**Project 2.1- State-Wise Development Analysis In India**

* Developing system to handle the incoming log feed and store the information in Hadoop Cluster (Flume)
* Analyze the data and understand the progress
* Store the results in Hbase/RDBMS

1. Find out the districts who achieved 100 percent objective in BPL cards .Export the results to mysql using sqoop.

2. Write a Pig UDF to filter the districts which have reached 80% of objectives of BPL cards. Export the results to MySQL using Sqoop.

Dataset - <https://drive.google.com/file/d/0Bxr27gVaXO5sUjd2RWFQS3hQQUE/view> - StatewStatewiseDistrictwisePhysicalProgress.xml

StatewiseDistrictwisePhysicalProgress.xmltrictwisePhysicalProgress.xml

StatewiseDistrictwisePhysicalProgress.xml

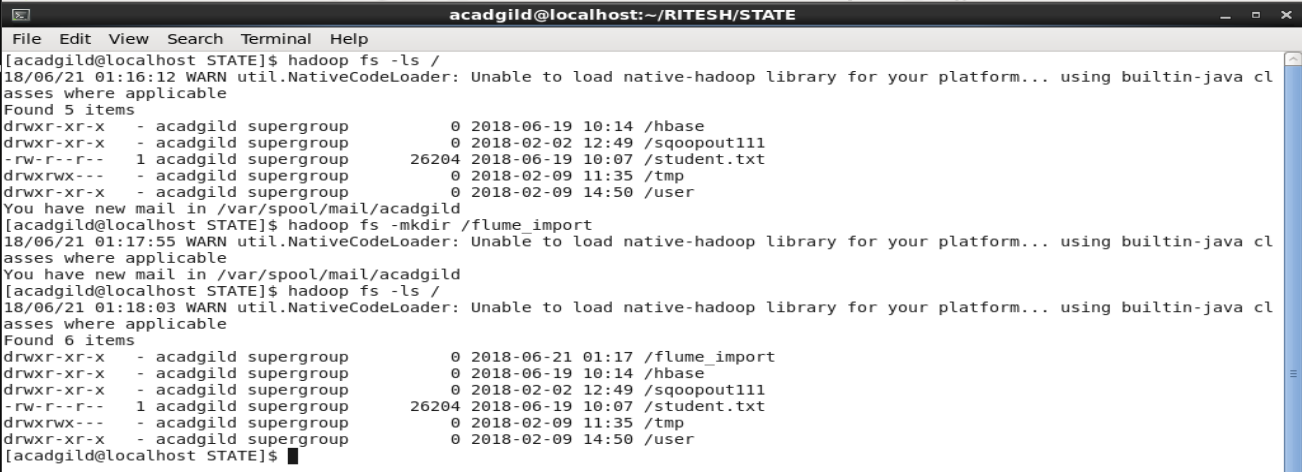
Solution -

1st Phase

* First Checked the folders in hdfs , then created a folder 'flume\_import' in it.

**hadoop fs -ls /**

**hadoop fs -mkdir /flume\_import**



* Then using flume we transfered the data from the local file system that is spool directory to HDFS.

**flume-ng agent -n agent1 -g /home/aadgild/RITESH/STATE/filecopy.conf**





This flume process goes on continuosly to fetch data from local file system to HDFS.

The flume configuration file **filecopy.conf**

**agent1.sources = mysrc**

**agent1.sinks = hdfsdest**

**agent1.channels = mychannel agent1.sources.mysrc.type = exec**

**agent1.sources.mysrc.command = hadoop dfs -put /home/acadgild/RITESH/STATE/spool/StatewiseDistrictwisePhysicalProgress.xml /flume\_import**

**agent1.sinks.hdfsdest.type = hdfs**

**agent1.sinks.hdfsdest.hdfs.path = hdfs://localhost:9000/flume\_import**

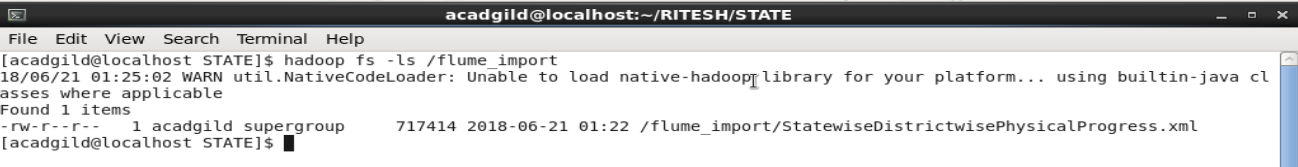
**agent1.channels.mychannel.type = memory**

