# Big Data Hadoop Training

Project: Titanic Data Analysis Solution

The data set contains information about passengers who boarded Titanic ship. It contains data points like:

* Passenger’s age
* Their native place
* Details of who survived
* Fare details of various travel classes
* Number of casualties from various classes etc.

**DATA SET DESCRIPTION**

Column 1 : PassengerId

Column 2 : Survived (survived=0 & died=1)

Column 3 : Pclass

Column 4 : Name

Column 5 : Sex

Column 6 : Age

Column 7 : SibSp

Column 8 : Parch

Column 9 : Ticket

Column 10 : Fare

Column 11 : Cabin

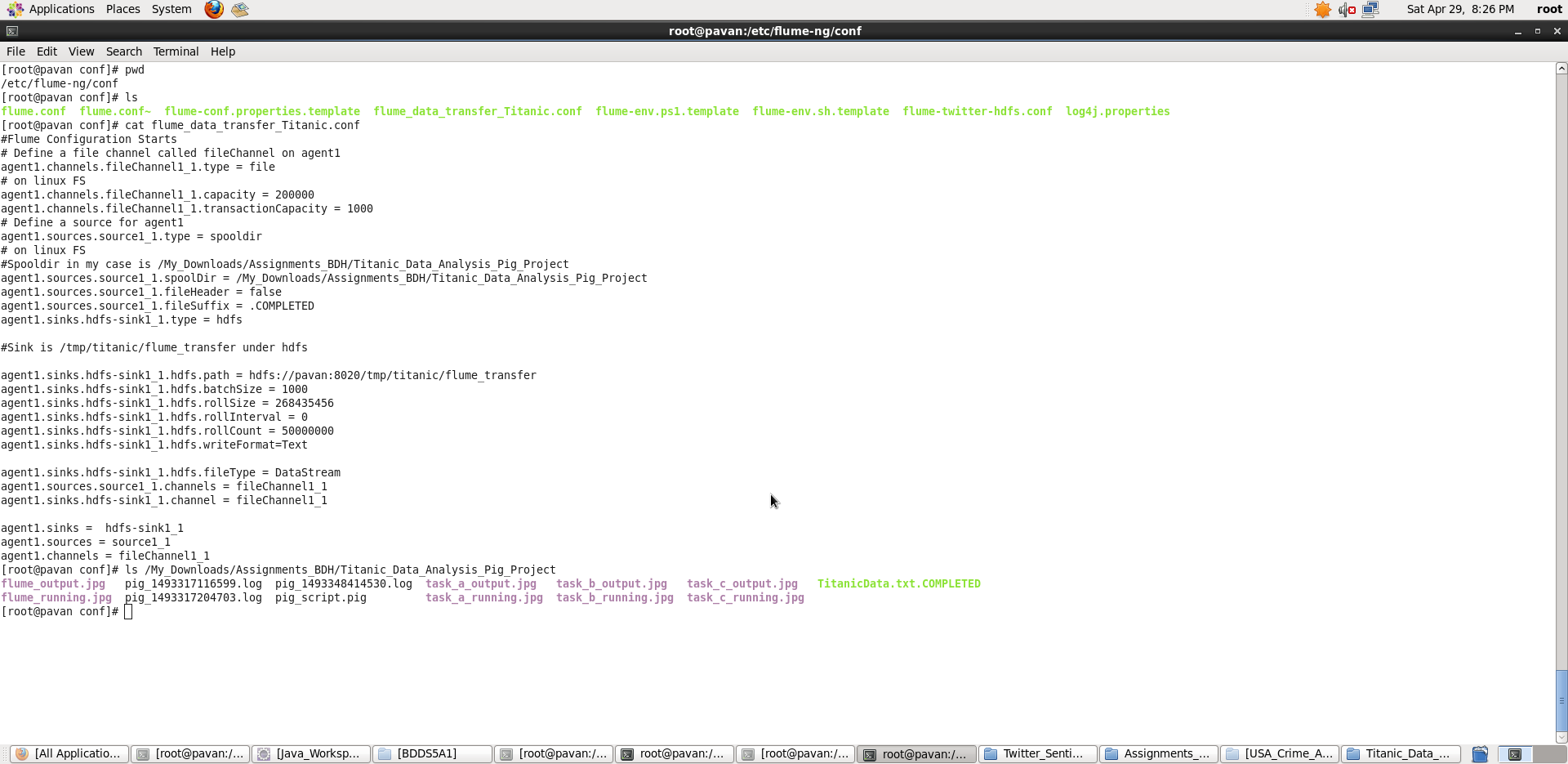
Column 12 : Embarked

**Input files stored in HDFS:**

**Q) You need to copy the data set into HDFS using flume and send the screen shot of that with the project solution**

**A)**  We will be using spool directory as our source and HDFS as destination.  
Here we will create an agent for flume (here data is fetched from Local File System).

**Flume\_data\_transfer\_titanic.conf file: (put it in /etc/flume-ng/conf directory)**



Note: This is a basic conf file configuring the source, channel and sink.

1) agent1.sources.source1\_1.spoolDir is set with input path as in local file system path.

