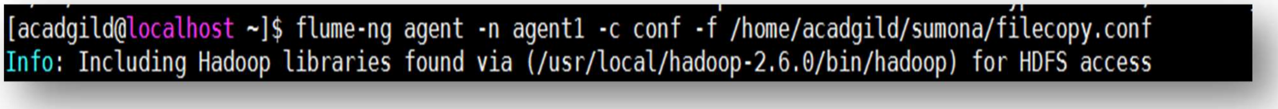
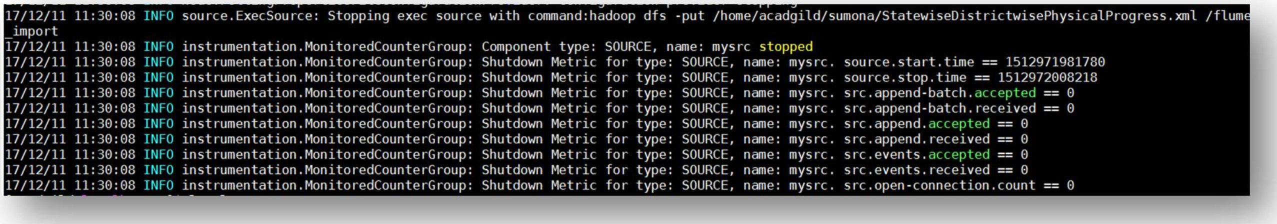
Project 2

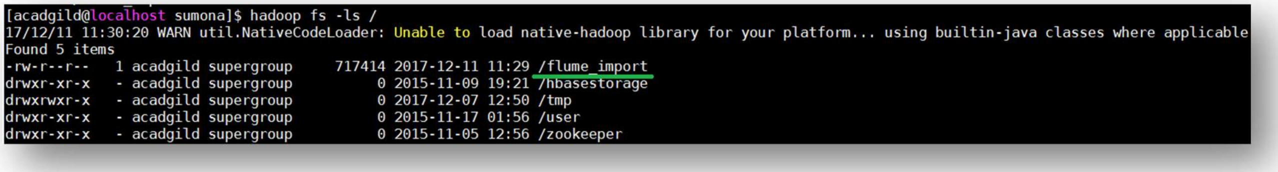
Copy dataset from local file system to HDFS using flume.



```
[acadgild@localhost ~]$ flume-ng agent -n agent1 -c conf -f /home/acadgild/sumona/filecopy.conf
Info: Including Hadoop libraries found via (/usr/local/hadoop-2.6.0/bin/hadoop) for HDFS access
```

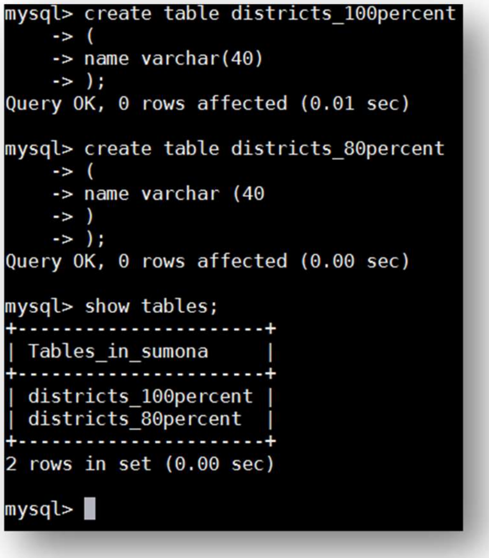


```
17/12/11 11:30:08 INFO source.ExecSource: Stopping exec source with command:hadoop dfs -put /home/acadgild/sumona/StatewiseDistrictwisePhysicalProgress.xml /flume
_import
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Component type: SOURCE, name: mysrc stopped
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. source.start.time == 1512971981780
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. source.stop.time == 1512972008218
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.append-batch.accepted == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.append-batch.received == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.append.accepted == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.append.received == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.events.accepted == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.events.received == 0
17/12/11 11:30:08 INFO instrumentation.MonitoredCounterGroup: Shutdown Metric for type: SOURCE, name: mysrc. src.open-connection.count == 0
```



```
[acadgild@localhost sumona]$ hadoop fs -ls /
17/12/11 11:30:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 5 items
-rw-r--r--   1 acadgild supergroup     717414 2017-12-11 11:29 /flume_import
drwxr-xr-x   - acadgild supergroup          0 2015-11-09 19:21 /hbasestorage
drwxrwxr-x   - acadgild supergroup          0 2017-12-07 12:50 /tmp
drwxr-xr-x   - acadgild supergroup          0 2015-11-17 01:56 /user
drwxr-xr-x   - acadgild supergroup          0 2015-11-05 12:56 /zookeeper
```