**agent1.sources.mysrc.channels = mychannel**

**agent1.sinks.hdfsdest.channel = mychannel**

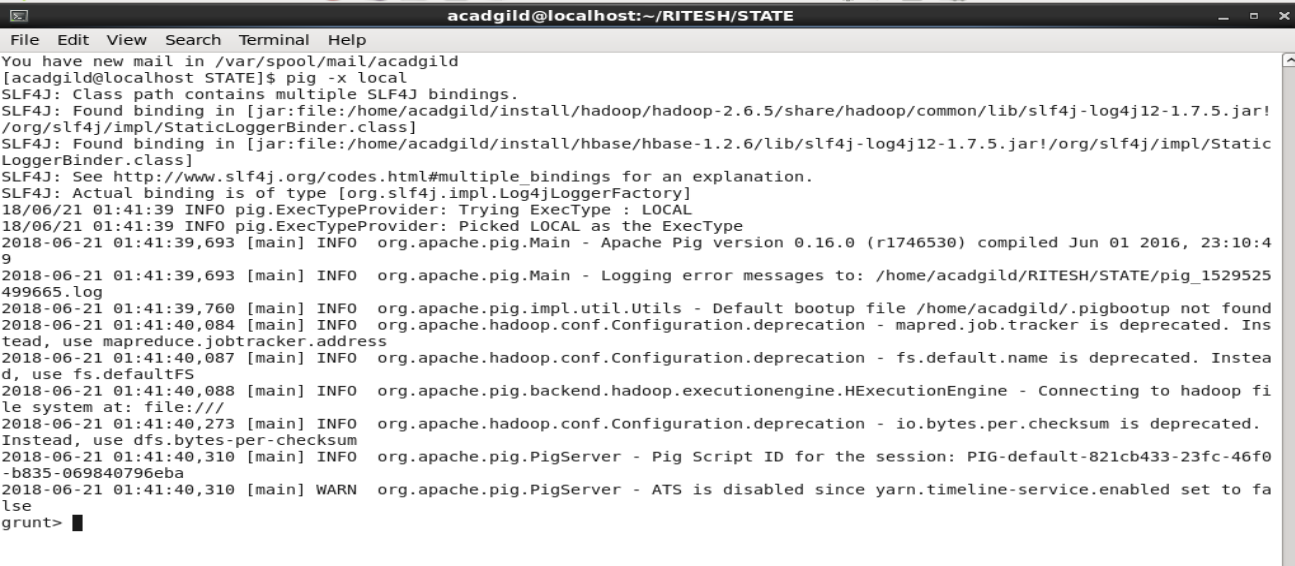
* Then the flume import folder is checked wheteher data got imported.

**hadoop fs -ls /flume\_import**



* Startedthe pig shell

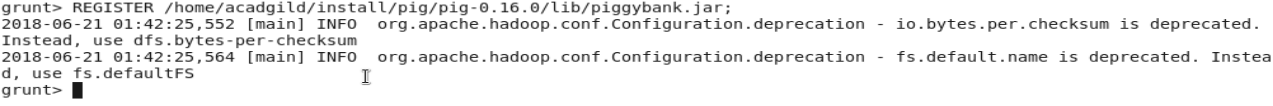
**pig -x local**



1.

* Registering the **piggybank** jar that contains the executables for various pig functions. to parse XML.

**REGISTER /home/acadgild/install/pig/pig-0.16.0/lib/piggybank.jar;**



* Defining the XML Parse function as **Xpath**

**DEFINE XPath org.apache.pig.piggybank.evaluation.xml.XPath;**



* Loading the data in from HDFS and then using the XML Loader function to load the data into the relation with every starting tag ‘row’ as one line of type: chararray with the name **x.**

**data = load 'hdfs://localhost:8020/flume\_import/StatewiseDistrictwisePhysicalProgress.xml' using org.apache.pig.piggybank.storage.XMLLoader('row') as (x:chararray);**



* Generating the rows in relation by using the XML Parser **XPath.** Every required tag for the anlysis under the main tag **row** will be separated by the tag name and given a pseudo name in the relation.

**req\_data = foreach data generate XPath(x,'row/State\_Name') as state, XPath(x,'row/District\_Name') as DistrictName, XPath(x,'row/Project\_Objectives\_IHHL\_BPL') as POIHHLBPL, XPath(x,'row/Project\_Performance-IHHL\_BPL') as PPIHHLBPL;**



* required fields are filtered if null is present.

**filter\_null = filter req\_data by (PPIHHLBPL is not null) AND (POIHHLBPL is not null);**



* Filter the data where performance objective is equal to performance achiedved.

**centPercentObjective = filter filter\_null by((int)PPIHHLBPL==(int)POIHHLBPL);**



* List is generated containg state , districts , performance objective and performance achieved of BPL cards.