Here <input\_path> is

< /My\_Downloads/Assignments\_BDH/Titanic\_Data\_Analysis\_Pig\_Project>  
2) agent1.sinks.hdfs-sink1\_1.hdfs.path is set with output path as in HDFS path

Here <output\_path> is

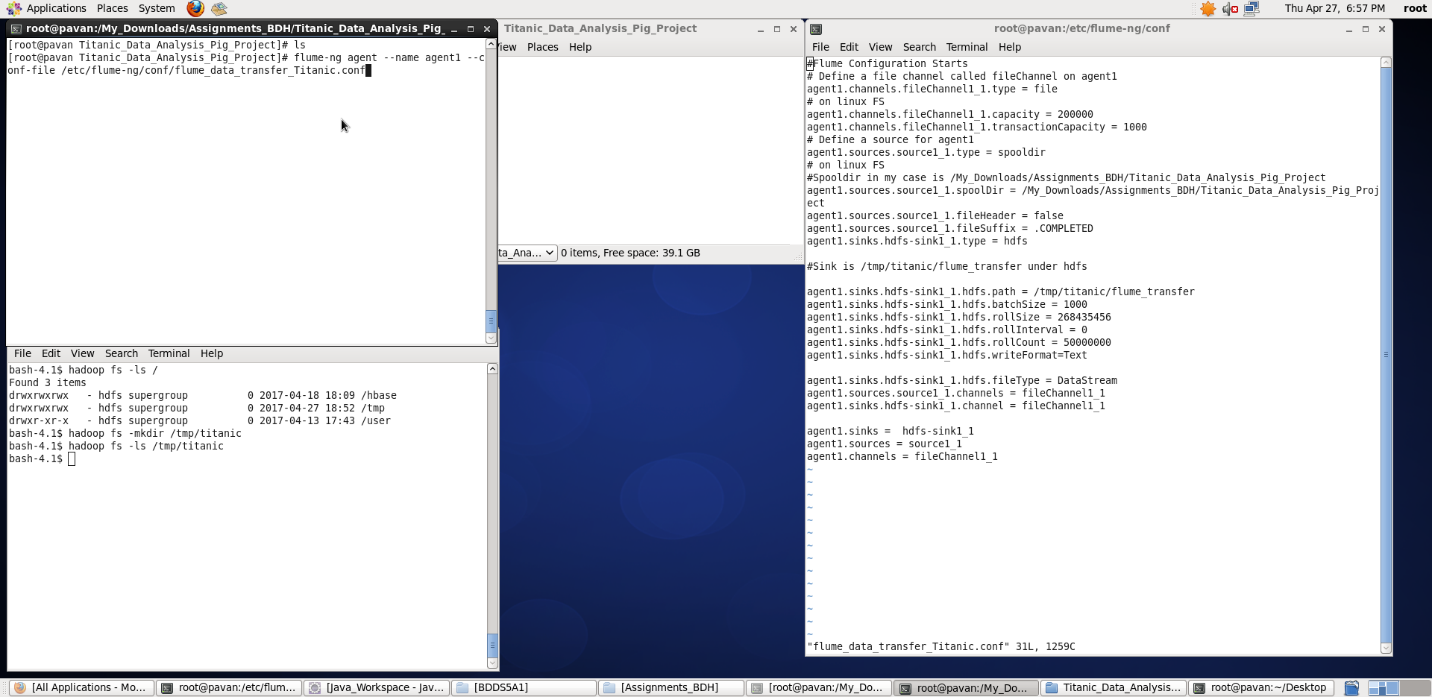
<hdfs://pavan:8020/tmp/titanic/flume\_transfer>

Note: Hostname : pavan , PortNo: 8020 to connect to my HDFS.