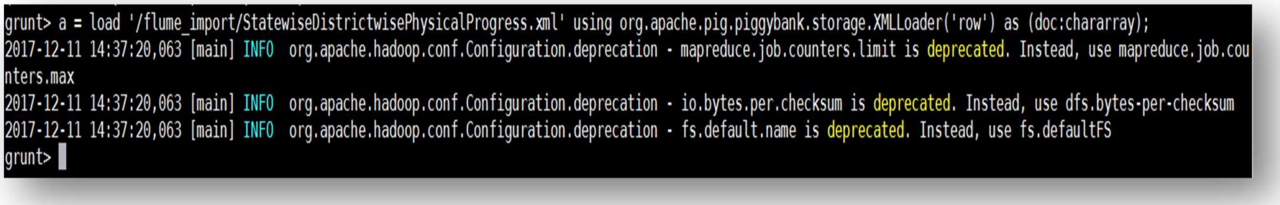
Created tables in MySQL



```
mysql> create table districts_100percent
    -> (
    -> name varchar(40)
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> create table districts_80percent
    -> (
    -> name varchar (40
    -> )
    -> );
Query OK, 0 rows affected (0.00 sec)

mysql> show tables;
+---------------------+
| Tables_in_sumona    |
+---------------------+
| districts_100percent |
| districts_80percent  |
+---------------------+
2 rows in set (0.00 sec)

mysql> 
```

Loaded the XML file into PIG

```
a = load '/flume_import/StatewiseDistrictwisePhysicalProgress.xml' using pig.XML.newloader('row') as (doc:chararray);
```



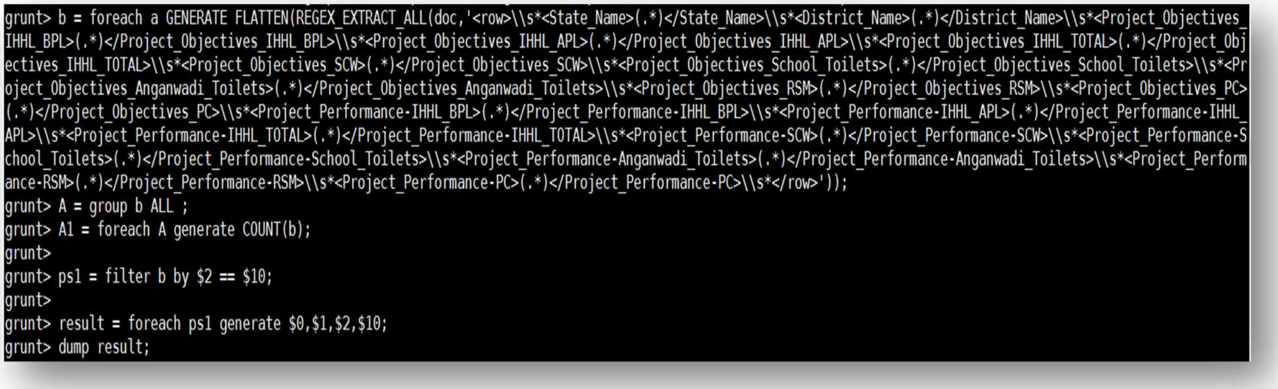
```
grunt> a = load '/flume_import/StatewiseDistrictwisePhysicalProgress.xml' using org.apache.pig.piggybank.storage.XMLLoader('row') as (doc:chararray);
2017-12-11 14:37:20,063 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.cou
nters.max
2017-12-11 14:37:20,063 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-12-11 14:37:20,063 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt>
```

1. Find out the districts who achieved 100 percent objective in BPL cards

b = Column names of the XML file

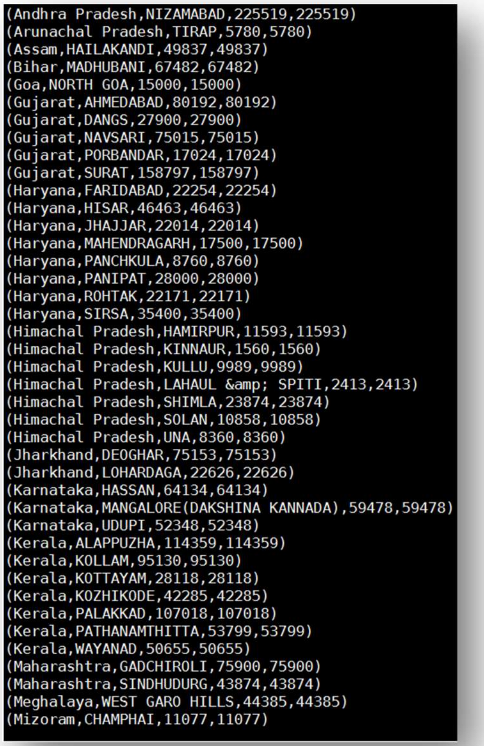
```
b = foreach a GENERATE
FLATTEN(REGEX_EXTRACT_ALL(doc,'<row>\\s*<State_Name>(.*)</State_Name>\\s*<District_Name>(.*)</District_Name>\\s*<Project_Objectives_IHHL_BPL>(.*)</Project_Objectives_IHHL_BPL>\\s*<Project_Objectives_IHHL_APL>(.*)</Project_Objectives_IHHL_APL>\\s*<Project_Objectives_IHHL_TOTAL>(.*)</Project_Objectives_IHHL_TOTAL>\\s*<Project_Objectives_SCW>(.*)</Project_Objectives_SCW>\\s*<Project_Objectives_School_Toilets>(.*)</Project_Objectives_School_Toilets>\\s*<Project_Objectives_Anganwadi_Toilets>(.*)</Project_Objectives_Anganwadi_Toilets>\\s*<Project_Objectives_RSM>(.*)</Project_Objectives_RSM>\\s*<Project_Objectives_PC>(.*)</Project_Objectives_PC>\\s*<Project_Performance-IHHL_BPL>(.*)</Project_Performance-IHHL_BPL>\\s*<Project_Performance-IHHL_APL>(.*)</Project_Performance-IHHL_APL>\\s*<Project_Performance-IHHL_TOTAL>(.*)</Project_Performance-IHHL_TOTAL>\\s*<Project_Performance-SCW>(.*)</Project_Performance-SCW>\\s*<Project_Performance-School_Toilets>(.*)</Project_Performance-School_Toilets>\\s*<Project_Performance-Anganwadi_Toilets>(.*)</Project_Performance-Anganwadi_Toilets>\\s*<Project_Performance-RSM>(.*)</Project_Performance-RSM>\\s*<Project_Performance-PC>(.*)</Project_Performance-PC>\\s*</row>'));
```

```
A = group b ALL ;

A1 = foreach A generate COUNT(b);

ps1 = filter b by $2 == $10 * 80/100;

result = foreach ps1 generate $0,$1,$2,$10;
```