**districts = foreach centPercentObjective generate state,DistrictName,POIHHLBPL,PPIHHLBPL;**



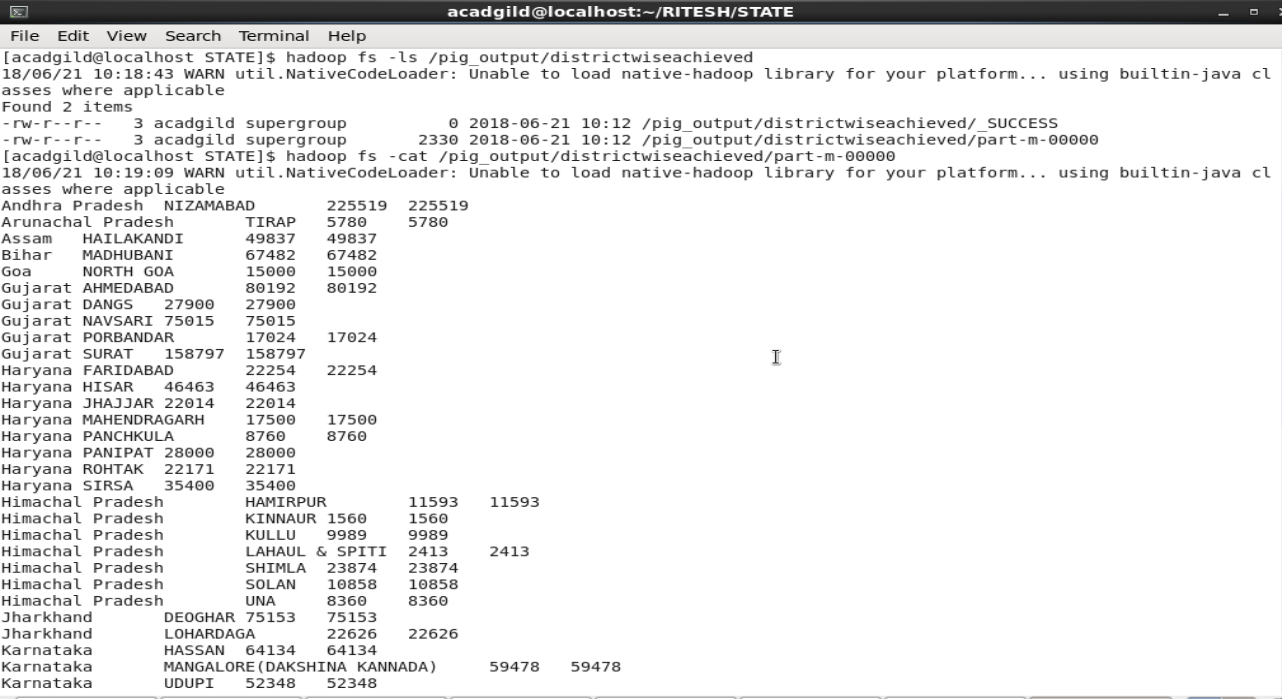
* Stored the final results into HDFS

**store districts into 'hdfs://localhost:8020/pig\_output/districtwiseachieved' using PigStorage(',');**



* Checked the content of the pig output in hdfs.

**hadoop fs -ls /pig\_output/districtwiseachieved**

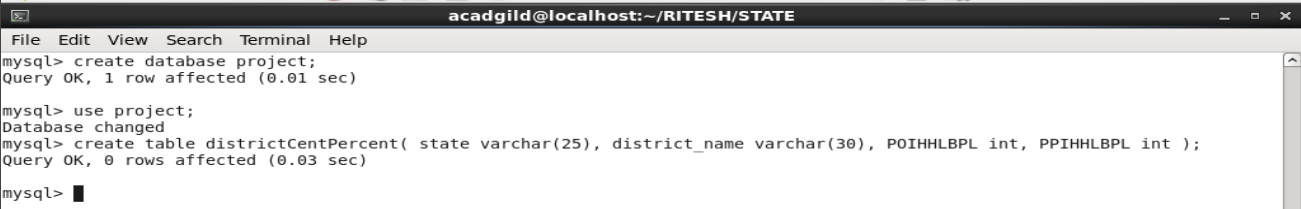


* Created a database project and a table 'districtCentPercent'.