**Create specified input and output paths and give read/write permissions accordingly.**

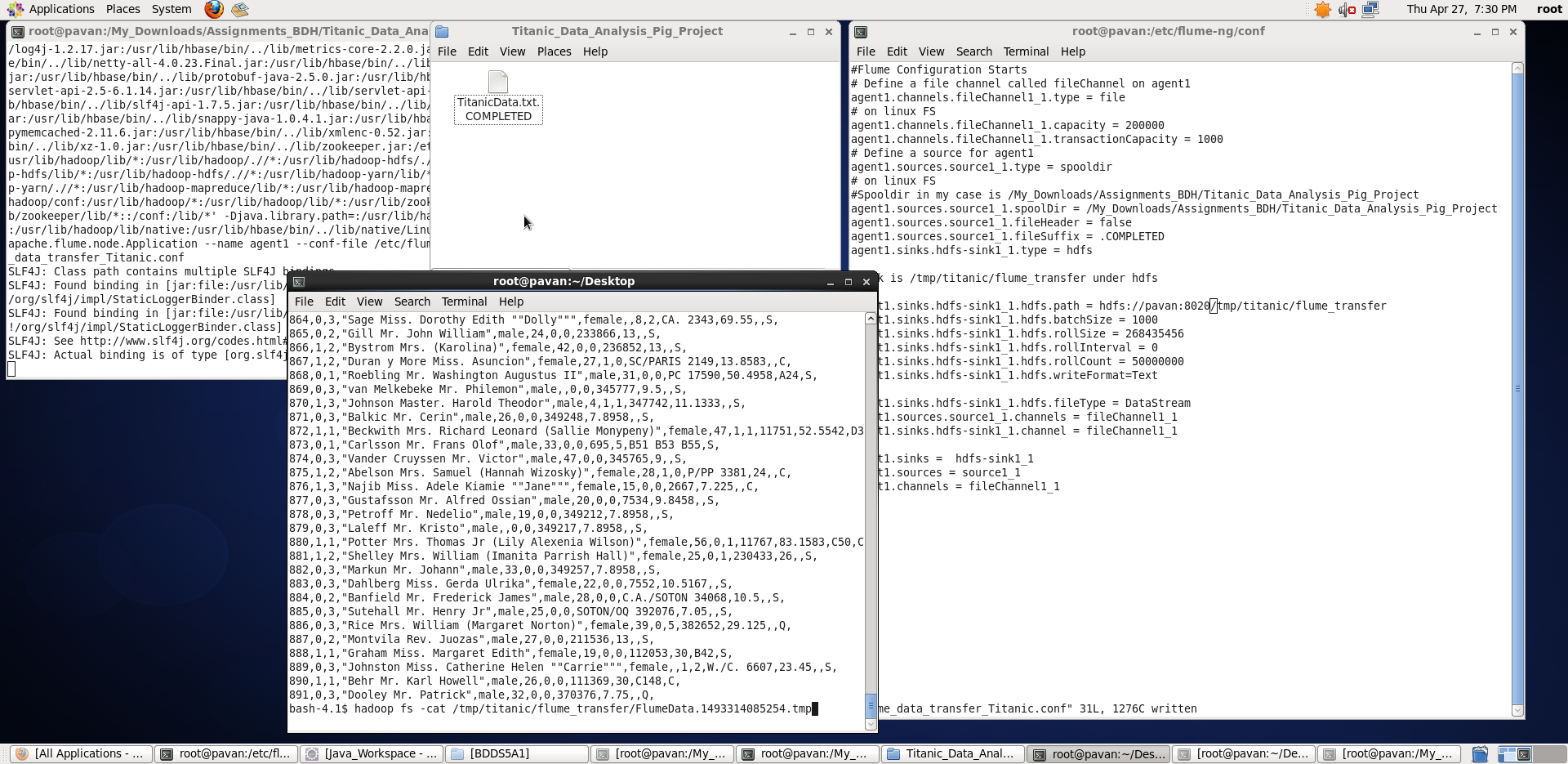
**We will now open another terminal and start flume agent by the following   
Command:**

**flume-ng agent –-name agent1 –-conf-file /etc/flume-ng/conf/flume\_data\_transfer\_Titanic.conf**

****

**Note: I didn’t place my input dataset inside the specified spool directory yet. Once the flume starts running, we place the dataset which immediately gets copied into HDFS.**

**Once, the copying is done, it marks the input files as .COMPLETED (we specified in flume\_data\_transfer\_Titanic.conf file)**

****

**We can see the dataset being copied into specified output directory in HDFS using the command:**

**$ hadoop fs –cat /tmp/titanic/flume\_transfer/FlumeData.1493314085254.tmp**

**Now, we work on problem statements in which I am loading the data from HDFS.**

**(a) In this problem statement we will find the average fare of each class.**

A) **Pig Latin Script**:

REGISTER '/usr/lib/pig/piggybank.jar';

titanic\_dataset = LOAD 'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'UNIX') AS (PassengerID:int,Survived:int,Pclass:int,Name:chararray,Sex:chararray,Age:int,SibSp:int,Parch:int,Ticket:chararray,Fare:double,Cabin:chararray,Embarked:chararray);

group\_by\_class = GROUP titanic\_dataset BY Pclass;

avg\_fare\_per\_class = FOREACH group\_by\_class GENERATE group, AVG(titanic\_dataset.Fare);