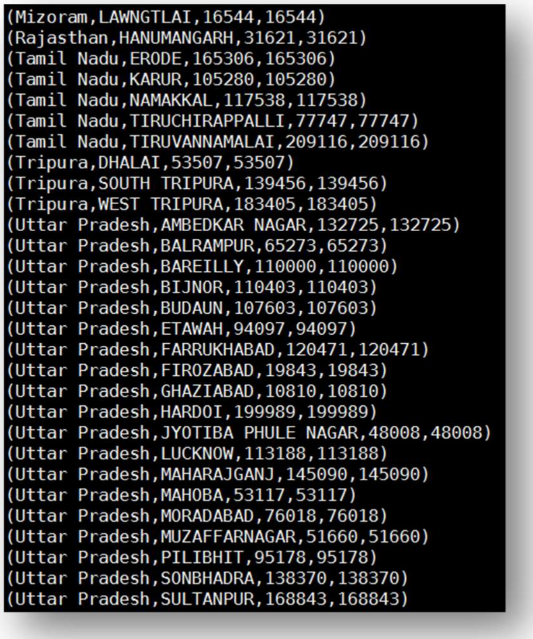


```
grunt> b = foreach a GENERATE FLATTEN(REGEX_EXTRACT_ALL(doc,'<row>\\s*<State_Name>(.*)</State_Name>\\s*<District_Name>(.*)</District_Name>\\s*<Project_Objectives_
IHHL_BPL>(.*)</Project_Objectives_IHHL_BPL>\\s*<Project_Objectives_IHHL_APL>(.*)</Project_Objectives_IHHL_APL>\\s*<Project_Objectives_IHHL_TOTAL>(.*)</Project_Obj
ectives_IHHL_TOTAL>\\s*<Project_Objectives_SCW>(.*)</Project_Objectives_SCW>\\s*<Project_Objectives_School_Toilets>(.*)</Project_Objectives_School_Toilets>\\s*<Pr
oject_Objectives_Anganwadi_Toilets>(.*)</Project_Objectives_Anganwadi_Toilets>\\s*<Project_Objectives_RSM>(.*)</Project_Objectives_RSM>\\s*<Project_Objectives_PC>
(.*)</Project_Objectives_PC>\\s*<Project_Performance-IHHL_BPL>(.*)</Project_Performance-IHHL_BPL>\\s*<Project_Performance-IHHL_APL>(.*)</Project_Performance-IHHL_
APL>\\s*<Project_Performance-IHHL_TOTAL>(.*)</Project_Performance-IHHL_TOTAL>\\s*<Project_Performance-SCW>(.*)</Project_Performance-SCW>\\s*<Project_Performance-S
chool_Toilets>(.*)</Project_Performance-School_Toilets>\\s*<Project_Performance-Anganwadi_Toilets>(.*)</Project_Performance-Anganwadi_Toilets>\\s*<Project_Perform
ance-RSM>(.*)</Project_Performance-RSM>\\s*<Project_Performance-PC>(.*)</Project_Performance-PC>\\s*</row>'));
grunt> A = group b ALL ;
grunt> A1 = foreach A generate COUNT(b);
grunt>
grunt> ps1 = filter b by $2 == $10;
grunt>
grunt> result = foreach ps1 generate $0,$1,$2,$10;
grunt> dump result;
```