**create database project;**

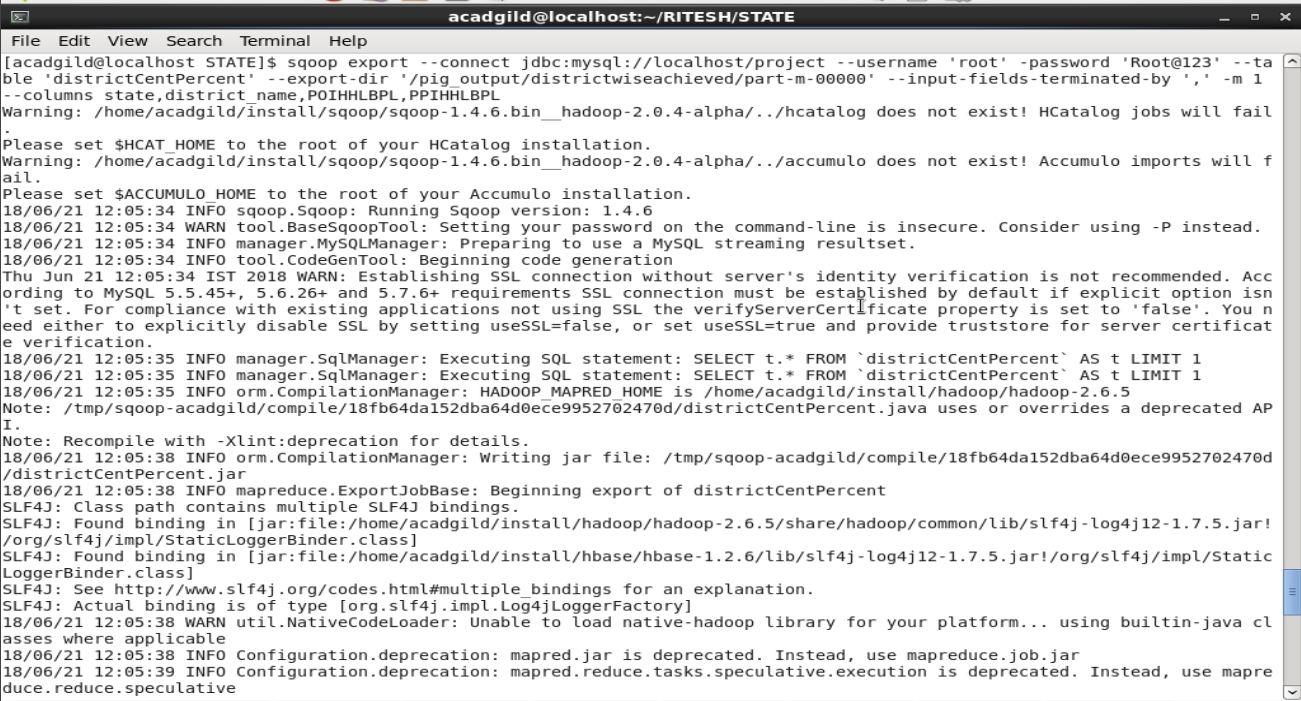
**use project;**

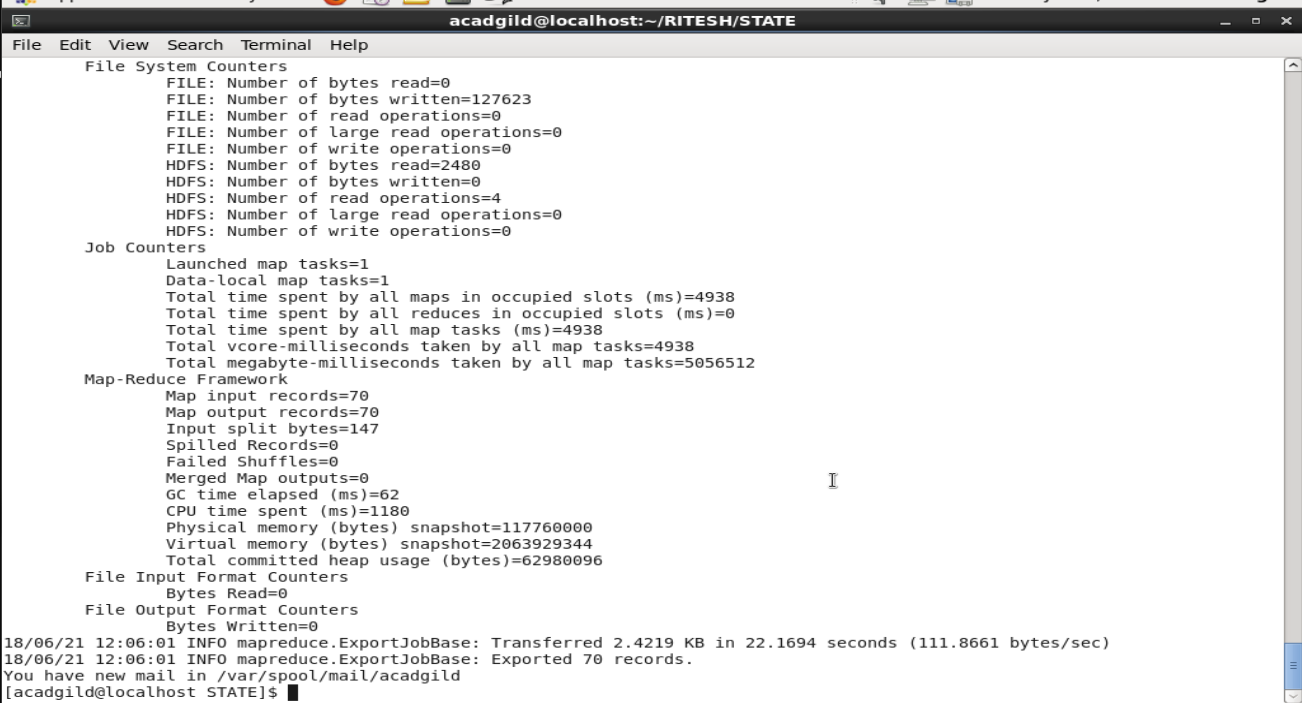
**create table districtCentPercent(state varchar(25), district\_name varchar(30), POIHHLBPL int , PPIHHLBPL int);**



* Using sqoop we exported the data from HDFS to mysql.