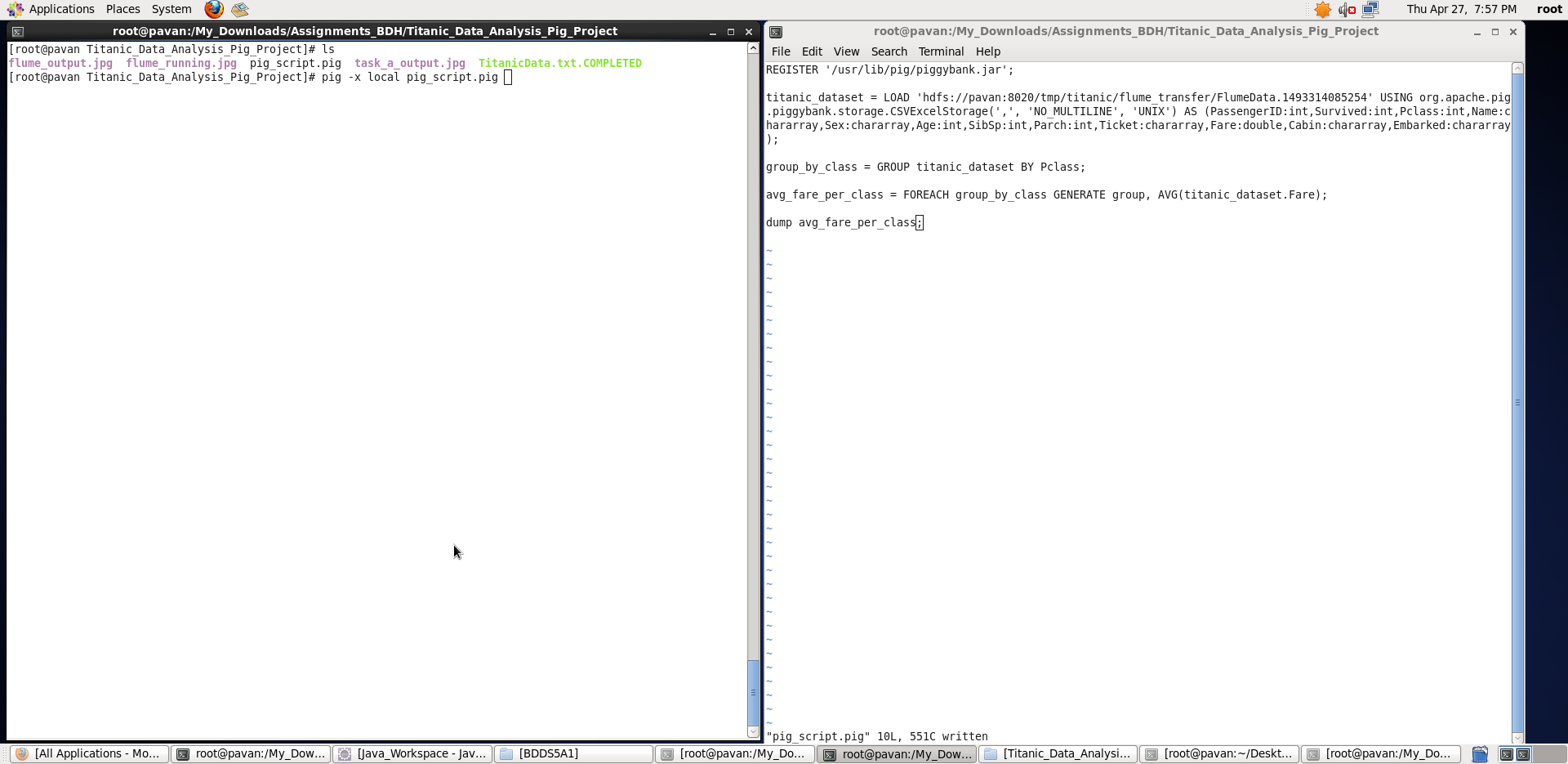
dump avg\_fare\_per\_class;

**Explanation :**

* Loaded data from HDFS file location: **'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254'** into “titanic\_dataset” relation. Here ‘,’ is the delimiter and schema is provided as mentioned in the question
* Then, all records are grouped with respect to their Pclass.
* Now, foreach key – Pclass , I have generated the average fare of that particular Pclass using the AVG – average function.
* **Outputting only “Pclass and its Avg Fare” as per Question using dump command.**