Output:



```
(Andhra Pradesh,NIZAMABAD,225519,225519)
(Arunachal Pradesh,TIRAP,5780,5780)
(Assam,HAILAKANDI,49837,49837)
(Bihar,MADHUBANI,67482,67482)
(Goa,NORTH GOA,15000,15000)
(Gujarat,AHMEDABAD,80192,80192)
(Gujarat,DANGS,27900,27900)
(Gujarat,NAVSARI,75015,75015)
(Gujarat,PORBANDAR,17024,17024)
(Gujarat,SURAT,158797,158797)
(Haryana,FARIDABAD,22254,22254)
(Haryana,HISAR,46463,46463)
(Haryana,JHAJJAR,22014,22014)
(Haryana,MAHENDRAGARH,17500,17500)
(Haryana,PANCHKULA,8760,8760)
(Haryana,PANIPAT,28000,28000)
(Haryana,ROHTAK,22171,22171)
(Haryana,SIRSA,35400,35400)
(Himachal Pradesh,HAMIRPUR,11593,11593)
(Himachal Pradesh,KINNAUR,1560,1560)
(Himachal Pradesh,KULLU,9989,9989)
(Himachal Pradesh,LAHAUL &amp; SPITI,2413,2413)
(Himachal Pradesh,SHIMLA,23874,23874)
(Himachal Pradesh,SOLAN,10858,10858)
(Himachal Pradesh,UNA,8360,8360)
(Jharkhand,DEOGHAR,75153,75153)
(Jharkhand,LOHARDAGA,22626,22626)
(Karnataka,HASSAN,64134,64134)
(Karnataka,MANGALORE(DAKSHINA KANNADA),59478,59478)
(Karnataka,UDUPI,52348,52348)
(Kerala,ALAPPUZHA,114359,114359)
(Kerala,KOLLAM,95130,95130)
(Kerala,KOTTAYAM,28118,28118)
(Kerala,KOZHIKODE,42285,42285)
(Kerala,PALAKKAD,107018,107018)
(Kerala,PATHANAMTHITTA,53799,53799)
(Kerala,WAYANAD,50655,50655)
(Maharashtra,GADCHIROLI,75900,75900)
(Maharashtra,SINDHUDURG,43874,43874)
(Meghalaya,WEST GARO HILLS,44385,44385)
(Mizoram,CHAMPHAI,11077,11077)
```

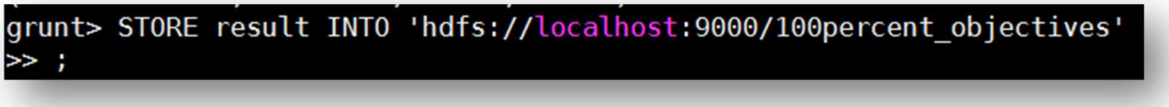


```
(Mizoram,LAWNGTLAI,16544,16544)
(Rajasthan,HANUMANGARH,31621,31621)
(Tamil Nadu,ERODE,165306,165306)
(Tamil Nadu,KARUR,105280,105280)
(Tamil Nadu,NAMAKKAL,117538,117538)
(Tamil Nadu,TIRUCHIRAPPALLI,77747,77747)
(Tamil Nadu,TIRUVANNAMALAI,209116,209116)
(Tripura,DHALAI,53507,53507)
(Tripura,SOUTH TRIPURA,139456,139456)
(Tripura,WEST TRIPURA,183405,183405)
(Uttar Pradesh,AMBEDKAR NAGAR,132725,132725)
(Uttar Pradesh,BALRAMPUR,65273,65273)
(Uttar Pradesh,BAREILLY,110000,110000)
(Uttar Pradesh,BIJNOR,110403,110403)
(Uttar Pradesh,BUDAUN,107603,107603)
(Uttar Pradesh,ETAWAH,94097,94097)
(Uttar Pradesh,FARRUKHABAD,120471,120471)
(Uttar Pradesh,FIROZABAD,19843,19843)
(Uttar Pradesh,GHAZIABAD,10810,10810)
(Uttar Pradesh,HARDOI,199989,199989)
(Uttar Pradesh,JYOTIBA PHULE NAGAR,48008,48008)
(Uttar Pradesh,LUCKNOW,113188,113188)
(Uttar Pradesh,MAHARAJGANJ,145090,145090)
(Uttar Pradesh,MAHOBA,53117,53117)
(Uttar Pradesh,MORADABAD,76018,76018)
(Uttar Pradesh,MUZAFFARNAGAR,51660,51660)
(Uttar Pradesh,PILIBHIT,95178,95178)
(Uttar Pradesh,SONBHADRA,138370,138370)
(Uttar Pradesh,SULTANPUR,168843,168843)
```

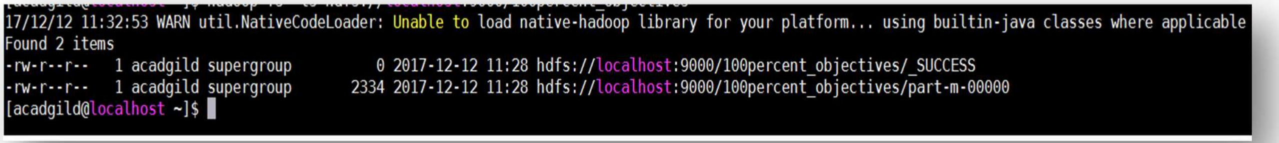
Exported this output to MySQL

First, we shall store the result into the folder created in Hadoop

STORE result INTO 'hdfs://localhost:9000/100percent_objectives'