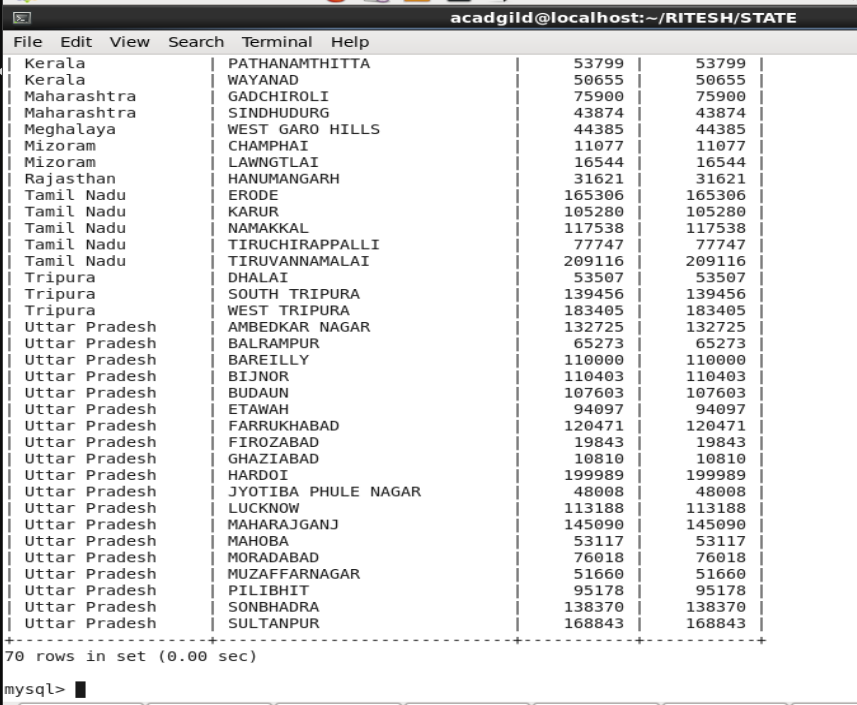
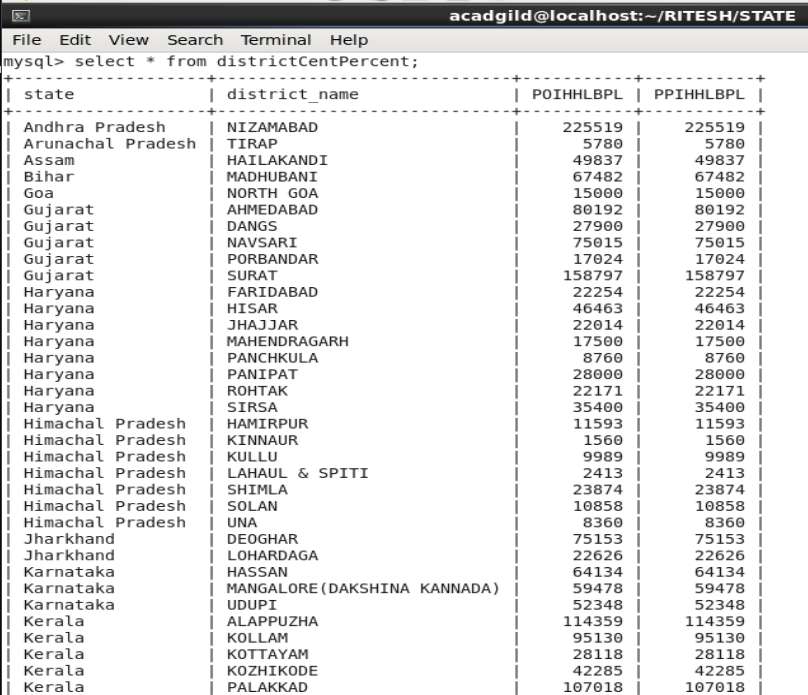
**sqoop export –connect jdbc:mysql://localhost/project –username 'root' -password** [**'Root@123**](mailto:'Root@123)**' –table 'districtCentPercent' –export-dir '/pig\_output/districtwiseachieved/part-m-00000' –input-fields-terminated-by ',' -m 1 –columns state,district,name,POIHHLBPL,PPIHHLBPL**





* We checked the content of the table in mysql.

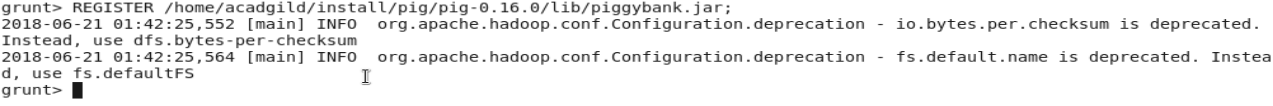
**select \* from districtCentPercent;**



2.

* Registering the **piggybank** jar that contains the executables for various pig functions. to parse XML.

**REGISTER /home/acadgild/install/pig/pig0.16.0/lib/piggybank.jar;**



* Defining the XML Parse function as **Xpath**

**DEFINE XPath org.apache.pig.piggybank.evaluation.xml.Xpath;**



* Registering pig UDF jar.

**REGISTER '/home/acadgild/RITESH/STATE/Filter80.jar';**



**Filter80 code:**

package pigUDF;

import java.io.IOException;

import org.apache.pig.EvalFunc;

import org.apache.pig.data.Tuple;

public class Filter80 extends EvalFunc<Double>{

public Double exec(Tuple input) throws IOException {

double performance = (double)input.get(0);

double objective = (double)input.get(1);

double percentage = (performance/objective)\*100;

if(percentage>80.0)

return percentage;

else

return 0.0;

}

}

* Loading the data in from HDFS and then using the XML Loader function to load the data into the relation with every starting tag ‘row’ as one line of type: chararray with the name **x.**

**data = load 'hdfs://localhost:8020/flume\_import/StatewiseDistrictwisePhysicalProgress.xml' using org.apache.pig.piggybank.storage.XMLLoader('row') as (x:chararray);**



* Generating the rows in relation by using the XML Parser **XPath.** Every required tag for the anlysis under the main tag **row** will be separated by the tag name and given a pseudo name in the relation.

**req\_data = foreach data generate XPath(x,'row/State\_Name') as state, XPath(x,'row/District\_Name') as DistrictName, XPath(x,'row/Project\_Objectives\_IHHL\_BPL') as POIHHLBPL, XPath(x,'row/Project\_Performance-IHHL\_BPL') as PPIHHLBPL;**

****

* required fields are filtered if null is present.

**filter\_null = filter req\_data by (PPIHHLBPL is not null) AND (POIHHLBPL is not null);**



* List is generated where it contains sate,district name, percentage completion of BPL cards.

