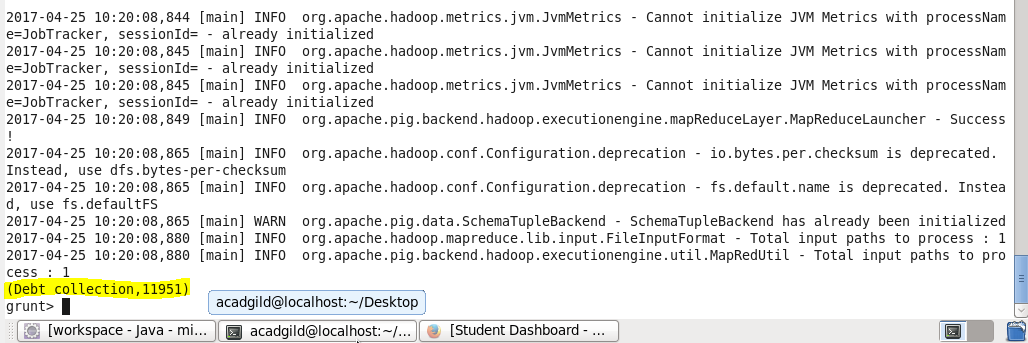
**Pig command -**

****

**Logic –**

* **filter by product =”DEBT COLLECTION”**
* **USED SUBSTRING FUNCTION TO GENERATE YEAR as “DAT”**
* **filter by DAT==2015**
* **MADE COUNT and display using dump**

**Output :**

****