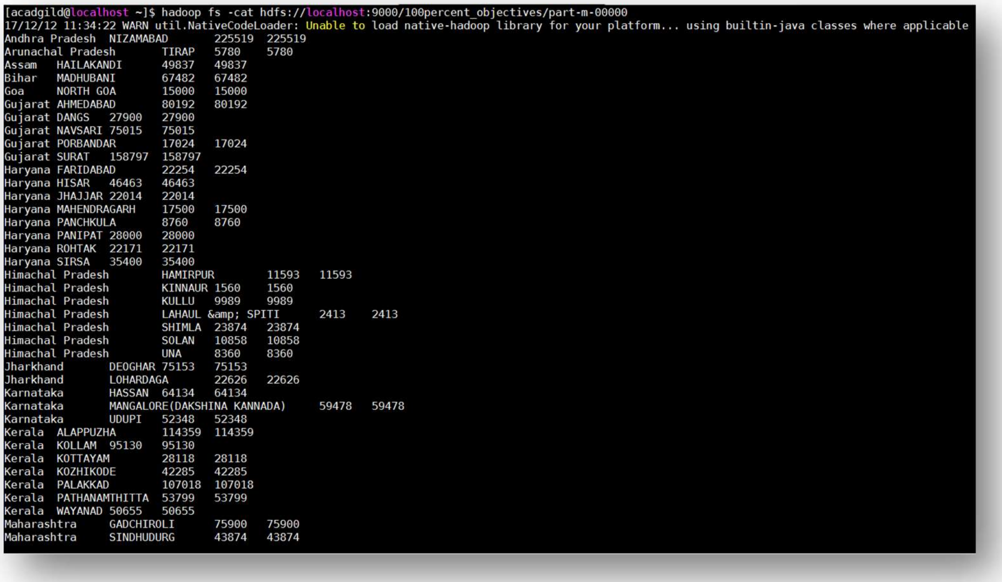


```
grunt> STORE result INTO 'hdfs://localhost:9000/100percent_objectives'
>> ;
```



```
17/12/12 11:32:53 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2017-12-12 11:28 hdfs://localhost:9000/100percent_objectives/_SUCCESS
-rw-r--r--   1 acadgild supergroup       2334 2017-12-12 11:28 hdfs://localhost:9000/100percent_objectives/part-m-00000
[acadgild@localhost ~]$ 
```

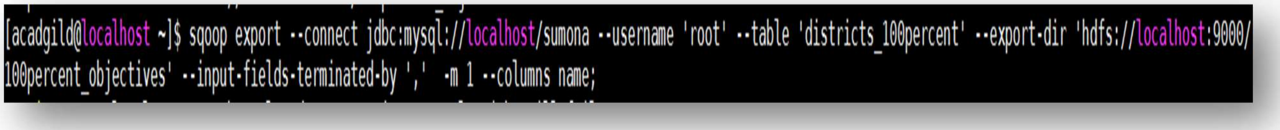
Let us cat part-m-00000 and check if the data has been loaded.



```
[acadgild@localhost ~]$ hadoop fs -cat hdfs://localhost:9000/100percent_objectives/part-m-00000
17/12/12 11:34:22 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Andhra Pradesh  NIZAMABAD       225519  225519
Arunachal Pradesh       TIRAP   5780    5780
Assam   HAILAKANDI      49837   49837
Bihar   MADHUBANI       67482   67482
Goa     NORTH GOA       15000   15000
Gujarat AHMEDABAD       80192   80192
Gujarat DANGS   27900   27900
Gujarat NAVSARI 75015   75015
Gujarat PORBANDAR       17024   17024
Gujarat SURAT   158797  158797
Haryana FARIDABAD       22254   22254
Haryana HISAR   46463   46463
Haryana JHAJJAR 22014   22014
Haryana MAHENDRAGARH    17500   17500
Haryana PANCHKULA       8760    8760
Haryana PANIPAT 28000   28000
Haryana ROHTAK  22171   22171
Haryana SIRSA   35400   35400
Himachal Pradesh        HAMIRPUR        11593   11593
Himachal Pradesh        KINNAUR 1560    1560
Himachal Pradesh        KULLU   9989    9989
Himachal Pradesh        LAHAUL &amp; SPITI      2413    2413
Himachal Pradesh        SHIMLA  23874   23874
Himachal Pradesh        SOLAN   10858   10858
Himachal Pradesh        UNA     8360    8360
Jharkhand       DEOGHAR 75153   75153
Jharkhand       LOHARDAGA       22626   22626
Karnataka       HASSAN  64134   64134
Karnataka       MANGALORE(DAKSHINA KANNADA)     59478   59478
Karnataka       UDUPI   52348   52348
Kerala  ALAPPUZHA       114359  114359
Kerala  KOLLAM  95130   95130
Kerala  KOTTAYAM        28118   28118
Kerala  KOZHIKODE       42285   42285
Kerala  PALAKKAD        107018  107018
Kerala  PATHANAMTHITTA  53799   53799
Kerala  WAYANAD 50655   50655
Maharashtra     GADCHIROLI      75900   75900
Maharashtra     SINDHUDURG      43874   43874
```