**percentObjective = foreach filter\_null generate state,DistrictName,ROUND\_TO(pigUDF.Filter80((double)PPIHHLBPL,(double)POIHHLBPL),2) as BPL\_Percent;**



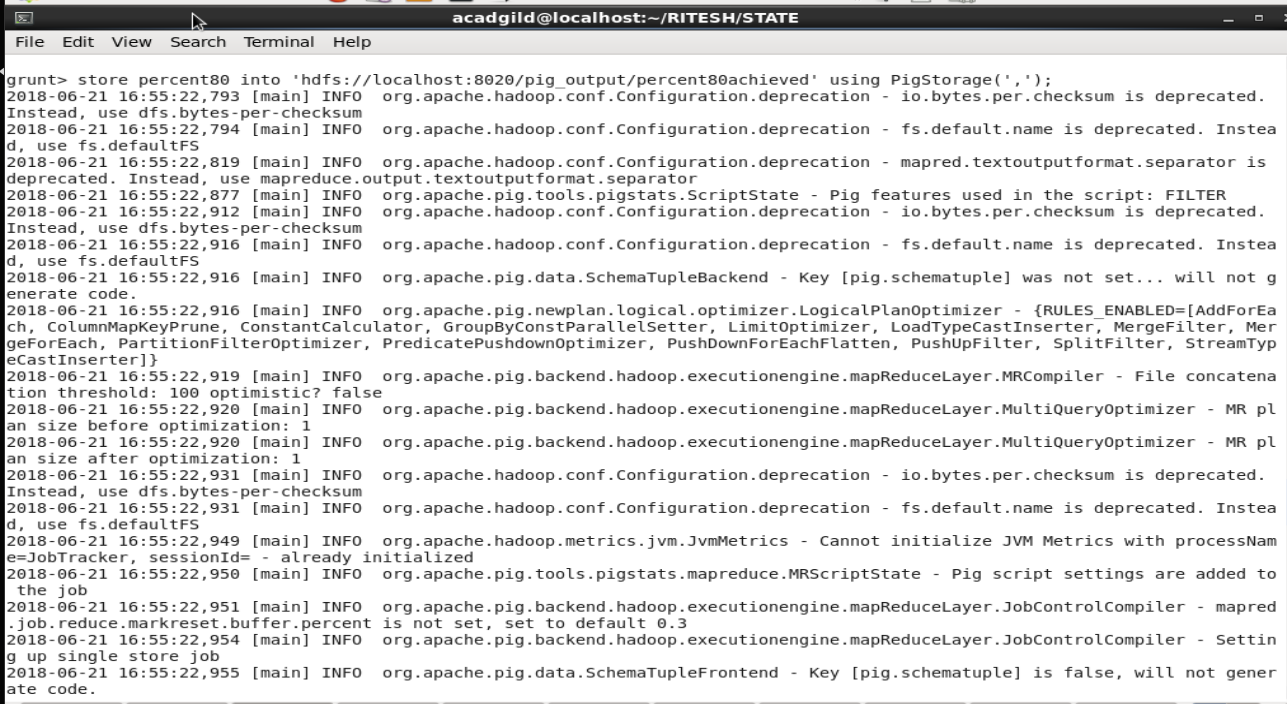
* Filtered the data where BPL cards completion is greater than 80

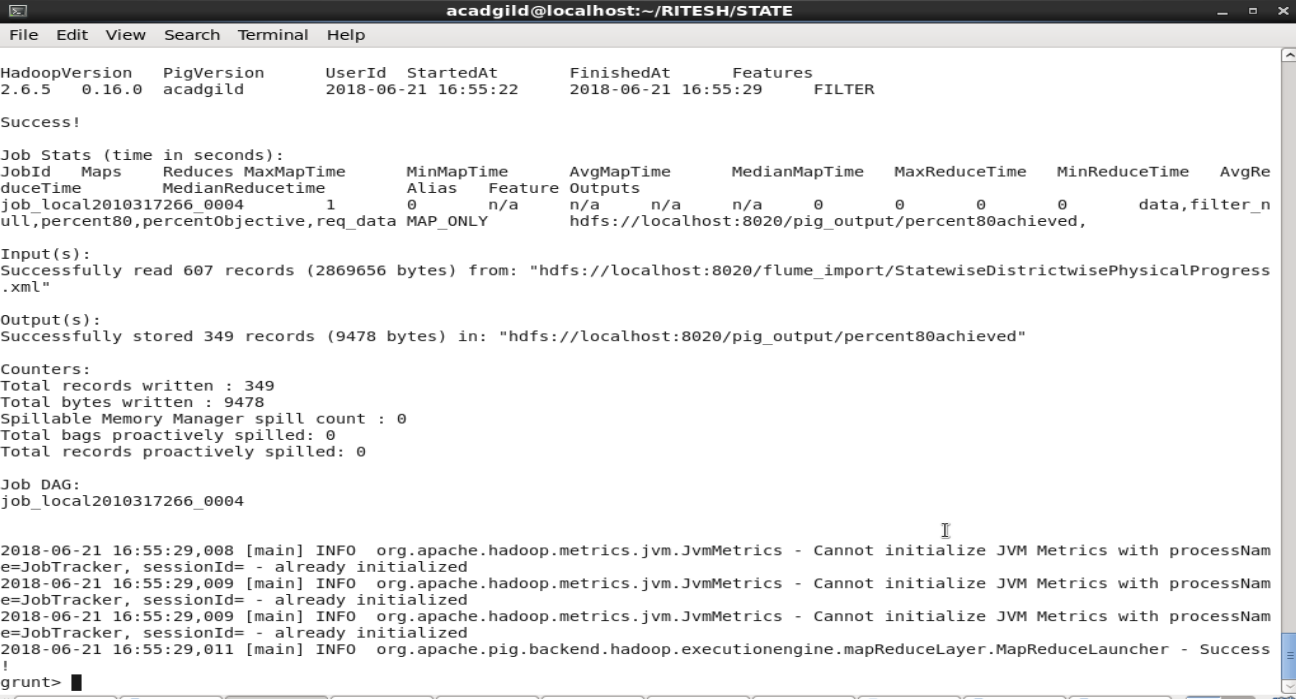
**percent80 = FILTER percentObjective BY BPL\_Percent>=80.00;**