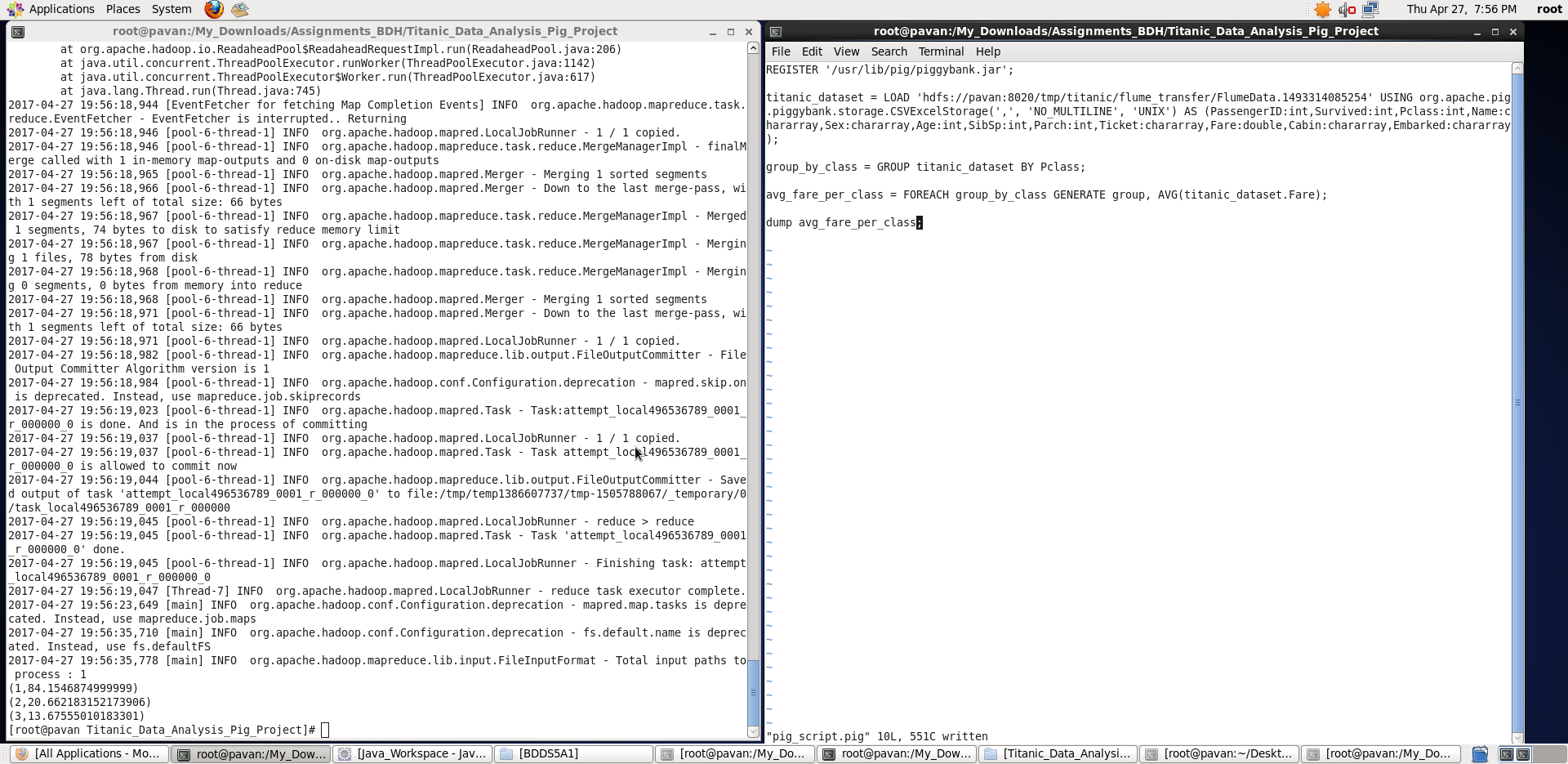
Screenshots

Running **Pig Latin Script in Local Mode** inside /My\_Downloads/Assignments\_BDH/Titanic\_Data\_Analysis\_Pig\_Project/ directory



Output displayed: **<Pclass> <Avg\_Fare>**

**Note: We can store the output if required in a file using “store” command.**



**b) In this problem statement we will find the number of people alive in each class and are embarked in Southampton.**

**Pig Latin Script:**

REGISTER '/usr/lib/pig/piggybank.jar';

titanic\_dataset = LOAD 'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'UNIX') AS (PassengerID:int,Survived:int,Pclass:int,Name:chararray,Sex:chararray,Age:int,SibSp:int,Parch:int,Ticket:chararray,Fare:double,Cabin:chararray,Embarked:chararray);

people\_alive\_and\_embarked\_S = FILTER titanic\_dataset BY (Survived==0) AND (Embarked=='S');

grouping\_them\_by\_class = GROUP people\_alive\_and\_embarked\_S BY Pclass;

people\_alive\_embark\_S\_per\_class = FOREACH grouping\_them\_by\_class GENERATE group, COUNT(people\_alive\_and\_embarked\_S);

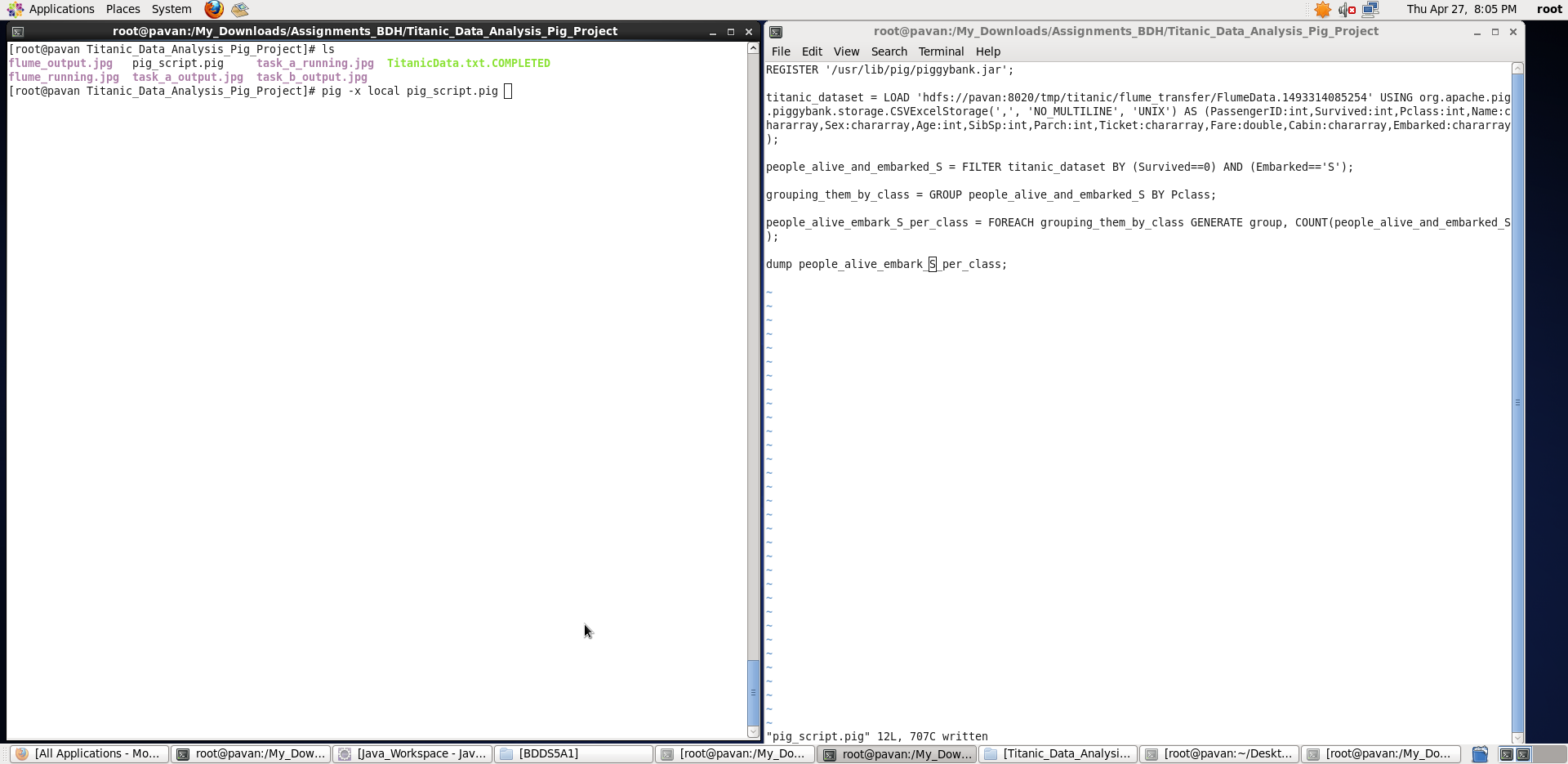
dump people\_alive\_embark\_S\_per\_class;