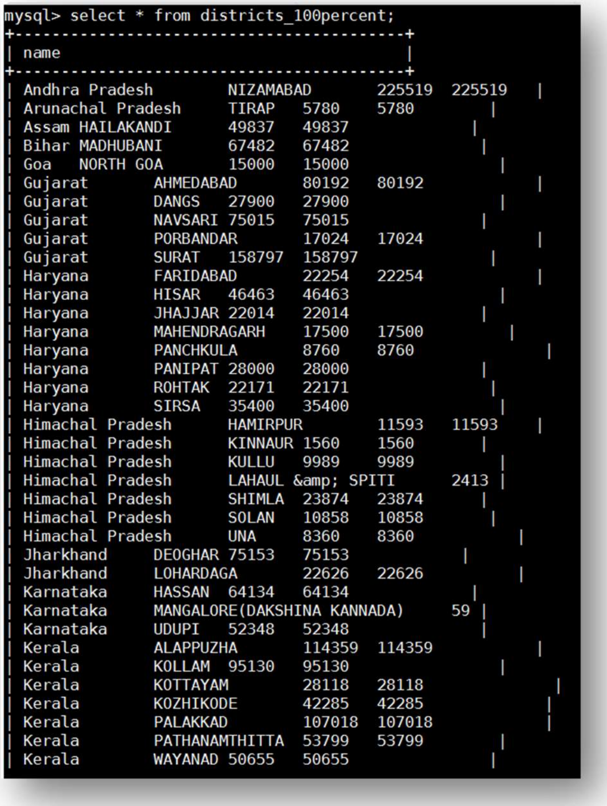
Use sqoop to export the data from HDFS to MySQL

sqoop export --connect jdbc:mysql://localhost/sumona --username 'root' --table 'districts_100percent' --export-dir 'hdfs://localhost:9000/100percent_objectives' --input-fields-terminated-by ',' -m 1 --columns name;



```
[acadgild@localhost ~]$ sqoop export --connect jdbc:mysql://localhost/sumona --username 'root' --table 'districts_100percent' --export-dir 'hdfs://localhost:9000/
100percent_objectives' --input-fields-terminated-by ','  -m 1 --columns name;
```

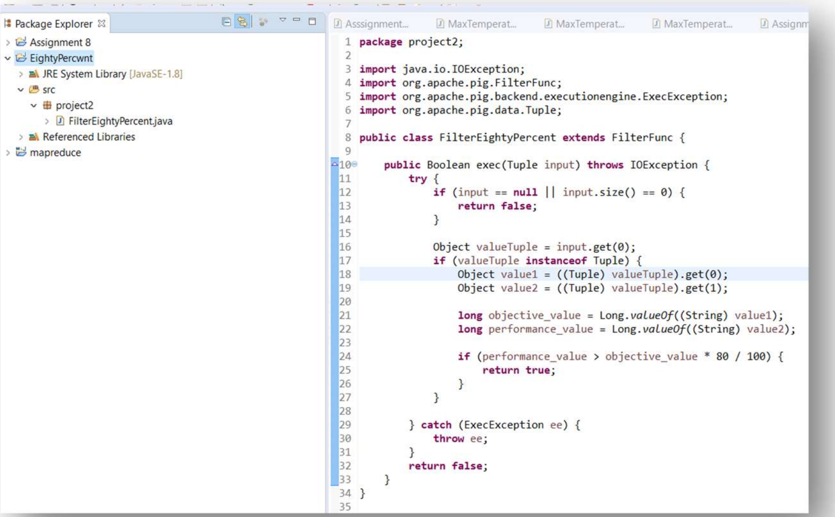
Now let us check the table in MySQL



```
mysql> select * from districts_100percent;
+------------------------------------------+
| name                                     |
+------------------------------------------+
| Andhra Pradesh        NIZAMABAD       225519  225519   |
| Arunachal Pradesh     TIRAP   5780    5780        |
| Assam HAILAKANDI      49837   49837             |
| Bihar MADHUBANI       67482   67482              |
| Goa   NORTH GOA       15000   15000                |
| Gujarat       AHMEDABAD       80192   80192            |
| Gujarat       DANGS   27900   27900                |
| Gujarat       NAVSARI 75015   75015              |
| Gujarat       PORBANDAR       17024   17024            |
| Gujarat       SURAT   158797  158797              |
| Haryana       FARIDABAD       22254   22254            |
| Haryana       HISAR   46463   46463                |
| Haryana       JHAJJAR 22014   22014              |
| Haryana       MAHENDRAGARH    17500   17500         |
| Haryana       PANCHKULA       8760    8760              |
| Haryana       PANIPAT 28000   28000              |
| Haryana       ROHTAK  22171   22171               |
| Haryana       SIRSA   35400   35400                |
| Himachal Pradesh      HAMIRPUR        11593   11593    |
| Himachal Pradesh      KINNAUR 1560    1560       |
| Himachal Pradesh      KULLU   9989    9989         |
| Himachal Pradesh      LAHAUL &amp; SPITI      2413 |
| Himachal Pradesh      SHIMLA  23874   23874      |
| Himachal Pradesh      SOLAN   10858   10858       |
| Himachal Pradesh      UNA     8360    8360           |
| Jharkhand     DEOGHAR 75153   75153            |
| Jharkhand     LOHARDAGA       22626   22626          |
| Karnataka     HASSAN  64134   64134             |
| Karnataka     MANGALORE(DAKSHINA KANNADA)     59 |
| Karnataka     UDUPI   52348   52348              |
| Kerala        ALAPPUZHA       114359  114359           |
| Kerala        KOLLAM  95130   95130                |
| Kerala        KOTTAYAM        28118   28118              |
| Kerala        KOZHIKODE       42285   42285             |
| Kerala        PALAKKAD        107018  107018            |
| Kerala        PATHANAMTHITTA  53799   53799        |
| Kerala        WAYANAD 50655   50655               |
```