* Store the data into hdfs

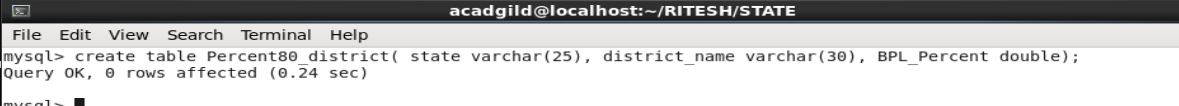
**store percent80 into 'hdfs://localhost:8020/pig\_output/percent80achieved' using PigStorage(',');**





* Created the table 'Percent80\_district'

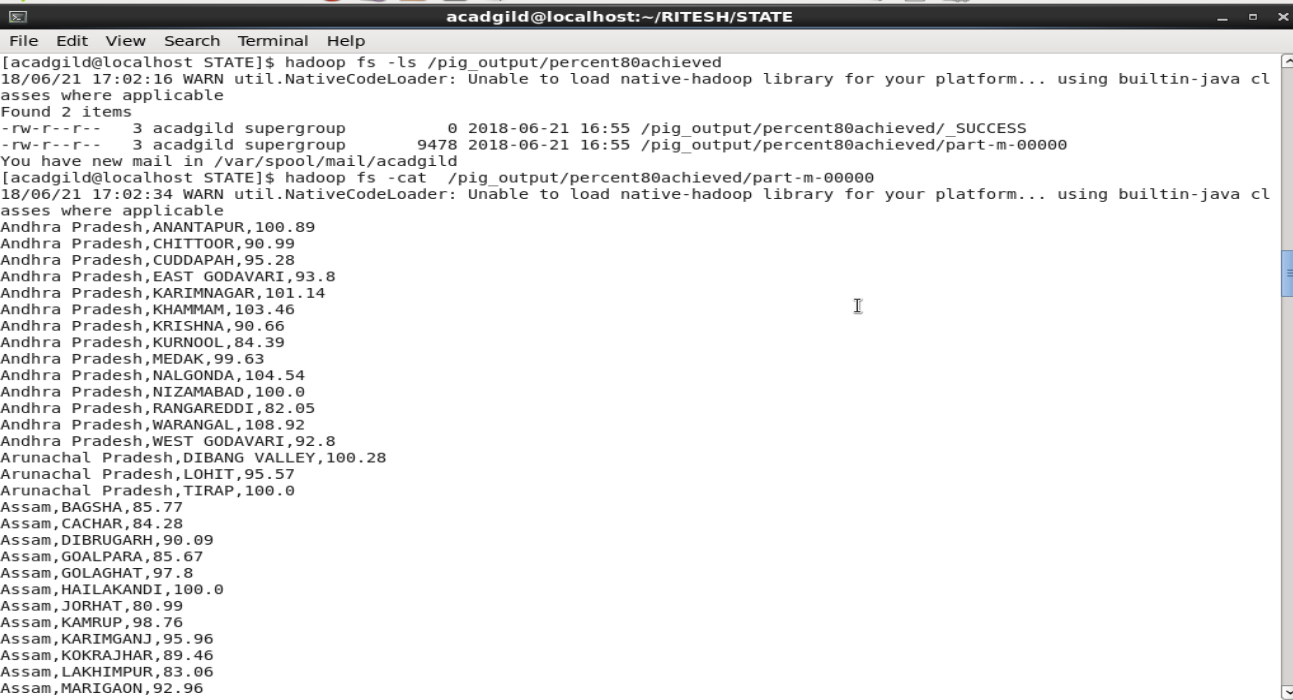
**create table districtCentPercent(state varchar(25), district\_name varchar(30),BPL\_percent double);**



* Checked the wheteher the pig output was stored or not in HDFS.