**Explanation :**

* Loaded data from HDFS file location: **'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254'** into “titanic\_dataset” relation. Here ‘,’ is the delimiter and schema is provided as mentioned in the question
* Then, filtering the dataset based on the conditions Survived==0 (implies they are alive) and Embarked==’S’ (implies they are embarked @ Southampton)
* Then, the filtered records are grouped with respect to their Pclass
* Now, for each key – Pclass, I have used COUNT function to count the number of passengers that are alive and embarked @ Southampton per each class.
* **Outputting only “Pclass and number of people alive & are embarked in Southampton” as per Question using dump command.**

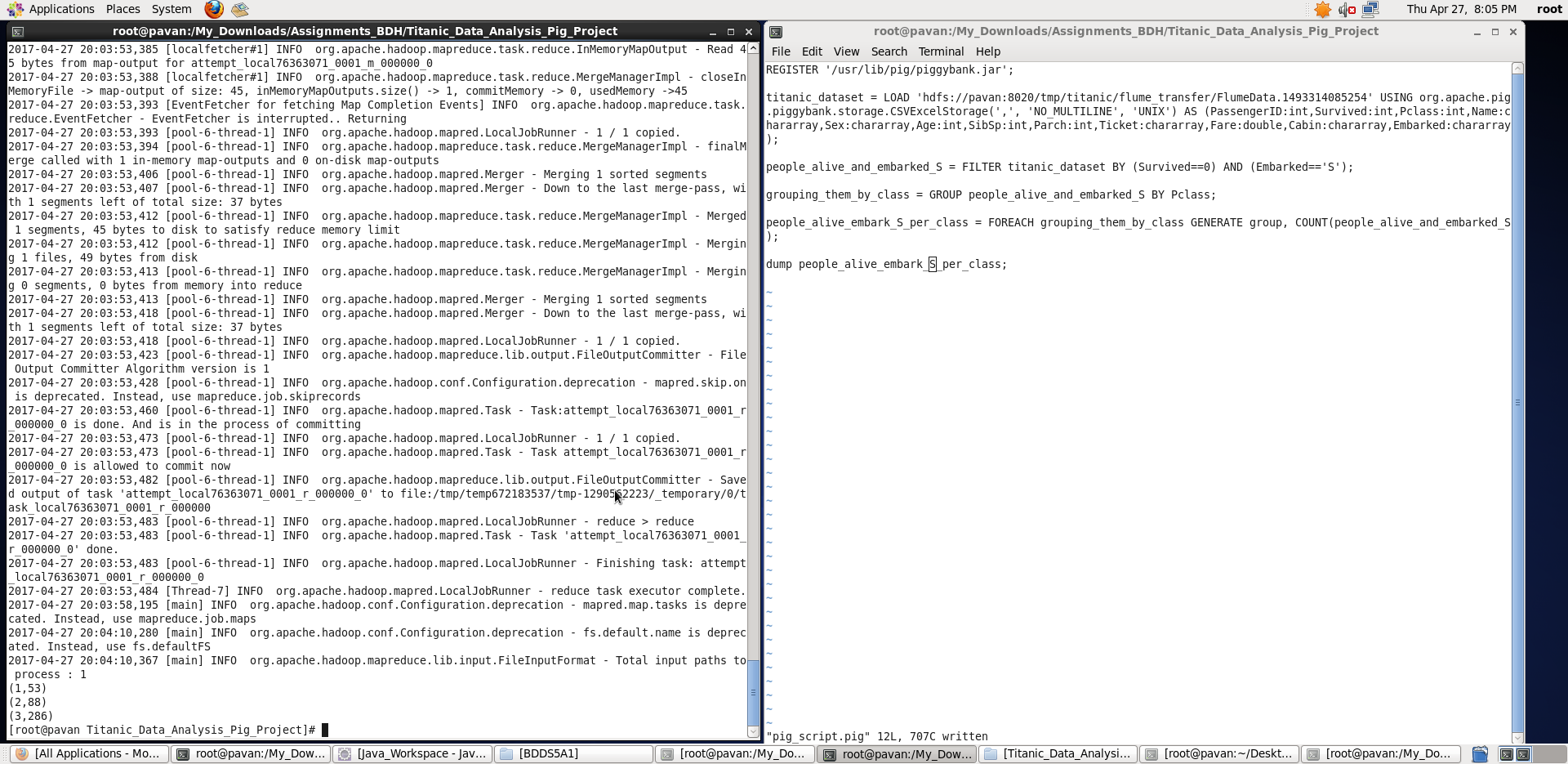
Screenshots

Running **Pig Latin Script in Local Mode** inside /My\_Downloads/Assignments\_BDH/Titanic\_Data\_Analysis\_Pig\_Project/ directory