2. Write a Pig UDF to filter the districts which have reached 80% of objectives of BPL cards.

Created an UDF to find the districts having reached 80% of Objectives of BPL cards.



```java
package project2;

import java.io.IOException;
import org.apache.pig.FilterFunc;
import org.apache.pig.backend.executionengine.ExecException;
import org.apache.pig.data.Tuple;

public class FilterEightyPercent extends FilterFunc {

    public Boolean exec(Tuple input) throws IOException {
        try {
            if (input == null || input.size() == 0) {
                return false;
            }

            Object valueTuple = input.get(0);
            if (valueTuple instanceof Tuple) {
                Object value1 = ((Tuple) valueTuple).get(0);
                Object value2 = ((Tuple) valueTuple).get(1);

                long objective_value = Long.valueOf((String) value1);
                long performance_value = Long.valueOf((String) value2);

                if (performance_value > objective_value * 80 / 100) {
                    return true;
                }
            }

        } catch (ExecException ee) {
            throw ee;
        }
        return false;
    }
}
```

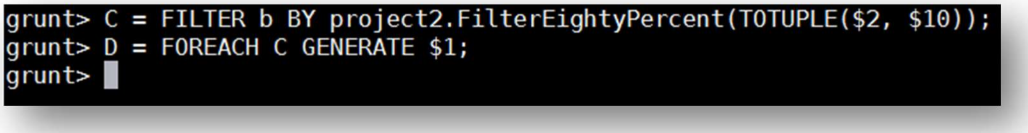
Extract the UDF as jar and register it in pig

REGISTER /home/acadgild/sumona/project2.jar;

Now PIG commands to find districts which have reached 80% of objectives of BPL cards

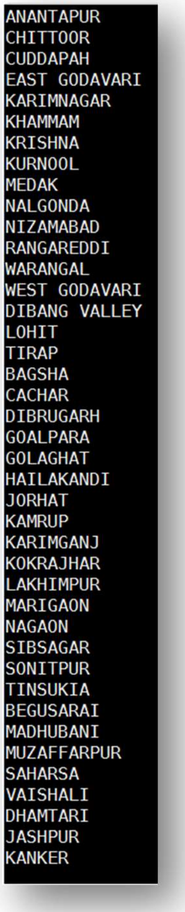
C = FILTER b BY project2.FilterEightyPercent(TOTUPLE($2, $10));

D = FOREACH C GENERATE $1;



```
grunt> C = FILTER b BY project2.FilterEightyPercent(TOTUPLE($2, $10));
grunt> D = FOREACH C GENERATE $1;
grunt> 
```

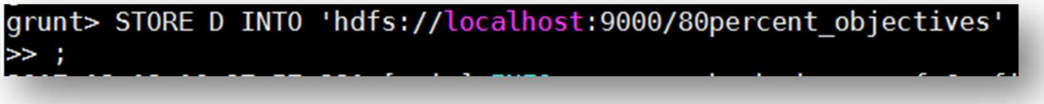
Output:



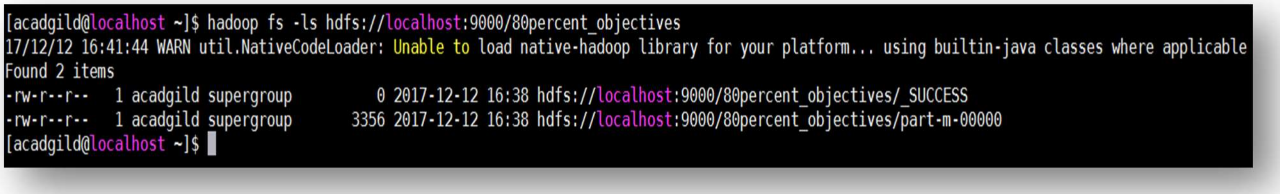
```
ANANTAPUR
CHITTOOR
CUDDAPAH
EAST GODAVARI
KARIMNAGAR
KHAMMAM
KRISHNA
KURNOOL
MEDAK
NALGONDA
NIZAMABAD
RANGAREDDI
WARANGAL
WEST GODAVARI
DIBANG VALLEY
LOHIT
TIRAP
BAGSHA
CACHAR
DIBRUGARH
GOALPARA
GOLAGHAT
HAILAKANDI
JORHAT
KAMRUP
KARIMGANJ
KOKRAJHAR
LAKHIMPUR
MARIGAON
NAGAON
SIBSAGAR
SONITPUR
TINSUKIA
BEGUSARAI
MADHUBANI
MUZAFFARPUR
SAHARSA
VAISHALI
DHAMTARI
JASHPUR
KANKER
```