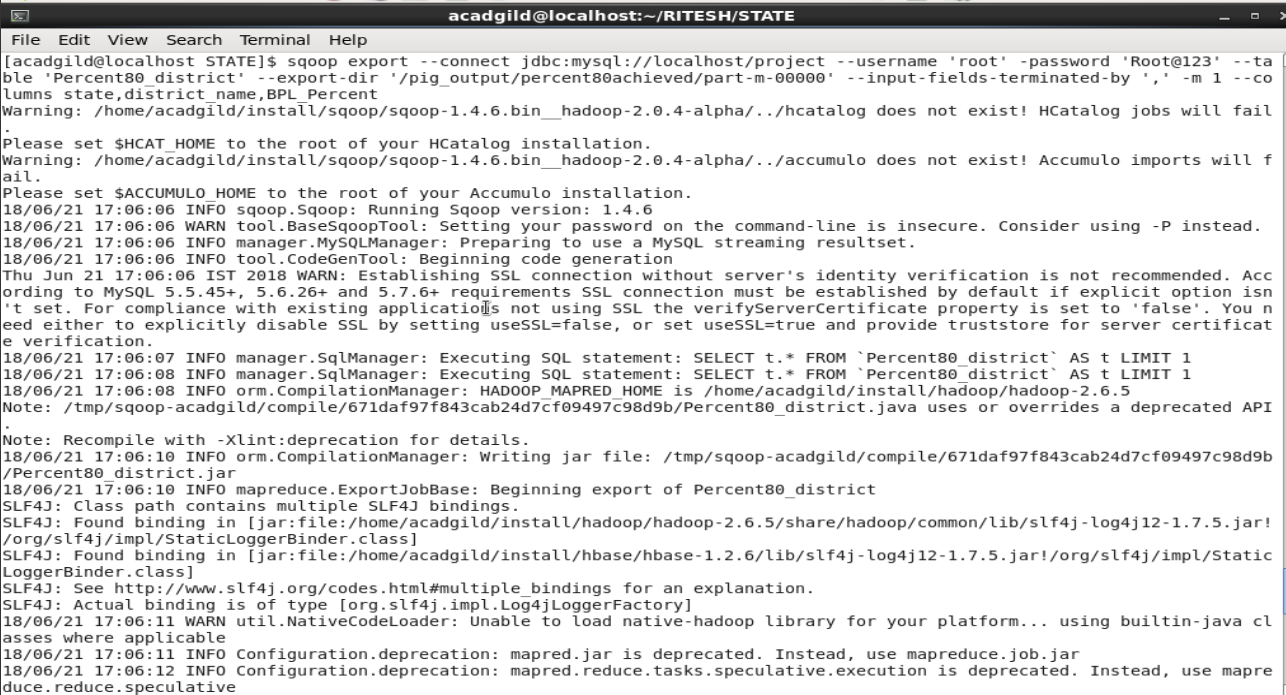
**hadoop fs -ls /pig\_output/percent80achieved**

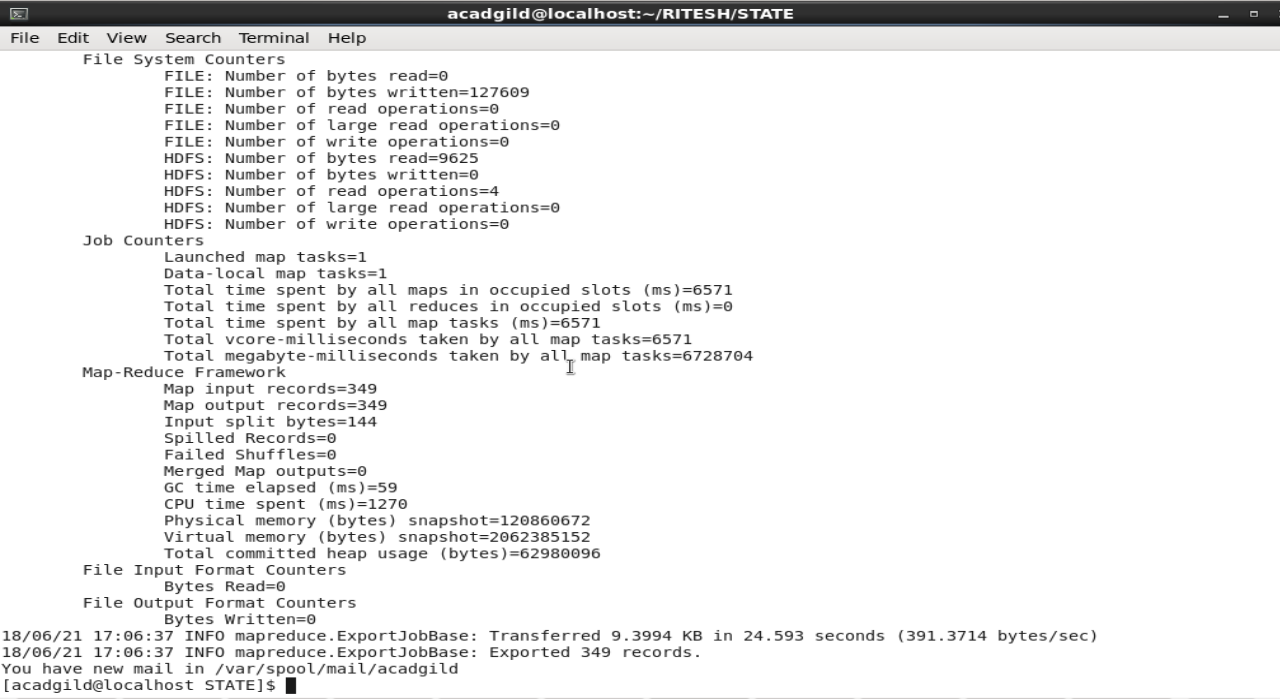
**hadoop fs -cat /pig\_output/percent80achieved/part-m-00000**



* Using sqoop we exported the data from HDFS to mysql.

**sqoop export –connect jdbc:mysql://localhost/project –username 'root' -password** [**'Root@123**](mailto:'Root@123)**' –table 'Percent80\_district' –export-dir '/pig\_output/percent80achieved/part-m-00000' –input-fields-terminated-by ',' -m 1 –columns state,district,name,BPL\_Percent**





* Checked the content of the table in mysql

**select \* from Percent80\_district;**