Output displayed: **<Pclass> < people alive in each class and are embarked in Southampton>**

**Note : We can store the output if required in a file using “store” command.**



c) **In this problem statement we will find out number of male and female people died in each class.**

**Pig Latin Script:**

REGISTER '/usr/lib/pig/piggybank.jar';

titanic\_dataset = LOAD 'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'UNIX') AS (PassengerID:int,Survived:int,Pclass:int,Name:chararray,Sex:chararray,Age:int,SibSp:int,Parch:int,Ticket:chararray,Fare:double,Cabin:chararray,Embarked:chararray);

filter\_died\_people = FILTER titanic\_dataset BY Survived==1;

group\_by\_class\_and\_sex = GROUP filter\_died\_people BY (Pclass, Sex);

no\_male\_female\_died\_per\_class = FOREACH group\_by\_class\_and\_sex GENERATE

FLATTEN(group) AS (Pclass, Sex), COUNT(filter\_died\_people);

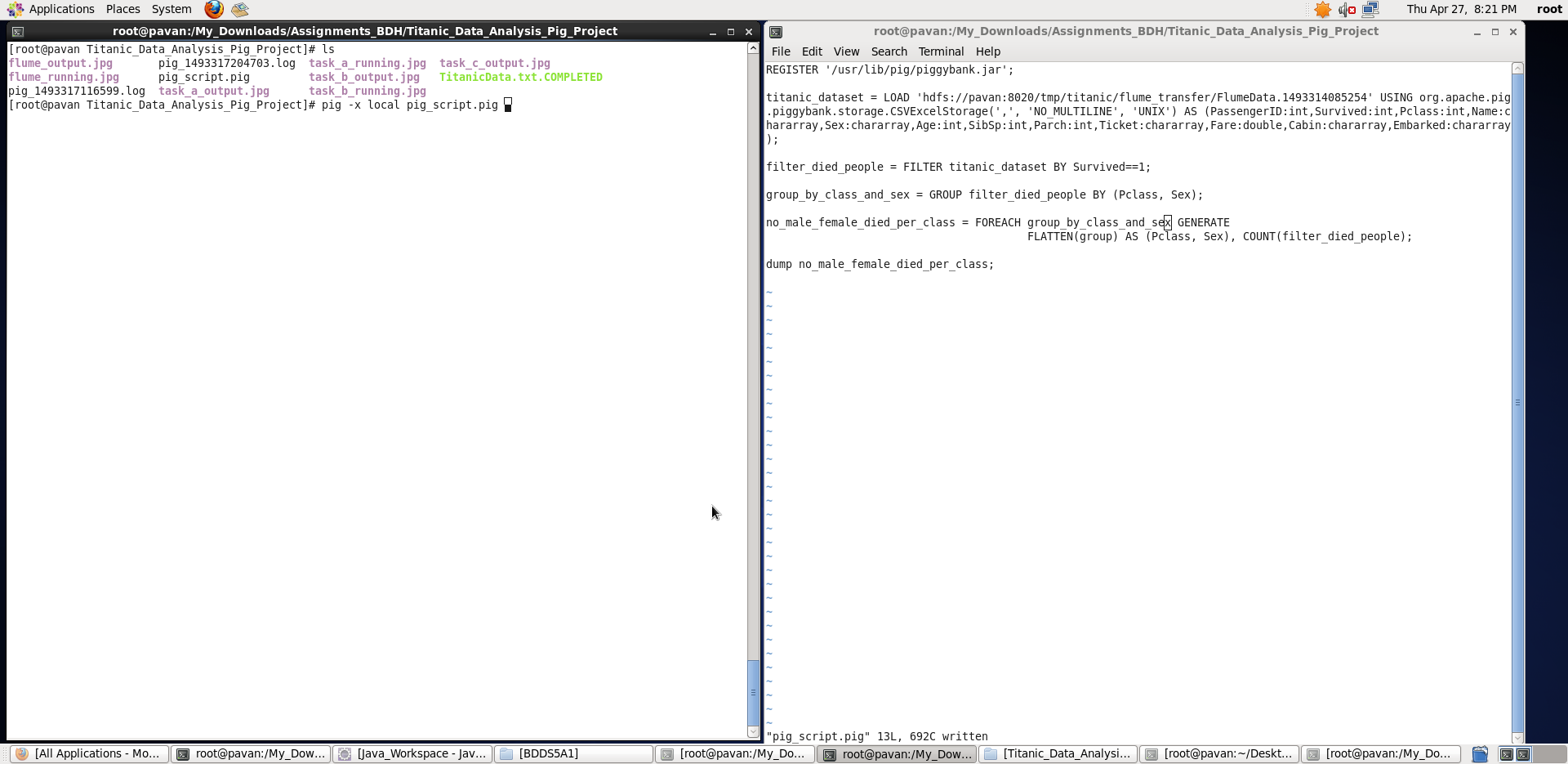
dump no\_male\_female\_died\_per\_class;