Store the output in the HDFS dir

STORE D INTO 'hdfs://localhost:9000/80percent_objectives'



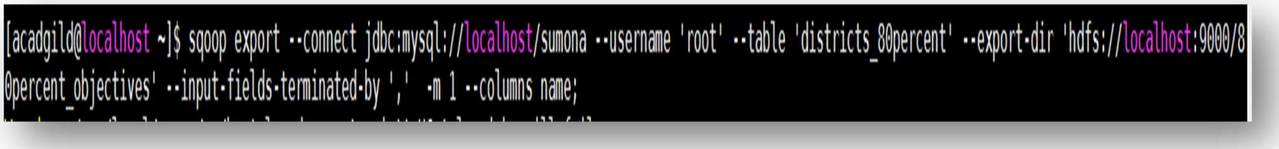
```
grunt> STORE D INTO 'hdfs://localhost:9000/80percent_objectives'
>> ;
```



```
[acadgild@localhost ~]$ hadoop fs -ls hdfs://localhost:9000/80percent_objectives
17/12/12 16:41:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2017-12-12 16:38 hdfs://localhost:9000/80percent_objectives/_SUCCESS
-rw-r--r--   1 acadgild supergroup       3356 2017-12-12 16:38 hdfs://localhost:9000/80percent_objectives/part-m-00000
[acadgild@localhost ~]$ 
```

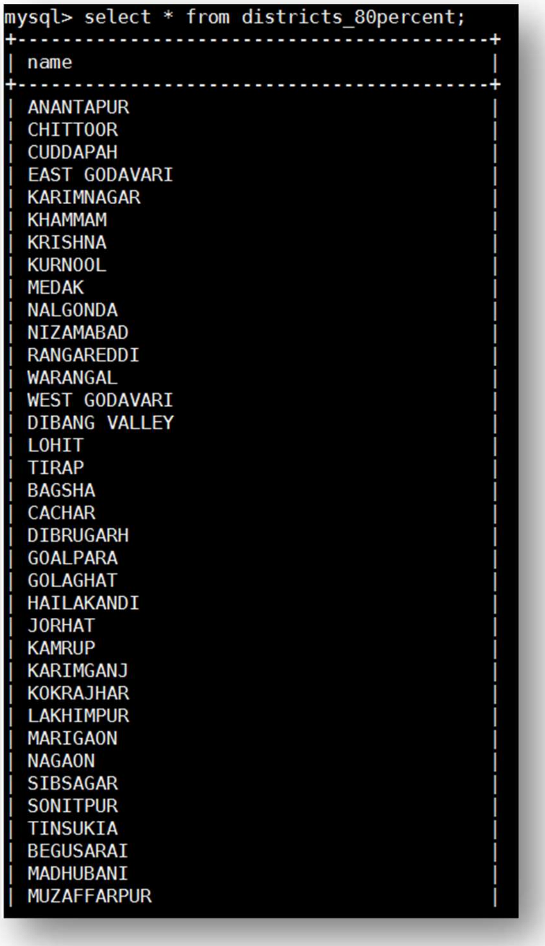
Now, we shall export the data from HDFS to MySQL using sqoop

sqoop export --connect jdbc:mysql://localhost/sumona --username 'root' --table 'districts_80percent' --export-dir 'hdfs://localhost:9000/80percent_objectives' --input-fields-terminated-by ',' -m 1 --columns name;



```
[acadgild@localhost ~]$ sqoop export --connect jdbc:mysql://localhost/sumona --username 'root' --table 'districts_80percent' --export-dir 'hdfs://localhost:9000/8
0percent_objectives' --input-fields-terminated-by ',' -m 1 --columns name;
```

Now we shall check the tables in MySQL



```
mysql> select * from districts_80percent;
+------------------------------------------+
| name                                     |
+------------------------------------------+
| ANANTAPUR                                |
| CHITTOOR                                 |
| CUDDAPAH                                 |
| EAST GODAVARI                            |
| KARIMNAGAR                               |
| KHAMMAM                                  |
| KRISHNA                                  |
| KURNOOL                                  |
| MEDAK                                    |
| NALGONDA                                 |
| NIZAMABAD                                |
| RANGAREDDI                               |
| WARANGAL                                 |
| WEST GODAVARI                            |
| DIBANG VALLEY                            |
| LOHIT                                    |
| TIRAP                                    |
| BAGSHA                                   |
| CACHAR                                   |
| DIBRUGARH                                |
| GOALPARA                                 |
| GOLAGHAT                                 |
| HAILAKANDI                               |
| JORHAT                                   |
| KAMRUP                                   |
| KARIMGANJ                                |
| KOKRAJHAR                                |
| LAKHIMPUR                                |
| MARIGAON                                 |
| NAGAON                                   |
| SIBSAGAR                                 |
| SONITPUR                                 |
| TINSUKIA                                 |
| BEGUSARAI                                |
| MADHUBANI                                |
| MUZAFFARPUR                              |
```