**Explanation :**

* Loaded data from HDFS file location: **'hdfs://pavan:8020/tmp/titanic/flume\_transfer/FlumeData.1493314085254'** into “titanic\_dataset” relation. Here ‘,’ is the delimiter and schema is provided as mentioned in the question
* Then, filtering those records of people who died Survived==1 implies “Died”
* Now, grouping the filtered records as per Pclass and also Sex. Note: There are 3 Pclasses and 2 categories in Sex, so total 6 groups with key (Pclass,Sex).
* Now, foreach group , we flatten the key and count the number of died people belonging to that particular group using COUNT function.
* **Outputting only “Pclass, Sex and number of died people of that category” as per Question using dump command.**

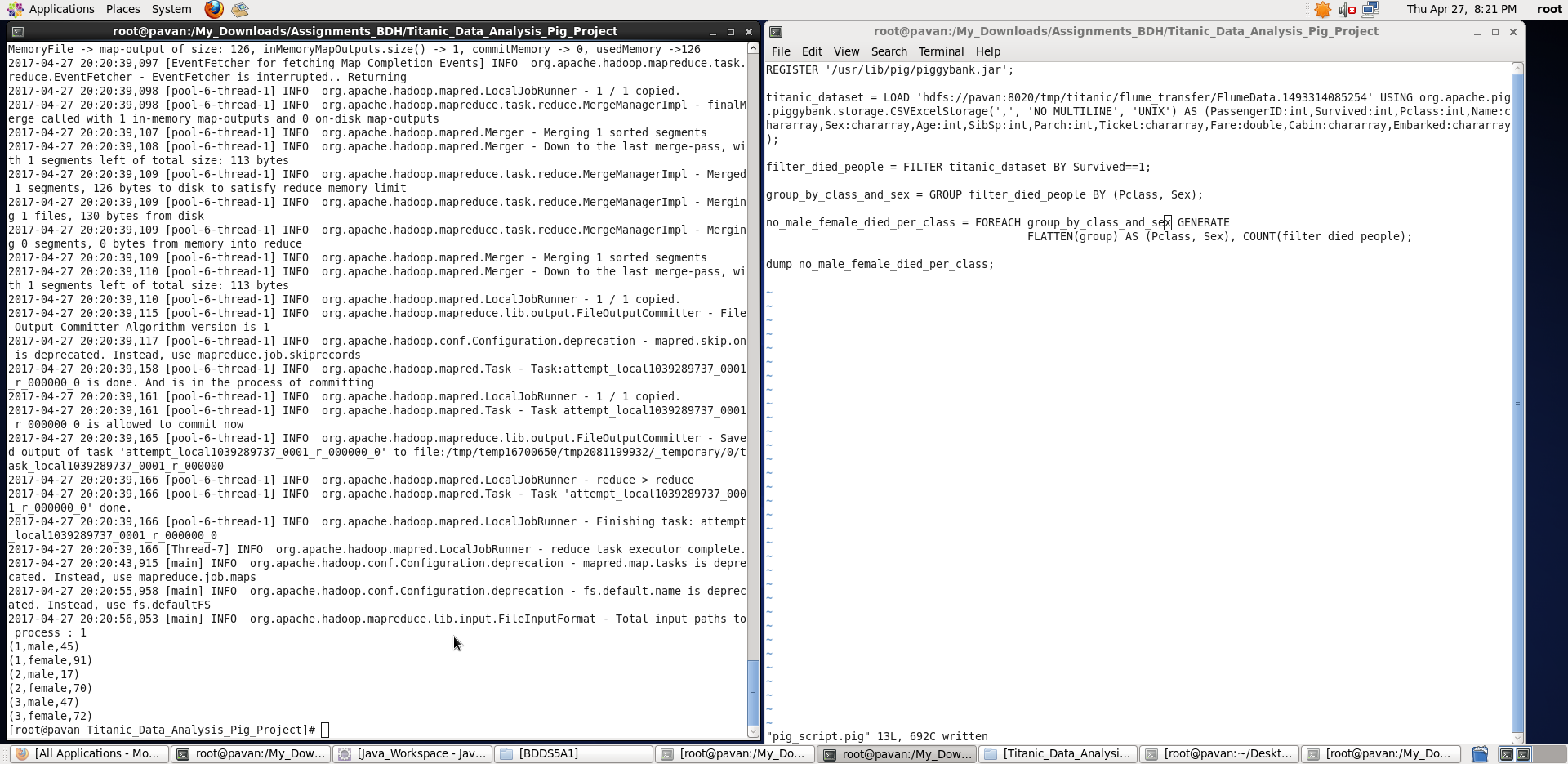
Screenshots

Running **Pig Latin Script in Local Mode** inside /My\_Downloads/Assignments\_BDH/Titanic\_Data\_Analysis\_Pig\_Project/ directory



Output displayed: **<Pclass> <Sex> <number of died people>**

**Note : We can store the output if required in a file using “store” command.**



Thus, with the help of Pig Latin script, I have displayed the results for all the 3 tasks.